Watermarking 3D CAPD models for topology verification

Zhiyong Su[a,*], Weiqing Li[b], Jianshou Kong[a], Yuewei Dai[a], Weiqing Tang[c,d]

[a]*School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China*
[b]*School of Computer Science, Nanjing University of Science and Technology, Nanjing 210094, China*
[c]*Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China*
[d]*Beijing Zhongke Fulong Computer Technology Co., Ltd, Beijing 100085, China*

Abstract

CAPD (Computer-Aided Plant Design) mainly focuses on providing an effective and efficient platform for designers to concentrate on the topology of tremendous number of plant components under complex constraints rather than just shapes. However, in the literature, none of existing watermarking schemes for CAD models have mentioned the problem of topology protection for CAPD models yet. In this paper, a semi-fragile watermarking algorithm for topology verification of CAPD models is presented. We first discuss the problem of topology authentication of CAPD models. Then a subset of the model's connection points are selected as mark points according to the mark point selecting principle. Watermarks are embedded in mark points to keep them in a predefined relationship with neighboring connection points so that any changes will ruin the relationship between the marked connection

*Corresponding author. Tel.: +8625 84315467; fax: +8625 84317332.
Email address: suzhiyong@njust.edu.cn (Zhiyong Su)

points and neighboring connection points. To the best of our knowledge, our algorithm is the first semi-fragile and blind scheme that can authenticate the topology of CAPD models. Experimental results show 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 simplification.
Keywords: Semi-fragile watermarking, Watermarking, CAPD, Process plant

1. Introduction

Computer-Aided Plant Design (CAPD), designed specifically for process plant engineers and designers, is an automatic solution provided for helping increase productivity, accuracy, and collaboration to meet the challenges of complex plant design projects. With the faster and higher demands of high quality products at low prices in a timely manner, companies are increasingly focusing on a collaborative design approach with each other to survive in a competitive global environment. And the design, engineering and construction of process plants involves multidisciplinary team effort. As a result, CAPD models may be copied and distributed frequently with an increase during the collaboration. Therefore, the protection of copyright and integrity is critical to companies when sharing models with its collaborators. Digital watermarking has been considered as an efficient solution to solve this problem, and thus has been paid significant attention in recent years [1].

A CAPD model can be completely described by three kinds of information: the geometry information describes the shape and 3D positions (co-

ordinates) of all plant components, the topology information provides the adjacency relations between different plant components, while the engineering information refers to design constraints, engineering disciplines and so on. Even though the graphical representation of the results may look like detailed engineering plant design, it has to be kept in mind that the aim of CAPD systems is only to optimize the plant layout[2]. The objective in plant layout design is 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 copyright and integrity protecting for CAPD models.

However, the watermarking techniques proposed for CAD models in the literature mainly focus on addressing the problem of watermarking geometry information. And among them the topology protecting problem for CAPD models, to our best knowledge, has not been mentioned in any reported literature. Hence, the issue of defining watermarking for topology information of CAPD models is still an unsolved problem. To address the issue of verifying the integrity of the topology information of CAPD models, we propose a semi-fragile watermarking scheme in this paper. We prefer embedding watermarks into a subset of a models 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 Section2; 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

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 CAD models especially for CAPD models. In this section we review some related work about watermarking 3D and 2D CAD models.

2.1. Watermarking for 3D CAD models

Watermarking schemes for 3D CAD models mainly target NURBS curves, subdivision surfaces, CSG models and CAD-based drawings represented by various geometric objects in some layers such as LINEs, ARCs, POLYGONs and 3DFACEs.

Ohbuchi et al.[6] present a watermarking scheme for 3D NURBS curves using reparameterization. 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[7]. A fragile watermarking schemes for authenticating CSG models was proposed by Fornaro and Sanna [8]. 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. present a method for watermarking T-spline curves and

surfaces by using knot insertion[9]. 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[10]. Reuter et al.[11] introduce 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. 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. A digital watermarking technique for 3D design drawing was proposed by Park et al. [12]. The scheme uses LINEs and 3DF ACEs based on vertex in CAD system to prevent infringement of copyright from unlawfulness reproductions and distribution. Kwon et al. also propose a watermarking scheme for 3D CAD drawings[13, 14]. 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. [15]. The proposed scheme embeds the watermark into the geometric distribution of POLYLINE, 3DFACE, and ARC objects in the main layers.

2.2. Watermarking for 2D CAD drawings

A watermarking scheme for a 2D architectural design drawing using LINEs and ARCs based on vertex was presented in [16]. It embeds the watermark into the LINE's length and ARC's angle information. Kwon et al.[17] propose a digital watermarking for 2D CAD drawings. The watermark is embedded by using a self-adaptive algorithm related to the length of LINEs, angles of ARCs, and radii of CIRCLEs. It is robust against various attacks, such as geometrical transformation. A semi-fragile watermarking algorithm for authenticating 2D CAD engineering graphics based on log-polar transformation was proposed in [18]. The vertices are divided into groups, and the vertices for the watermark in each group are mapped to the log-polar coordinate system. Then the watermark is embedded in the mantissa of the real-valued log-polar coordinates via bit substitution. It is robust against incidental global operations such as rotation, translation, and scaling, and can detect and locate malicious attacks such as entity modification and entity addition/deletion. Peng et al. also propose a reversible watermarking scheme for 2D CAD engineering graphics based on improved difference expansion[19]. The watermark is embedded into the scale factor of the relative coordinates by using improved difference expansion technology.

3. CAPD Models

Process plants are complex facilities mainly consisting of equipments and pipelines (including pipes and piping components). Fig. 1 shows an example of a solid model of hydrogenation plant. We first introduce the geometry and topology modeling of CAPD models in this section. Then we give a

brief description of the topology verification problem.

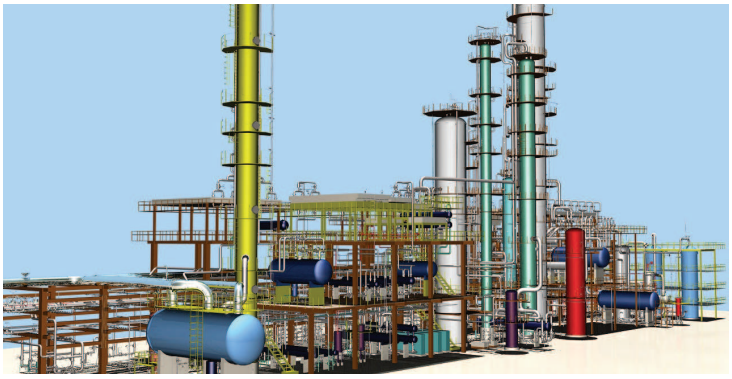


Figure 1: An example of a CAPD model(hydrogenation plant).

3.1. Parametric geometry modeling of plant components

CAPD systems often involve more than just shapes. They mainly focus on providing an effective and efficient platform to concentrate on the layout of tremendous number of plant components (equipments, pipes and piping components) under complex constraints. Plant components are normally modeled by basic entities, such as box, cylinder, prism, sphere and so on. Fig. 2 shows some entities used in CAPD systems.

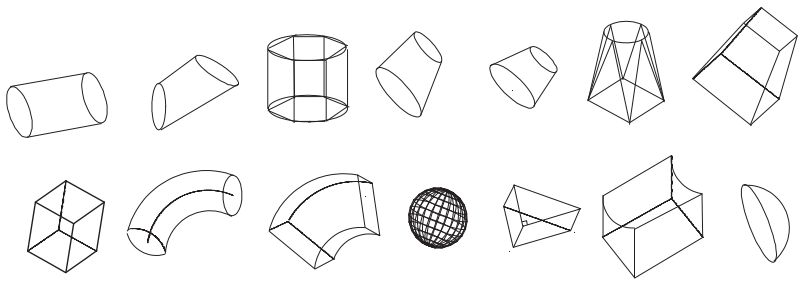


Figure 2: Some entities used in CAPD systems.

In order to support the automatic generation of isometrics, orthographics, and other construction documents, which directly exchanged with the 3D model, CAPD models are designed by using a parametric geometry modeling method. Only their geometric parameters and types rather than meshes are stored in the file. An example of a cone entity is shown in Fig. 3. 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.

ID	**A81**
Type	Cone
Geometric parameters	H: 10.0
	D1: 24.0
	D2: 12.0
	P1: 0.0,0.0,0.0
	P2: 0.0,10.0,0.0

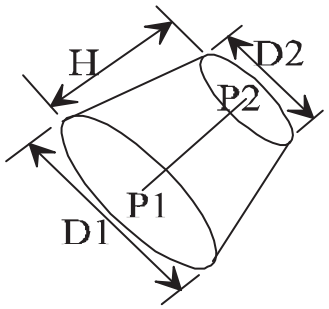


Figure 3: An example of a cone entity.

3.2. Topology modeling of plant components

Plant layout is an important part of plant design. It is concerned with the most economical spatial arrangement of process vessels and equipment and their interconnections that satisfies construction, operation, maintenance, and safety requirements. This is an important aspect in the design of process plants. A good layout will ensure that the plant functions correctly and will provide an economically acceptable balance between the many, often conflicting, design constraints which are derived from safety, construction, maintenance and operational considerations[4].

The layout poses significant limitations on the type, size and location of plant components. Positions of plant components can be simply described 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 between two plant components, but it should also describe their corresponding interconnection ends. Only the two ends of different plant components which satisfy pipe diameter, end type, pressure rating, and flow direction requirements can then be connected.

End connection can be mainly represented in two formats: connection points [20] 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 the most widely used and effective representation for topology modeling.

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. 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. Fig. 4 shows the structure of connection points.

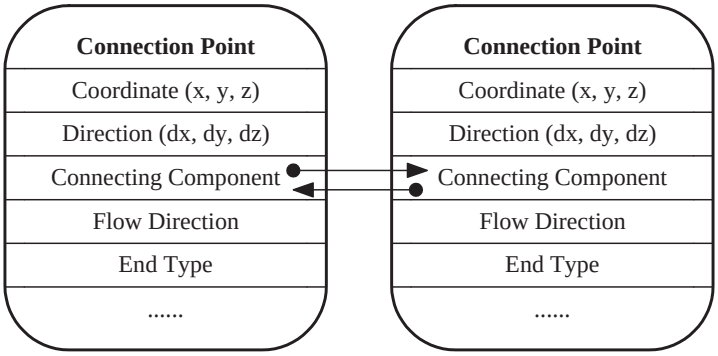
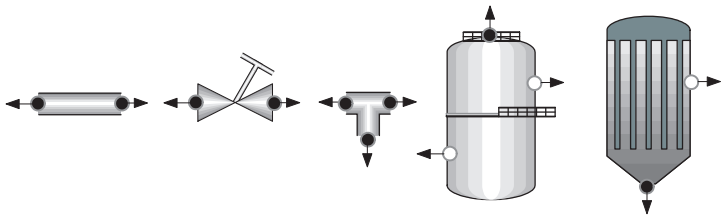
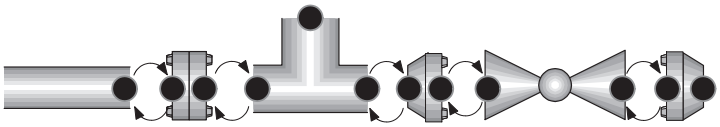


Figure 4: The structure of connection points.



(a)



(b)

Figure 5: 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. (a) Connection points of some selected plant components. (b) Connection points of a simple pipeline.

3.3. Problem statement

The problem of topology authentication of CAPD models consists of two aspects. One is joint plant components authentication. The other is 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 two joint plant components are modified or not. That is to say that, for each plant component, the problem of topology authentication targets to verify not only the joint components, but also the exact joint ends, since a plant component usually have more than one joint ends.

4. Proposed scheme

4.1. Embedding targets

We argue that, for topology verification, connection points are the best candidate for data embedding for the following reasons.

- First, the topology relation among different plant components is described by their connection points.
- Second, each end 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.

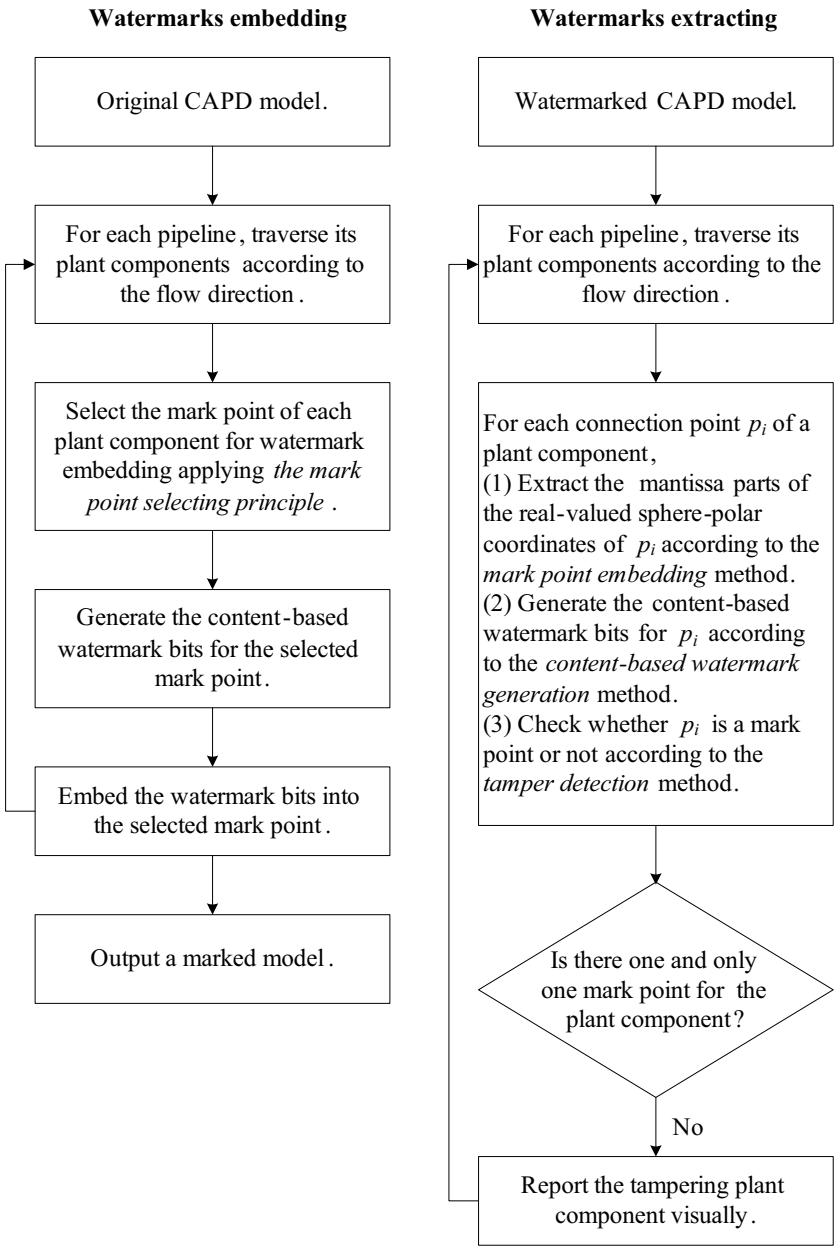


Figure 6: Overview of the proposed semi-fragile watermarking scheme.

- Third, connection points are within 3D CAPD models. There are two main kinds of displaying mode for CAPD systems: solid mode and wireframe mode. There is no doubt that they are fully invisible to viewers in the solid mode. For the wireframe mode, one could also hardly perceive the slightly modification of connection points due to their small size and little contribution to the final scene.

4.2. Overview of the method

Fig. 6 shows the flowcharts of the proposed watermark embedding and extraction procedures.

In the watermark embedding part, for each pipeline, the scheme first selects the mark point for each plant component following the *mark point selecting principle*. After that, the watermark embedding part modifies the mantissa of the real-valued sphere-polar coordinates via bit substitution for each mark point according to the *watermarks embedding method*. Finally it generates the watermarked models.

In the watermark extracting part, for each pipeline, we traverse all of the connection points of each plant component to find whether the plant component has one and only one mark point. The scheme first extract the mantissa of the real-valued sphere-polar coordinates (r,θ,φ). Then the watermark bits for the connection point is generated according to the *content-based watermark generation method*. Finally, we check whether the connection point is a mark point or not according to the *tamper detection method*. Only if the plant component has one and only one mark point, then can we make sure that the plant component has been not changed. Otherwise, the tampering plant component is located accurately and reported visually.

In the following part we call the connection points to be watermarked as mark points and the other points as non-mark points.

4.3. Mark point selecting principle

In this section we describe how to select the mark points. Let L be a given pipeline with N_c plant components.$C_i(i \in [0, N_c-1])$ denotes the plant component with N_i^p connection points. $P_{i,j}(j \in [0, N_i^p - 1])$ denotes the connection points of C_i.

We first define the 1-ring neighbors of $P_{i,j}$ by using the following terminology.

The 1-ring neighbors of $P_{i,j}$ is the set of its joint connection point $P_{m,n}$ and the other connection points of C_i. We can denote the 1-ring neighbors of $P_{i,j}$ by

$$N(P_{i,j}) = \{P_{m,n} | m \in [0, N_c - 1], n \in [0, N_m^p - 1], m \neq i\} \bigcup \{P_{i,k} | k \in [0, N_i^p - 1], k \neq j\} \quad (1)$$

We traverse each pipeline of the model, according to its flow direction, from one end to the other end and choose mark points following the discipline below.

- Each plant component has one and only one mark point.
- One of the two connection points of two joint ends should be selected as a mark point.
- The connection point chosen as a mark point must have no mark point among its 1-ring neighboring connection points. Once a connection

point has been chosen as a mark point, its 1-ring neighboring connection points are no longer eligible.

Note that those isolated plant components, which have no joint plant components, are not taken into consideration for topology protection in our scheme.

Fig. 7 gives an example of our mark points selection of a simple pipeline. The black points $p_{i,0}(i \in [0,8])$ are mark points while the white points are non-mark points. For example, the the 1-ring neighbors of $p_{4,0}$ is $N(p_{4,0}) = \{p_{3,1}\} \bigcup \{p_{4,1}, p_{4,2}\} = \{p_{3,1}, p_{4,1}, p_{4,2}\}$.

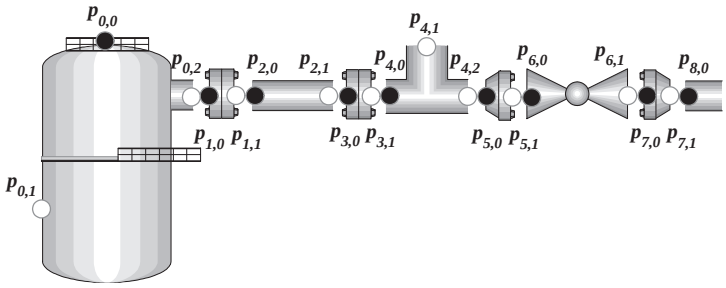


Figure 7: Illustration of mark points selection. Black points are selected mark points, while white points are non-mark points.

The union of all mark points should cover the whole model so that all of the plant components are protected. However, there might still be some plant components with no mark points according to the above mark point selecting principle. These plant components always locate at the end of a pipeline according to its flow direction. Fig. 8 shows an example of a plant component C_5 with no mark point assigned in a pipeline. For these plant components, we randomly select one of their connection points, which has

not been included in their joint mark point's 1-ring neighbors, as a mark point. Take the plant component C_5 for example, $p_{5,1}$ is selected as a mark point since $p_{5,0}$ has been included in the 1-ring neighbors of its joint mark point $p_{4,1}$.

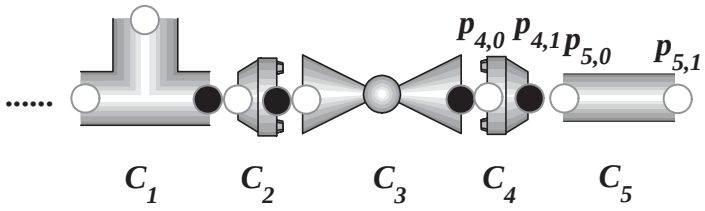


Figure 8: Illustration of a plant component C_5 with no mark point assigned in a pipeline.

Then it can be guaranteed that the 1-ring neighbors of all mark points cover the whole model according to our mark point selecting principle. And the selected mark points are uniformly distributed in the model so that all plant components are protected. Therefore it results in high locating accuracy for each joint end.

4.4. Content-based watermark generation

In order to improve the accuracy of tampering localization, a content-based watermark generation method is proposed by exploiting the topology information.

Given that a mark point is p, and its corresponding plant component is C. We define the joint plant components of C as $C_i(1 \leq i \leq n_c^{max})$. n_c^{max} represents the maximum number of joint components of C. And it is set as 4 in this paper since most of the common plant components have no more than 3 ends in general except the component 'Cross' with 4 ends. Note that

the isolated plant components are not taken into account. Let the handle value of $C_i(1 \leq i \leq n_c^{max})$ be $h_i(1 \leq i \leq n_c^{max})$ respectively. These unique handle values are not changed even if the entity is modified[18]. And they are involved in the construction of the watermark bits in order to protect the interconnection among C and its joint plant components C_i.

We generate b bits watermark strings for p according to the handle values $h_i(1 \leq i \leq n_c^{max})$ provided that the handle value h_i is arranged in ascending order. Then the b-bit watermark strings w for p can be generated by

$$\begin{aligned} w &= w_1 + ... + w_n, \\ w_i &= Sel(hash(h_i), b_i, K), i \in [1, n], \end{aligned} \tag{2}$$

where

$$n = \begin{cases} n_c^{max} - 1 & ,n_c^{max} > 1 \\ 1 & ,n_c^{max} = 1 \end{cases},$$

$$\begin{cases} b_1 = b, & n = 1 \\ b_i = \lfloor b/n \rfloor, b_n = b - \sum_{i=1}^{n-1}(b_i), i \in [1, n-1], & n > 1 \end{cases},$$

$$b = \sum_{i=1}^{n}(b_i),$$

$K = max(h_i) = h_{n_c^{max}}$, $hash()$ is a hash function, $Sel(f, b_i, K)$ is a function that selects b_i bits from the mantissa parts of the floating-point number f in a random fashion under the control of a secret key K.

4.5. The mark point embedding

For a selected mark point $p_i(x_i, y_i, z_i)$, we first find its 1-ring neighbors $N(p_i)$ according to the mark point selecting principle described in Section

4.3. Then we we calculate the center (x_c, y_c, z_c) of its 1-ring neighboring points by

$$\begin{cases} x_c = \dfrac{1}{(|N_{p_i}|-1)} \displaystyle\sum_{p_j \in N_{p_i}} (p_j)_x \\ y_c = \dfrac{1}{(|N_{p_i}|-1)} \displaystyle\sum_{p_j \in N_{p_i}} (p_j)_y \\ z_c = \dfrac{1}{(|N_{p_i}|-1)} \displaystyle\sum_{p_j \in N_{p_i}} (p_j)_z \end{cases} \tag{3}$$

where $N(p_i)$ is the set of p_i's neighboring points and $|N(p_i)|$ is the size of $N(p_i)$.

Given that (x_c, y_c, z_c) is the origin of the spherical polar coordinate system after transformation, we transform the cartesian coordinates (x_i, y_i, z_i) of p_i into sphere-polar coordinates $(r_i, \theta_i, \varphi_i)$

$$\begin{cases} r_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2} \\ \theta_i = \arccos(\dfrac{(z_i - z_c)}{r_i}) \\ \varphi_i = \arctan(\dfrac{(x_i - x_c)}{(y_i - y_c)}) \end{cases} \tag{4}$$

Afterwards we embed the same b-bit watermark bits w_i of p_i into r_i, θ_i and φ_i (treated as IEEE-754 double-precision floating-point numbers) by modifying their bit notations. The modified bit positions are parameters of our scheme, namely, P_L and P_H. P_L and P_H are the distance from the point to the lower bit and to the higher bit respectively.

$$b = P_L - P_H + 1. \tag{5}$$

We select b bits each from the mantissa parts of r_i, θ_i and φ_i according to P_H and P_L, and substitute them with the same b-bit watermark bits w_i.

Higher positions may cause lower invisibility, and lower positions may cause lower robustness against malicious attacks. Finally, we perform the reverse spherical polar transformation on the watermarked mark point.

We assign the watermark embedding point p_i as a mark point so that we can recognize it as a mark point in the watermark extraction stage.

4.6. Tamper detection

In this section, we describe the procedure used to accurately locate the tampering joint ends of plant components for each pipeline in the model.

For each plant component C_i in a pipeline, we traverse all its ends to see if it has a mark point. For each connection point $p_{i,j}$ of C_i, we first find its 1-ring neighbors. Then we apply a sphere-polar transformation to $p_{i,j}$ according to Eq. 4 and take b bits from the mantissa parts of r, θ and φ according to P_H and P_L. Let the three bit sequences generated be $w^r_{i,j}$, $w^\theta_{i,j}$ and $w^\varphi_{i,j}$ respectively. Afterwards, we calculate the watermark bits $w_{i,j}$ for $p_{i,j}$ according to the content-based watermark generation method. Finally we verify if $p_{i,j}$ is a mark point and do topology verification by checking the boolean of tag

$$tag = Bool(w^r_{i,j} == w_{i,j} || w^\theta_{i,j} == w_{i,j} || w^\varphi_{i,j} == w_{i,j}) \tag{6}$$

Only if the predefined condition $tag = true$ is satisfied, then can we set $p_{i,j}$ as a mark point

If C_i has no mark point or more than one mark points, it suggests that the topology relationship among C_i and its joint plant components has been changed since each plant component has one and only one mark point.

5. Performance discussion and experimental results

5.1. Discussion on tamper detection and localization

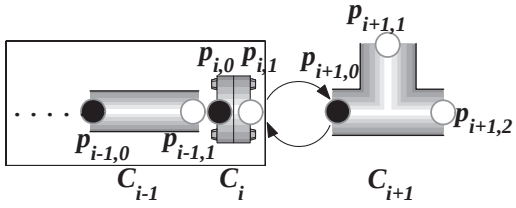
The objective of our scheme is to, for each plant component, authenticate its joint components and furthermore the exact joint ends. Therefore, we discuss the performance on detecting the modifications on the model from the following two aspects: components specific attacks and joint ends specific attacks. Components attacks mainly include adding and deleting components. While joint ends attacks include disconnecting the two joint ends geometrically and logically respectively. These modifications are common in practical design process. As discussed later, the proposed scheme in this paper can detect these kinds of modifications on CAPD models.

5.1.1. Components modification

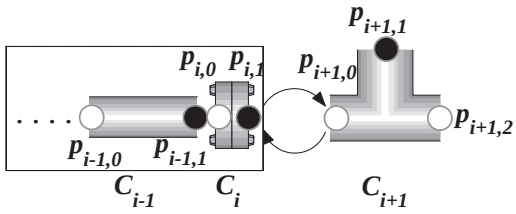
- **Components addition**. Without loss of generality, we assume that C_{i+1} is the component to be added, and C_{i+1} is connected with an existing component C_i. Therefore, the newly added component C_{i+1} will have no watermarked mark point according to the mark point selecting and embedding principle described in Section 4.3 and Section 4.5 respectively.

  During the watermark verification stage, there exist two situations when a component is added which is shown in Fig. 9. One is that $p_{i,1}$ is a non-mark point, then $p_{i+1,0}$ will be inevitably selected as a mark point shown in Fig. 9(a) according to the mark point selecting principle. Thus the extracted mantissa parts from $p_{i+1,0}$ will not match Eq. 6 since no watermark bits have been embedded into it. The other

is that $p_{i,1}$ is a mark point shown in Fig. 9(b). Then the newly added component C_{i+1} will change the centroid of the neighborhood of the mark point $p_{i,1}$. As a result, the extracted watermark will not match Eq. 6. Meanwhile, one of the two connection points $p_{i+1,1}$ and $p_{i+1,2}$ of C_{i+1} will be selected as a mark point according to the mark point selecting principle. And the extracted mantissa parts will not match Eq. 6 too, since it's not watermarked. As a consequence, the topology modification between C_{i+1} and C_i will be detected and located accurately for both of the two situations above.



(a)



(b)

Figure 9: Illustration of detecting and localizing component addition/deltetion.

- **Components deletion**. Component deletion modifies the topology

relation of the model. Without loss of generality, we also assume that C_{i+1} is the component to be deleted, and C_{i+1} is connected with an existing component C_i. Generally, component deletion can be classified into two categories shown in Fig. 9. The first category is that $p_{i,1}$ is a non-mark point. Then during the watermarking verification, the generated watermark for $p_{i,0}$ will be different from the original watermark according to the watermark generation principle described in Section 4.4. The second category is that $p_{i,1}$ is a mark point. Thus, during the watermarking verification, the 1-ring neighbors and its centroid will be changed. And the generated watermark for $p_{i,1}$ will be modified too. So that for all situations this will lead to incorrect verification, and thus the topology modification induced by components deletion is detected.

5.1.2. Joint ends modification

- **Disconnect the two joint ends geometrically**. This kind of topology attack separates one end from the other end geometrically while keeps their topology relation logically. At least one connection point will be alerted under this situation. The modified connection point may be a mark point or a non-mark point. The spherical polar coordinates (r, θ, φ) of the point will be changed if it is a mark point. While the coordinates (x_c, y_c, z_c) will be modified if it is a non-mark point, which will also lead to the modification of the value (r, θ, φ). Therefore the extracted watermark will be different from the original, and subsequently the modification is detected.

- **Change the topology relation between two joint ends logically**. This kind of topology attack changes the connection between two joint

ends logically while keeps the coordinates of their connection points unmodified. As described in Section 4.3, one of the two joint ends must have a mark point. Thus the 1-ring neighbors of the mark point will be modified. And the original watermark generated for the mark point according to the method described in the Section 4.4 will be alerted too. Consequently, the extract watermark will be different from the original and the attack will be detected.

5.2. Discussion on robustness against non-malicious attacks

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

5.2.1. Robustness against similarity transformation

The objective of performing spherical polar transformation is to obtain invariance against rotation, uniform scaling and translation (RST) in this paper. We analyze the robustness against similarity transformation in this section provided that $P(x, y, z)$ is a selected mark point and $O(x_c, y_c, z_c)$ is the geometric center of P's 1-ring neighboring connection points in the cartesian coordinate system. O corresponds to the origin of the spherical polar coordinate system. (r, θ, φ) is the sphere-polar coordinates of P after spherical polar transformation.

- **Rotation** Without loss of generality, we rotate P and O about an arbitrary axis by the angle ϕ, provided that the axis passes through the origin, since we are not considering translation at the moment. The axis

is defined by a unit vector $n = (n_x, n_y, n_z)$. Then the new coordinates of P and O after rotating about n can be written $P^r(x^r, y^r, z^r) = PR$ and $O^r(x_c^r, y_c^r, z_c^r) = OR$ respectively.

The matrix R can be constructed by three basis vectors R_1,R_2, and R_3:

$$R = [R_1, R_2, R_3] \tag{7}$$

where

$$R_1 = \begin{bmatrix} n_x^2(1-\cos\phi) + \cos\phi \\ n_x n_y(1-\cos\phi) - n_z \sin\phi \\ n_x n_z(1-\cos\phi) + n_y \sin\phi \end{bmatrix}$$

$$R_2 = \begin{bmatrix} n_x n_y(1-\cos\phi) + n_z \sin\phi \\ n_y^2(1-\cos\phi) + \cos\phi \\ n_y n_z(1-\cos\phi) - n_x \sin\phi \end{bmatrix}$$

$$R_3 = \begin{bmatrix} n_x n_z(1-\cos\phi) - n_y \sin\phi \\ n_y n_z(1-\cos\phi) + n_x \sin\phi \\ n_z^2(1-\cos\phi) + \cos\phi \end{bmatrix}$$

The corresponding new spherical polar coordinates $(r^r, \theta^r, \varphi^r)$ can then be computed by

$$\begin{cases} r^r = \sqrt{(x^r - x_c^r)^2 + (y^r - y_c^r)^2 + (z^r - z_c^r)^2} \\ \theta^r = \arccos(\dfrac{(z^r - z_c^r)}{r^r}) \\ \varphi^r = \arctan(\dfrac{(x^r - x_c^r)}{(y^r - y_c^r)}) \end{cases} \tag{8}$$

then we get

$$r^r = r \tag{9}$$

which means the component r is not changed under rotation. Thus, our scheme is invariant to rotation.

- **Uniform Scaling** We apply scale independent of the coordinate system used by scaling in an arbitrary direction. We will define $n(n_x, n_y, n_z)$ to be the unit vector parallel to the direction of scale, and k be the scale factor to be applied about the plane that passes through the origin and is perpendicular to n. Then the new coordinates of P and O after scaling about n can be written $P^s(x^s, y^s, z^s) = PS$ and $O^s(x_c^s, y_c^s, z_c^s) = OS$ respectively. The matrix S can be constructed by three basis vectors S_1,S_2, and S_3:

$$S = [S_1, S_2, S_3] \tag{10}$$

where

$$S_1 = \begin{bmatrix} n_x^2(k-1)+1 \\ n_x n_y(k-1) \\ n_x n_z(k-1) \end{bmatrix}$$

$$S_2 = \begin{bmatrix} n_x n_y(k-1) \\ n_y^2(k-1)+1 \\ n_y n_z(k-1) \end{bmatrix}$$

$$S_3 = \begin{bmatrix} n_x n_z(k-1) \\ n_y n_z(k-1) \\ n_z^2(k-1)+1 \end{bmatrix}$$

The corresponding new spherical polar coordinates $(r^s, \theta^s, \varphi^s)$ can then

be computed by

$$\begin{cases} r^s = \sqrt{(x^s - x_c^s)^2 + (y^s - y_c^s)^2 + (z^s - z_c^s)^2} \\ \theta^s = \arccos(\dfrac{(z^s - z_c^s)}{r^s}) \\ \varphi^s = \arctan(\dfrac{(x^s - x_c^s)}{(y^s - y_c^s)}) \end{cases} \tag{11}$$

then we have

$$\begin{cases} \theta^s = \theta \\ \varphi^s = \varphi \end{cases} \tag{12}$$

which means the component θ and φ keep unchanged under uniform scaling. So the proposed scheme is robust against uniform scaling.

- **Translation** Given a translation by a vector $v = (\Delta x, \Delta y, \Delta z)$, the new coordinates of P and O can be written $P^t(x^t, y^t, z^t) = (x + \Delta x, y + \Delta y, z + \Delta z)$ and $O^t(x_c^t, y_c^t, z_c^t) = (x_c + \Delta x, y_c + \Delta y, z_c + \Delta z)$ respectively. Then the corresponding new spherical polar coordinates $(r^t, \theta^t, \varphi^t)$ can be computed by

$$\begin{cases} r^t = \sqrt{(x^t - x_c^t)^2 + (y^t - y_c^t)^2 + (z^t - z_c^t)^2} \\ \theta^t = \arccos(\dfrac{(z^t - z_c^t)}{r^t}) \\ \varphi^t = \arctan(\dfrac{(x^t - x_c^t)}{(y^t - y_c^t)}) \end{cases} \tag{13}$$

then we have

$$\begin{cases} r^t = r \\ \theta^t = \theta \\ \varphi^t = \varphi \end{cases} \tag{14}$$

Table 1: Information about the three CAPD models.

Model	Connection Points	Mark Points
Carton board	13964	4654
Hydrogenation	32624	10847
Styrene	38198	12732

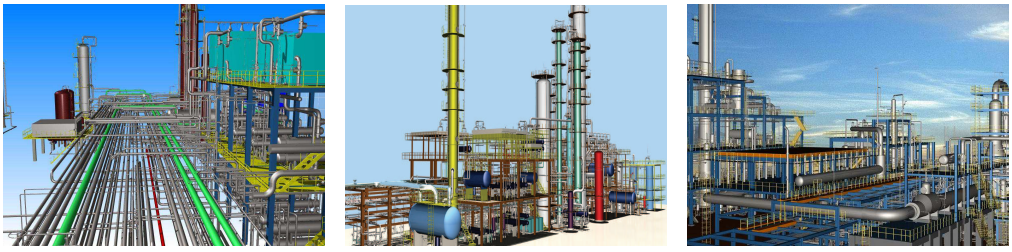
which means the component r, θ, and φ are the same. Therefore, we can see that our scheme is invariant to translation.

5.2.2. Robustness against simplification

As the complexity of process plant models increases, the enormous size of these CAD data sets poses a number of challenges in terms of interactive display and manipulation. Simplification is a key technology to reduce the model complexity and improve the rendering performance for large scale complex CAPD models. However, connection points and topological relation among plant components will not be influenced by simplification since it can only change the levels of detail of entities. 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 simplification.

5.3. Experimental results

In this section, we evaluate our semi-fragile watermarking scheme against various attacks on a set of CAPD models. Fig. 10 shows the three models used for experiments. Table 1 gives the detailed information about the three models. Parameters are set as follows: $P_L = 18$, $P_H = 10$.



(a) (b) (c)

Figure 10: Three CAPD models used for experiments. (a)Carton board plant; (b)Hydrogenation plant; (c)Styrene plant.

5.3.1. Tamper detection and localization evaluation

Fig. 11 illustrates our scheme on a hydrogenation plant model and the detected changes by our scheme. Fig. 11(a) shows part of the original hydrogenation plant model. The hydrogenation plant has 32624 connection points; 10847 connection points are selected as mark points. Fig. 11(b) is the same view of the watermarked model, which is visually identical with the original model. Fig. 11(c), Fig. 11(e), Fig. 11(g) and Fig. 11(i) show a close view of part of the marked hydrogenation plant model that has been illegally changed 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. 11(d), Fig. 11(f), Fig. 11(h) and Fig. 11(j) illustrate the located suspicious plant components in red. From Fig. 11(d), Fig. 11(f), Fig. 11(h) and Fig. 11(j) we can find that the regions in red are exactly where the tampering opera-

tions happen. The experimental results verify the accuracy of our locating procedure.

5.3.2. 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. The robustness 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^s and the originally embedded one w_i^o as given by the following equation [21]:

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

where $\overline{w^s}$ and $\overline{w^o}$ indicate the averages of the watermark bit sequence w_i^s and w_i^o respectively. This correlation value measures the similarity between two watermark bit sequences and varies between -1 (orthogonal strings) and $+1$ (the same strings).

For each plant component, if the values of BER and $Corr$ are 0 and 1 respectively, then we can set the plant component as untampered. Otherwise the plant component is detected as tampered. Let N_c be the number of plant components in a model and N_m be the number of plant components detected as tampered. Table 2 details 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.

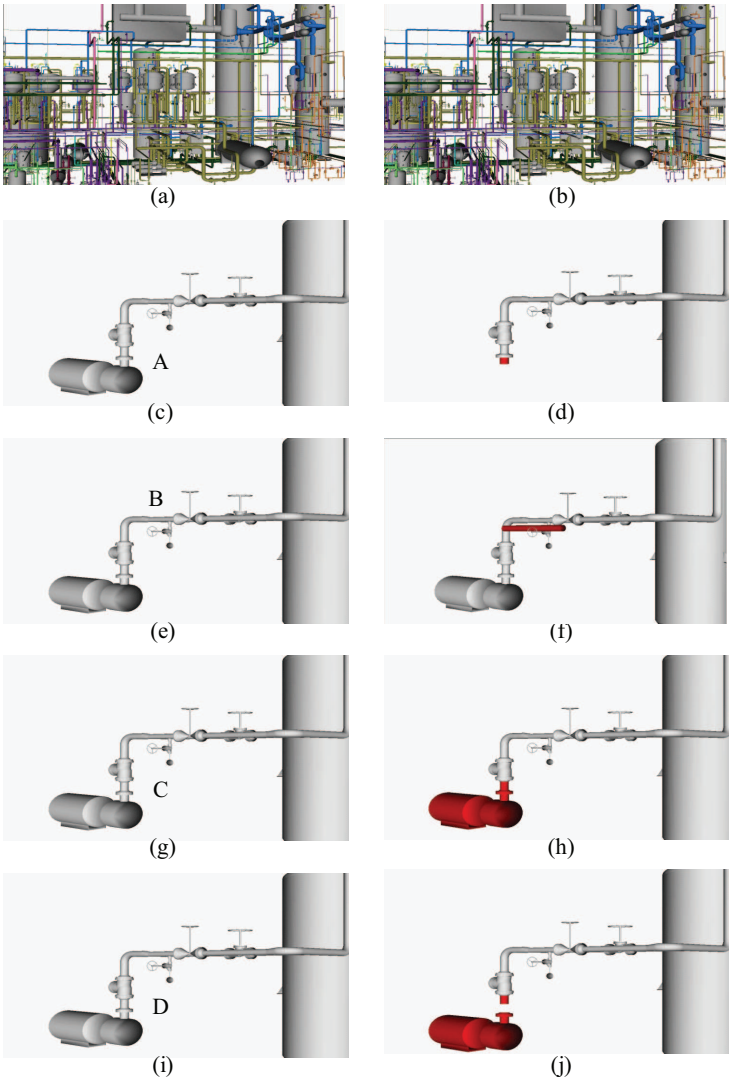


Figure 11: The proposed scheme works on a hydrogenation plant model.(a) Part of the he original model. (b) Part of the watermarked model. (c)(e)(g)(i) Several attacks on the model. Label A denotes the regions of joint components deletion. Label B denotes the regions of joint components addition. Label 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. (d)(f)(h)(j) 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
Simplification			
(80% triangles)	0	0	0
(60% triangles)	0	0	0
(40% triangles)	0	0	0

5.3.3. Imperceptibility evaluation

We measure the objective distortion of plant components and connection points induced by watermarking between the original and watermarked CAPD models by Metro [22] in terms of maximum root mean square error(MRMS) and PSNR (peak signal-to-noise ratio) [15] 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 model and watermarked model, 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.

From the Table 3, we can see that the MRMS values of plant components are all 0, since our scheme doesn't modify the geometric parameters of plant components. The PSNR values of connection points are also listed in Table 3.

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

Model	connection points	MRMS	PSNR(dB)
Carton board	13964	0	70.23
Hydrogenation	32624	0	84.47
Styrene	38198	0	81.97

According to the embedding algorithm described in Section 4.5, we know that the embedding distortion of connection points depends on the bit positions where the watermarks are embedded and the length b of the watermark. And the PSNR values are inversely proportional to the number of watermark bits b.

Note that our approach 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 alert the geometry or topology information of the model. As a consequence, our scheme will have no influence on the design and automatic generation of various construction documents. Thus, our scheme is also functionally imperceptible.

6. Conclusion

This paper is the first paper, to the best of our knowledge, to report and analyze the problem of topology verification for CAPD models. A semi-fragile and blind watermarking scheme is proposed to address the problem of authenticating topology integrity. We generate the content-based watermark

bits for each plant component by taking their topology relation into consideration. And the watermark bits are embedded into a subset of the model's connection points which are uniformly distributed in the model. Theoretical analysis and experimental results show that our scheme has a strong ability to detect and locate malicious attacks such as components modification and joint ends modification. And it is proven to be robust against various non-malicious attacks, such as transformation, rotation, uniform scaling and simplification.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (NO.61170250, NO.61103201). The models used in this paper are the courtesy of Beijing Zhongke Fulong Computer Technology Co., Ltd. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

References

[1] 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.

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

[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] 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.

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

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

[9] Bin W, Ri-jing P, Zhi-qiang Y, Shan-chao Y, Xiao-qing F, Zhi-geng. P. Watermarking t-spline surfaces. In: Proceedings of the 11th IEEE international conference on communication technology proceedings. HangZhou, China; 2008, p. 773–6.

[10] 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.

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

[12] Park HK, Lee SH, Kwon KR. Blind watermarking for copyright protection of 3d cad drawing. In: Proceedings of the 8th International Conference on Advanced Communication Technology. Gangwon-Do, Korea; 2006, p. 253–6.

[13] Kwon KR, Lee SH, Lee EJ, Kwon SG. Watermarking for 3d cad drawings based on three components. Lecture Notes in Computer Science 2006;4109:217–25.

[14] 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.

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

[16] Jang BJ, Moon KS, Huh Y, Kwon KR. A new digital watermarking for architectural design drawing using lines and arcs based on vertex. Lecture Notes in Computer Science 2004;2939:196–203.

[17] Kwon K, Jang B, Lee E, Huh Y. Copyright protection of architectural cad drawing using the multiple watermarking schemes. In: Proceedings of the IEEE international conference on multimedia and expo. Taipei, Taiwan; 2004, p. 871–4.

[18] 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.

[19] Peng F, zhou Lei Y, Long M, ming Sun X. A reversible watermarking scheme for two-dimensional cad engineering graphics based on improved difference expansion. Computer-Aided Design 2011;43(8):1018–24.

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

[21] Wang K, Lavoue G, Denis F, Baskurt A. Robust and blind mesh watermarking based on volume moments. Computers & Graphics 2011;35(1):1–19.

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