Topology authentication for CAPD models based on laplacian coordinates

Abstract

The intellectual property protection for 3D CAPD (computer-aided plant design) models features their intrinsical complex topology relation. This paper discusses digital watermarking technology for 3D CAPD models defined by using parametric solids, which may offer a solution to topology authentication. We first analyze the geometrical and topological structures of CAPD models, followed by discussion on the topology protection problem. Then we propose an effective semi-fragile watermarking method for topology authentication based on Laplacian coordinates and quantization index modulation (QIM) against several attacks. We compute the custom Laplacian coordinate vector for each mark connection point according to the topological relation among the joint plant components. The content-based watermark for each mark connection point is generated from selected attributes of its joint plant component. Watermarks are inserted into the coordinates of mark connection points by adjusting the lengths of their Laplacian coordinate vectors. Experimental results demonstrate that our approach not only can detect and locate malicious topology attacks such as components modification and joint ends modification, but also is robust against various non-malicious attacks such as similarity transformations and level-of-detail(LOD).
Keywords: Semi-fragile watermarking, Fragile watermarking,

Watermarking, Laplacian coordinates, Topology authentication

1. Introduction

Today's market is characterized by increasing competition. Companies need to find ways and means of reducing project costs and diminishing resources within basic and detail engineering, while at the same time sustaining optimum productivity. And this calls for improvements in process plant design [1]. Process plants are complex facilities mainly consisting of various plant components, such as equipments and pipelines which include pipes and piping components. In order to facilitate process plant design processes, research is actively carried out for developing methodologies and technologies of collaborative computer-aided plant design (CAPD) systems to support design teams geographically dispersed based on the quickly evolving information technologies. The CAPD system is an automatic solution provided for helping increase productivity, accuracy, and collaboration to meet the challenges of complex plant design projects. And it often refers to the automation technologies, work practices and business rules supporting the engineering and design of plants. In a collaborative CAPD system, designers and engineers inevitably share their work with globally distributed colleagues. Therefore, it is essential to confirm the integrity of all models for companies when sharing models with their collaborators. Digital semi-fragile watermarking provides a simple and reasonable solution for the integrity check of CPAD models [2].

Generally, we can describe the CAPD model by three kinds of information completely: the geometry information, the topology information and the engineering information. The geometry information describes the shape and 3D positions of all plant components. The topology information provides the complex topology relations among different plant components. The engineering information refers to design constraints, engineering disciplines and so on. Unlike the traditional mechanical computer aided design (CAD) industry which mainly concentrates on the geometric modeling, the CAPD systems mainly focuses on optimizing the plant layout[1]. Plant layout design devotes to find the most economical spatial arrangement of process vessels and equipment and their interconnecting pipes that satisfies construction, operation, maintenance, and safety requirements[3]. This is an important aspect in the design of process plants since a good layout will ensure that the plant functions correctly and will provide an economically acceptable balance between the many, often conflicting, design constraints [4]. Therefore, the topology information protecting is a significant part of intellectual property protection for CAPD models.

However, in the literature, existing watermarking schemes mainly target traditional mechanical 2D CAD drawings or 3D CAD models. Furthermore, these watermarking techniques mainly concentrate on the geometry information protection. Thus topology protection for CAPD models is still in its infancy and offers very interesting potentials for improvements because of their intrinsical complex topology. In this paper, we propose a semi-fragile watermarking scheme for addressing the issue of verifying the integrity of the topology information for CAPD models. The topology information is taken into consideration for both of the watermark generation and embedding. And the content-based watermarks are embedded in a subset of the model's connection points to keep them in a predefined relationship with neighboring connection points so that any changes will ruin the relationship between the marked connection points and neighboring connection points.

The rest of this paper is organized as follows. We review some related works in Section 2. Section 3 gives a brief introduction of CAPD models. Section 4 describes the proposed scheme. Experimental results that demonstrate our watermarking scheme performance are presented in Section 5. Conclusions follow in Section 6.

2. Related work

We review some related works about watermarking 3D CAD models in this section.

Digital watermarking techniques for 3D models have been widely studied since Ohbuchi first proposed a watermarking scheme for 3D models[5]. However, relatively few watermarking algorithms have been proposed for 3D CAD models especially for CAPD models. Watermarking schemes for 3D CAD models mainly target CAD-based drawings, NURBS curves, subdivision surfaces, CSG models, etc..

A CAD drawing can be represented by various geometric objects in some layers such as LINEs, ARCs, POLYGONs and 3DFACEs, which include the basic components of vertex, angle, radius, and so on. Park et al. proposed a digital watermarking scheme for 3D CAD drawings [6]. The scheme uses LINEs and 3D FACEs based on vertex in CAD system to prevent infringement of copyright from unlawfulness reproductions and distribution. Kwon et al. also proposed a watermarking scheme for 3D CAD drawings[7, 8]. The approach arbitrarily selects the LINE, FACE, and ARC components and embeds the watermark into the difference in length between the reference line and the connected lines in the case of line components, the circular radius in the case of the arc components, and the length ratio of two sides in the case of the face components. These schemes require the index and order of embedding components and the original point coordinates for watermark extraction. Therefore, they cannot detect watermarks when the components of the drawing are rearranged. A robust watermarking scheme based on geometric features with k-means++ clustering for the 3D CAD drawings was presented by Lee et al. [9]. The proposed scheme embeds the watermark into the geometric distribution of POLYLINE, 3DFACE, and ARC objects in the main layers. Ohbuchi et al. presented a watermarking scheme for 3D NURBS curves using reparameterization [10]. Their method is robust under affine transformations, but not under Möbius reparameterization. Lee et al. also present a method for watermarking NURBS data using two-dimensional virtual images[11]. A fragile watermarking schemes for authenticating CSG models was proposed by Fornaro and Sanna [12]. It computes the watermark from selected attributes of the model and stores it in one or more places into the model itself. Weng et al. proposed a method for watermarking T-spline curves and surfaces by using knot insertion[13]. In order to watermark subdivision surfaces, Cheung et al. present a robust non-blind watermarking scheme using modulating spectral coefficients of the subdivision control mesh[14]. Reuter et al. introduced a method to extract Shape-DNA, a numerical fingerprint or signature, of any 2d or 3d manifold (surface or solid) by taking the eigenvalues (i.e. the spectrum) of its Laplace-Beltrami operator

[15]. It uses the sequence of eigenvalues (spectrum) of the Laplace operator of a planar domain or 3d solid or the Laplace-Beltrami operator of a surface or parameterized solid in Euclidean space as a fingerprint.

3. CAPD Models

We give a brief introduction of the geometrical and topological modeling of CAPD models in this section.

3.1. Geometrical modeling

CAPD systems mainly focus on providing an effective and efficient platform to concentrate on the layout of tremendous number of plant components under complex constraints rather than shapes. Plant components are created using CSG (Constructive Solid Geometry) representation by combining basic solid entities which have simple shape such as sphere, cylinder, cone, etc..

In order to support the automatic generation of construction documents, such as isometrics, orthographics, etc., which directly exchanged with the model, plant components are defined using parameters. The main sections of a solid entity in the CAPD file are handle, entity type and geometric parameters. An example of a sphere entity is shown in Fig. 1. Plant components placed in a design model are parametric objects with a high degree of intelligence. Designers progressively construct a highly intelligent design database by placing instances of parametric components into the model.

3.2. Topological modeling

The layout poses significant limitations on the type, size and location of plant components. Positions of plant components can be simply described

Handle	4e
Entity Type	Sphere
Geometric Parameters	P: 10.2, 34.5, 1 R: 20.8

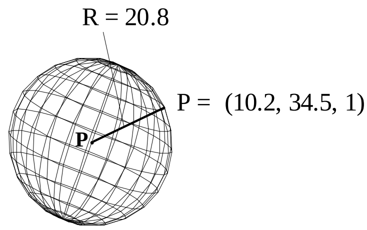


Figure 1: An example of a sphere entity.

by their absolute cartesian coordinates. But how to represent the interconnections among plant components is a key issue of CAPD systems. Not only should the layout represent the interconnection among two plant components, but it should also describe their interconnection ends. Only the two ends of different plant components which satisfy the specific requirements, such as pipe diameter, end type, pressure rating, and flow direction, can then be connected.

End connection can be mainly represented in two formats: connection points [16] and the order of plant components stored in the CAPD file. This paper aims to watermark CAPD models which describe the end connection by connection points since this format is one of the most widely used and effective representation for topological modeling.

Fig. 2 shows the main structure of connection points. Each connection point has the same attributes including geometry information, topology constraint, handle value and various engineering properties. And each connection point may have one joint connection point at most. In general, a connection point is defined as the center point of the end face. And it is added, deleted and transformed along with its corresponding plant component in CAPD systems. Connection points can be classified into two kinds:

invariant connection points and variant connection points. Invariant connection points have just to do with the structure of their plant components. While variant connection points are concomitant with some operations. For example, a new connection point will be added at the joint when we inserting a nozzle to an equipment. Unlike pipes and piping components, the number of connection points of equipments may hold is unlimited in theory. Fig. 3(a) shows connection points of some selected plant components. Fig. 3(b) shows the connection points of a simple pipeline. The interconnection between the two joint plant components C_1 and C_2 is represented through their connection points $P_{1,1}$ and $P_{2,0}$ respectively.

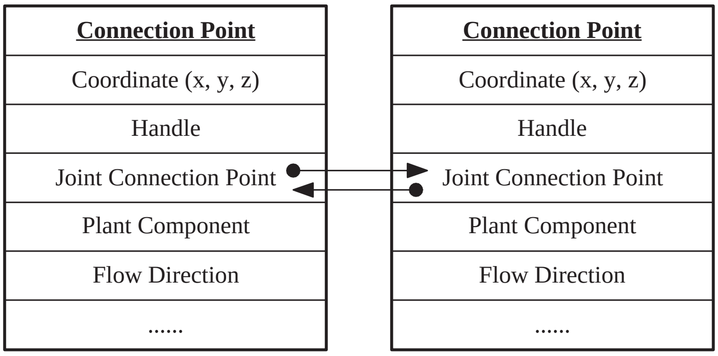


Figure 2: The structure of connection points.

In this paper, we discuss the problem of topology authentication for CAPD models from the following two aspects: joint plant components authentication and joint ends authentication. Joint plant components authentication aims to make sure that whether the joint plant components of each plant component are changed or not. While joint ends authentication further verifies whether the exact joint ends between the two joint plant components are modified or not. That is to say that, for each plant component, the

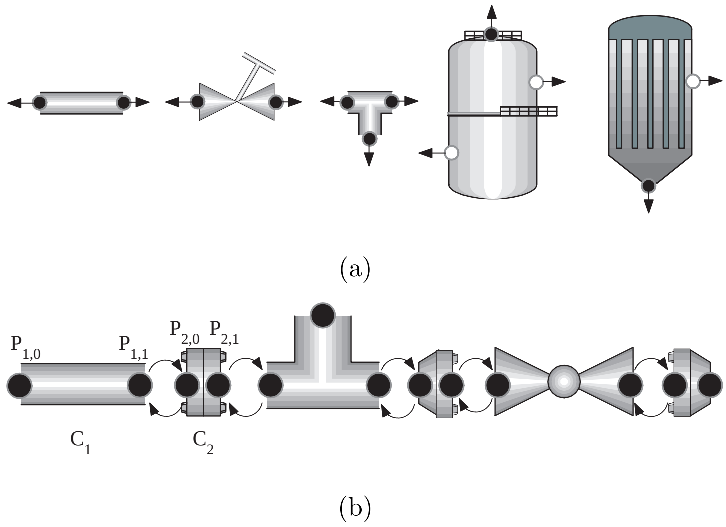


Figure 3: Examples of connection points of individual plant components and a simple pipeline. Black points are invariant connection points while white points are variant connection points. Note that all the connection points are scaled for better illustration. (a) Connection points of some selected plant components. (b) Connection points of a simple pipeline.

problem of topology authentication targets to verify not only its joint components, but also the exact joint ends, since a plant component usually has more than one joint ends.

4. The watermarking scheme for topology authentication

This section describes our topology verification method inspired on traditional Laplacian operators.

4.1. Overview of the method

The proposed watermarking scheme consists of two separate procedures, the *embedding procedure* and the *extraction procedure*, which is shown in Fig.

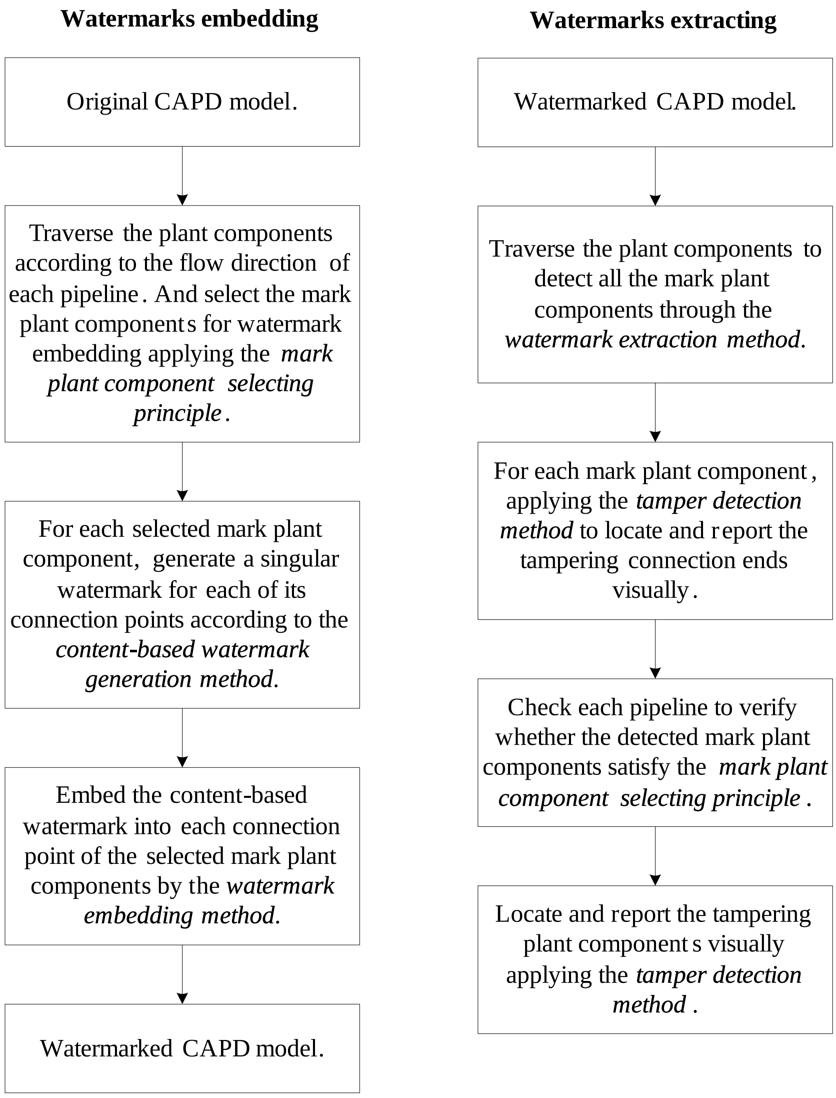


Figure 4: Overview of the proposed semi-fragile watermarking scheme.

4. The overview of the proposed watermarking scheme is described as follows.

In the watermark embedding stage, we first traverse the plant components of each pipeline according to its flow direction and select the mark plant components following the *mark component selecting principle*. Then, for each selected mark plant components, we generate a singular content-based watermark for each of its connection points, which are also called mark connection points, according to the *content-based watermark generation method*. After that, we calculate the Laplacian coordinate vector through the *computation of laplacian coordinate method* for each mark connection point. Finally, the content-based watermark is embedded into each mark connection point through modifying the length of its Laplacian coordinate vector based on the *watermarks embedding method*.

In the watermark extraction stage, the scheme uses the *watermarks extracting method* to detect and label all the mark plant components of the watermarked model. For each mark plant component, we first extract the embedded watermarks according to the *watermark extraction method* for each of its connection points. Second, we use the *content-based watermark generation method* to calculate the content-based watermarks for each of its connection point. Third, we verify the topology integrity of each joint end of the mark plant component by comparing the extracted watermarks with the calculated content-based watermarks applying the *tempering detection method*. Last, we report the tampering joint ends of the mark plant component visually. For each pipeline, we verify whether the detected mark plant components satisfy the *mark component selecting principle*. For those plant components which do not meet the *mark component selecting principle*, we locate and report them as suspicious tampering plant components visually.

4.2. Watermark targets

The objective of our scheme is to insert the watermark bits into the model to verify not only the joint plant components, but also the exact joint connection ends.To embed the watermark bits, a difficulty arise in our case:the geometrical parameters of plant components should be kept unmodified. Otherwise, the modification will inevitably lead to generate construction documents incorrectly. In other words, that means the watermarks should not be embedded into the geometrical parameters of CAPD models.

To resolve this issue, we argue that the connection points are the best candidates for data embedding because of the following reasons. First, the topological relation among different plant components is described by their connection points. Second, each end face of plant components has one and only one associated connection point. And connection points are by definition the least likely to be removed among the types of data objects that exist in CAPD models. Moreover, the deletion of connection points will inevitably lead to generate construction documents incorrectly.

4.3. Mark plant components selecting principle

We describe how to select the mark plant components in this section. We initially set all plant components as *non-mark* components and traverse each pipeline of the whole model to get eligible plant components for watermark embedding according to the flow direction following the discipline below.

- One of the two joint plant components should be selected as a mark plant component.

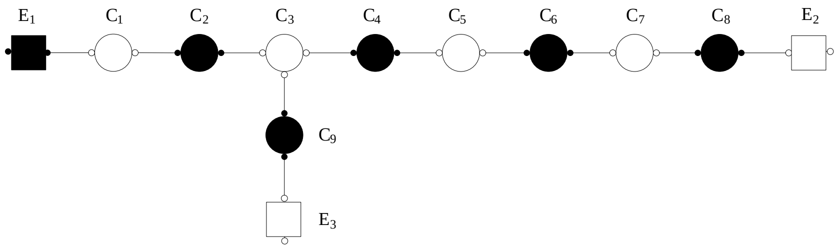
- The plant component chosen as a mark plant component must have no mark components among its 1-ring neighboring components. Once a plant component has been chosen as a mark component, its 1-ring neighboring components are no longer eligible.

This principle is quite simple, and Fig. 5 shows two different selection results for a same abstracted CAPD model. The union of the mark plant components and their 1-ring neighborhood covers all the plant components of the model. All the connection points of the selected mark plant components are set as mark connection points and then used for watermark embedding. Thus, it can be guaranteed that the mark plant components and their mark connection points are uniformly distributed in the models. Experimental data in Table 1 show that our principle selects around 50% of plant components and connection points as mark plant components and mark connection points respectively. And this can result in high locating accuracy which will be discussed in Section 5.2.

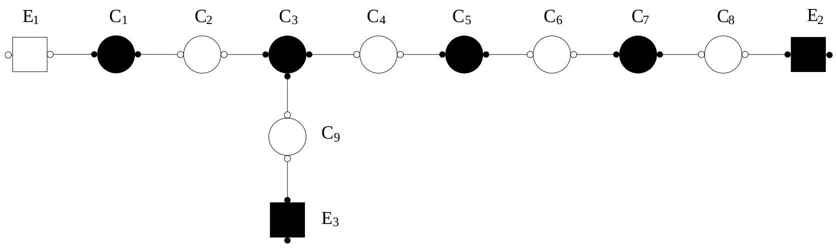
4.4. Content-based watermark generation

We generate a content-based watermark for each mark connection point taking some singular properties of its joint connection point and joint plant component into consideration using a deterministic chaotic map.

Let C_m be a selected mark plant component with n connection points. Plant component C_{m+1} is one of the joint plant components of C_m. $P_{m,i}$ is a mark connection point of C_m ($i \in [0, n-1]$). $P_{m+1,j}$ denotes a connection point of C_{m+1}. Assume that the joint connection point of $P_{m,i}$ is $P_{m+1,j}$. We denote the handle value of $P_{m+1,j}$ as $pHandle_{m+1,j}$. The handle value is



(a)



(b)

Figure 5: Two different selection results of mark plant components for a same simple abstracted CAPD model. The circular nodes represent pipe components while the rectangular nodes represent equipments. The black nodes are selected mark plant components while the white nodes are non-mark plant components.

involved in the construction of the watermarks, since each object in CAPD models has an unique handle value and it is not changed even if the object is modified[17].Let the total number of joint plant components of C_{m+1} be d_{m+1}. And it is also involved in the watermark generation. The chaotic map used in this paper for the watermark generation is a well-known logistic function shown as follows:

$$f(x_n) = x_{n+1} = ax_n(1 - x_n), \tag{1}$$

where a is a positive number that acts as a function seed, and x_n is a number between 0 and 1, representing the current value of the mapping in time with an initial value x_0 [18]. When $a > 3.5699456$, the sequence iterated with an initial value is chaotic. Different sequences will be generated with different initial values since the logistic function is extremely sensitive to initial conditions. The complicated but deterministic properties of the map make it ideally suited for watermark generation [19, 20, 21].

In order to generate the watermark $w_{m,i}$ for $P_{m,i}$, the handel value $pHandle_{m+1,j}$ of $P_{m+1,j}$ is first converted into a positive float number $h_{m,i}$ $(0 < h_{m,i} < 1)$ by

$$h_{m,i} = hash(pHandle_{m+1,j}), \tag{2}$$

where $hash()$ is a hash function. Then the logistic function, shown in Eq.1, is seeded with an initial starting value of $x_0 = h_{m,i}$, and iterated, and a final float value $f_{m,i}$ is calculated. After that we generate the watermark $w_{m,i}$ $(0 < w_{m,i} < 1)$ by

$$w_{m,i} = d_{m+1} \times f_{m,i}. \tag{3}$$

It should point out that there may be some mark connection points which have no joint connection points. In general, those selected mark plant components at the start or end position of a pipeline may have one or more mark connection points with no joint connection points. Take the mark plant component E_1 in Fig.5(a) for example, it has two mark connection points but only one of them has a joint connection point. We assume that $P_{m,i}$ has no joint connection point. And its handle is denoted as $pHandle_{m,i}$. Let the total number of joint plant components of C_m be d_m. In order to generate a watermark $w_{m,i}$ for $P_{m,i}$, then the positive number $h_{m,i}$ $(0 < h_{m,i} < 1)$ for

$P_{m,i}$ is calculated by

$$h_{m,i} = hash(pHandle_{m,i}). \tag{4}$$

And the watermark $w_{m,i}$ $(0 < w_{m,i} < 1)$ is finally generated by

$$w_{m,i} = d_m \times f_{m,i}. \tag{5}$$

4.5. The watermark embedding

In order to embed the watermark, we first calculate the Laplacian coordinate vector δ for each mark connection point. Then we alert the Laplacian length l, computing a new length $\hat{l}$ carrying the watermark. Finally, the new Laplacian vector $\hat{\delta}$ with length $\hat{l}$ is realized through a minimization process, and eventually the corresponding Cartesian coordinate is computed.

4.5.1. The computation of laplacian coordinates

For each connection point $P_{m,i}$ of C_m, we first define its neighboring connection points using the following terminology: the neighboring connection points $N(P_{m,i})$ is the set of all the connection points of the joint plant components of $P_{m,i}$. $P_{m,i}$ is conventionally represented using absolute Cartesian coordinates, denoted by $P_{m,i} = (x_{m,i}, y_{m,i}, z_{m,i})$. Fig. 6 shows an example of the neighboring connection points of a mark connection point. The mark component C_1 is a flange while its joint component C_2 is a valve. $P_{1,1}$ is a mark connection point of C_1. The neighboring connection points $N(P_{1,1}) = \{P_{2,0}, P_{2,1}, P_{2,2}\}$, where $P_{2,0}$, $P_{2,1}$, and $P_{2,2}$ are connection points of C_2.

Then, we define the *differential* or $\delta - coordinates$ of $P_{m,i}$ to be the difference between the absolute coordinates of $P_{m,i}$ and the center of mass of the neighboring connection points of $P_{m,i}$,

$$\delta_{m,i} = (x'_{m,i}, y'_{m,i}, z'_{m,i}) = P_{m,i} - \frac{1}{d_{m,i}} \sum_{P_{m,j} \in N(P_{m,i})} P_{m,j} \tag{6}$$

where $d_{m,i} = |N(P_{m,i})|$ is the number of neighboring connection points of $P_{m,i}$. $\delta_{m,i}$ is also called the Laplacian coordinate of $P_{m,i}$. The length of the Laplacian coordinate vector is then selected as the watermark carrier for topology protection

$$l_{m,i} = \|\delta_{m,i}\| = \sqrt{(x'_{m,i})^2 + (y'_{m,i})^2 + (z'_{m,i})^2}. \tag{7}$$

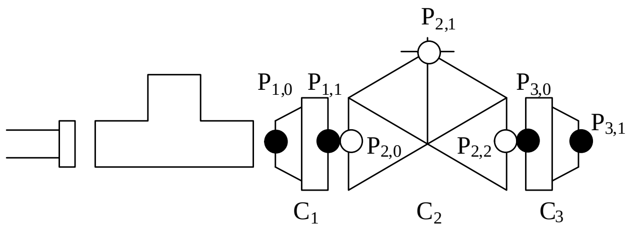


Figure 6: Illustration of the neighboring connection points of a mark connection point. C_1 is a flange and it is mark plant component while its joint component C_2 is a valve. The black point $P_{1,1}$ is a mark connection point of C_1 and its neighboring connection points are the white connection points $P_{2,0}$, $P_{2,1}$, and $P_{2,2}$ of C_2.

As mentioned in Section 4.4, there may be some mark connection points with no joint plant components. For these mark connection points, we define the connection points of the plant component they subject to as their neighboring connection points. For example, $P_{3,1}$ is a mark connection point of C_3 with no joint plant components in Fig.6. Its neighboring connection points $N(P_{3,1}) = \{P_{3,0}, P_{3,1}\}$, where $P_{3,0}$ and $P_{3,1}$ are all subject to C_3.

4.5.2. Quantization-based modulation

After the calculation of Laplacian length, we describe our QIM based watermark embedding method in this section.

We notice that the lengths of the Laplacian coordinates, unlike the Laplacian coordinates themselves, are invariant under both translation and rotation, but sensitive to uniform scaling. Therefore, two float factors S and f are predefined as the keys for watermark embedding and extraction. The initial value of S is set as the radius of the bounding sphere of the original model. They are used to calculate the quantization step Δ,

$$\Delta = \frac{R}{S} \times f, \tag{8}$$

where R is the radius of the bounding sphere of the model. It is obvious that the quantization step Δ has a ratio to the radius of the bounding sphere of the model. That is, a model with larger or smaller size will have larger or smaller quantization step. Thus we can achieve uniform scaling invariance.

Fig. 7 illustrates how a watermark $w_{m,i}$ is embedded in the length $l_{m,i}$. At first, we initialize the integer quotient $Q_{m,i}$ by $Q_{m,i} = \lfloor l_{m,i}/\Delta \rfloor$ with the quantization step size Δ, where $\lfloor \cdot \rfloor$ represents the floor function. The remainder $R_{m,i}$ is defined by $R_{m,i} = l_{m,i} - Q_{m,i} \times \Delta$. In general, $l_{m,i}$ cannot be completely divided by Δ. In that case, the remainder R_i is discarded by adjusting the the length $l_{m,i}$ such that $l^e_{m,i}$ can be divided by Δ

$$l^e_{m,i} = l_{m,i} - R_{m,i}. \tag{9}$$

Then we embed $w_{m,i}$ into $l_{m,i}$

$$\hat{l_{m,i}} = l^e_{m,i} + w_{m,i} \times \Delta \tag{10}$$

where $\hat{l_{m,i}}$ represent the length after embedding, $0 < w_{m,i} < 1$.

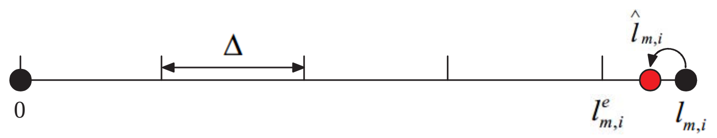


Figure 7: With the quantization step Δ, a watermark $w_{m,i}$ can be embedded by modifying the length $l_{m,i}$ to $\hat{l}_{m,i}$.

4.5.3. Distortion minimization

We discuss the calculation of the new Laplacian coordinate $\hat{\delta}_{m,i}$ after the computation of the embedded length $\hat{l}_{m,i}$ in this section. This is an undetermined problem and we solve it by minimizing the distance between the Laplacian coordinates before and after watermarking. We minimize the distance for each connection point by

$$(x'_{m,i} - \hat{x}'_{m,i})^2 + (y'_{m,i} - \hat{y}'_{m,i})^2 + (z'_{m,i} - \hat{z}'_{m,i})^2 = \|\delta_{m,i} - \hat{\delta}_{m,i}\|^2 \tag{11}$$

subject to

$$(\hat{x}'_{m,i})^2 + (\hat{y}'_{m,i})^2 + (\hat{z}'_{m,i})^2 = (\hat{l}_{m,i})^2 \tag{12}$$

This minimization problem is equivalent to finding a point $(\hat{x}'_{m,i}, \hat{y}'_{m,i}, \hat{z}'_{m,i})$ which is closest to the given point $(x'_{m,i}, y'_{m,i}, z'_{m,i})$ on a sphere C of radius $\hat{l}_{m,i}$ centered at the origin. We can take the point $(\hat{x}'_{m,i}, \hat{y}'_{m,i}, \hat{z}'_{m,i})$ as the projection of $(x'_{m,i}, y'_{m,i}, z'_{m,i})$ on C. As C is centered at the origin, the projection of $(x'_{m,i}, y'_{m,i}, z'_{m,i})$ on it is given by

$$\begin{cases} \hat{x}'_{m,i} = \dfrac{x'_{m,i}\hat{l}_{m,i}}{\sqrt{(x'_{m,i})^2 + (y'_{m,i})^2 + (z'_{m,i})^2}} \\ \hat{y}'_{m,i} = \dfrac{y'_{m,i}\hat{l}_{m,i}}{\sqrt{(x'_{m,i})^2 + (y'_{m,i})^2 + (z'_{m,i})^2}} \\ \hat{z}'_{m,i} = \dfrac{z'_{m,i}\hat{l}_{m,i}}{\sqrt{(x'_{m,i})^2 + (y'_{m,i})^2 + (z'_{m,i})^2}} \end{cases} \tag{13}$$

Finally, the Cartesian coordinate of the watermarked connection point can be computed from its Laplacian coordinate $(\hat{x}'_{m,i}, \hat{y}'_{m,i}, \hat{z}'_{m,i})$ according to Eq. 6.

In our scheme, the distortion induced by watermark embedding depends on the quantization step Δ. From the Eq. 8, we can see that the larger the key value f, the larger the induced distortion, since the key value S is set to the radius R of the bounding sphere of the original model initially. Therefore, the maximum distortion from each mark connection point can be controlled by setting the key value f according to the precision requirement.

4.6. The watermark extraction and tamper detection

In the watermark extraction stage, we initially set all plant components and connection points as mark plant components and mark connection points respectively. S and f are the keys for malicious-change detection. And they are employed to calculate the quantization step Δ with Eq. 8. We first check and find out all of the mark plant components of the model. Then we apply the *mark plant components selecting principle* to detect and locate the tampered regions.

For each plant component C_m with n connection points, we check each of its connection points $P_{m,i}$ $(0 \leq i \leq n-1)$ to see whether it is a mark connection point or not. We first compute the length $l_{m,i}$ of the Laplacian coordinate vector of $P_{m,i}$ with Eq.6 and Eq. 7. Then we extract the embedded watermark with the quantization step Δ by

$$w'_{m,i} = \frac{(l_{m,i} - \lfloor \frac{l_{m,i}}{\Delta} \rfloor \times \Delta)}{\Delta} \tag{14}$$

In order to see whether $P_{m,i}$ is a mark connection point, the content-based watermark $w_{m,i}$ for $P_{m,i}$ is generated according to the *content-based watermark generation method* described in Section 4.4. Thus, $w_{m,i}$ and $w'_{m,i}$ should satisfy $w_{m,i} = w'_{m,i}$ if $P_{m,i}$ is a mark connection point. We label C_m as a mark plant component if it has at least one mark connection point. Otherwise, C_m is set to be a non-mark plant component.

After the labeling of mark connection points and mark plant components, we next detect and locate the tampered plant components and connection points applying the *mark plant components selecting principle.* For each mark plant component, we set it as an unmodified plant component only if all of its connection points are mark connection points. Otherwise, we label its non-mark connection points and their joint plant components as suspicious regions. For each pipeline of the model, we traverse its plant components according to its flow direction and check if the labeled mark plant components satisfy the *mark plant components selecting principle.* We set those plant components which do not meet the *mark plant components selecting principle* as tampered plant components.

(a) (b) (c)

Figure 8: Three CAPD models used for experiments. (a)Carton board plant; (b)Hydrogenation plant; (c)Styrene plant.

5. Experimental results and discussion

To validate the feasibility of our topology verification algorithm, we first give some experimental results and then discuss its performance later in this section.

5.1. Experimental results

We evaluated the proposed semi-fragile watermarking scheme on a set of 3D CAPD models with various unauthorized attacks and three of them are shown in Fig. 8. Table 1 lists the detailed information about the three models. The following parameter settings are used in our experiments. The logistic function used for the watermark generation is seeded with a value $a = 4$ for 5000 iterations. The key value f is set to 10^{-3} according to the model precision and S is equal to the radius of the bounding sphere of each model listed in Table 1.

From Table 1 we can find that our approach selects around 50% of the plant components as mark plant components. And the watermark bits are embedded into nearly 50% of the connection points.

Table 1: Lists of three CAPD models used in our experiments and their detail information including plant components(PCs), connection points(CPs), mark plant components(MPCs), mark connection points(MCPs) and radius.

Model	PCs	CPs	MPCs	MCPs	Radius(m)
Carton board	6810	13964	3365	7002	118.890
Hydrogenation	15570	32624	8145	16556	104.380
Styrene	18912	38198	9652	19484	86.321

5.1.1. Tamper detection and localization evaluation

Fig. 9(a) and Fig. 9 (c) show a close view of part of the original hydrogenation plant model rendered in solid and wireframe mode respectively. The hydrogenation plant model has 15570 plant components and about 52.3% of them are selected as mark plant components. Fig. 9(b) and Fig. 9(d) are the same view of part of the watermarked model rendered in solid and wireframe mode respectively, which are visually identical with the original model.

Fig. 10 illustrates that our scheme accurately detects and locates several kinds of attacks simultaneously on a hydrogenation plant model. Fig. 10(a), Fig. 10(c), Fig. 10(e) and Fig. 10(g) show a close view of the regions of the watermarked hydrogenation plant model before being attacked illegally by joint components modification and joint ends modification respectively. The regions labeled A, B, C, and D denote the regions of joint components addition, joint components deletion, disconnecting the two joint ends geometrically and changing the topology relation between two joint ends logically, respectively. Our scheme locates these changed regions by setting all detected suspicious plant components as suspicious regions. Fig. 10(b), Fig. 10(d), Fig. 10(f) and Fig. 10(h) illustrate the located suspicious plant components

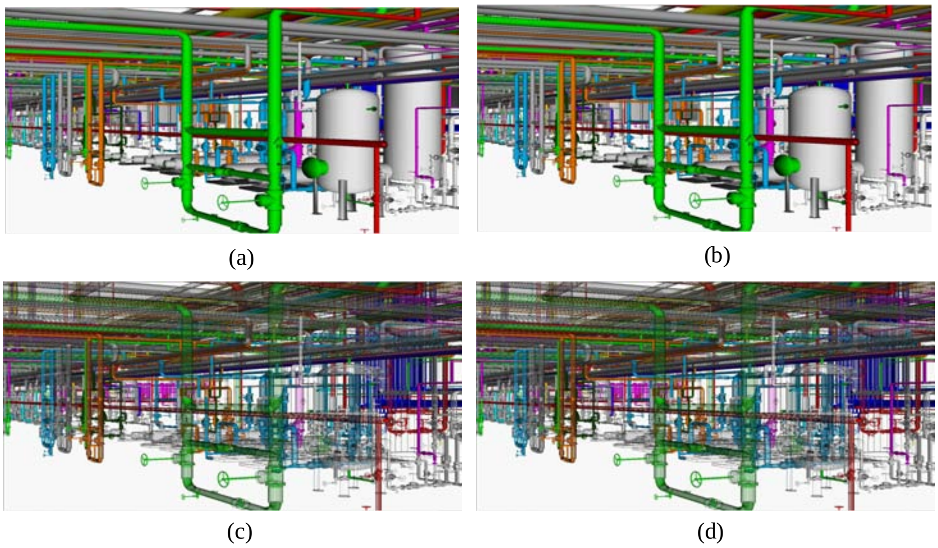


Figure 9: One example of semi-fragile watermarking. (a)(c) A close view of part of the original model rendered in solid and wireframe mode respectively. (b)(d) A close view of part of the watermarked model rendered in solid and wireframe mode respectively.

in red. From Fig. 10(b), Fig. 10(d), Fig. 10(f) and Fig. 10(h) we can find that the regions in red are exactly where the tampering operations happen. The experimental results verify the accuracy of our locating procedure.

5.1.2. Robustness evaluation

We evaluated the robustness against various operations provided by CAPD systems that can be considered to be non-malicious attacks on the design model. These non-malicious attacks include rotating, uniform scaling, transformation and LOD. The robustness of our semi-fragile watermarking scheme is evaluated in terms of the BER (bit error rate) of the extracted watermark bit sequence, as well as the correlation coefficient $Corr$ between the extracted binary sequence $\{w_i^e\}$ and the originally embedded one $\{w_i^o\}$ as given by the

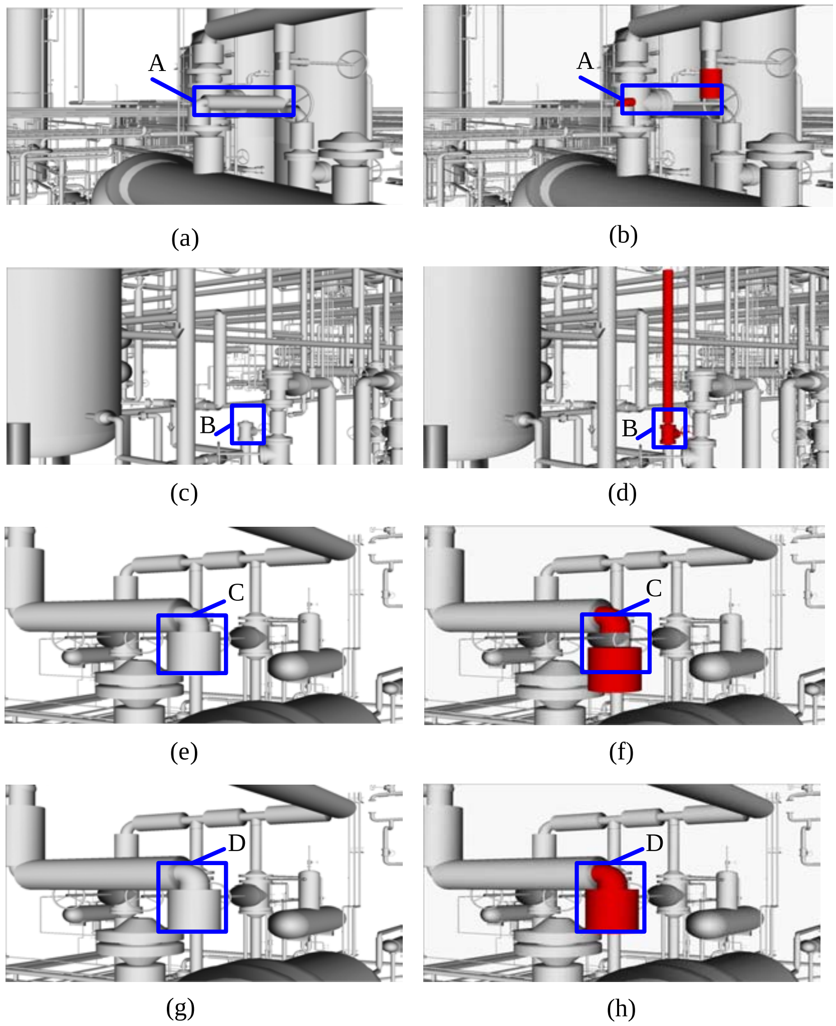


Figure 10: The proposed scheme works on a hydrogenation plant model.(a)(c)(e)(g) The regions before being attacked. Label A denotes the regions of joint components deletion. Label B denotes the regions of joint components addition. Label C denotes the regions of disconnecting the two joint ends geometrically, and label D denotes the region of changing the topology relation between two joint ends logically. (b)(d)(f)(h) Our scheme accurately locates these attacks visually.

Table 2: N_m/N_c of the three CAPD models after various non-malicious attacks.

Attacks	Carton board	Hydrogenation	Styrene
RST	0	0	0
LOD			
(80% triangles)	0	0	0
(60% triangles)	0	0	0
(40% triangles)	0	0	0

following equation [22]:

$$Corr = \frac{\sum_{i=0}^{n-1}(w_i^e - \overline{w^s})(w_i^o - \overline{w^o})}{\sqrt{\sum_{i=0}^{n-1}(w_i^e - \overline{w^s})^2} \times \sqrt{\sum_{i=0}^{n-1}(w_i^o - \overline{w^o})^2}}, \tag{15}$$

where $\overline{w^e}$ and $\overline{w^o}$ are, respectively, the averages of the watermark bit sequence $\{w_i^e\}$ and $\{w_i^o\}$.

For each plant component, if the values of BER and $Corr$ are, respectively, 0 and 1, then we can set the plant component as untampered. Otherwise the plant component is detected as tampered. Let N_c be the total number of plant components in a model and N_m be the number of plant components detected as tampered. Table 2 presents the N_m/N_c of the three models after various non-malicious attacks. And we can find that our scheme is robust against these non-malicious operations.

5.1.3. Imperceptibility evaluation

For evaluating the subject imperceptibility, we compare the original hydrogenation plant model and the watermarked hydrogenation plant model rendered in solid and wireframe mode respectively. Fig. 9 shows a close view of part of the original and watermarked hydrogenation plant model. And we can see the imperceptibility of the watermarked connection points.

In order to measure the objective distortion of the watermarked CAPD models induced by watermarking, we use the Metro [23] in terms of maximum root mean square error(MRMS) for plant components and PSNR (peak signal-to-noise ratio) [9] for connection points respectively.

$$PSNR = 10 \lg \frac{MAX^2}{MSE}, \tag{16}$$

where

$$MAX = max\|P_i - o\|, i \in [0, N-1],$$
$$MSE = \frac{1}{N} \sum_{i=0}^{N-1} \|P_i - P_i'\|,$$

P_i and P_i' are the corresponding connection points in the original and watermarked model respectively, o is the geometric center of the model, N is the number of connection points, $\|P_i - P_i'\|$ is the Euclidean distance between these two connection points. Table 3 lists the MRMS values of plant components and the PSNR values of connection points.

From the Table 3, we can see that the MRMS values are all 0, since our scheme prefers the connection points, which are integral parts of CAPD models, instead of the geometric parameters of plant components themselves as watermark carriers. That means we need not modify the geometric parameters of plant components. Therefore our scheme has no influence on the geometry shape of CAPD models.

Although the connection points, compared with the large scale plant components, are nearly not seen by viewers because of their small size and little

Table 3: The MRMS values of plant components and PSNR values of connection points between the original and watermarked models.

Model	MRMS	PSNR(dB)
Carton board	0	68.56
Hydrogenation	0	81.03
Styrene	0	79.64

contribution to the final scene even rendered in wireframe mode, we still give the PSNR values of connection points here. The impact of watermark embedding on connection points could be tuned by the quantization step size Δ. According to our *watermark embedding method* described in Section 4.5, our scheme just slightly adjust the positions of mark connection points. And the topology relation will not be alerted too. As a consequence, our scheme will have no influence on the design and automatic generation of various construction documents. Thus, our scheme is functionally imperceptible.

5.2. Discussion on tamper detection and localization

We analysis the performance of our scheme on detecting and locating the tampered regions on the model from the following two aspects: attacks against plant components and attacks against joint ends, both of which are common operations in practical design process. Components attacks mainly include adding, deleting and replacing plant components. While joint ends attacks mainly include separating the two joint ends geometrically, disconnecting the two joint ends logically and replacing the joint end.

5.2.1. Components modification

- **Components addition**. Without loss of generality, there mainly exist three situations when adding plant components into the model which is shown in Fig. 11. Plant components are represented by rectangular nodes. The black nodes are mark plant components and their connection points are watermarked. The white nodes are non-mark plant components. The plant components to be added are represented by red nodes.

  First, Fig. 11(a) shows that a new plant component A_1 is added and it is connected with an existing non-mark plant component C_1. This kind of attacks modifies the topological relation of C_1. And it changes the total number of joint plant components of C_1 from one to two. During the watermark extraction stage, $P_{2,1}$ is labeled as a mark connection point. Then C_2 is set as a mark plant component. However, the watermark for $P_{2,0}$ generated according to the *content-based watermark generation method* is different from the extracted original embedded one. Thus the topological modification of C_1, as well as $P_{2,0}$, is detected.

  Second, Fig. 11(b) shows that a new plant component A_1 is added and it is connected with an existing mark plant component C_2. Thus A_1 becomes the joint plant component of $P_{2,1}$. During the watermark extraction stage, $P_{2,0}$ is labeled as a mark connection point. Then C_2 is set as a mark plant component. But the watermark for $P_{2,1}$ generated according to the *content-based watermark generation method* is different from the extracted original embedded one. Therefore the topological modification of C_2, as well as $P_{2,1}$, is detected.

Third, Fig. 11(c) shows that two new plant components A_1 and A_2 are added, and A_1 is inserted between the non-mark plant component C_1 and the mark plant component C_2 while A_2 is inserted between the non-mark plant component C_3 and the mark plant component C_2. These attacks modify the topological relation of C_1, C_2 and C_3. During the watermark extraction stage, all the connection points are labeled as non-mark connection points according to the *watermark extraction and tamper detection method*. And then all the plant components are set as non-mark plant components. As a result, all the plant components are labeled as tampered plant components since they do not satisfy the *mark plant components selecting principle*. And Subsequently the topological modification of C_1 , C_2 and C_3 are detected and located accurately.

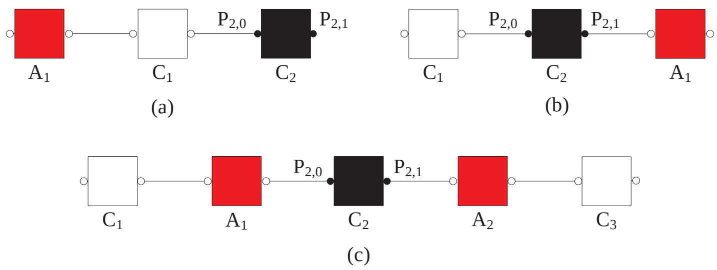


Figure 11: Illustration of detecting and localizing components addition. The black nodes are mark plant components. The white nodes are non-mark plant components. The red nodes represent the added plant components.

- **Components deletion**. These attacks modify the topological relation of the model. There are two main situations when deleting plant components from the model shown in Fig.12: mark plant components deletion and non-mark plant components deletion.

In Fig.12(a), a mark plant component D_1 is deleted. Thus the total number of joint plant components of C_2 changes from two to one. C_1 is set as a mark plant component since $P_{1,0}$ is labeled as a mark connection point during the watermark verification stage. The generated watermark for $P_{1,1}$ is different from the extracted original one according to the *content-based watermark generation method*. As a result, the topological modification of C_2 is detected and located accurately.

In Fig.12(b), a non-mark plant component D_1 is deleted. Thus no joint plant component is assigned to the mark connection point $P_{2,1}$. C_2 is set as a mark plant component because $P_{2,0}$ is labeled as a mark connection point during the watermark verification stage. However, the generated watermark for $P_{2,1}$ is different from the extracted original one according to the *content-based watermark generation method*. As a result, the topological modification of C_2, as well as $P_{2,1}$, induced by components deletion is detected and located accurately.

- **Components replacing**. Two main situations arise when replacing plant components from the model: replacing mark plant components and replacing non-mark plant components.

  In Fig.13(a), a mark plant component C_3 is replaced with a plant component R. During the watermarking extraction stage, C_1 is labeled as a mark plant component. However, the extracted watermarks from R inevitably do not match the original embedded ones since the coordi-

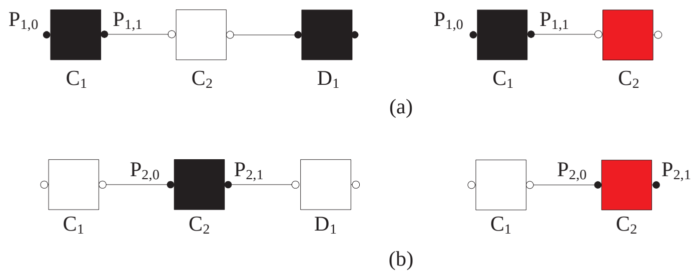


Figure 12: Illustration of detecting and localizing component deletion. The black nodes are mark plant components. The white nodes are non-mark plant components. The suspicious plant components are represented by red nodes.

nates of the connection points of R are different from the coordinates of the watermarked connection points of C_3. And thus it is labeled as a non-mark plant component. As a result, R and C_2 are set as suspicious plant components since both of them are non-mark plant components applying *the mark plant components selecting principle*.

In Fig.13(a), a non-mark plant component C_3 is replaced with a plant component R. During the watermarking extraction stage, C_2 is labeled as a mark plant component since $P_{2,0}$ is set as a mark connection point. The generated watermark for $P_{2,1}$ is different from the extracted original one because the handle value of R is different from the handle value of C_3. Hence the modification of the topological relation between $P_{2,1}$ and R induced by components replacing is detected and located accurately.

Note that we just take the same kind of plant components into consideration, since different kind of plant components may not only induce different handle values and coordinates but also induce different number of connection points. These attacks can be detected relatively easily.

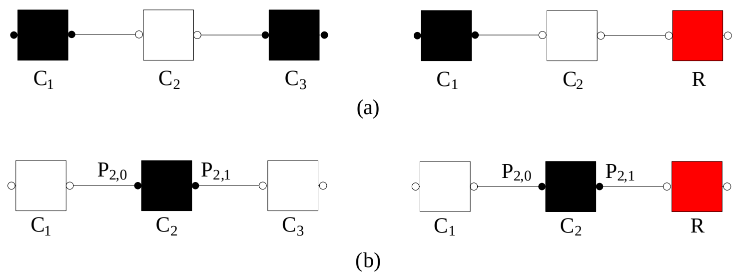


Figure 13: Illustration of detecting and localizing component replacing. The black nodes are mark plant components. The white nodes are non-mark plant components. The red nodes represent the plant components after replacing.

5.2.2. Joint ends modification

We discuss the attacks on the two joint connection ends in this section. These two attacked connection ends subject to two different joint plant components. And one should be a mark connection point while the other should be a non-mark connection point according to the *mark plant components selecting principle.*

- **Disconnect the two joint ends geometrically**. This kind of attacks separates one connection end from the other connection end geometrically while keeps their topology relation logically. During the watermark extraction stage, the generated watermark for the attacked mark connection point is identical to the original embedded one since non topological modification is induced.However, the Laplacian coordinate vector of the attacked mark connection point is different from the original one because of the geometrical modification of the two attacked joint plant components. Therefore, the extracted watermark is not match the original embedded one. As a result, the attacked connection end and its joint plant component are detected.

- **Change the topology relation between two joint ends logically**. This kind of topology attacks changes the topological relation between the two joint ends logically. Thus the joint connection point of the mark connection point is alerted. This modification leads to the difference between the embedded watermark and the calculated watermark during the watermark extraction stage. Consequently, the attacked two joint ends are detected.

5.3. Discussion on robustness against non-malicious attacks

A good semi-fragile watermarking scheme should be invariant to translation, rotation, uniform scaling and LOD operations. These operations do not change the integrity of the original model and should not be regarded as malicious attacks.

5.3.1. Robustness against similarity transformation

These similarity transforming operations modify the coordinates of the model. Our scheme prefers the lengths of the Laplacian coordinates to the Laplacian coordinates themselves of the mark connection points as watermark carriers. Thus it is invariant under both translation and rotation.In order to resist the uniform scaling operation, the radius of the bounding sphere of the model is involved in the *quantization-based modulation* stage for watermark embedding. That is, a model with larger or smaller size will have larger or smaller quantization steps. Thus we can achieve uniform scaling invariance.

5.3.2. Robustness against level-of-detail

For the past several years, the widespread use of collaborative CAPD systems and the reuse of existing CAPD data in new designs have created a data explosion in many application areas. And this has resulted in large databases of complex CAPD models. As the complexity of CAPD models increases, the enormous size of these CAD data sets poses a number of challenges in terms of interactive display and manipulation. Thus, CAPD systems must employ methods for filtering out as efficiently as possible the data that isn't contributing to a particular image. LOD is a key technology to reduce the model complexity and improve the rendering performance for large scale complex CAPD models. A LOD model is a compact description of multiple representations of a single shape and is the key element for providing the necessary degrees of freedom to achieve runtime adaptivity. However, connection points and topological relation among plant components will not be influenced by LOD since it can only change the details of entity surfaces. Therefore, the 1-ring neighboring points set of each mark connection point will not be affected. Subsequently it will not change the centroid of the neighborhood of mark points. As a result, our scheme is robust against LOD.

6. Conclusion

This paper presents digital watermarking as a possible topology authentication tool to provide security to 3D CAPD models. Both of the topological relation and singular attributes of plant components are taken into consideration for the watermark generation and embedding. The watermarks are embedded into the coordinates of mark connection points by adjusting the lengths of their Laplacian coordinate vectors. Theoretical analysis and experimental results show that our semi-fragile scheme has a strong ability to detect and locate malicious attacks which are common operations in practical design process. Meanwhile, our scheme can exactly preserve the geometric shape of plant components and hence has no effect on the automatic generation of construction documents.

[1] Burdorf A, Kampczyk B, Lederhose M, Schmidt-Traub H. Capd-computer-aided plant design. Computers and Chemical Engineering 2004;28(1-2):73–81.

[2] Wang K, Lavoué G, Denis F, Baskurt A. A comprehensive survey on three-dimensional mesh watermarking. IEEE Transactions on Multimedia 2008;10(8):1513–27.

[3] Guirardello R, Swaney R. Optimization of process plant layout with pipe routing. Computers & Chemical Engineering 2005;30(1):99–114.

[4] Georgiadisa M, Macchietto S. Layout of process plants: A novel approach. Computers & Chemical Engineering 1997;21(Supplement 1):S337–42.

[5] Ohbuchi R, Masuda H, Aono M. Watermarking three-dimensional polygonal models. In: Proceedings of the ACM Multimedin. Seattle, USA; 1997, p. 261–72.

[6] Park HK, Lee SH, Kwon KR. Blind watermarking for copyright protection of 3d cad drawing. In: Proceedings of the 8th International Con-

ference on Advanced Communication Technology. Gangwon-Do, Korea; 2006, p. 253–6.

[7] Kwon K, Lee S, Lee E, Kwon S. Watermarking for 3d cad drawings based on three components. Lecture Notes in Computer Science 2006;4109:217–25.

[8] Kwon K, Chang H, Jung GS, Moon K, Lee S. 3d cad drawing watermarking based on three components. In: Proceedings of the IEEE International Conference on Image Processing. Atlanta, GA, USA; 2006, p. 1385–8.

[9] Lee SH, Kwon KR. Cad drawing watermarking scheme. Digital Signal Processing 2010;20(5):1379–99.

[10] Ohbuchi R, Masuda H, Aono M. A shape-preserving data embedding algorithm for nurbs curves and surfaces. In: Proceedings of the Computer Graphics International. Alberta, Canada; 1999, p. 180–7.

[11] Lee JJ, Cho NI, Lee SU. Watermarking algorithms for 3d nurbs graphic data. EURASIP Journal on Applied Signal Processing 2004;2004(14):2142–52.

[12] Fornaro C, Sanna A. Public key watermarking for authentication of csg models. Computer-Aided Design 2000;32(12):727–35.

[13] Weng B, Ri J, Yao Z, Yang S, Feng X, Pan. Z. Watermarking t-spline surfaces. In: Proceedings of the 11th IEEE international conference on communication technology proceedings. HangZhou, China; 2008, p. 773–6.

[14] Cheung YM, Wu HT. A sequential quantization strategy for data embedding and integrity verification. IEEE Transactions on Circuits and Systems for Video Technology 2007;17(8):1007–16.

[15] Reuter M, Wolter FE, Peinecke N. Laplace-beltrami spectra as 'shape-dna' of surfaces and solids. Computer-Aided Design 2006;38(4):342–66.

[16] Dow M. Integration of calculation models and cad systems in building services design. Computer-Aided Design 1987;19(5):226–32.

[17] Peng F, Guo RS, Li CT, Long M. A semi-fragile watermarking algorithm for authenticating 2d cad engineering graphics based on log-polar transformation. Computer-Aided Design 2010;42(12):1207–16.

[18] Marek M, Schreiber I. Chaotic behaviour of deterministic dissipative. Cambridge University Press; 1991.

[19] Zhang L, Liao X, Wang X. An image encryption approach based on chaotic maps. Chaos, Solitons and Fractals 2005;24(3):759–65.

[20] Mooney A, Keating J, Heffernan D. A detailed study of the generation of optically detectable watermarks using the logistic map. Chaos, Solitons and Fractals 2006;30(5):1088–97.

[21] Mooney A, Keating J, Heffernan D. Performance analysis of chaotic and white watermarks in the presence of common watermark attacks. Chaos, Solitons and Fractals 2009;42(1):560–70.

[22] Wang K, Lavoue G, Denis F, Baskurt A. Robust and blind mesh

watermarking based on volume moments. Computers & Graphics 2011;35(1):1–19.

[23] Cignoni P, Rocchini C, Scopigno R. Metro: measuring error on simplified surfaces. Computer Graphics Forum 1998;17(2):167–74.