Communications in Information and Systems
Vol. 1, No. 2, pp. 225–240, April 2001

006

THE CONSISTENCY PROBLEM ON CONTENT-BASED PICTORIAL DESCRIPTION IN PICTORIAL DATABASE SYSTEMS*

QING-LONG ZHANG†, SHI-KUO CHANG‡, AND STEPHEN S.-T. YAU†

Abstract. In this paper we propose the consistency problem on content-based pictorial description, which is of common interest for Pictorial Database Systems. Then we suggest a framework for content-based Pictorial Database Systems. We describe major components of Pictorial Database Systems, and demonstrate how they work together to facilitate content-based picture indexing and retrieval. We finally discuss the consistency problem for spatial relationships in a picture.

1. Introduction. With the interest in multimedia systems over the past few years, content-based image retrieval has attracted the attention of researchers across several disciplines [10]. Applications that use image databases include office automation, computer-aided design, robotics, geographic data processing, remote sensing and management of earth resources, law enforcement and criminal investigation, medical pictorial archiving and communication systems, and defense. While the use of indexing to allow database accessing has been well-established in traditional database systems, content-based picture indexing techniques need to be developed for facilitating pictorial information retrieval from a pictorial database.

Tanimoto [15] suggested the use of picture icons as picture indexes, thus introducing the concept of iconic indexing. Subsequently, Chang et al. [4] developed the concept of iconic indexing by introducing the 2D string representation of the image. Since then, the 2D string approach has been studied further in the literature. A detailed summary on the 2D string approach was given by Chang and Jungert [3]. Other methods on image representation and retrieval can be found in the literature (see, e.g., [2, 7, 8, 9, 11, 12, 14]). Sistla et al. [14] developed a rule system for reasoning about spatial relationships in picture retrieval systems. One obvious distinction between the work of Sistla et al. [14] and the work such as [3, 4, 9] is that the spatial operators in [14] are defined by absolute spatial relationships among objects, while the spatial operators in the other approaches are defined by relative spatial relationships among objects. Consider, for example, two significant objects A and B in a real picture. Then the spatial relationship "A is left of B" (written as "A *left-of* B") in [4] means that the position of the centroid of A is left of that of B (and we say "A *left-of* B" is relative), whereas in [14] it means that A is absolutely left of B (and we say "A *left-of* B" is absolute). Note that the operator *left-of* has the weaker meaning in [4] than in [14] in the sense that "A *left-of* B" is true in [4] whenever it is true in [14], and "A

---

*Received Sep 1, 2000; accepted for publication Mar 14, 2001.

†Control and Information Laboratory, Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, 322 Science and Engineering Offices, 851 South Morgan Street, Chicago, Illinois 60607, USA. The first author's current address: Lucent Technologies – Bell Labs Innovations, 2000 N. Naperville Road, P.O. Box 3033, Naperville, IL 60566. E-mail: qlzhang@lucent.com, yau@uic.edu.

‡Department of Computer Science, University of Pittsburgh, Pittsburgh, PA 15260, USA; E-mail: chang@cs.pitt.edu.

left-of B" is not necessarily true in [14] when it is true in [4]. Spatial relationships may be classified into directional and topological relationships. The 2D string approach developed by Chang et al. [4] is based on (relative) directional spatial relationships: *left-of, right-of, above*, and *below*. Egenhofer and Franzasa [5, 6] proposed that there are eight fundamental (absolute) topological spatial relationships that can hold between two planar regions: *disjoint, contains, inside, meet, equal, covers, covered-by,* and *overlap*. Spatial relationships used in [14] are (absolute) directinal or (absolute) topological. Spatial relationships proposed in our work [16, 17, 18, 19, 21] are more general, can be (absolute) directional, (relative) directional, or (absolute) topological.

In this paper, we intend to formulate a model for Content-based Image Database Systems (CIDBS) and, for the first time, to address the important consistency problem about content-based image indexing and retrieval.

The rest of this paper is organized as follows. In Section 2, we propose the consistency problem on content-based pictorial description. In Section 3, we present a framework for content-based Image Database Systems. We describe major components of Image Database Systems, and demonstrate how they work together to facilitate content-based image indexing and retrieval. In Section 4, we discuss the consistency problem for spatial relationships in a picture. Conclusions and future research are given in Section 5.

2. The Consistency Problem. A real picture is assumed to be associated with some content-based meta-data about that picture, that is, information about objects in the picture and absolute/relative spatial relationships among them. The meta-data about a real picture might contain certain incorrect information about the picture, which is introduced during the image capture stage, possibly because of limitations of existing image-processing algorithms or manual errors. So it is natural for us to ask that: *Is all the meta-data information about a real picture, generated during the image capture stage, true or correct with regard to the original real picture itself?* In general, it is hard for us to answer this question. Instead of answering this general question, we should specifically address the following, *Does the meta-data about a picture contain certain contradictory information across the entire database having all meta-data information about pictures?* We call it the *consistency* problem. The consistency problem is of common interest for the Content-based Image Database Systems, and the Consistency Checking Mechanism is dedicated to verify/maintain consistency of meta-data information about pictures across the entire database.

The consistency problem is crucial for the following reasons. First, because of their limitations, current image-processing algorithms might generate certain minor inaccurate information which might cause inconsistency in the database.Second, existing image-processing algorithms may not be able to detect all objects and their spatial relationships in a picture. The missing information may have to be introduced manually. Sometimes even a picture may be captured completely manually. This manual process may have captured some inaccurate or uncertain information into the meta-data describing the picture contents. This may occur when a picture can not be specified precisely by the user. Third, the consistency problem may happen to an image query. It is possible to have contradictory content-based information description about a picture in an image query the user provides through the User Interface during the image retrieval stage. This may particularly occur with a casual or novice user, and is more likely when more objects are involved in an image query. Consider, for example, a picture description in an image query: there are four objects A, B, C, and D, satisfying the following four absolute spatial relationships, A *left-of* B, C *left-of* D, B *overlaps* C, and A *overlaps* D. Note that A *left-of* D can be deduced from A *left-of* B, B *overlaps* C, and C *left-of* D using the Deductive Rule II in Section 4.1. Then the derived absolute spatial relationship A *left-of* D contradicts the given absolute spatial relationship A *overlaps* D, yielding that the given image query is contradictory. This type of contradictory image queries needs to be detected/corrected during the image query formulation and before they are submitted for the Image Matching component. Otherwise, it will be a disaster if a contradictory image query is submitted for the image-matching in a huge image database, since it will particularly waste time for applications such as law enforcement and criminal investigation, and consume a lot of real-time system resources spending on image-matching where no matching is possible.

Consider, for example, two different pictures which both have one same person inside them. Different people might recognize the same person for his/her age (one important property for a person object) slightly differently. Assume the system divides the age ranges by 10 years each (certain threshold or criterion determined by the application domain), for example, a person with age among 30–39 and a person with age among 40–49, etc. Two users now are required to manually capture these two pictures and then enter the meta-data into database. One user may recognize his/her age among 30–39 for one picture, while another user may recognize his/her age among 40–49 for another picture. This may occur particularly when the person is around 40 years old. Database now contains two contradictory person objects (because their age properties are different), because we know these two person objects are the same. These two person objects should be captured same regardless of whoever does the image capture work. This kind of consistency problem about objects is for lack of certain common background information such as age for a particular person. If we know the person in those two pictures is famous and well-known, and if we maintain his/her background information (assume it is available) in our Content-based Image Database System, then any user will capture the person in those two pictures as a same person object by simply checking the stored background information.

This above example indicates that, one object in a picture should not cause any inconsistency problem among objects in the same picture, while one object in a picture may cause the inconsistency problem with another object in another picture across the entire database.

In contrast, with the consistency problem of objects, one spatial relationship in a picture should not cause any inconsistency problem with another spatial relationship in another picture across the entire database, while one spatial relationship in a picture may cause the inconsistency problem among spatial relationships in the same picture.

For example, the set $\{A$ *left-of* B, B *left-of* $C\}$ is consistent, while the set $\{A$ *overlaps* B, A *outside* $B\}$ is inconsistent. Note that, given a finite set of absolute spatial relationships $\boldsymbol{F}$ for a real picture, the non-contradictory meaning of $\boldsymbol{F}$ with regard to the real picture here coincides with the logical consistent meaning defined in the rule system where, $\boldsymbol{F}$ is said to be consistent if there exists a picture that satisfies all the absolute spatial relationships in $\boldsymbol{F}$. Thus, for a real picture, we are interested in capturing a non-contradictory or consistent finite set of spatial relationships for the picture during the image capture stage.

Note that, it is possible that the set of spatial relationships captured for a picture is consistent but contains certain spatial relationships which are not true/correct with respect to the original picture. Consider, for example, a real picture with three objects A, B, and C. Assume it contains two relative spatial relationships A *left-of* B and A *left-of* C, but not B *left-of* C. Now if the meta-data for this picture contains all these three relative spatial relationships, i.e., A *left-of* B, A *left-of* C and B *left-of* C. This may occur because of the similar reasons specified earlier. It is obvious that these three relative spatial relationships in the meta-data are consistent or non-contradictory but one of them B *left-of* C is not true/correct with respect to the original picture. This example demonstrates the distinction between the consistency problem of meta-data and the correctness problem of meta-data with respect to the original picture. Because the consistency of meta-data is automatically implied by the correctness of meta-data with respect to the original picture. Thus, the latter problem is more general and harder than the former one. This example also demonstrates that the latter problem could not be detected by the consistency checking mechanism, which is dedicated to detect the consistency problem. Sometimes the latter problem might be avoided/corrected by using more accurate image-processing algorithms or with human being's careful help during the image capture stage.

The Consistency Checking Mechanism may also need to be applied to an image query the user provides through the User Interface during the image retrieval stage.

The consistency problem of spatial relationships, in the meta-data information about a real picture, will be further discussed in Section 4.

3. Content-based Image Database Systems.

A Content-based Image Database System (CIDBS) will consist of at least the following seven major components: Image Capture Mechanism, Consistency Checking Mechanism, Image Indexing, Spatial Reasoning, Database, Image Matching, and Human-Computer Interface.

3.1. Image Capture Mechanism.

Image Capture Component consists of image-processing algorithms that are used to capture content-based information about a picture. Given a real picture as an input, Image Capture Component will invoke its image-processing algorithms to automatically generate some content-based meta-data about that picture, that is, information about objects in the picture and absolute/relative spatial relationships among them. With limitations of current image-processing algorithms, this meta-data information is possibly generated semi-automatically by image-processing algorithms with human being's help or completely manually, through Human-Computer Interface.

An object in a real picture corresponds to a significant element of the image. Depending on the application, the significant elements of the image can be pixels, lines, regions, etc. A spatial relationship among objects is relative if it is determined by the position of the centroid of its objects. A spatial relationship is absolute if it is determined by the absolute position of its objects in the image. The following various absolute spatial relationships are of common interest in pictorial databases: *left-of, right-of, in-front-of, behind, above, below, inside, outside,* and *overlaps*. Only the first six spatial operators are considered for relative spatial relationships, since *inside, outside,* and *overlaps* operators are not applicable. Note that the first six spatial operators are directional and the last three spatial operators are topological.

3.2. Consistency Checking Mechanism. After the Image Capture Component processes a picture and generates its content-based meta-data information, this meta-data is sent to the Consistency Checking Component. Consistency Checking Mechanism will then be invoked to verify the consistency of the meta-data information, introduced in Section 2.

3.3. Image Indexing. After the consistency checking of meta-data information about a picture, the Consistency Checking Component will send this meta-data to the Image Indexing component. Image Indexing Component then will invoke its image indexing algorithms to index the image based on this meta-data. We use Iconic Indexing for content-based Image Indexing. Iconic Indexing algorithms generate the 2D string representation for the image as an image index.

For a complete description of our iconic indexing development, the reader may refer to [16, 18, 19, 21].

3.4. Spatial Reasoning. Spatial reasoning is an important component in pictorial retrieval systems. There are two approaches to handling spatial relationships: the well-known one is to use algorithms on which most earlier work such as [3, 4, 9] is based, and the recent one [14] is to construct deductive rules that allow spatial relationships to be deduced. Sistla et al. [14] developed a system of rules $\mathcal{R}$ on reasoning about the following absolute spatial relationships: *left-of, right-of, in-front-of, behind, above, below, inside, outside,* and *overlaps*.

With the system of rules $\mathcal{R}$, the *deduction* problem is to deduce new spatial relationships from a given set $\boldsymbol{F}$ of spatial relationships. More precisely, we are interested in generating all deducible spatial relationships from $\boldsymbol{F}$ (i.e., the maximal set of $\boldsymbol{F}$). The *reduction* problem is to eliminate redundant spatial relationships from a given set $\boldsymbol{F}$ of spatial relationships. More precisely, we are interested in finding all nonredundant spatial relationships in $\boldsymbol{F}$ (i.e., the minimal set of $\boldsymbol{F}$). It was shown in [16, Chapter 2] that both the deduction and reduction problems can be solved by efficient algorithms. We use both the deduction mechanism and reduction mechanism for our proposed Spatial Reasoning component. Both the deduction mechanism and reduction mechanism can be considered to be reverse procedures of each other, and should be invoked by the Image Indexing Component that generates content-based image indexes and the query-processing mechanism in the Image Matching Component that retrieves images by content.

3.5. Database. Database Storage Device in Database component is the repository where physical images, and their image indexes based on meta-data information about them, are stored. Database Management System is needed to handle communications between the outside and Database which contains all meta-data information about images and physical images.

One important question is how to design the Database Component for fast storing and fetching information to/from the Database Storage Device.

3.6. Image Matching. Image Matching is an important component that retrieves images by content. For our proposed framework, the query-processing mechanism in the Image Matching Component will use various 2D string matching algorithms for the pictorial retrieval [3, 4, 18, 21].

The advanced picture-matching algorithms need to be developed for improving the performance of pictorial retrieval. Similarity-based picture-matching algorithms also need to be devised for developing approximate pictorial retrieval that retrieves images, which are most similar to a query image.

3.7. Human-Computer Interface. Human-Computer Interface component is used to communicate between the human-being and the Content-based Image Database System (CIDBS). The User Interface has to be made as simple as possible for a casual or novice user to interact with the CIDBS. An instructive and user-friendly graphical interface environment should be developed to guide the user step by step in performing a picture indexing during the image indexing stage, and in specifying the content-based information about the picture the user has in mind during the image retrieval stage. The interface should then have a knowledge-based support for query formulation.

At this point, we have described the seven major components of a Content-based Image Database System (CIDBS). Now we begin to demonstrate how they work together to facilitate content-based image indexing and retrieval.

Figure 3.1 is the block diagram of a Content-based Image Database System (CIDBS). In this Figure 3.1, the left-side part represents an image indexing flow while the right-side part represents an image retrieval flow.

3.8. Image Indexing Flow. In this Section, we intend to demonstrate how a Content-based Image Database System (CIDBS) performs the image indexing work for a real picture.

For a real picture as an input, the Human-Computer Interface in a CIDBS first sends a request for capturing the picture to the Image Capture Component. The Image Capture Component will then invoke the Image Capture Mechanism to generate the content-based meta-data information about the picture. With limitations of existing image-processing algorithms, this meta-data information is possibly generated semi-automatically by image-processing algorithms with human being's help or completely manually, through the Human-Computer Interface.

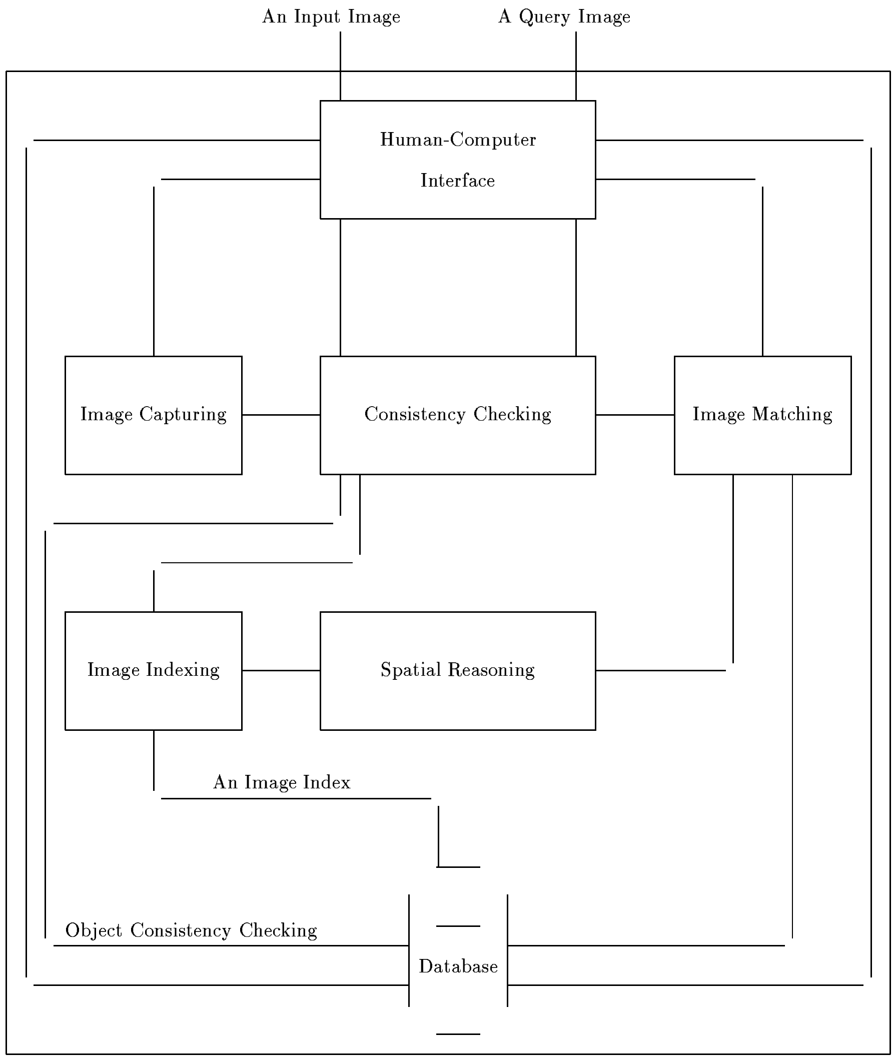

FIG. 3.1. *Block diagram of a content-based image database system.*

After the meta-data about the picture is captured, the Image Capture Component will send this meta-data to the Consistency Checking Component. The Consistency Checking Mechanism will then be invoked to verify the consistency of meta-data across the entire Database (so this step will involve the Database Component). It will perform the consistency checking among only those spatial relationships in this meta-data for the picture, while performing the consistency checking of objects in this meta-data across the entire Database.

If certain inconsistency in the meta-data is detected, the Consistency Checking Mechanism will temporarily stop and this inconsistency will be reported to the human-being for special assistance through the Human-Computer Interface. This possibly requires more accurate image-processing algorithms and/or careful manual help to recapture the picture until the inconsistency in the meta-data about that picture is solved. Certain inconsistency in the meta-data may also be detected and corrected automatically by the Consistency Checking Mechanism if the Consistency Checking Component is equipped with certain special recovery procedures. After the consistency of meta-data is verified, the Consistency Checking Component will send this meta-data to the Image Indexing Component.

After the meta-data about the picture is received, the Image Indexing Component will generate the image index for that picture based on this meta-data. The Deduction and Reduction Mechanism in the Spatial Reasoning component will also be invoked to generate the compact/minimal image index at the Image Indexing stage. Our iconic indexing approach will generate the 2D string representation for the image as an image index.

After an image index for the picture is produced, the Image Indexing Component will send the image index to the Database Component. Database Management System will place the image index (e.g., the 2D string representation for our iconic indexing approach) for the picture and its physical image to the database repository. An *Acknowledgment of Completion* message will be sent from the Database to the Human-Computer Interface to indicate the completion of image indexing for the input picture.

This finishes the image indexing flow.

3.9. Image Retrieval Flow. In this Section, we intend to demonstrate how a Content-based Image Database System (CIDBS) performs the image retrieval work for an image query.

An image query is inputted through the Human-Computer Interface to the Consistency Checking Component. The Consistency Checking Mechanism will be invoked to verify the consistency among spatial relationships in the content-based description of the query image. Note that it is not necessary to check the consistency among objects in the content-based description of the query image. If certain inconsistency among spatial relationships is detected, the error will be reported to the user through the Human-Computer Interface for correction of the image query. After the inconsistency among spatial relationships is resolved, the user may resubmit the modified image query through the Human-Computer Interface.

Note that, using a visual representation of an image query in the Human-Computer Interface sometimes might avoid the inconsistency problem of spatial relationships in the query, since the visual representation automatically preserves the consistency of its spatial relationships. Then it is proposed that the User Interface will have a mechanism to support the consistent query formulation from the visual representation of an image query.

After the consistency among spatial relationships is verified, the image query will be sent to the Image Matching Component. The query-processing mechanism will then be invoked to perform picture-matching between the query image and an image fetched from the Database, based on their content-based meta-data information. This picture-matching process may also invoke the Deduction and Reduction Mechanism in the Spatial Reasoning component to regenerate the information about redundant spatial relationships. Finally, a finite set (possibly null) of images matching the query image will be sent to the Human-Computer Interface.

This finishes the image retrieval flow.

4. The Consistency Problem for Spatial Relationships in a Picture. We [20] have considered the consistency problem for spatial relationships in a picture, and have used the mathematically simple matrix representation approach to present an efficient (i.e., polynomial-time) algorithm for consistency checking of spatial relationships. In this Section, we briefly discuss the main result in [20].

4.1. The Rules for Reasoning about Absolute Spatial Relationships. Here let us present the system of rules $\mathcal{R}$, rules I–VIII, introduced in [14], for reasoning about absolute spatial relationships.

I. (Transitivity of *left-of, above, behind,* **and** *inside***)** For each $x \in \{$*left-of, above, behind, inside*$\}$, we have

$A\ x\ C :: A\ x\ B, B\ x\ C$

II. For each $x \in \{$*left-of, above, behind*$\}$, we have

$A\ x\ D :: A\ x\ B, B\ \mathit{overlaps}\ C, C\ x\ D$

III. For each $x \in \{$*left-of, above, behind, outside*$\}$, we have the following two types of rules.

(a) $A\ x\ C :: A\ \mathit{inside}\ B, B\ x\ C$

(b) $A\ x\ C :: A\ x\ B, C\ \mathit{inside}\ B$

IV. (Symmetry of *overlaps* **and** *outside***)** For each $x \in \{$*overlaps, outside*$\}$, we have

$A\ x\ B :: B\ x\ A$

V. For each $x \in \{$*left-of, above, behind*$\}$, we have

$A\ \mathit{outside}\ B :: A\ x\ B$

VI. $A\ \mathit{overlaps}\ B :: A\ \mathit{inside}\ B$

VII. $A\ \mathit{overlaps}\ B :: C\ \mathit{inside}\ A,\ C\ \mathit{overlaps}\ B$

VIII. $A\ \mathit{inside}\ A ::$

Notice that, in these rules, we exclude the relationship symbols *right-of, below*, and *in-front-of*, since they are duals of *left-of, above*, and *behind*, respectively. They can be handled by simply introducing additional rules that relate them to their duals (see rules IX–XI in [14]).

4.2. Maximal Sets of Spatial Relationships. Without loss of generality, we can assume that, for a set of spatial relationships $\boldsymbol{E}$, the maximal set of $\boldsymbol{E}$ defined below involves only those objects appearing in $\boldsymbol{E}$.

DEFINITION 4.1. *Given a set* $\boldsymbol{E}$ *of spatial relationships, a superset* $\boldsymbol{F} \supseteq \boldsymbol{E}$ *is called a maximal set of* $\boldsymbol{E}$ *under the system of rules* $\mathcal{R}$ *if* (*i*) *each* $r \in \boldsymbol{F}$ *is deducible from* $\boldsymbol{E}$ *using the rules in* $\mathcal{R}$, *and* (*ii*) *no proper superset of* $\boldsymbol{F}$ *satisfies condition* (*i*).

Proposition 4.2 establishes the existence and uniqueness of the maximal set.

PROPOSITION 4.2. *Given a set* $\boldsymbol{E}$ *of spatial relationships, there exists exactly one maximal set* $\boldsymbol{F}$ *of* $\boldsymbol{E}$ *under* $\mathcal{R}$.

Proof: For each possible relationship AxB, where objects A and B appear in $\boldsymbol{E}$ and $x \in$ {*left-of, above, behind, inside, outside, overlaps*}, we put it into $\boldsymbol{F}$ if and only if it is deducible from $\boldsymbol{E}$ under $\mathcal{R}$. Then $\boldsymbol{F}$ satisfies the required properties. □

Proposition 4.3 establishes the close connection of consistency between a set $\boldsymbol{E}$ of spatial relationships and the maximal set of $\boldsymbol{E}$ under $\mathcal{R}$.

PROPOSITION 4.3. *Given a set $\boldsymbol{E}$ of spatial relationships, $\boldsymbol{E}$ is consistent if and only if the maximal set of $\boldsymbol{E}$ under $\mathcal{R}$ is consistent.*

Proof: It is obvious that $\boldsymbol{E}$ is consistent if the maximal set of $\boldsymbol{E}$ under $\mathcal{R}$ is consistent. Conversely, if $\boldsymbol{E}$ is consistent, then the maximal set of $\boldsymbol{E}$ under $\mathcal{R}$ must be consistent, since the set of rules $\mathcal{R}$ is sound for two-dimensional and three-dimensional pictures. □

4.3. Directed Graph and Transitive Closure.

A *directed graph* (or *digraph*) G on the set of vertices $V = \{v_1, v_2, \ldots, v_n\}$ is a subset of $V \times V$, the members of G being called arcs. A graph is called *acyclic* if and only if it contains no cycles or loops. A graph G is said to be *transitive* if, for every pair of vertices u and v, not necessarily distinct, $(u, v) \in G$ whenever there is a directed path in G from u to v. The *transitive closure* of G, denoted by G^T, is the least subset of $V \times V$ that contains G and is transitive.

The following fact 4.4 is stated in [16, Chapter 2].

FACT 4.4. *It takes the same equivalent time complexity to compute the transitive reduction of a graph, or to compute the transitive closure of a graph, or to perform Boolean matrix multiplication.*

Notice that we can easily compute the transitive closure of a graph G using efficient standard algorithms with time complexity $O(n^3)$ and space complexity $O(n^2)$, where n is the total number of vertices in G (see, e.g., [1]).

We assume that a graph G is represented by its adjacency matrix M. For simplicity, sometimes we identify a graph G with its adjacency matrix M, and also use M^T to denote adjacency matrix of the transitive closure G^T. For a set $\boldsymbol{E}$ of "x" relationships, where $x \in$ {*left-of, above, behind, inside, outside, overlaps*}, we also associate it with its adjacency matrix, the matrix with a 1 in row i and column j if the relationship "(the ith object) x (the jth object)" is in $\boldsymbol{E}$ and a 0 there otherwise, and identify $\boldsymbol{E}$ with its adjacency matrix. However, the intended meaning will be clear from the context.

DEFINITION 4.5. *Let $\boldsymbol{SR}$ be a set of spatial relationships and x be a relationship symbol chosen from {left-of, above, behind, inside}. A dependency graph derived by x (and $\boldsymbol{SR}$ implicitly) is defined as a directed graph G_x, its vertex set is the set of all objects involved in $\boldsymbol{SR}$, and an arc (A, B) is in G_x if and only if AxB is in $\boldsymbol{SR}$.*

Note that, from Rule VIII, any relationship A *inside* A is always redundant for any involved object A and thus could be deleted from $\boldsymbol{SR}$ immediately. Further, all of them must be added into the maximal set of $\boldsymbol{SR}$ when we generate it. Therefore, we can assume that the derived dependency graph G_{inside} does not include any arc (A, A). Now it is obvious that four derived dependency graphs, $G_{\textit{left-of}}, G_{above}, G_{behind}$, and G_{inside} are acyclic for any consistent set $\boldsymbol{SR}$ of spatial relationships.

Let $\boldsymbol{E}$ be a set of spatial relationships and x be a relationship symbol. We will use $\boldsymbol{E}^x$ to denote the subset of all "x" relationships that are in $\boldsymbol{E}$.

4.4. Consistency Checking Algorithms. Now we begin to present the algorithms for consistency checking of spatial relationships.

The 2D string approach for Iconic Indexing developed by Chang et al. [4] considers only relative spatial relationships among objects, that is, it considers only relative spatial relationships involving *left-of, above,* and *behind* (for three-dimensional pictures only). Our proposed GC-2D string approach [18, 21] considers both relative and absolute spatial relationships. Note that there are no interactions among *left-of, above,* and *behind* relationships. Let us consider a set of only relative spatial relationships $\boldsymbol{E}$. We can detect the consistency of $\boldsymbol{E}$ in the following way. First, check whether $\boldsymbol{E}$ contains one self-contradictory relationship AxA for some object A involved in $\boldsymbol{E}$ and $x \in \{$*left-of, above, behind*$\}$. It is obvious that $\boldsymbol{E}$ is inconsistent if $\boldsymbol{E}$ contains one self-contradictory relationship AxA. Now if $\boldsymbol{E}$ doesn't contain any self-contradictory relationship AxA, then compute the transitive closure G_x^T of G_x for each $x \in \{$*left-of, above, behind*$\}$, where G_x is the dependency graph derived by x (and $\boldsymbol{E}$). It is clear that $\boldsymbol{E}$ is inconsistent if and only if G_x is cyclic, if and only if G_x^T contains a loop (A, A) for some object A involved in $\boldsymbol{E}$, if and only if G_x^T contains two arcs (A, B) and (B, A) for two different objects A and B involved in $\boldsymbol{E}$, where x is either *left-of, above,* or *behind.* Note that the required time complexity is dominated by applying the transitive closure algorithm.

Let $\boldsymbol{E}$ be a set of spatial relationships among objects in the content-based metadata information about a picture. Note that *inside, outside,* and *overlaps* operators are not applicable for relative spatial relationships, and an absolute spatial relationship involving *left-of, above,* and *behind* is also true as a corresponding relative spatial relationship. Thus, in order to verify the consistency of $\boldsymbol{E}$, we need to do the following two consistency checkings. One is to check the consistency of the set of those absolute spatial relationships in $\boldsymbol{E}$. The rest of this paper is devoted to this. The other is to check the consistency of the union set of relative spatial relationships already in $\boldsymbol{E}$ and those corresponding relative spatial relationships which, as absolute spatial relationships, are in the maximal set of $\boldsymbol{E}$ under $\mathcal{R}$. This can be done efficiently as shown above.

Similar to the consistency checking of only relative spatial relationships as shown above, we clearly have an efficient procedure for detecting the consistency of relative and/or absolute spatial relationships involving only *left-of, above,* and *behind.*

From now on, let us consider only absolute spatial relationships in the meta-data information about a picture.

Given two different objects A and B, we say *A and B have a pair of contradictory spatial relationships* if at least one of the following six conditions holds:

1. A *inside* B and B *inside* A.
2. $A\ x\ B$ and $B\ x\ A$ for some $x \in \{$*left-of, above, behind*$\}$.
3. A *outside* B and A *overlaps* B.
4. A *overlaps* B and $A\ x\ B$ for some $x \in \{$*left-of, above, behind*$\}$.

5. A *inside* B and $A\ x\ B$ for some $x \in \{$*left-of, above, behind*$\}$.
6. A *outside* B and A *inside* B.

Each condition is respectively called type-i, where $1 \leq i \leq 6$. (Note that these are all possible cases of contradictory pairs.)

Given a set $\boldsymbol{E}$ of absolute spatial relationships, we say $\boldsymbol{E}$ *contains one pair of contradictory spatial relationships* if there exist two objects A and B having a pair of contradictory spatial relationships in $\boldsymbol{E}$. We say $\boldsymbol{E}$ *contains a self-contradictory spatial relationship* if there exists one object A such that $\boldsymbol{E}$ contains either one of the following spatial relationships: A *left-of* A, A *above* A, A *behind* A, and A *outside* A.

It is obvious that any set $\boldsymbol{E}$ of absolute spatial relationships is inconsistent if $\boldsymbol{E}$ contains one pair of contradictory spatial relationships. It is also obvious that $\boldsymbol{E}$ is inconsistent if $\boldsymbol{E}$ contains a self-contradictory spatial relationship.

Given a set $\boldsymbol{SR}$ of absolute spatial relationships, we will follow the process of generating the maximal set of $\boldsymbol{SR}$ under $\mathcal{R}$ (see [16, Chapter 2]), to detect whether the maximal set of $\boldsymbol{SR}$ under $\mathcal{R}$ contains one pair of contradictory spatial relationships. And if the maximal set of $\boldsymbol{SR}$ under $\mathcal{R}$ doesn't contain any pair of contradictory spatial relationships, our proposed procedure will finally generate the maximal set of $\boldsymbol{SR}$ under $\mathcal{R}$.

Before the beginning of detection algorithm, first check whether $\boldsymbol{SR}$ contains a self-contradictory spatial relationship. If $\boldsymbol{SR}$ contains the spatial relationship $A\ x\ A$ for some object A involved in $\boldsymbol{SR}$ and $x \in \{$*left-of, above, behind, outside*$\}$, then $\boldsymbol{SR}$ is inconsistent. Also note that, from Rules VIII and VI, any relationships A *inside* A and A *overlaps* A are always redundant for any involved object A and thus could be deleted from $\boldsymbol{SR}$ immediately. Therefore, we can assume that $\boldsymbol{SR}$ does not contain $A\ x\ A$ for $x \in \{$*left-of, above, behind, inside, outside, overlaps*$\}$.

The correctness of the following detection algorithm, for checking whether the maximal set of $\boldsymbol{SR}$ under $\mathcal{R}$ contains one pair of contradictory spatial relationships, can be found in [20]. In this algorithm, addition '+', subtraction '−', and multiplication '∗' denote Boolean matrix addition, subtraction, and multiplication, respectively. The following algorithm assumes that we already have efficient standard algorithm for computing the transitive closure G^T of a given directed graph G. The algorithm for computing G^T of G is represented by $\mathbf{TranC}(G, G^T)$, where G is a directed graph as input and G^T is a directed graph as output of **TranC**. For each $x \in \{$*left-of, above, behind, inside, outside, overlaps*$\}$, all sets of "x" relationships are identified with their associated adjacency matrices. Let I be an $n \times n$ identity matrix, where n is the number of all objects involved in $\boldsymbol{SR}$. Then I can denote either the set $\{A$ *inside* $A \mid A$ is any involved object$\}$ if the intended relationship is *inside* or the set $\{A$ *overlaps* $A \mid A$ is any involved object$\}$ if the intended relationship is *overlaps*. We will also use M' to denote the transpose matrix of a given matrix M.

Algorithm. *Detect whether the maximal set of a given set of absolute spatial relationships contains one pair of contradictory spatial relationships.*

Input: a given set $\boldsymbol{SR}$ of absolute spatial relationships.

```
Output: NO if the maximal set of $\boldsymbol{SR}$ doesn't contain any pair of contradictory spatial
        relationships, and the maximal set of $\boldsymbol{SR}$ is also produced;
        YES, otherwise.
/* Assume $\boldsymbol{SR}$ doesn't contain any $AxA$ for $x \in \{$left-of, above, behind, inside, outside,
    overlaps$\}$ */
Step (0). Check whether $\boldsymbol{SR}$ contains one pair of contradictory spatial relationships.
          If YES, halt. Otherwise, continue.
Step (1). Generate inside relationships
          /* $G_{inside}$ denotes the dependency graph derived by inside and $\boldsymbol{SR}$ */
          (1a). Compute $\boldsymbol{INSIDE}^{+} = G^{T}_{inside}$ by calling algorithm TranC($G_{inside}$,
                $\boldsymbol{INSIDE}^{+}$). Check whether $\boldsymbol{INSIDE}^{+}$ contains a loop. If YES, halt.
                Otherwise, continue.
          (1b). $\boldsymbol{INSIDE} = \boldsymbol{INSIDE}^{+} + I$.
Step (2). Generate overlaps relationships
          /* $\boldsymbol{O}_0 = \boldsymbol{SR}^{overlaps}$ denotes the subset of all overlaps relationships in $\boldsymbol{SR}$ */
          (2a). $\boldsymbol{O}_1 = \boldsymbol{O}_0 + \boldsymbol{O}'_0$, $\boldsymbol{O}_2 = \boldsymbol{INSIDE} + \boldsymbol{INSIDE}'$, and set $M_{12} = \boldsymbol{O}_1 + \boldsymbol{O}_2$;
          /* $M_{in}$ is the adjacency matrix of $\boldsymbol{INSIDE}^{+}$ */
          (2b). $\boldsymbol{O}_3 = M'_{in} * M_{12}$, $\boldsymbol{O}_4 = \boldsymbol{O}'_3$ and $\boldsymbol{O}_5 = \boldsymbol{O}_3 * M_{in}$;
          (2c). $\boldsymbol{OVERLAPS} = M_{12} + \boldsymbol{O}_3 + \boldsymbol{O}_4 + \boldsymbol{O}_5$, and set
                $\boldsymbol{OVERLAPS}^{+} = \boldsymbol{OVERLAPS} - I$.
Step (3). Generate left-of, above, and behind relationships
          /* $G_x$ denotes the dependency graph derived by $x$ and $\boldsymbol{SR}$ */
          For each $x \in \{$left-of, above, behind$\}$, go through (3a)–(3c):
          (3a). Compute $G^{T}_{x}$ by calling algorithm TranC($G_x, G^{T}_{x}$).
                Check whether $G^{T}_{x}$ contains a loop. If YES, halt. Otherwise, continue.
          /* $M_{ov}$ is the adjacency matrix of $\boldsymbol{OVERLAPS}^{+}$ */
                Check whether $G^{T}_{x} \cup M_{ov}$ contains one pair of type-4 contradictory spatial
                relationships.
                If YES, halt. Otherwise, continue.
          (3b). $M_x = G^{T}_{x} * M_{ov}$, and compute $M^{T}_{x}$ by calling algorithm TranC($M_x, M^{T}_{x}$),
                then set $M_{x2} = G^{T}_{x} + M^{T}_{x} * G^{T}_{x}$.
                Check whether $M_{x2}$ contains a loop. If YES, halt. Otherwise, continue.
                Check whether $M_{x2} \cup M_{ov}$ contains one pair of type-4 contradictory spatial
                relationships.
                If YES, halt. Otherwise, continue.
          (3c). $\boldsymbol{MAX}(x) = M_{x2} + M_{in} * M_{x2} + M_{x2} * M'_{in} + M_{in} * M_{x2} * M'_{in}$.
                Check whether $\boldsymbol{MAX}(x) \cup M_{ov}$ contains one pair of type-4 contradictory
                relationships.
                If YES, halt. Otherwise, continue.
Step (4). Generate outside relationships
          /* $U_0 = \boldsymbol{SR}^{outside}$ denotes the subset of all outside relationships in $\boldsymbol{SR}$ */
          (4a). $\boldsymbol{U}_1 = \boldsymbol{U}_0 + \boldsymbol{U}'_0$, $\boldsymbol{U}_2 = \boldsymbol{MAX}(\textit{left-of}) + \boldsymbol{MAX}(\textit{above}) + \boldsymbol{MAX}(\textit{behind})$
                and reset $\boldsymbol{U}_2 = \boldsymbol{U}_2 + \boldsymbol{U}'_2$, then $U_{12} = \boldsymbol{U}_1 + \boldsymbol{U}_2$.
                Check whether $U_{12} \cup M_{ov}$ contains one pair of type-3 contradictory spatial
                relationships.
                If YES, halt. Otherwise, continue.
          (4b). $\boldsymbol{U}_3 = M_{in} * U_{12}$, $\boldsymbol{U}_4 = \boldsymbol{U}'_3$ and $\boldsymbol{U}_5 = M_{in} * \boldsymbol{U}_4$;
```

```
(4c). $\boldsymbol{OUTSIDE} = U_{12} + \boldsymbol{U}_3 + \boldsymbol{U}_4 + \boldsymbol{U}_5$.
      Check whether $\boldsymbol{OUTSIDE} \cup M_{ov}$ contains one pair of type-3
      contradictory relationships.
      If YES, halt and output YES. Otherwise, halt and output NO.
/* End of the detection algorithm */
```

Note that if the above detection algorithm outputs YES, we are certain that $\boldsymbol{SR}$ is inconsistent. But it does not tell us exactly the questionable relationship(s) in $\boldsymbol{SR}$ causing the inconsistency. Consider, for example, $\boldsymbol{SR} = \{A \textit{ above } B, B \textit{ above } C, C \textit{ above } A\}$. Then $\boldsymbol{SR}$ is inconsistent. Deleting either one of the three relationships in $\boldsymbol{SR}$ will make the left two relationships in $\boldsymbol{SR}$ consistent. Thus, the user may be required to help resolve the inconsistency of $\boldsymbol{SR}$ when the inconsistency of $\boldsymbol{SR}$ is detected and reported to the Human-Computer Interface.

The above detection algorithm yields the following theorem and corollary [20].

THEOREM 4.6. *There exists an efficient algorithm to detect whether, given a set* $\boldsymbol{SR}$ *of absolute spatial relationships, the maximal set of* $\boldsymbol{SR}$ *under* $\mathcal{R}$ *contains one pair of contradictory spatial relationships. The time required by it is at most a constant factor more than the time to compute the transitive reduction of a graph or to compute the transitive closure of a graph or to perform Boolean matrix multiplication, and thus is always bounded by time complexity* $O(n^3)$ *(and space complexity* $O(n^2)$*), where* n *is the number of all involved objects.*

COROLLARY 4.7. *The above detection algorithm can completely answer whether a given set of three-dimensional absolute spatial relationships is consistent.*

For two-dimensional pictures, we will not have the relationship symbol *behind* and the rules referring to it in $\mathcal{R}$. Similarly, we can use the above algorithm (discarding those computations involving *behind* relationships) to detect whether, given a set $\boldsymbol{SR}$ of two-dimensional absolute spatial relationships, the maximal set of $\boldsymbol{SR}$ under $\mathcal{R}$ contains one pair of contradictory spatial relationships. However, our proposed algorithm might fail to detect inconsistency of the description of absolute spatial relationships (involving *inside, outside,* and *overlaps*) for certain planar pictures, while at least checking their maximal sets under $\mathcal{R}$ don't contain any pair of contradictory spatial relationships.

The detailed algorithm given above can be directly programmed into executable computer codes.

5. Conclusions and Future Research. In this paper we have proposed the consistency problem on content-based pictorial description in pictorial database systems. Then we briefly discuss the main result, shown in [20], on the consistency problem for spatial relationships in a picture. We have also suggested a framework for Content-based Image Database Systems (CIDBS). Future research is required to further investigate this CIDBS model for facilitating fast image indexing and retrieval.

While the data consistency problem has been well addressed in traditional database systems, the consistency problem about content-based multimedia indexing and retrieval needs to be investigated. The consistency problem will also arise when multimedia data sources are merged. Our proposed approach for the image database case is an attempt to begin addressing this important issue.

REFERENCES

[1] A. V. AHO, J. E. HOPCROFT AND J. D. ULLMAN, *Data Structures and Algorithms,* Addison-Wesley, Reading, Massachusetts, 1983.

[2] C. C. CHANG AND S. Y. LEE, *Retrieval of Similar Pictures on Pictorial Databases*, Pattern Recognition, 24:7(1991), pp. 675–680.

[3] S.-K. CHANG AND E. JUNGERT, *Symbolic Projection for Image Information Retrieval and Spatial Reasoning,* Academic Press, 1996.

[4] S.-K. CHANG, Q. Y. SHI AND C. W. YAN, *Iconic Indexing by 2-D Strings*, IEEE Trans. Pattern Anal. Machine Intell., 9:3(1987), pp. 413–428.

[5] M. J. EGENHOFER AND R. D. FRANZASA, *Point-set Topological Spatial Relations*, J. Geogr. Inf. Syst., 5:2(1991), pp. 161–174.

[6] M. J. EGENHOFER AND R. D. FRANZASA, *On the Equivalence of Topological Relations*, J. Geogr. Inf. Syst., 9:2(1995), pp. 133–152.

[7] E. A. EL-KWAE AND M. R. KABUKA, *A Robust Framework for Content-based Retrieval by Spatial Similarity in Image Databases*, ACM Transactions on Information Systems, 17:2(1999), pp. 174–198.

[8] V. N. GUDIVADA, *On Spatial Similarity Measures for Multimedia Applications*, Proceedings of SPIE – Storage and Retrieval for Still Images and Video Databases III, (SPIE, Bellingham, WA), San Jose, CA, 363–372, Feb. 9–10, 1995.

[9] V. N. GUDIVADA AND V. V. RAGHAVAN, *Design and Evaluation of Algorithms for Image Retrieval by Spatial Similarity*, ACM Transactions on Information Systems, 13:2(1995), pp. 115–144.

[10] V. N. GUDIVADA AND V. V. RAGHAVAN, *Content-based Image Retrieval Systems*, IEEE Computer, September, 1995, pp. 18–22.

[11] T.-Y. HOU, P. LUI AND M. Y. CHUI, *A Content-based Indexing Technique Using Relative Geometry Features*, Proceedings of SPIE – Image Storage and Retrieval Systems, the International Society for Optical Engineering, Vol. 1662, 1992.

[12] D. PAPADIAS, T. SELLIS, Y. THEODORIDIS AND M. J. EGENHOFER, *Topological Relations in the World of Minimum Bounding Rectangles: A Study with R-trees*, Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data (SIGMOD '95), (M. Carey and D. Schneider, eds. ACM Press, New York, NY), San Jose, CA, 92–103, May 23–25, 1995.

[13] A. PENTLAND, R. W. PICARD AND S. SCLROFF, *Photobook: Content-based Manipulation of Image Databases*, Proceedings of SPIE – Storage and Retrieval for Image and Video Database II, (SPIE, Bellingham, WA), San Jose, CA, pp. 34–47, Feb. 6–10, 1994.

[14] A. P. SISTLA, C. YU AND R. HADDAD, *Reasoning about Spatial Relationships in Picture Retrieval Systems*, Proceedings of the 20th International Conference on Very Large Databases, Santiago, Chile, pp. 570–581, 1994.

[15] S. L. TANIMOTO, *An Iconic Symbolic Data Structuring Scheme*, Pattern Recogn. and Artificial Intell., Academic Press, New York, pp. 452–471, 1976.

[16] Q.-L. ZHANG, *A Unified Framework for Iconic Indexing of Spatial Relationships in Image Databases*, Ph.D. Thesis, Department of Mathematics, Statistics, and Computer Science, Univ. of Illinois, Chicago, 1996.

[17] Q.-L. ZHANG, S.-K. CHANG AND S. S.-T. YAU, *A Unified Approach to Indexing Images in Image Databases*, Proceedings of the First International Workshop on Image Databases and Multi-Media Search, Amsterdam, The Netherlands, pp. 99–106, August 22–23, 1996.

[18] Q.-L. ZHANG, S.-K. CHANG AND S. S.-T. YAU, *A Unified Approach to Iconic Indexing, Retrieval, and Maintenance of Spatial Relationships in Image Databases*, Journal of Visual Communication and Image Representation, Special Issue on Indexing, Storage, Retrieval

and Browsing of Images and Video, Academic Press, 7:4(1996), pp. 307–324.

[19] Q.-L. Zhang, S.-K. Chang and S. S.-T. Yau, *An Experimental Result in Image Indexing using GEP-2D Strings*, Image Databases and Multimedia Search (A. Smeulders and R. Jain, eds.), World Scientific Pub. Co., pp. 127–146, 1997.

[20] Q.-L. Zhang, S.-K. Chang and S. S.-T. Yau, *On Consistency Checking of Spatial Relationships in Content-based Image Database Systems*, Preprint, 2000.

[21] Q.-L. Zhang and S. S.-T. Yau, *A New Iconic Indexing for 2D and 3D Scenes*, Proceedings of the 2nd Chinese World Congress on Intelligent Control and Intelligent Automation (CWCICIA 97), Xi'an, P. R. China, pp. 1667–1672, June 23–27, 1997.