CrossMark

A unified framework for authenticating topology integrity of 2D heterogeneous engineering CAD drawings

**Zhiyong Su[1] · Lang Zhou[2] · Yaobin Mao[1] · Yuewei Dai[1] ·
Weiqing Tang[3]**

Abstract This paper presents a novel unified framework for authenticating topology integrity of 2D heterogeneous engineering CAD drawings. Topology information, through which a variety of engineering reports are generated, plays the most important role in the engineering CAD field. However, topology integrity authentication for engineering CAD drawings is still in its infancy and few efforts were made in the literature. By efficiently extracting topological and geometric features regardless of their heterogeneity with respect to geometrical shape and topology representation, the proposed framework supports verifying topology integrity for various heterogeneous engineering CAD drawings, such as process flow diagrams (PFD), piping and instrument drawings (P&ID), piping isometric drawings (ISO) as well as sectional drawings. Topology authentication is achieved through embedding local topological features into geometric features by introducing a generic and effective semi-fragile watermarking scheme. A novel descriptor, called topology local binary patterns (T-LBP), is proposed to extract local topological features of heterogeneous drawings. Theoretical analysis and experiments have demonstrated the discrimination power, robustness and sensitivity of the proposed T-LBP descriptor. We also carry out further experiments to prove that the proposed framework can not only detect and locate malicious topological modifications, but also yield strong robustness against various topology preserving modifications.

Keywords Topology integrity · Topology authentication · Local binary patterns · Engineering CAD · Heterogeneous · Watermarking

✉ Zhiyong Su
  suzhiyong@njust.edu.cn

[1] School of Automation, Nanjing University of Science and Technology, Nanjing, China

[2] College of Information Engineering, Nanjing University of Finance and Economics, Nanjing, China

[3] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China

✷ Springer

1 Introduction

Over the last ten decades, the rapid improvement of the collaborative design technology has witnessed the huge popularity increase in digital graphic documentations in the computer-aided design (CAD) field. While image, audio and video have dominated the digital media, graphic documentations are increasingly becoming an important form of media. Engineering CAD drawings, as a kind of digital graphic documentations, are very important industrial art work and are extensively used in the Architecture, Engineering and Construction (AEC) industry. Take the process industry as an example, a variety of heterogeneous engineering CAD drawings, such as process flow diagrams (PFD), piping and instrument drawings (P&ID), piping isometric drawings (ISO) as well as sectional drawings, are produced during the whole life cycle of the plant design. Collaborative design is the process where multidisciplinary designers and engineers participate in design decision-making and share product information across enterprise boundaries in an Internet-enabled distributed environment. During the collaboration, while digital drawings offer unusual flexibility in creation, manipulation, transport, storage, and retrieval, they unfortunately also bring some unique problems. They can be easily edited, replicated and distributed through networks or through stored media [8]. Therefore, protecting engineering CAD drawings against malicious tampering is of great importance and a challenge, since the integrity of engineering CAD drawings is one of the crucial factors of the engineering quality.

Topology information plays an important role in the design of CAD drawings [16]. Digital contents of these heterogeneous drawings comprise geometry information, engineering information and topology information. Geometry information refers to the shape and positions of components. Engineering information depicts design constraints, engineering disciplines, etc. Topology information describes complex topological relation among joint components. The design of engineering CAD drawings focuses on the relative topological relation among various joint components rather than the graphical representation. For instance, plant design primarily focuses on optimizing the plant layout [1, 5]. The objective of plant layout design is to determine the most economical spatial arrangement of process vessels and equipment and their interconnecting pipes that satisfy construction, operation, maintenance, and safety requirements. This is different from traditional mechanical CAD which concentrates on geometric modeling. However, to the best of our knowledge, little attention has been devoted to the problem of topology integrity authentication for engineering CAD drawings.

This paper dedicates to present a novel unified framework for tackling the topology integrity authentication problem for various 2D heterogeneous engineering CAD drawings, taking the plant design as an example. The biggest challenge facing the framework is to cope with the heterogeneity of these drawings with regard to geometrical shape and topology representation. Graphical symbols of components vary from drawing to drawing or design standard to design standard. Besides, representations of topological relation of components may differ from CAD tool to CAD tool. The essence of the proposed framework is to extract topological and geometric invariant features from heterogeneous engineering CAD drawings. Topological features are then employed to facilitate the generation of topology-sensitive watermarks. Topology authentication is implemented by embedding topology-sensitive watermarks into constructed geometric invariants. The main contributions of this paper can be classified as the following:

(1) A unified topology integrity authentication framework is presented . This framework supports verifying topology integrity for various heterogeneous engineering CAD

drawings, such as process flow diagrams (PFD), piping isometric drawings, piping and instrument drawings (P&ID) as well as sectional drawings in the process industry, by introducing a generic and effective semi-fragile watermarking scheme.

(2) A novel and rich descriptor for local topology structure called T-LBP is proposed to generalize the local topological relation among neighboring components of various heterogeneous drawings. The proposed T-LBP descriptor extends the conventional LBP beyond texture features to topology characteristics for the first time.

(3) Geometric invariants, which are robust against both global and local similarity transformations, are constructed and selected as watermark carriers for watermarks embedding. Topology-sensitive watermarks are generated based on the computed T-LBP descriptors and then embedded into these watermark carriers for topology authentication.

(4) The proposed framework achieves robustness against several topology preserving operations in addition to the common merits (such as global and local similarity transformations) of previous watermarking algorithm: object copying, object rearrangement and file format converting. Therefore, it is a general algorithm and applicable to various heterogeneous engineering CAD drawings following different design standards in industry practices.

The rest of the paper is organized as follows. Section 2 describes the interesting problem discussed in this paper. Section 3 reviews some related works. Details of the proposed framework is described in Section 4. The experimental results and performance discussions have been presented in Section 5. We conclude the article and point to main directions of future research in Section 6.

2 Problem statement

In this section, we start with a brief introduction of the heterogeneity of engineering CAD drawings in the process industry. Then, we discuss some general topology preserving operations, which are common functions in practice. The design of the proposed framework should take these two aspects into consideration.

Several kinds of 2D engineering CAD drawings are produced during the whole life cycle of the plant design. The process plant consists of various components, including equipments, piping components and pipes. Piping components cover fittings, valves, flanges, gasket, etc. In this paper, without loss of generality, we take process flow diagrams, piping and instrument drawings, piping isometric drawings and sectional drawings as examples to discuss the topology authentication problem. PFDs show the flow of chemicals and the major equipments involved in the process. A P&ID includes more details than a PFD. It includes major and minor flows, control loops and instrumentation. The piping isometric drawing is a detailed orthographic drawing. It represents the details of the 3D structure of the pipe in the form of a 2D diagram. A sectional drawing provides a view of the whole or part of a model as though it had been cut along some imaginary plane. The drawing is used to show the relative position of specified components of process plants.

2.1 Heterogeneity of engineering CAD drawings

We investigate the heterogeneity of engineering CAD drawings from the following two aspects: geometrical shape and topology representation.

2.1.1 Heterogeneity of geometrical shape

In the case of the same kind of drawings produced by different CAD tools, the same component may come in different shapes and sizes. This is because that the design standards which process plant drawings should follow differ from country to country. In order to increase productivity and shorten project design cycles, CAD tools used in the process industry always provide hundreds of catalogs representing either dimensional standards or manufacturer specific components. These catalogs include hundreds of thousands of items used to create specific specifications for the project requirements.

In terms of different kinds of drawings produced during different design stages, the same component is represented by different graphic symbols in different drawings. And, regarding geometry, there is no association between those different shapes of the same component.

In addition, multiple scales may be employed for engineering CAD drawings. To achieve a satisfactory appearance and fit, it is common to apply similarity transformations on certain individual components to revise their dimensions or locations without altering their topological relation. Performing these changes will create a cleaner and more legible drawing and further facilitate the annotation for various components.

2.1.2 Heterogeneity of topology representation

The topological relation between two joint components falls into two categories, geometric topological relation or logical topological relation. Geometric topological relation simply indicates the connection relationship in geometric space. Logical topological relation is the core concern of the topological structure in the process industry. The two joint components should meet the specific requirements, such as pipe diameter, end type, pressure rating and flow direction. To represent the topological relation between joint components, two popular formats are widely adopted in commercial softwares nowadays: the storing order of components in the file [14] as well as connection points [3] which are based on handle values.

2.2 Topology preserving operations

Engineering CAD drawings can be easily edited intentionally or non-intentionally using CAD tools. Here, we discuss some general topology preserving operations from the following two aspects: handle value insensitive operations and handle value sensitive operations. The handle value is a unique code for each entity in the drawing. Every object entity has its own handle value that is allocated in the order of entity generation. The topology integrity authentication scheme should survive in these non-malicious operations.

2.2.1 Handle value insensitive operations

Handle value insensitive operations cover global and local similarity transformations as well as stretching operations on pipes. In order to achieve a satisfactory appearance and fit, local similarity transformations are often applied on certain individual components. These operations, including rotation, translation and uniform scaling, are taken in order to revise dimensions or locations of selected components without altering their topological relation. And, they will not change handle values of modified components. Components which occupy a large space in the drawing will inevitably influence the projection of other
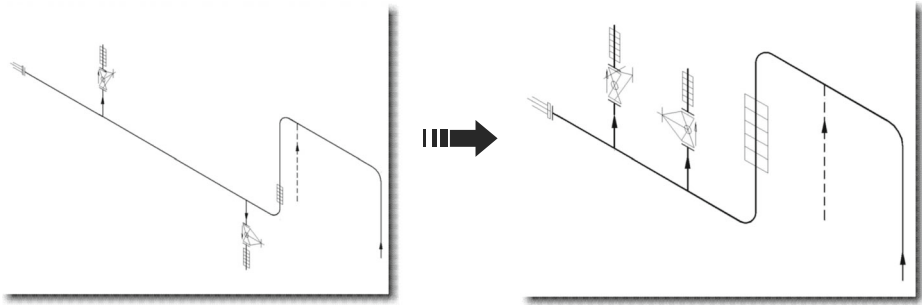
Fig. 1 An example of topology-preserving geometric modifications. To make the isometric cleaner and more legible, local similarity transformations are applied on specified components in the left isometric. Two isometric drawings are equivalent regarding their functions

ones. Thus, these components with larger dimensions should be scaled down. In the same way, those ones with smaller dimensions should be scaled up to avoid any inconvenience when reading the drawings. The regions with dense pipelines should be expanded, and those regions with sparse pipelines should be compressed. Besides, pipes may be stretched with different ratios due to local similarity transformations applied on certain piping components discussed above.

Figure 1 gives an example of topology preserving geometric modifications on a simple piping isometric. Local similarity transformations are performed on specified components in the left drawing, respectively. Consequently, some pipes are stretched due to those operations. These changes are made to create a cleaner and more legible isometric and further facilitate the annotation for various components. Therefore, two drawings in Fig. 1 are equivalent regarding their functions.

2.2.2 Handle value sensitive operations

In this paper, handle value sensitive operations refer to those editing functions including object copying, object rearrangement and file format converting [7]. These editing functions are also common operations during the design cycle in practice. And, they are performed based on the premise of retaining topological information of components in this paper. Handle values of components can be easily reallocated or lost when these operations are applied on partial or entire drawings Lots of conventional watermarking schemes have to know handle values for watermarks embedding or extraction [12, 15]. The topology authentication scheme should be robust against these editing functions if topological information and types of attacked components are kept.

3 Related works

Current study of content integrity authentication for 2D and 3D graphical models has been directed towards techniques applicable to geometric information protection and authentication in the literature [7, 10, 12, 13, 18, 20]. Ohbuchi et al. [10] first proposed that the CAD model should be recognized as a multimedia data type, along with such data types as sound, text, still image, and movie. And they discussed digital watermarking technology for

geometric CAD data, which may offer a solution to the security related issues of authentication, tamper detection, and intellectual property management. Multiple approaches for watermarking mechanical CAD models defined by using parametric curves and surfaces are then presented. Lee et al. [7] presented a robust watermarking scheme based on k-means++ for CAD drawings. The proposed scheme clustered the target objects in the selected layers by using k-means++ and embedded the watermark into the geometric distribution of POLYLINE, 3DFACE, and ARC objects in the main layers. Peng et al. [12, 13] have been made some contributions on reversible watermarking for 2D engineering graphics. A semi-fragile watermarking algorithm for authenticating 2D CAD engineering graphics based on log-polar transformation was proposed in [12]. The watermark is embedded in the mantissa of the real-valued log-polar coordinates via bit substitution. Two reversible watermarking schemes based on difference histogram shifting were put forward in [13]. Difference histograms are constructed according to the characteristics of the coordinates and phases in 2D CAD engineering graphics. The proposed schemes have great potential to be applied for content authentication or secret communication of 2D CAD engineering graphics. Xiao et al. [20] proposed a combined reversible watermarking scheme for 2D CAD engineering graphics based on improved quantization index modulation (IQIM) and improved difference expansion. The proposed scheme can solve the embedding-limitation problem existing in IQIM technique and increase the watermark embedding capacity greatly so that every vertex of the 2D CAD engineering graphics can be utilized for embedding watermark.

Recently, a semi-fragile and blind watermarking scheme was proposed to address the topology integrity authentication problem of piping isometric drawings [15]. Topology authentication is achieved by embedding topology sensitive watermarks into selected areas of drawings. Handle values of entities are employed in both watermarks generation and extraction procedures. The proposed scheme is fragile to those non-malicious handle value sensitive operations discussed above. These limitations further restrict the generality of the scheme in industry practices.

To summarize, existing watermarking techniques for 2D CAD drawings typically target the geometric information protection and authentication. Moreover, they are primarily designed to resist global similarity transformation attacks, such as rotation, translation and uniform scaling, which are applied to the entire drawings. The motivation behind this paper stands upon the need of developing new general and robust tools to support the topology integrity authentication for various 2D heterogeneous engineering CAD drawings, which even produced by different CAD tools in industry practices.

4 The unified framework for topology authentication

4.1 Overview of the framework

The block diagram of the proposed framework for authenticating heterogeneous engineering CAD drawings is shown in Fig. 2. Major computational modules of the proposed framework involve topological feature extraction, geometric feature extraction and topology authentication. Topological feature extraction modules firstly construct the directed topological graph for the input drawing. Then, the T-LBP descriptor is calculated for each node by virtue of the proposed topology local binary patterns. Geometric feature extraction modules aim to select embedding targets and then construct geometric invariants. Finally, topology authentication is resolved by embedding topology-sensitive watermarks generated based on
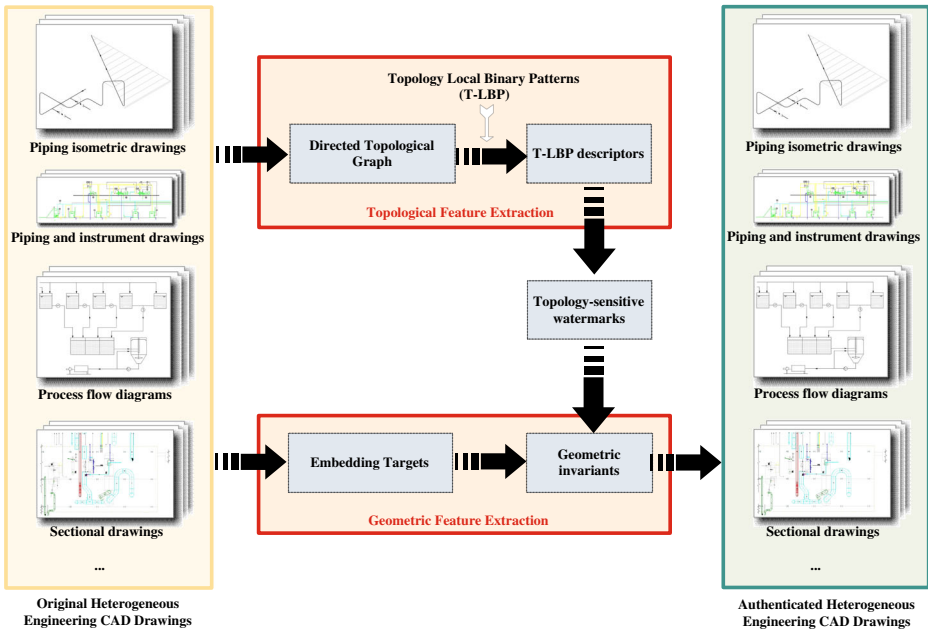
Fig. 2 The proposed framework

T-LBP descriptors into geometric invariants of selected components. The details of these key processing stages appear in Sections 4.2, 4.3 and 4.4.

4.2 Topological feature extraction

To extract topological features, we first present a unified representation of heterogeneous engineering CAD drawings as directed topological graphs. Then, we propose a T-LBP descriptor to efficiently summarize the local topological relation among neighboring components regardless of the heterogeneity with respect to geometric shape and topology representation, which is inspired by the conventional LBP for texture analysis purposes. The most important properties of T-LBP are its generality, computational simplicity and tolerance regarding topology preserving geometric modifications.

4.2.1 Directed topological graph construction

In this subsection, we first describe the basic terminology and principles of constructing the directed topological graph. Then, we give a comprehensive example to illustrate the construction procedure.

The directed topological graph is a set of nodes and a collection of directed edges that each connects an ordered pair of nodes. The node refers to equipments or piping components. The directed edge corresponds to either the logical connection relationship between two joint nodes or the pipe between two neighboring nodes.

For a specified engineering CAD drawing, given that C is the set of its equipments and piping components, P is the set of its pipes and R is the set of connection relationship between two joint components of C, its directed topological graph is defined as

$D = (N, E)$, where $N = \{n|n \in C\}$ are nodes and $E = \{e|e \in P \bigcup R\}$ are directed edges. The direction of edges is identical with the flow direction of components. Considering two adjacent nodes n_i and n_j, the directed edge is written as $e_{ij} = < n_i, n_j >$, if the flow direction is from n_i to n_j. n_i is called an initial node and n_j is called a terminal node of the edge. e_{ij} indicates the connection relationship if n_i and n_j connect with each other directly. Otherwise, it represents the pipe between them if n_i and n_j connect with each other through a pipe.

For each node n_i, the in-degree of n_i is written by $deg^-(n_i)$, which is the number of edges with n_i as the terminated node. While the out-degree of n_i is written by $deg^+(n_i)$, which is the number of edges with n_i as the initial node. Then the degree of n_i is denoted by $deg(n_i) = deg^-(n_i) + deg^+(n_i)$. The set of pioneers of n_i is denoted by $\Gamma_D^-(n_i) = \{n_i^j|n_i^j \in V \wedge < n_i^j, n_i >\in E \wedge n_i \neq n_i^j\}$. While the set of successors of n_i is denoted by $\Gamma_D^+(n_i) = \{n_i^j|n_i^j \in V \wedge < n_i, n_i^j >\in E \wedge n_i \neq n_i^j\}$. Thus, the set of neighbors of n_i is written by $N_D(n_i) = \Gamma_D^-(n_i) \bigcup \Gamma_D^+(n_i)$.

Figure 3 gives an example of constructing the directed topological graph for a given piping isometric. Figure 3a shows the given piping isometric which is composed of 10 piping components and 12 pipes. The flow direction is labeled by arrows. Figure 3b is the constructed directed topological graph. The blue directed edges indicate the connection relationship between two joint piping components. While black directed edges represent that the couple of piping components are connected with each other through pipes. For example, the blue directed edge $e_{10,9}$ in Fig. 3b indicates that n_{10} and n_9 connected with each other directly as shown in Fig. 3a. The black directed edge $e_{5,4}$ denotes that n_5 and n_4 connected with each other through a pipe as shown in Fig. 3a too. Black nodes in Fig. 3a represent undefined piping components. Take the node n_4 for example, $deg^-(n_4) = 2$, $deg^+(n_4) = 1$, $deg(n_4) = 3$, $\Gamma_D^-(n_4) = \{n_5, n_8\}$, $\Gamma_D^+(n_4) = \{n_3\}$, $N_D(n_4) = \{n_5, n_8, n_3\}$.
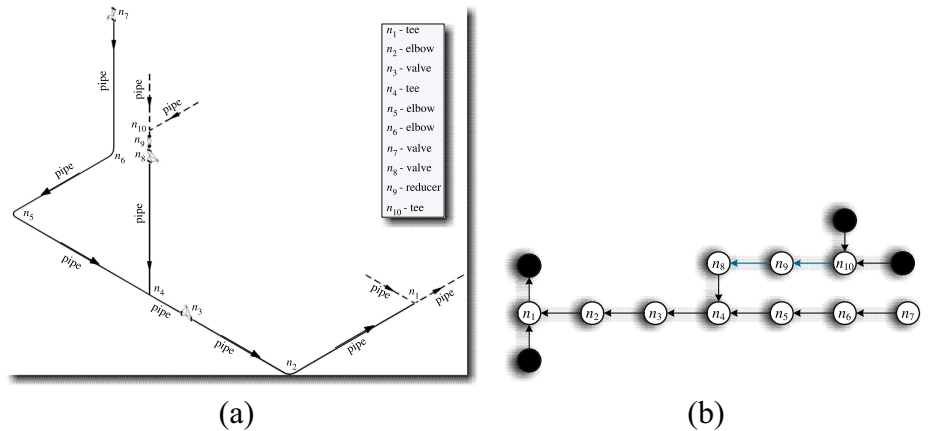

(a)   (b)

Fig. 3 An example of constructing the directed topological graph for a given piping isometric. **a** A piping isometric with 10 piping components and 12 pipes. The flow direction is represented by arrows. **b** The constructed directed topological graph. Nodes refer to piping components. Black nodes represent undefined piping components. The blue directed edge indicates that the two piping components are connected with each other directly. The black directed edge represents that the two piping components are connected with each other through a pipe

4.2.2 Uniform code for components

In order to encode and represent the topological relation among joint components in heterogeneous drawings uniformly, each kind of components is expressed with a uniform code represented by fixed m integers. These uniform codes are designed to be globally unique on the basis of component types, regardless of their handle values, geometric shapes and sizes in different design standards. The parameter m is determined by the total number of component types. The uniform code is also assigned to each node of the directed topological graph according to the type of its corresponding component. Taking the process plant for example, all components can be classified into four levels which means that m is set to 4. The first level covers pipes, piping components and equipments. The second level, in terms of piping components, includes valve, fitting, bend, flange, etc. The third level, in terms of valve, comprises gate valve, angle valve, etc. The fourth level, in terms of gate valve, covers double disc parallel seat, plug gate valve, etc. It is observed that 4 integers are enough to encode all the types of components. Hence, a globally unique code with 4 integers is assigned to every component in each level.

4.2.3 Topology local binary patterns

In this subsection, we first introduce the basic terminology of the LBP descriptor briefly. Then, we describe the proposed T-LBP descriptor with discussion and motivation on using the T-LBP to encode local topological features. Finally, we illustrate how to calculate the T-LBP descriptor based on the local topological relation.

We start by formulating the traditional LBP descriptor first introduced by Ojala et al. [11]. LBP has proved a simple yet powerful approach to efficiently summarize local structures by comparing each pixel with its neighboring pixels. Due to its excellent performance, LBP has been extensively studied in a wide array of fields and has demonstrated superior performance in several comparative studies [2, 6, 9, 21]. Conventional LBP descriptor extracts information which is invariant to local gray scale variations in the image. It is computed at each pixel location, considering the values of a small circular neighborhood (with radius R pixels) around the value of a central pixel c. Formally, given a central pixel c located at coordinate (x_c, y_c), the value of the LBP code of c is given by

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c) \times 2^p \qquad (1)$$

with

$$s(g_p - g_c) = \begin{cases} 1, g_p - g_c \geq 0 \\ 0, g_p - g_c < 0 \end{cases}, \qquad (2)$$

where P is the number of neighbor pixels of c whose distances to c do not exceed the radius R, g_c and g_p are the intensities of c and a neighboring pixel p respectively. Figure 4 shows an example of the LBP computation for a typical 3×3 neighborhood corresponding to a small gray-scale image portion.

The proposed T-LBP descriptor extends the conventional LBP beyond texture features to topological features. The main idea behind T-LBP is that, for a given node, we treat it as the central pixel. And, its adjacent nodes are regarded as neighbor pixels. The uniform code of each node serves as the pixel value. Directions between the given node and its adjacent nodes correspond to the signs of the differences of pixel values in the traditional LBP. Its procedure can be described as follows:
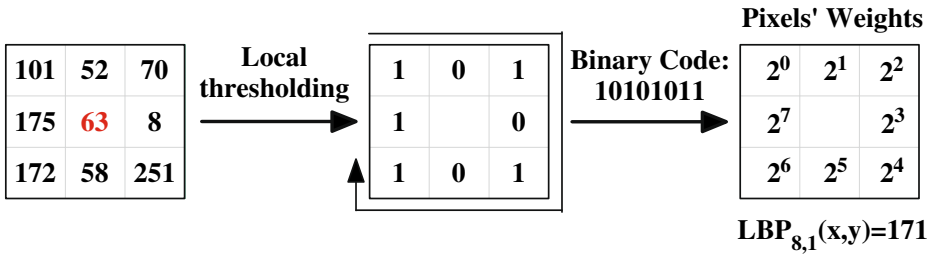
Pixels' Weights

101	52	70
175	**63**	8
172	58	251

Local thresholding →

1	0	1
1		0
1	0	1

Binary Code: 10101011 →

2^0	2^1	2^2
2^7		2^3
2^6	2^5	2^4

$LBP_{8,1}(x,y)=171$

Fig. 4 Illustration of the original LBP descriptor

(1) For a given drawing, we first construct its directed topological graph D. For each node n_i of D, we get its sets of pioneers $\Gamma_D^-(n_i)$ and successors $\Gamma_D^+(n_i)$, respectively.

(2) After that, by reference to the original LBP descriptor, we define a texture unit represented by 9 elements for n_i since its maximum number of neighboring nodes will not exceed 9 in process plants. In the constructed texture unit, n_i serves as the central pixel. Each node in $N_D(n_i)$ corresponds to a neighboring pixel of the central pixel. And the values of the pixels are set to the uniform codes of corresponding nodes, respectively. These uniform codes also act as the weights of the corresponding pixels. Generally, in addition to the above active pixels, there may be some unused neighboring pixels due to the varying numbers of neighboring nodes of n_i.

(3) In the binary case, the pixel is marked following the topological relation and flow direction based on the constructed directed topological graph. If the node in $N_D(n_i)$ is a pioneer of n_i, then its corresponding pixel is marked as 1. Otherwise, the pixel is marked as -1.

(4) Finally, the T-LBP descriptor $T - LBP_{deg(n_i),1}(n_i)$ is calculated based on the generated binary code and the weights given to the corresponding pixels.

$$T - LBP_{deg(n_i),1}(n_i) = \sum_{j=0}^{deg(n_i)-1} f(n_i^j) \times 10^{g(j) \times m \times s(n_i^j, n_i)} \tag{3}$$

with

$$s(u,v) = \begin{cases} 1, & <u,v> \in E \\ -1, & <v,u> \in E \end{cases} \tag{4}$$

$$g(j) = \begin{cases} j, & j < deg^-(n_i) \\ j - deg^-(n_i) + 1, & deg^-(n_i) \le j \le deg(n_i) - 1 \end{cases}, \tag{5}$$

where $n_i^j \in N_D(n_i)$, $f(n_i^j)$, which serves as the weight, is the uniform code of the component n_i^j with m digits.

In order to reach a rotation invariant T-LBP descriptor, for each node n_i, its neighboring nodes are classified and sorted according to their flow direction and uniform codes. Firstly, the nodes in $N_D(n_i)$ are classified as pioneers $\Gamma_D^-(n_i)$ and successors $\Gamma_D^+(n_i)$ according to the edge direction. Then, nodes in $\Gamma_D^-(n_i)$ and $\Gamma_D^+(n_i)$ are arranged in descending order according to their uniform codes separately. Finally, we can get a unique arranged set $N_D(n_i)$ by grouping the two ordered sets $\Gamma_D^-(n_i)$ and $\Gamma_D^+(n_i)$ together sequentially.

According to (3), the T-LBP descriptor depends on uniform codes, flow direction and local connection relationship of components. Therefore, it achieves strong discrimination power on local topology relation. Furthermore, it is also by definition invariant against
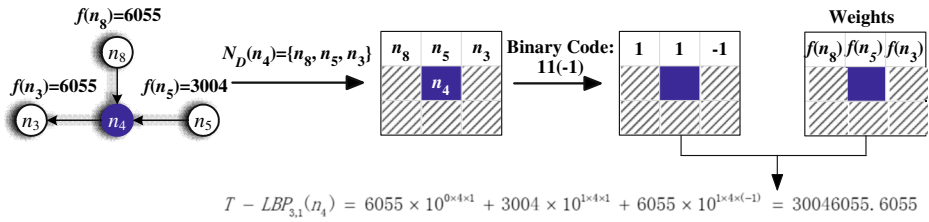
$$T - LBP_{3,1}(n_4) = 6055 \times 10^{0 \times 4 \times 1} + 3004 \times 10^{1 \times 4 \times 1} + 6055 \times 10^{1 \times 4 \times (-1)} = 30046055.6055$$

Fig. 5 Encoding the topological relation of the node n_4 based on the T-LBP descriptor. The texture unit with 4 active pixels is first constructed based on the ordered neighboring set $N_D(n_4)$. Then, the binary code is generated. The weight of each active pixel is set to the uniform code of its corresponding component. Finally, the T-LBP descriptor is calculated based on the binary code and weights

any topology preserving geometric modifications and sensitive to malicious topological modifications.

Figure 5 gives an example of the T-LBP descriptor computation. Taking the node n_4 in Fig. 3b as an example, we first get its sets of pioneers $\Gamma_D^-(n_4) = \{n_5, n_8\}$ and successors $\Gamma_D^+(n_4) = \{n_3\}$, respectively. $\Gamma_D^-(n_4)$ and $\Gamma_D^+(n_4)$ are arranged to be $\Gamma_D^-(n_4) = \{n_8, n_5\}$ and $\Gamma_D^+(n_4) = \{n_3\}$ according to their uniform codes, respectively. Then, we get the ordered set $N_D(n_4) = \Gamma_D^-(n_4) \bigcup \Gamma_D^+(n_4) = \{n_8, n_5, n_3\}$. After that, we construct a texture unit with 4 active pixels for n_4 which serves as the central pixel. Consequently, we get the binary code "11(−1)" and set the weights of active pixels to the uniform codes of corresponding components, respectively. Finally, the transformation of the binary code to the T-LBP descriptor denoted as a decimal value is achieved by applying (3).

4.3 Geometric feature extraction

In order to extract geometric features of components in heterogeneous drawings, we first select appropriate embedding targets. Then, we construct geometric invariants of these targets. The geometric feature extraction method presented in this paper can be applied to various graphical symbols of same components.

4.3.1 Embedding targets selection

We argue that equipments and piping components are the best candidates for data embedding among the types of data objects introduced in Section 2.1 for the following reasons. First, it's observed that equipments and piping components show more complex geometric shape than pipes which are simply represented by straight lines in most cases. Second, pipes may be inevitably stretched with different ratios when similarity transformation operations are applied on their joint components. It is reasonable to employ geometrical primitives of equipments and piping components to embed watermarks to resist certain classes of geometric transformations. Therefore, we can exploit more redundancy in equipments and piping components to embed topological features. For the above reason, we prefer equipments and piping components to be embedding targets.

An eligible triangle is constructed for each embedding target. We observe that either equipments or piping components consist of at least three non-collinear vertices. Therefore, we can always construct some candidate triangles for further selection. Assume that an embedding target is comprised of a set of vertices V. Let v_i denote the ith vertex of V. The eligible triangle construction and selection methods are described as follows:
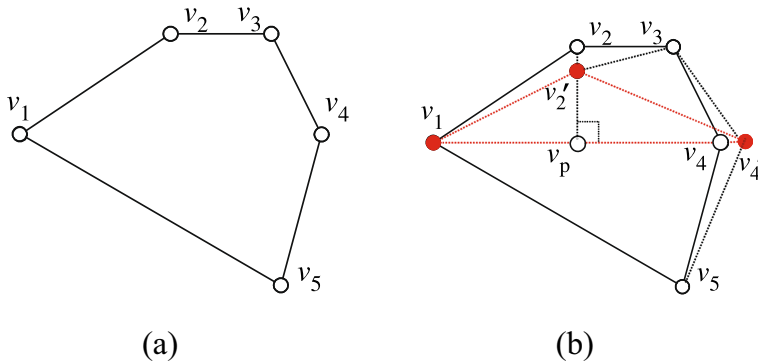
Fig. 6 An example of constructing the eligible triangle for an embedding target. **a** The original convex hull $S = \{v_1, v_2, v_3, v_4, v_5\}$ of an embedding target. **b** The constructed triangle $v_1 v_2' v_4'$ which is highlighted in red. The new convex hull are $S' = \{v_1, v_2', v_3, v_4', v_5\}$

(1) Firstly, we traverse the vertex set V to find its convex hull $CH(V)$. The convex hull is defined as the minimal convex set containing V. A lot of convex hull algorithms can be employed. In this paper, the classical convex hull algorithm proposed by Graham is preferred [4]. The triangle is chosen as the candidate if there are only three vertices in $CH(V)$.

(2) Secondly, we calculate the length of all the edges and diagonals of the convex hull $CH(V)$ and find the longest one. If there is more than one edge (diagonal) with the longest length, we select one of them arbitrarily. Then the selected one is lengthened to achieve the longest length by adjusting the coordinates of one of its two vertices along the direction of the edge (diagonal) they reside on.

(3) Finally, we construct the eligible triangle through finding another vertex from $CH(V)$ with the minimum vertical distance to the selected longest edge (diagonal). If there is more than one vertex with the minimum distance, we randomly select one and shorten the vertical distance from it to the selected longest edge (diagonal) by slightly adjusting the coordinates of the vertex along its vertical direction.

Figure 6 shows an example of the eligible triangle construction for an embedding target. The original convex hull of the embedding target is illustrated in Fig. 6a. The diagonals with the longest length are $v_1 v_4$ and $v_2 v_5$. First, the diagonal $v_1 v_4$ is randomly selected and lengthened by adjusting the coordinates of v_4 along the direction of $v_1 v_4$. Then, we calculate the distances from each vertex to $v_1 v_4'$ and find out the vertex with the minimum distance. Since the distance from v_2 to $v_1 v_4'$ is equal to the distance from v_3 to $v_1 v_4'$, we randomly select one vertex, such as v_2, from them and then shorten the vertical distance from it to $v_1 v_4'$ by slightly adjusting the coordinates of v_2 along its vertical direction $v_2 v_p$. Finally, we get the eligible triangle $v_1 v_2' v_4'$.

4.3.2 Geometric invariant construction

We construct a geometric invariant for each embedding target, given its selected embedding triangle T which is denoted $\triangle ABC$. First, we select the longest side as the base of the triangle of T. If there is more than one side with the longest length, we randomly select one
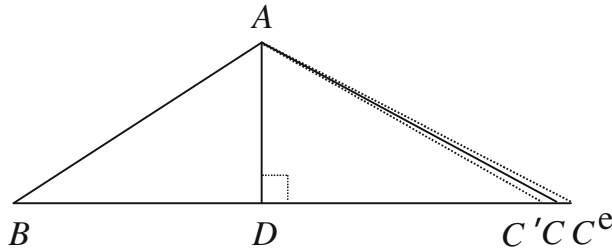
Fig. 7 Illustration of embedding the watermark by slightly perturbing the location of C to C^e

side and move one of its two vertices outwards with a shift distance δ, following the direction of the side on which it resides. Without loss of generality, as illustrated in Fig. 7, we assume that AD is an altitude of T, the vertex D is the foot of the altitude AD, and BC is the base of the altitude AD. Then, we select the length ratio AD/BC to be the watermark carrier. The selected geometric invariant is very favorable for watermarking since it is preserved under any global and local similarity transformations.

4.4 Topology authentication

This section describes the proposed T-LBP based topology authentication scheme through the digital watermarking technique [17] in detail. The proposed method consists of two parts, which are the watermark embedding and the watermark extraction. In the watermark embedding part, we firstly describe the watermark generation method based on the T-LBP descriptor. Then, we detail how to embed the topology information into the length ratio of the embedding primitive. In the watermark extraction part, we firstly extract the embedded watermark from its embedding primitive. Then, we compute its T-LBP descriptor according to its current topological relation. Finally, the topology integrity is verified through comparing the extracted watermark with the watermark calculated based on the current T-LBP descriptor.

4.4.1 Watermark Generation

Watermarks are generated based on T-LBP descriptors. These topology-sensitive watermarks are then embedded into selected areas of embedding targets.

To generate topology-sensitive watermarks, a chaotic function called Kent map is first predefined as the key for the watermark generation. The Kent map is one of the most studied discrete chaotic maps.

$$y_{k+1} = \begin{cases} y_k/a, & 0 \le y_k \le a \\ (1 - y_k)/(1 - a), & a < y_k \le 1 \end{cases} \tag{6}$$

where $0 < a < 1, k \ge 1$. y_k is a number between 0 and 1 if $y_0 \in [0, 1]$, and it is the current value of the mapping in time with an initial value y_0. When the Kent map is seeded with a 'function seed' a, and iterated, chaotic behavior is witnessed in general. Different sequences will be generated with different initial values since the Kent map is extremely sensitive to initial conditions.

We generate a unique watermark w_i ($0 < w_i < 1$) for each node n_i. The T-LBP descriptor $T - LBP(n_i)$ is first computed. Let I_i and D_i be the integer part and the decimal part of $T - LBP(n_i)$, respectively

$$T - LBP(n_i) = I_i + D_i. \tag{7}$$

Then, the watermark w_i is calculated by:

$$w_i = kent(I_i, D_i), \tag{8}$$

where $kent(x, y)$ is a function which generates w_i from iterating the Kent map for x times with the initial value y and the 'function seed' a_i.

$$a_i = \varepsilon + A_i, \tag{9}$$

where ε and A_i are control parameters defined for each node n_i. And, they also serve as private keys to enhance the security of our scheme. A_i depends on the connection type that n_i and its neighboring nodes n_i^j ($n_i^j \in N_D(n_i)$) use. And it takes the form

$$A_i = 0.B_0...B_{deg(n_i)-1} \tag{10}$$

where $B_j = 0$ ($0 \leq j \leq deg(n_i) - 1$) if n_i and n_i^j connect with each other through a pipe. Otherwise, $B_j = 1$ if n_i and n_i^j connect with each other directly.

4.4.2 Watermark embedding

We embed the topology-sensitive watermark into the geometric invariant through the QIM (Quantization Index Modulation) method with a quantization step size Δ.

Suppose that the selected length ratio $r = AD/BC$. Topology-sensitive watermarks embedding methods are described as follows:

(1) At first, r is partitioned by the step size Δ. In general, r cannot be completely divided by Δ. In that case, the remainder is discarded by adjusting the location of C to C' such that $r' = AD/BC'$ can be divided by Δ.

(2) At last, we embed the watermark w into r' by changing the C' location to C^e as illustrated in Fig.7.

$$r^e = r' + w \times \Delta = \lfloor r/\Delta \rfloor \times \Delta + w \times \Delta. \tag{11}$$

where $\lfloor \cdot \rfloor$ represents the floor function.

4.4.3 Watermark extraction

For each embedding target, the detailed watermark extraction procedures consist of the following steps:

(1) Find its convex hull and construct the eligible triangle T.
(2) Select the length ratio r^e (AD/BC) to be the geometric invariant of T.
(3) Extract the embedded watermark w^e from r^e with the quantization step size Δ by

$$w^e = \frac{r^e - \lfloor r^e/\Delta \rfloor \times \Delta}{\Delta}. \tag{12}$$

After extracting the watermark w^e, we check the topology integrity of the embedding target according to the following steps:

(1)   We first calculate the T-LBP descriptor according to its current topological relation.
(2)   Then, the new watermark w^c is generated based on the current T-LBP descriptor.
(3)   Finally, we use w^e and w^c to check the authentication. All corresponding w^e and w^c should satisfy

$$|w^e - w^c| < \tau. \tag{13}$$

where the parameter τ is involved to address the numerically instable problem [19]. That is, not satisfying (13) suggests at least one neighboring component of the embedding target has been changed. Thus, for any piping component that cannot satisfy the relation, we set its neighboring components as suspicious. There is no way for a forger to modify a model and keep the relationship unchanged without the keys.

5 Performance discussions and experimental results

We provide performance analysis and experimental results of the proposed framework in this section. The experiments address the performance assessment of T-LBP, the ability of tamper detection and localization, robustness to various attacks, and watermark invisibility.

5.1 Experimental drawings and settings

The proposed framework was applied on a large number of engineering CAD drawings of process plants. We provide the results of five selected engineering CAD drawings shown in
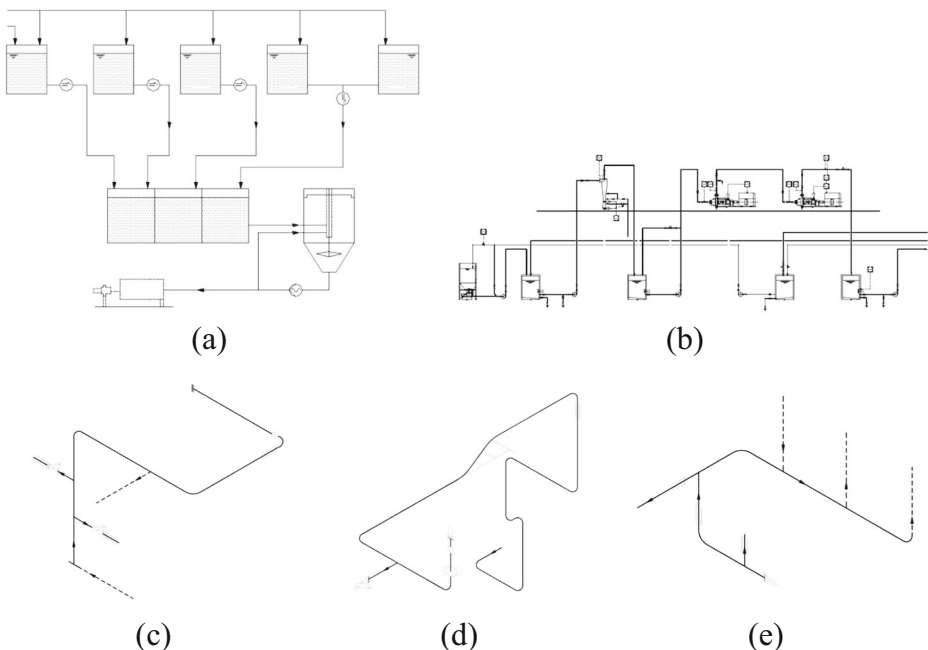

Fig. 8   Selected engineering CAD drawings used in the experiments. **a** A PFD drawing. **b** A P&ID drawing. **c–e** Piping isometric drawings

Table 1 RMSE values between the watermarked engineering CAD drawings and the original engineering CAD drawings

Drawings	Equipments	Pipes	Piping Components	RMSE($\times 10^{-3}$)
PFD	10	40	5	0.093
P&ID	8	113	66	0.132
ISO1	0	14	14	0.254
ISO2	0	16	15	0.106
ISO3	0	15	11	0.029

Fig. 8. The number of equipments, piping components and pipes of each engineering CAD drawing are provided in Table 1.

5.2 Optimization of the parameters

Several parameters are required during the watermark embedding and detection stage in our scheme.

The parameter τ is involved in (13) for comparing the extracted watermarks with the original ones in order to determine if the topology integrity has been destroyed. This is because that the embedding and extraction procedures are based on the floating-point arithmetic. The authentication scheme may fail to work properly due to the numerically instable problem, which is common in the floating-point arithmetic [19]. In order to find a proper τ, we conduct experiments on 60 watermarked drawings which are attacked by various topology preserving modifications. These components are divided into 6 groups (G1-G6) evenly. We apply global and local similarity transformations on 6 groups, respectively. Suppose that the original embedded watermark is w and the extracted watermark is w^e. The numerical error ξ is obtained by $\xi = |w - w^e|$. Table 2 lists the maximum numerical errors of extracted watermarks from each test embedding target of 6 groups (G1-G6) under various non-malicious attacks mentioned above. From Table 2 we can see that the maximum numerical errors induced by computational errors are within the interval $[0, 0.248 \times 10^{-3}]$. Thus, the parameter τ is recommended to be larger than 0.248×10^{-3}. In this paper, τ is set to 0.4×10^{-3}.

Table 2 Maximum numerical errors ($\times 10^{-3}$) of extracted watermarks from each test embedding target of 6 groups (G1-G6) under various non-malicious attacks, including global rotation (G-R), global scaling (G-S), global translation (G-T), local rotation (L-R), local scaling (L-S), local translation (L-T)

	G-R	G-S	G-T	L-R	L-S	L-T
1	0.130	0.028	0.012	0.094	0.248	0.164
2	0.102	0.140	0.024	0.100	0.090	0.181
3	0.213	0.128	0.178	0.223	0.210	0.115
4	0.052	0.076	0.084	0.007	0.121	0.085
5	0.099	0.012	0.127	0.112	0.085	0.193
6	0.137	0.166	0.010	0.164	0.194	0.212
7	0.065	0.150	0.123	0.232	0.162	0.090
8	0.087	0.200	0.078	0.177	0.009	0.124
9	0.009	0.119	0.152	0.092	0.132	0.095
10	0.207	0.008	0.159	0.102	0.093	0.137
	G1	G2	G3	G4	G5	G6

There also exist some additional parameters in the scheme. The uniform code of the component is represented by fixed 4 integers since it is observed that 4 integers are enough to encode all the types of components in the graphical symbol library. The shift distance δ is set to $\Delta/2$. The parameter Δ is set to 0.001 according to the drawings' precision. The control parameter ε in (9) is set to 0.45.

5.3 Evaluation of T-LBP

5.3.1 Discrimination power assessment

The discrimination performance of the proposed T-LBP descriptor, as shown in (3), depends on the uniform codes, flow direction and local connection relationship of components. Each kind of component in the graphical symbol library is assigned a uniform code which is represented by fixed 4 integers. These uniform codes are designed to be globally unique on the basis of component types. In order to make the T-LBP descriptor invariant to component rearrangement, for each component, its neighboring components are arranged according to their uniform codes and flow direction. Therefore, the T-LBP descriptor can achieve strong discrimination power.

Figure 9 illustrates directed topological graphs and computed T-LBP descriptors of the same piping component n_2 with a different neighboring piping components or flow direction. The piping component n_3 in Fig. 9a is replaced by a new piping component n_4 with a different uniform code. This modification results in the changing of the T-LBP descriptor as
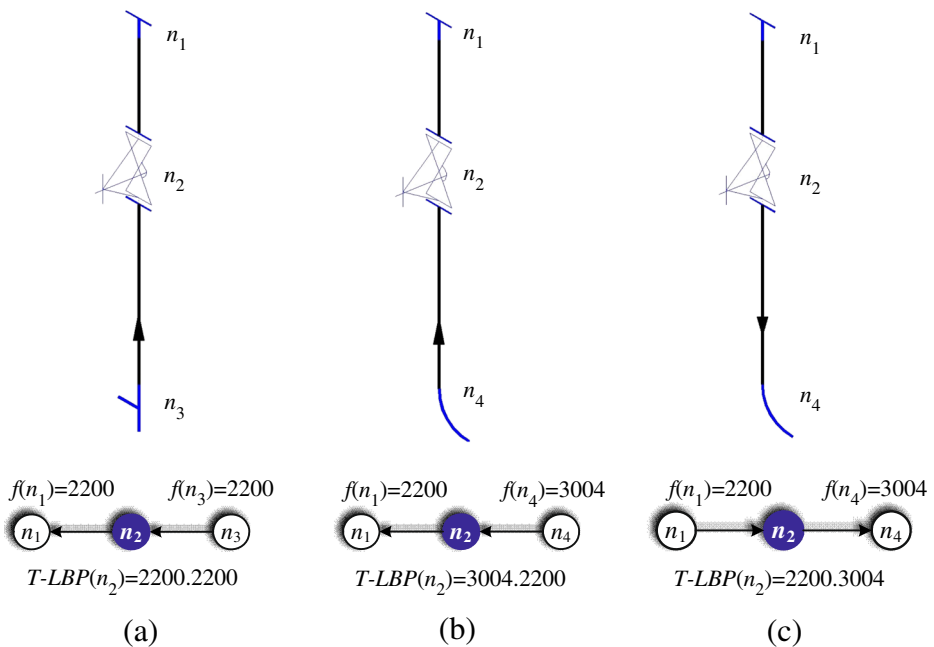

Fig. 9 Directed topological graphs and T-LBP descriptors of the same piping component n_2 with different neighboring piping components or flow direction. The piping component n_3 in (**a**) is replaced by a different piping component n_4 in (**b**). The flow direction of all components in (**b**) is diverted as shown in (**c**)

shown in Fig. 9b. Then, we divert the flow direction of all components in Fig. 9b. Figure 9c illustrates the variation of the directed topological graph and the T-LBP descriptor.

5.3.2 Robustness assessment

The T-LBP descriptor is proposed to extract topology characteristics of embedding targets for topology integrity authentication. Thus, it should be invariant to topology preserving geometric modifications, including global similarity transformations and local similarity transformations and the stretching operation on pipes. These operations are applied to create a cleaner and more legible drawings and further facilitate the annotation for various components. They just affect the position, dimension and orientation of components. The directed topological graph remains the same. Hence, as discussed in Section 5.3.1, the T-LBP descriptor keeps fixed when suffering these operations. And for this reason, the descriptor is used to generate topology-sensitive watermarks for the authentication.
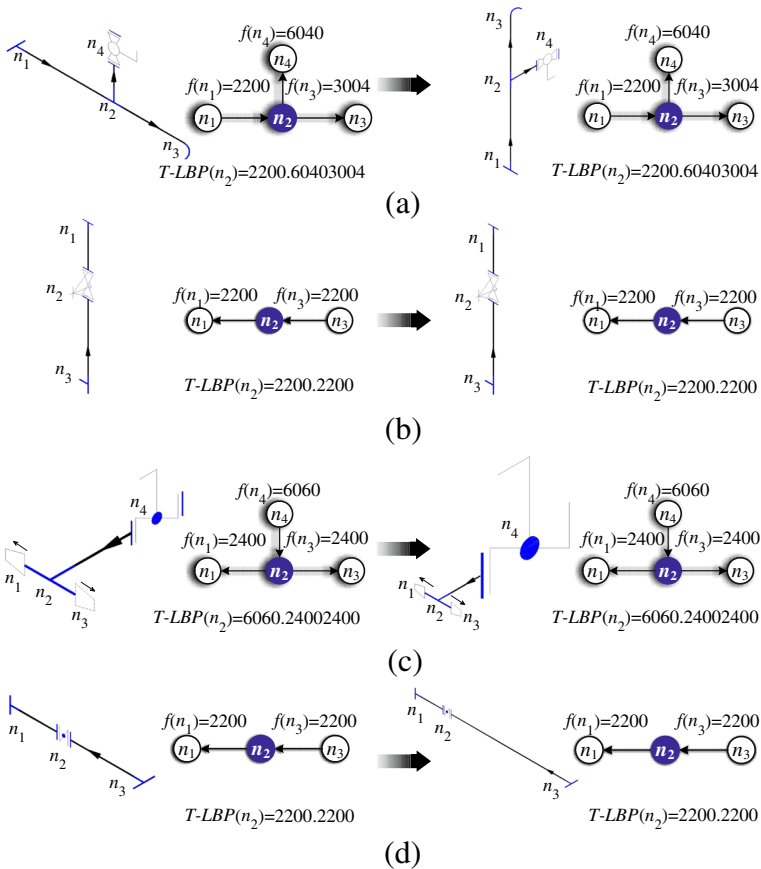

Fig. 10 Robustness of the T-LBP descriptor against topology preserving geometric modifications. **a** Local and global rotation. **b** Global scaling. **c** Local scaling. **d** Stretching on pipes

Figure 10 presents some examples of T-LBP descriptors of selected piping components under various topology preserving geometric modifications. In Fig. 10a, first, only the piping components n_2 and n_4, as well as the pipe between them are rotated 90 degrees. Then, the entire drawing is rotated 90 degrees too. In Fig. 10b, we apply global uniform scaling on the entire drawing. We also perform local scaling on the piping component n_4 as shown in Fig. 10c. At last, the pipe between piping components n_2 and n_3 is stretched as shown in Fig. 10d. From Fig. 10 we can find that directed topological graphs of attacked drawings keep constant which further contribute to the same T-LBP descriptors.

5.3.3 Sensitivity assessment

To authenticate topology integrity, the T-LBP descriptor should not only be immune to topology preserving geometric modifications, but also be sensitive to malicious topological modifications. These modifications cover adding components, deleting components
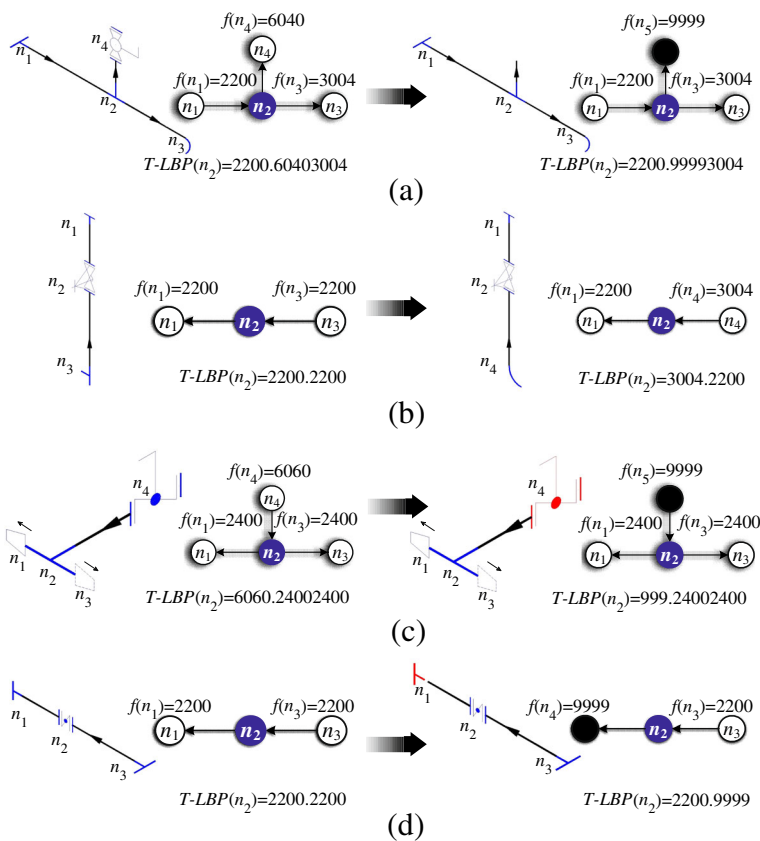

Fig. 11 Sensitivity of the proposed T-LBP descriptor under various topological attacks. **a** Deleting n_4. **b** Replacing n_3 with n_4. **c** Disconnecting n_4 from its joint component logically. **d** Disconnecting n_1 from its joint component logically and geometrically

and modifying topological relation logically. For a given piping component, these topology modifications change either the number of neighboring components or the uniform codes. Consequently, as seen from (3), the T-LBP descriptor will be changed. Therefore, the descriptor is fragile to the above malicious topology modifications.

Figure 11 illustrates the sensitivity of the proposed T-LBP descriptor under various malicious topological attacks. Black nodes are undefined nodes whose default uniform codes are 9999. The piping component n_4 in Fig. 11a is deleted from the drawing. Consequently, the directed topological graph of the attacked drawing is modified and so does the T-LBP descriptor. In Fig. 11b, the piping component n_3 is replaced by a new piping component n_4 with a different uniform code. This modification results in the changing of the directed topological graph and T-LBP descriptor as shown in Fig. 11b. The piping component n_4 in Fig. 11c, which is labeled in red, is disconnected from its joint component logically. In Fig. 11d, the piping component n_4 which is marked in red is disconnected from its joint component logically and geometrically. These operations change both the directed topological graphs and T-LBP descriptors of the attacked drawings.

5.4 Tamper detection and localization assessment

We discuss the performance of the proposed scheme on detecting and locating malicious topological modifications on the drawing in this section. Malicious topological modifications include adding components, deleting components, and modifying the local topological relation logically.

5.4.1 Adding components

On the basis of component type, adding components may be classified as pipe addition or non-pipe component (piping component and equipment) addition.

In terms of pipe addition, three situations arise. The first is that the added pipe is connected with a piping component. The second is that the added pipe is used to connect two disconnected piping components. These attacks change directed topological graphs of attacked piping components and lead to the modifications of their T-LBP descriptors. The last situation is that two joint piping components are disconnected first and then connected with each other through the added pipe. This kind of attack keeps directed topological graphs and T-LBP descriptors of referred piping components constant. However, they change the control parameter A used in (9). In short, all the attacks discussed above will result in the modification of watermarks of attacked piping components during the verification stage. Consequently, the topological relation of attack piping components will be regarded as tampered.In terms of non-pipe component addition, the added component should be connected with existing pipes or non-pipe components. This kind of attack changes the topological relation of modified non-pipe components. Thus, their directed topological graphs and T-LBP descriptors are changed. As a result, calculated watermarks of those modified non-pipe components will be different from the extracted ones during the extraction stage. Therefore, the topological relation between those modified non-pipe components and their joint components will be labeled as tampered.

5.4.2 Deleting components

This kind of attack is divided into two cases: deleting pipes and deleting non-pipe components (piping component and equipment).

In the case of pipe deletion, it disconnects the pipe from its joint non-pipe components. In the case of non-pipe component deletion, it disconnects the non-pipe component from its joint pipes or non-pipe components. Thus, the topological relation of the involved non-pipe components is modified. Then, the modified directed topological graph gives rise to the changing of the T-LBP descriptors of the referred non-pipe components. Consequently, during the extraction stage, the difference between extracted watermarks and newly calculated ones is induced. Therefore, the topological relation between these piping components and their joint components will be considered as tampered.

5.4.3 Modifying local topological relation logically

Modifying local topological relation of components involves various operations, such as disconnecting two joint components logically, and connecting two disconnected components logically. These operations inevitably bring about the alteration of directed topological graphs of referred non-pipe components. Hence, the T-LBP descriptors of the referred non-pipe components will be changed too. Consequently, watermarks generated based on the new T-LBP descriptors differ from the extracted watermarks. Therefore, these involved non-pipe components will be located accurately during the verification stage.

5.5 Evaluation of robustness

We assess the robustness against various topology preserving operations discussed in Section 2.2, which are typical operations in applications for engineering CAD drawings of process plants.

The true positive rate (TPR) is used to evaluate the robustness against various topology preserving modifications and their combination. Given that N is the total number of watermarked non-pipe components and N^t is the number of non-pipe components which are detected as tampered. We apply the formula

$$TPR = \frac{N^t}{N} \tag{14}$$

to compute the true positive rate.

5.5.1 Robustness against handle value insensitive operations

Global similarity transformations applied on the entire drawing are common operations which do not destroy topology integrity of components. Thus, these operations do not change T-LBP descriptors and further the embedded watermarks. Meanwhile, length ratios which are preferred as watermark carriers are invariant to similarity transformations. As a result, the proposed scheme is robust against global similarity transformations.

Local similarity transformations performed on specified individual components are characteristic operations in applications for engineering CAD drawings of process plants as discussed in Section 2.2. These operations alert the layout of drawings while keep the topological relation constant. Therefore, T-LBP descriptors and embedded watermarks remain the same. Besides, watermark carriers are preserved under similarity transformations. As a result, the proposed scheme can resist these local similarity transformations on specified individual components.

Pipes may be stretched with different ratios when local similarity transformations are performed on specified components. In order to achieve the robustness against this kind of
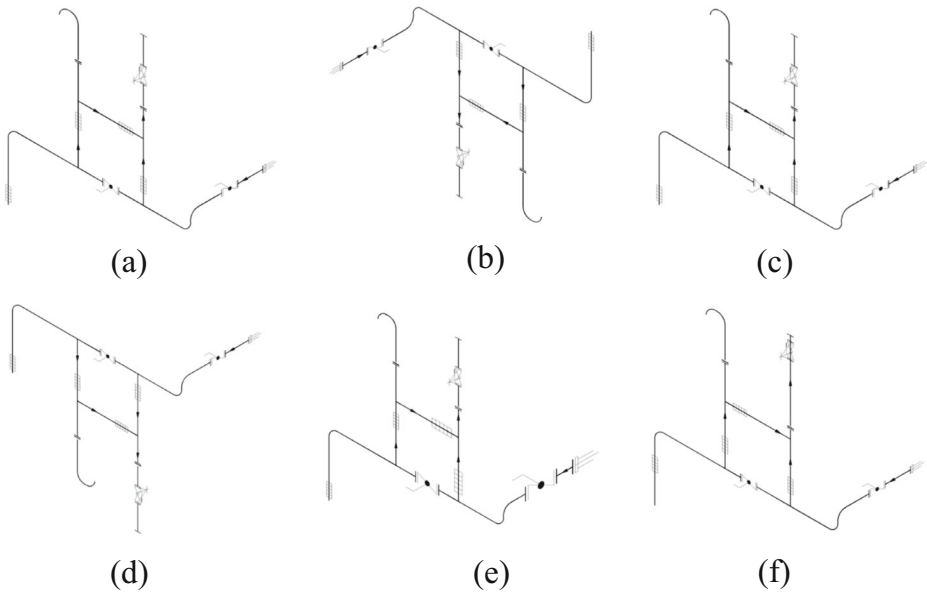
Fig. 12 Examples of global and local similarity transformations for robustness test. **a** Original drawing. **b** Global rotation by 180^0. **c** Global scaling with a factor 2.0. **d** Local rotation. **e** Local scaling. **f** Local translation

operation, we first map the pipe between two non-pipe components to the edge between two neighboring nodes in the directed topological graph. The edge indicates the connection relationship between two neighboring nodes. Therefore, this kind of operation does not modify T-LBP descriptors as well as watermarks. Second, we prefer to embed T-LBP based watermarks into non-pipe components rather pipes. By doing so, the watermark extraction will be free from these operations. As a result, embedded watermarks can be extracted accurately. By virtue of the above elaborate approaches, the proposed scheme is robust against the stretching operation on pipes.

We carry out a series of tests on test drawings to evaluate the robustness of our scheme. To evaluate the resistance against global similarity transformations, as illustrated in Fig. 12a–c, we apply rotation, uniform scaling and translation on the entire test drawings, respectively. In these tests, the parameters for various global transformations are set as the following: (a)G-R: rotation by 45^0, 90^0 and 180^0, respectively; (b)G-S: scaling with a factor 0.5, 2.0 and 10.0, respectively; (c)G-T: translating along X-axis and Y-axis by three arbitrary distances, respectively. To evaluate the robustness to local similarity transformations , we apply rotation (L-R), uniform scaling (L-S) and translation (L-T) on some selected non-pipe components (about 50 %) of each drawing respectively, as illustrated in Fig. 12d–f. TPR values of test drawings under the above attacks are all 0.0 which indicate that the proposed scheme can extract embedded watermarks from the attacked drawings accurately.

5.5.2 Robustness against handle value sensitive operations

As discussed in Section 2.2.2, handle value sensitive functions referred in this paper can reallocate handle values based on the premise of retaining topological information. These functions are common and non-malicious operations in industry practices.

In terms of object copying, components can be copied using mirror copying or array generation. In the mirror copying operation, handle values are allocated in all mirror-copied objects since the mirror-copied objects are added after the objects in the original drawing. In the array generation operation, new unique handle values are allocated to a multiple copied objects similar as mirror copying. Therefore, handle values of new objects generated by object copying are allocated depending on the handle values of previous objects. However, these copied objects have the same geometric and topological information of the original ones. In the case of object rearrangement, handle values can be easily reallocated in a lump by object rearrangement. In terms of file format converting, the drawing can be easily stored or converted to other formats through CAD tools. Handle values of objects may be reallocated or lost during format conversion. The proposed framework is robust against these operations since it does not require handle values for generating, embedding and extracting watermarks.

5.6 Evaluation of invisibility

The invisibility is affected by the distance distortion introduced in watermark carriers construction and watermark embedding. We can control the maximum distortion from each piping component and the maximum average distortion by setting the quantization step Δ according to the precision requirement. The larger the quantization step Δ, the larger the induced distortion.

We employ the RMSE (Root Mean Square Error) to measure the distortion between the watermarked drawings and the original drawings

$$RMSE = \frac{1}{N}\|v - v^{'}\|, \tag{15}$$

where v and $v^{'}$ are the corresponding vertices in the original drawing and the watermarked drawing respectively, and N denotes the total number of vertices in the drawing.

Table 1 shows statistics of geometry distortion of the test drawings. From Table 1 we can see that, in terms of the drawings' precision, the geometrical distortion between the original drawing and the watermarked drawing is very small. More importantly, the geometric distortion does not ruin the topological relation among joint components. Therefore, our scheme is visually and functionally imperceptible.

5.7 Performance comparison

To the best of our knowledge, in the literature, no related works which focus on the problem investigated in this paper have been reported except the proposed scheme designed specially for piping isometric drawings [15]. The main superiority of the proposed framework, compared with the previous work [15], can be summarized as follows.

First of all, the proposed framework is applicable to various 2D heterogeneous engineering CAD drawings including piping isometric drawings. This significant extension is achieved by proposing a novel T-LBP descriptor which discussed in Section 4.2.3. The T-LBP descriptor can generalize the local topological relation among neighboring components regardless of the heterogeneity with respect to geometrical shape and topology representation.

Second, the proposed framework is robust against handle value sensitive operations. The previous scheme in [15] was designed on the basis of handle values, which are involved in

watermark generation, geometric invariants construction and watermark extraction procedures. As a result, it can not resist those non-malicious handle value sensitive operations which lead to the modification of handle values. In this paper, the proposed framework is independent on handle values. Consequently, it is robust against those handle value sensitive operations discussed in this paper. Therefore, the proposed framework gets a better robust performance.

All in all, it is believed that the proposed framework is more generic and practical in industry practices.

6 Conclusions

In this paper, taking the plant design in the process industry as an example, we present a novel unified framework for authenticating topology integrity of 2D heterogeneous engineering CAD drawings. Heterogeneous drawings are firstly represented uniformly by directed topological graphs. Then, a novel descriptor called T-LBP is proposed to extract topological features which are further employed to compute topology-sensitive watermarks. Finally, topology authentication is achieved through embedding topology-sensitive watermarks into selected areas of engineering CAD drawings. An extensive set of experiments is carried out. We demonstrate that the T-LBP descriptor achieves good discrimination power, robustness against topology preserving geometrical modifications as well as sensitivity to various malicious topological attacks. The performance of the authentication scheme with respect to tamper localization and robustness is also clearly demonstrated. It's believed that the proposed framework is a general algorithm and applicable to various kinds of heterogeneous engineering CAD drawings in the AEC industry produced by different CAD tools in industry practices.

In this work, the authentication framework is designed based on the digital watermarking technique which embeds watermarks by changing the coordinates of drawings. Therefore, the precision of CAD drawings can be changed slightly by watermarking. In our future work, we intend to take the content-based hashing technique, which has been widely used in multimedia information security and retrieval, into consideration for topology integrity authentication and verification.

References

1. Burdorf A, Kampczyk B, Lederhose M, Schmidt-Traub H (2004) CAPD-Computer-aided plant design. Comput Chem Eng 28(1–2):73–81
2. Ciocca G, Cusano C, Schettini R (2015) Image orientation detection using lbp-based features and logistic regression. Multimedia Tools and Applications 74(9):3013–3034
3. Dow M (1987) Integration of calculation models and CAD systems in building services design. Comput Aided Des 19(5):226–232
4. Graham R (1972) An efficient algorithm for determining the convex hull of a finite planar set. Inf Process Lett 1(4):132–133
5. Guirardello R, Swaney R (2005) Optimization of process plant layout with pipe routing. Comput Chem Eng 30(1):99–114

6. Huang D, Shan C, Ardabilian M, Wang Y, Chen L (2011) Local binary patterns and its application to facial image analysis: A survey. IEEE Trans Syst Man Cybern Part C Appl Rev 41(6):765–781
7. Lee S, Kwon K (2010) CAD Drawing watermarking scheme. Digital Signal Process 20(5):1379–1399
8. Mazurczyk W, Szczypiorski K (2014) Advances in digital media security and right management. Multimedia Systems 20(2):101–103
9. Nanni L, Lumini A, Brahnam S (2012) Survey on LBP based texture descriptors for image classification. Expert Syst Appl 39(3):3634–3641
10. Ohbuchi R, Masuda H (2001) Managing CAD data as a multimedia data type using digital watermarking. In: Proceedings of the IFIP TC5 WG5.2 fourth workshop on knowledge intensive CAD to knowledge intensive engineering. Deventer, The Netherlands, The Netherlands, pp 103–116
11. Ojala T, Pietikäinen M., Harwood D (1996) A comparative study of texture measures with classification based on featured distributions. Pattern Recogn 29(1):51–59
12. Peng F, Guo R, Li C, Long M (2010) A semi-fragile watermarking algorithm for authenticating 2D CAD engineering graphics based on log-polar transformation. Comput Aided Des 42(12):1207–1216
13. Peng F, Liu Y, Long M (2014) Reversible watermarking for 2D CAD engineering graphics based on improved histogram shifting. Comput Aided Des 49(4):42–50
14. Su Z, Li W, Kong J, Dai Y, Tang W (2013) Watermarking 3D CAPD models for topology verification. Comput Aided Des 45(7):1042–1052
15. Su Z, Yang X, Liu G, Li W, Tang W (2015) Topology authentication for piping isometric drawing. Comput Aided Des 66(9):33–44
16. van der Meiden HA, Bronsvoort WF (2010) Tracking topological changes in parametric models. Comput Aided Geom Des 27(3):281–293
17. Wang Y, Hu S (2010) Optimization approach for 3d model watermarking by linear binary programming. Comput Aided Geom Des 27(5):395–404
18. Wang N, Men C (2013) Reversible fragile watermarking for locating tampered blocks in 2d vector maps. Multimedia Tools and Applications 67(3):709–739
19. Wang W, Zheng G, Yong J, Gu H (2008) A numerically stable fragile watermarking scheme for authenticating 3D models. Comput Aided Des 40(5):634–645
20. Xiao D, Hu S, Zheng H (2015) A high capacity combined reversible watermarking scheme for 2-D CAD engineering graphics. Multimedia Tools and Applications 74(6):2109–2126
21. Yang H, Cao Y, Su H, Fan Y, Zheng S (2014) The large-scale crowd analysis based on sparse spatial-temporal local binary pattern. Multimedia Tools and Applications 73(1):41–60

Zhiyong Su is currently an associate professor in the School of Automation at Nanjing University of Science and Technology, China. He received the B.S. and M.S. degrees from the School of Computer Science and Technology, Nanjing University of Science and Technology in 2004 and 2006, respectively, and received the Ph.D. from the Institute of Computing Technology, Chinese Academy of Sciences in 2009. His current interests include computer vision and computer graphics.

Lang Zhou received her B.S., M.S. and Ph.D. degrees from the School of Computer Science and Technology, Nanjing University of Science and Technology in 2004, 2006 and 2010, respectively. She joined the College of Information Engineering in Nanjing University of Finance and Economics, China, in 2010. Her current interests include pattern recognition and natural language processing.

Yaobin Mao is currently an associate professor in the School of Automation at Nanjing University of Science and Technology, China. He received the B.S. degrees from the School of Automation, Beijing Institute of Technology in 1993 and 1996, respectively, and received the Ph.D. from the School of Automation, Nanjing University of Science and Technology in 2003. His current interests include image processing, computer vision and information security.

Yuewei Dai is currently a professor in the School of Automation at Nanjing University of Science and Technology, China. He received his M.S. and Ph.D. degrees from Nanjing University of Science and Technology, in 1987 and 2002, respectively, all in automation. His research interests are in the areas of information security, signal and image processing.

Weiqing Tang received the B.S. and M.S. degrees from the Nanjing University of Science and Technology, China, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, China, in 1993. He is now a professor at the Institute of Computing Technology, Chinese Academy of Sciences, China. His research interests include computer graphics, computational geometry and CAD.