COMMUNICATIONS IN INFORMATION AND SYSTEMS
Vol. 4, No. 3, pp. 211-234, 2004

002

ON COMPLETENESS OF REASONING ABOUT PLANAR SPATIAL RELATIONSHIPS IN PICTORIAL RETRIEVAL SYSTEMS*

STEPHEN S.-T. YAU† AND QING-LONG ZHANG‡

Abstract. In this paper we consider the completeness problem of reasoning about planar spatial relationships in pictorial retrieval systems. We define a large class of two-dimensional scenes, the extended pseudo-symbolic pictures. The existing rule system $\mathcal{R}$ is proved to be complete for (extended) pseudo-symbolic pictures. We also introduce a new iconic indexing, the (extended) pseudo-2D string representation, for them. The (extended) pseudo-2D string has the good properties of the 2D string. It is unambiguous, like the augmented 2D string, and has a compact form suitable for image retrieval. We then present efficient algorithms to determine whether a given planar picture is (extended) pseudo-symbolic or not, and if it is, these algorithms also return its (extended) pseudo-2D string representation. Picture retrieval by (extended) pseudo-2D strings is also discussed.

1. Introduction. Image database systems have been much studied over the past 20 years. One of the most important problems in the design of image database systems is how images are stored in the image databases [5, 6, 9, 11, 12, 24]. 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 [25] suggested the use of picture icons as picture indexes, thus introducing the concept of iconic indexing. Subsequently, Chang et al. [11] developed the concept of iconic indexing by introducing the 2D string representation of the image. The 2D string approach is based on the idea that the spatial knowledge contained in a real picture can be suitably represented by a symbolic picture (i.e., a matrix of symbols) where every symbol corresponds to a significant element of the image. The position of a symbol in the grid corresponds to the position of the centroid of the represented significant element. Depending on the application, the significant elements of the image can be pixels, lines, regions, and objects, etc. A 2D string representing a symbolic picture is derived from the picture by orthogonally projecting its symbols by columns and by rows. This approach gives an efficient and natural way to construct iconic indexes for two-dimensional pictures. With the 2D string approach, the problem of pictorial information retrieval for 2D pictures becomes a problem of 2D

*The main results of this paper are from Chapter 3, the second author's Ph.D. Thesis, Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago.

†Fellow, IEEE, 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. Tel., Fax: 312-996-3065. E-mail: yau@uic.edu

‡Member, IEEE. 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. Current E-mail contact: zhangq@math.uic.edu

string subsequence matching [11, 21]. Since then, the 2D string approach has been studied further in the literature (see, e.g., [2, 13, 26]). Some forms of extensions of the 2D string approach can be found in [7, 8, 10, 19, 20]. For three-dimensional pictures, representations such as the octree [18, 22] were developed, and an extension of the 2D string to three dimensions was suggested in [12], and a unified approach to iconic indexing for 2D and 3D pictures was then proposed by Costagliola et al. [14]. Other methods on image representation and retrieval can be found in the literature (see, e.g., [4, 16, 17]).

Sistla et al. [23] developed a rule system $\mathcal{R}$ for reasoning about spatial relationships in picture retrieval systems. In their paper, a real picture is assumed to be associated with some content-based meta-data about that picture, that is, information about the objects in the picture, their properties, and the spatial or nonspatial relationships among them. This meta-data information is generated and stored in the database. Sistla et al. considered various spatial relationships: *left-of, right-of, in-front-of, behind, above, below, inside, outside,* and *overlaps*. For the first time, they presented a set of rules $\mathcal{R}$ that can be used to deduce new relationships from a given set of relationships. These rules are sound, and $\mathcal{R}$ is complete for 3D pictures. However, they presented a counterexample to show that $\mathcal{R}$ is incomplete for 2D pictures.

There are three obvious distinctions between the work of Sistla et al. [23] and the work such as [11, 15, 16] on handling spatial relationships. First, the sets of spatial operators are not identical. For example, the operators *overlaps, inside,* and *outside* in [23] are not present in the other approaches. Second, the operators in [23] are defined by absolute spatial relationships among objects, while the operators in the other approaches are defined by relative spatial relationships among objects. For example, consider 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 [11] 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 [23] 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 [11] than in [23] in the sense that "A *left-of* B" is true in [11] whenever it is true in [23], and "A *left-of* B" is not necessarily true in [23] when it is true in [11]. Third, the approach to handling spatial relationships in [23] is to construct rules that allow spatial relationships to be deduced, but the other studies are based mostly on algorithms.

In this paper we address the completeness problem, proposed in [23], of reasoning about planar spatial relationships in pictorial retrieval systems. We could have two ways to attack this two-dimensional completeness problem. The first way is to add some new deductive rules to the existing rule system $\mathcal{R}$ so as to make the extended rule system complete for planar pictures that satisfy the connectedness property (this property prevents an object in a picture from having disjoint parts). Because there might exist infinitely many types of counterexamples that make $\mathcal{R}$ incomplete, we might need to add infinitely many new rules to $\mathcal{R}$, but clearly this is inefficient and impractical. Thus, more generally, we wish to ask: *Does there exist an algorithm such that the existing rule system $\mathcal{R}$ along with this algorithm is complete for planar pictures?* It seems to us that this algorithm should be inefficient (i.e., not polynomial-time) if it exists. The second way is to find the set of all planar pictures for which $\mathcal{R}$ is complete. More precisely, we wish to identify those properties $\boldsymbol{P}$ such that $\mathcal{R}$ is complete for planar pictures that satisfy $\boldsymbol{P}$. However, such a property $\boldsymbol{P}$ is likely to be inefficiently decidable; that is, there might not exist an efficient algorithm to determine whether a given planar picture satisfies the property $\boldsymbol{P}$. Thus, we require that these properties $\boldsymbol{P}$ be efficiently decidable. One can see that these two ways are closely related. In this paper we consider the second way and present two efficiently decidable properties (more precisely, classes of planar pictures): *pseudo-symbolic pictures* and *extended pseudo-symbolic pictures*. The class of extended pseudo-symbolic pictures extends the class of pseudo-symbolic pictures. The extended pseudo-symbolic pictures represent a reasonably large class of planar pictures that have nice properties as symbolic pictures, and are very useful for representing planar pictures in many domain-dependent applications.

The rest of this paper is organized as follows. In Section 2, we present the definitions of symbolic picture and 2D string [11]. We also include the system of rules $\mathcal{R}$ [23] for two dimensions here. In Section 3, we modify the notion of symbolic pictures to the notion of *pseudo-symbolic pictures* in our situation. We introduce *pseudo-2D strings*, which in the form can be considered as a variation of the 2D string representation for symbolic pictures, to represent pseudo-symbolic pictures. The completeness property of the existing rule system $\mathcal{R}$ for planar pseudo-symbolic pictures is also shown in this section. In Section 4, we extend the notion of pseudo-symbolic pictures to the notion of *extended pseudo-symbolic pictures*, and we introduce an "almost" 2D string representation, called the *extended pseudo-2D string*, to represent extended pseudo-symbolic pictures. We then show the completeness property of the existing rule system $\mathcal{R}$ for planar extended pseudo-symbolic pictures. In Section 5, we propose efficient algorithms to determine whether a given planar picture is a pseudo-symbolic (or an extended pseudo-symbolic) picture, and if it is, these algorithms also return its corresponding pseudo-2D (or extended pseudo-2D) string representation. In Section 6, we discuss planar picture retrieval by (extended) pseudo-2D strings. Conclusions are given in Section 7.

2. Definitions and Facts. We first recall the definitions of symbolic picture and 2D string given in [11].

2.1. Symboic Pictures. We use $<_r$ and $<_a$, respectively, to represent relative and absolute spatial relationships involving *left-of* and *below*, as mentioned in the

Introduction. For simplicity, sometimes we identify $<_r$ with $<_a$ and just use $<$. However, the intended meaning will be clear from the context.

Definition 2.1. *Given a set V of symbols, a symbolic picture f over V is an $m \times n$ matrix, in which each slot of the matrix is assigned a* (*possibly empty*) *subset of V.*

Definition 2.2. *A* (*reduced*) *2D string (u, v) over V is defined as a pair of strings*

$$(x_1 y_1 x_2 y_2 \cdots y_{t-1} x_t, x_{p(1)} z_1 x_{p(2)} z_2 \cdots z_{t-1} x_{p(t)}),$$

where $x_i \in V$ and y_i, z_i are either $<_r$ or null symbols and $p : \{1, 2, \ldots, t\} \longrightarrow \{1, 2, \ldots, t\}$ is a permutation function.

A 2D string representing a symbolic picture is derived from the picture by orthogonally projecting its symbols by columns and by rows. The symbol $<_r$ is used to separate nonempty columns and rows. Empty columns and rows are ignored. Because of the possibility of multiple occurrences of a given symbol, this representation may cause ambiguity when a symbolic picture is reconstructed from its 2D string representation. The characterizations of ambiguous pictures under different 2D string representations can be found in [11, 13]. The ambiguity problem for the whole class of symbolic pictures can be solved by adding the permutation function to the 2D string (augmented 2D string). The following definition of non-redundant 2D string is given by Costagliola et al. [13].

Definition 2.3. *Let f be a symbolic picture and (u', v') be its reduced 2D string representation. Each substring between two consecutive $<_r$'s, or before the first or after the last $<_r$'s of u'* (v', *respectively*) *is called a local substring of u'* (v', *respectively*). *The non-redundant 2D string representation of f is a pair (u, v), where u* (v, *respectively*) *is obtained from u'* (v', *respectively*) *by replacing multiple occurrences of a same symbol in each local substring of u'* (v', *respectively*) *by exactly one occurrence of the symbol.*

Figure 1 shows an image and the symbolic picture f representing it. The set of symbols is $V = \{$c, r, s, t$\}$, where c, r, s, and t correspond to the objects *circle, rectangle, square,* and *right triangle,* respectively. The symbolic picture f can be represented by the 2D string (ct $<$ cr $<$ s, cc $<$ rs $<$ t), where $p = 13452$, by the augmented 2D string (ct $<$ cr $<$ s, cc $<$ rs $<$ t, 13452), and by the non-redundant 2D string (ct $<$ cr $<$ s, c $<$ rs $<$ t).

2.2. A System of Rules $\mathcal{R}$. Now we first recall the semantic definitions of absolute spatial relationships, introduced in [23], for two-dimensional pictures.

It is assumed that a 2-dimensional picture p consists of finitely many objects and each object in p corresponds to a nonempty set of points in the 2-dimensional Cartesian space, where each point is given by its two x- and y-coordinates. Let p be a picture in which objects A and B are contained. Now we define when p satisfies the

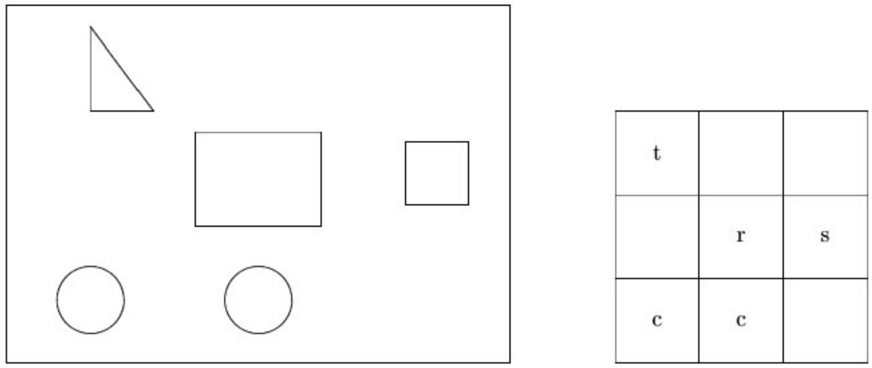


FIG. 1. *An image and its symbolic representation.*

following basic absolute spatial relationships: *left-of, right-of, below, above, inside, outside,* and *overlaps*.

- p satisfies the relationship A *left-of* B, stating that A is to the left of B in the picture p, iff the x-coordinate of each point in $p(A)$ is less than the x-coordinate of each point in $p(B)$.
- p satisfies the relationship A *below* B, stating that A is below B in the picture p, iff the y-coordinate of each point in $p(A)$ is less than the y-coordinate of each point in $p(B)$.
- p satisfies the relationship A *inside* B, stating that A is inside B in the picture p, iff $p(A) \subseteq p(B)$.
- p satisfies the relationship A *outside* B, stating that A is outside B in the picture p, iff $p(A) \cap p(B) = \emptyset$.
- p satisfies the relationship A *overlaps* B, stating that A overlaps B in the picture p, iff $p(A) \cap p(B) \neq \emptyset$.

The semantics of spatial relationship symbols *right-of* and *above* are defined similarly. Notice that the relationship symbols *right-of* and *above* are duals of *left-of* and *below*, respectively.

A finite set of spatial relationships $\boldsymbol{F}$ is said to be consistent if there exists a picture satisfying all the relationships in $\boldsymbol{F}$. A relationship r is said to be implied by a finite set of spatial relationships $\boldsymbol{F}$ if every picture satisfying all the relationships in $\boldsymbol{F}$ also satisfies the relationship r.

A deductive rule is in the following form

$$r :: r_1, r_2, \ldots, r_k$$

where r and r_i $(1 \leq i \leq k)$ are relationships and $k \geq 0$. The relationship r and the list of relationships $r_1, r_2, \ldots, r_k$ are called the *head* and the *body* of the rule, respectively.

A relationship r is said to be deducible in one step from a set of relationships $\boldsymbol{F}$ by using a rule, if r is the head of the rule and each relationship in the body of the rule is in $\boldsymbol{F}$. Let $\mathcal{R}$ be a set of rules. A relationship r is said to be deducible from a set of relationships $\boldsymbol{F}$ by using the rules in $\mathcal{R}$, if r is in $\boldsymbol{F}$, or there exists a finite sequence of relationships $r_1, r_2, \ldots, r_k$ $(= r)$, such that r_1 is deducible in one step from $\boldsymbol{F}$ by using a rule in $\mathcal{R}$, and for each $2 \leq i \leq k, r_i$ is deducible in one step from $\boldsymbol{F} \cup \{r_1, r_2, \ldots, r_{i-1}\}$ by using a rule in $\mathcal{R}$.

Now, a system of rules $\mathcal{R}$, rules I–VIII, introduced in [23], is presented as follows.

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

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

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

$A\ x\ D :: A\ x\ B, B$ *overlaps* $C, C\ x\ D$

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

(a) $A\ x\ C :: A$ *inside* $B, B\ x\ C$

(b) $A\ x\ C :: A\ x\ B, C$ *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, below*$\}$, we have

A *outside* $B :: A\ x\ B$

VI. A *overlaps* $B :: A$ *inside* B

VII. A *overlaps* $B :: C$ *inside* A, C *overlaps* B

VIII. A *inside* A ::

Notice that, in the above rules, we exclude the relationship symbols *right-of* and *above*, since they are duals of *left-of* and *below*, respectively. They can be handled by additional rules that simply relate them to their duals (see rules IX–X in [23]).

Unless it is otherwise stated, we will use $\mathcal{R}$ to represent the set of rules I–VIII given above.

3. Pseudo-Symbolic Pictures. In this section we introduce the definition of pseudo-symbolic picture based on the notion of symbolic picture. We consider the following basic spatial relationship symbols: *left-of, below, inside, outside,* and *overlaps.* We exclude the relationship symbols *right-of* and *above*, since they are duals of *left-of* and *below*, respectively.

The concept of a local scene plays a key role in pseudo-symbolic pictures and extended pseudo-symbolic pictures.

DEFINITION 3.1. *Given a set V of symbols, a local scene over V consists of a subset $U \subseteq V$ and a consistent set $\boldsymbol{F}$ of spatial relationships among symbols in U satisfying that $\boldsymbol{F}$ contains only inside, outside, and overlaps relationships, but no left-of or below relationships, and exactly one of "x outside y" and "x overlaps y" is in $\boldsymbol{F}$ for any two distinct symbols x and y in U.*

A local scene can be encoded in a compact and minimal form, in which the spatial relationships consist of only *inside* and *overlaps*. Since, for any two objects A and B, exactly one of A *overlaps* B and A *outside* B holds, so A *outside* B can be implied if A *overlaps* B is not given (note that we do not distinguish between CxD and DxC for any objects C and D, and $x \in \{outside, overlaps\}$). Furthermore, $\boldsymbol{F}$ can be reduced to a minimal set under the system of rules $\mathcal{R}$ (recall $\mathcal{R}$ is introduced in Section 2). Consider, for example, a local scene that consists of $U = \{A, B, C, D\}$ and $\boldsymbol{F} = \{A\, inside\, B, B\, inside\, C, A\, inside\, C, A\, overlaps\, B, B\, overlaps\, C, A\, overlaps\, C, C\, overlaps\, D, A\, outside\, D, B\, outside\, D\}$. Then, this local scene can be encoded by $e = \{A\ inside\ B, B\ inside\ C, C\ overlaps\ D\}$.

A pseudo-symbolic picture has the same form in matrix as a symbolic picture except that there are *inside, outside,* and *overlaps* relationships, but no *left-of* or *below*, among objects in each slot of the matrix.

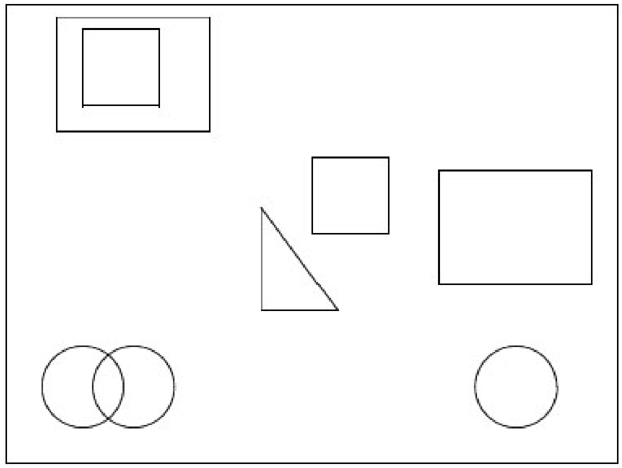
Definition 3.2. *Given a set V of symbols, a pseudo-symbolic picture (abbr. P-symbolic picture) f over V is an $m \times n$ matrix, in which each slot of the matrix is assigned a (possibly empty) local scene over V.*

Intuitively, in a pseudo-symbolic picture f, each slot (namely, a local scene) can be considered as a local significant and minimal unit in the sense that one object can overlap with, or be inside, or be outside another object, but they cannot be separately by either *left-of* or *below* relationships. Objects in one slot can always have either *left-of* or *below* relationships with objects in another slot.

In general, a real picture is assumed to be associated with some content-based meta-data about that picture, that is, information about the objects in the picture, their properties, and the spatial or nonspatial relationships among them. Due to the possibility of multiple occurrences of a given object, this meta-data representation may cause ambiguity. Hence, to avoid ambiguity, we associate multiple occurrences of the same object with different nonnegative integers starting at 0. Suppose, for example, an object A appears in a picture four times. Then each occurrence of A can be represented by A_0 (or simply A), A_1, A_2, and A_3, respectively.

Now we introduce a variation of the 2D string, called the *pseudo-2D string*, to represent the pseudo-symbolic pictures. Let f be a pseudo-symbolic picture. We first represent each nonblank slot by a super-symbol e_i ($i \geq 0$); that is, each e_i points to a local scene in the slot. The enumeration of the super-symbols is produced starting in the down-left position, proceeding by columns and ending with the super-symbol in the up-right position. Then f becomes a simple symbolic picture f_s over the set of super-symbols under the absolute spatial relationships "$<_a$", where each slot of f_s contains at most one super-symbol and different slots contain different super-symbols. We call f_s the *reduced symbolic picture* of f. As with 2D strings, the pseudo-2D string of f is obtained by projecting the super-symbols of f_s by columns and by rows.

Definition 3.3. *Let f be a pseudo-symbolic picture over a set V of symbols,*



s *inside* r		
	s_1 *outside* t	r_1
c *overlaps* c_1		c_2

e_1		
	e_2	e_4
e_0		e_3

FIG. 2. *An image, its pseudo-symbolic and reduced symbolic representations.*

f_s be the reduced symbolic picture of f, and (u, v) be the 2D string representation of f_s under the absolute spatial relationships "$<_a$". Then the pseudo-2D string (abbr. P-2D string) representation of f is just the pair (u, v), where each super-symbol in u or v points to a local scene over V.

Since, for a pseudo-2D string (u, v), super-symbols appearing in u (and v, resp.) are distinct, the reduced symbolic picture f_s can be uniquely reconstructed from the 2D string (u, v) (see [11]). Then the pseudo-symbolic picture f can be obtained from f_s by replacing each super-symbol in each nonblank slot of f_s by its corresponding local scene.

Figure 2 shows an image, the pseudo-symbolic picture g representing it, and the reduced symbolic picture g_s of g. The symbols c, r, s, and t and the set of symbols V are the same as those in Section 2. Three occurrences of the *circle* (c) are represented by c_0 (simply c), c_1, and c_2, respectively. Two occurrences of the *rectangle* (r) are represented by r_0 (simply r) and r_1, respectively. Two occurrences of the *square* (s) are represented by s_0 (simply s) and s_1, respectively. Then, the pseudo-2D string representation of g is $(u, v) = (e_0e_1 < e_2 < e_3e_4, e_0e_3 < e_2e_4 < e_1)$, where

e_0, e_1, e_2, e_3, and e_4, respectively, point to the local scenes {c *overlaps* c_1}, {s *inside* r}, {s_1 *outside* t}, "c_2", and "r_1".

We can derive a consistent set $\boldsymbol{F}$ of spatial relationships from a pseudo-symbolic picture f as follows: every relationship in each local scene is in $\boldsymbol{F}$, a relationship A *left-of* B (A *below* B, respectively) is in $\boldsymbol{F}$ if and only if $j < l$ ($i < k$, respectively), where A and B are in the (i, j)-slot and (k, l)-slot respectively. This set $\boldsymbol{F}$ is called the *derived set* of spatial relationships associated with f. We say that the set of rules $\mathcal{R}$ is *complete* for pseudo-symbolic pictures if it satisfies the following property for every derived set $\boldsymbol{F}$ associated with pseudo-symbolic pictures: every relationship implied by $\boldsymbol{F}$ is deducible from $\boldsymbol{F}$ by using the rules in $\mathcal{R}$.

THEOREM 3.4. *The set of rules $\mathcal{R}$ is complete for planar pseudo-symbolic pictures.*

Proof. Let $\boldsymbol{F}$ be the derived set of spatial relationships associated with a pseudo-symbolic picture f, and let r be a relationship implied by $\boldsymbol{F}$. By the definition of "implied by $\boldsymbol{F}$," the pseudo-symbolic picture f particularly satisfies the spatial relationship r, since f satisfies all the spatial relationships in $\boldsymbol{F}$ by the definition of "the derived set." Now we only need to verify that r is deducible from $\boldsymbol{F}$ by using the rules in $\mathcal{R}$ for each case $r = AxB$, where A and B are two involved objects in f and $x \in \{$*left-of, below, inside, outside, overlaps*$\}$.

Case I. Both A and B are in the same local scene.

By the definition of "the derived set", $\boldsymbol{F}$ contains every spatial relationship in each local scene of f. So $r \in \boldsymbol{F}$ and obviously r $(= AxB)$ is deducible from $\boldsymbol{F}$ by using the rules in $\mathcal{R}$, where $x \in \{$*inside, outside, overlaps*$\}$ (note that a local scene could not have *left-of* or *below* relationships).

Case II. A and B are in different local scenes.

Observe that both A *inside* B and A *overlaps* B could not be satisfied in f, since A and B are in different local scenes. Thus, the spatial relationship symbol x could not be *inside* or *overlaps*.

Let A and B are in the (i, j)-slot and (k, l)-slot respectively.

For the case $x =$ *left-of* ($x =$ *below*, respectively), $j < l$ ($i < k$, respectively) holds, since A *left-of* B (A *below* B, respectively) is satisfied in f. So, A *left-of* B (A *below* B, respectively) must be in $\boldsymbol{F}$ by the definition of "the derived set" and then is obviously deducible from $\boldsymbol{F}$ by using the rules in $\mathcal{R}$.

For the last case $x =$ *outside*, obviously A *outside* B is always satisfied in f (i.e., the assumption is always true), since A and B are in two distinct local scenes. Observe that at least one of the following four inequalities $i < k$, $i > k$, $j < l$, or $j > l$ holds; that is, at least one of the following four spatial relationships A *below* B, B *below* A, A *left-of* B, or B *left-of* A is satisfied in f and is in $\boldsymbol{F}$. So, $r = A$ *outside* B can be deduced from the above satisfied spatial relationship (i.e., $\boldsymbol{F}$) by using Rule V and possibly Rule IV in $\mathcal{R}$.

This completes the proof of this theorem.

□

4. Extended Pseudo-Symbolic Pictures. In this section we introduce an extension of pseudo-symbolic pictures, called *extended pseudo-symbolic pictures*.

Definition 4.1. *A regular partition on an $m \times n$ matrix is a collection of sets of slots $\mathcal{T} = \{T_\alpha \mid \alpha \in I\}$ such that*
(1) *for each set of slots T_α, there exist integers $1 \le i \le j \le m$ and $1 \le k \le l \le n$ such that*

$$T_\alpha = \{(x,y) \mid i \le x \le j, k \le y \le l\},$$

where each pair (x,y) denotes the (x,y)-slot in the given $m \times n$ matrix. We call i, j, k, and l, respectively, below-bound, above-bound, left-bound, and right-bound;
(2) *for any two distinct subscripts α, β in I, $T_\alpha \cap T_\beta = \emptyset$;*
(3) $\cup_{\alpha \in I} T_\alpha = \{(i,j) \mid 1 \le i \le m, 1 \le j \le n\}$*;*
(4) (*minimality of rows and columns*) *for $1 \le i \le m$ and $1 \le j \le n$, there exist $T_{\alpha 1}$ with below-bound i, $T_{\alpha 2}$ with above-bound i, $T_{\alpha 3}$ with left-bound j, and $T_{\alpha 4}$ with right-bound j.*

An example for examining the concept of a regular partition is the bottom-left figure in Figure 3, where each slot is assumed to be empty. It is a regular partition on a 4×4 matrix.

Definition 4.2. *Given a set V of symbols, an extended pseudo-symbolic picture* (*abbr. EP-symbolic picture*) *f over V is a regular partition on an $m \times n$ matrix $\mathcal{T}$ $= \{T_\alpha \mid \alpha \in I\}$, in which each T_α is assigned a* (*possibly empty*) *local scene over V such that $\mathcal{T}$ satisfies the following property, called the minimality condition of rows and columns: for any $1 \le i \le m$ and $1 \le j \le n$, there exist $T_{\alpha 1}$ with below-bound i, $T_{\alpha 2}$ with above-bound i, $T_{\alpha 3}$ with left-bound j, and $T_{\alpha 4}$ with right-bound j such that each of them is assigned a nonempty local scene.*

The minimality condition of rows and columns is necessary for compactness of an extended pseudo-symbolic picture f. The reason is as follows: suppose each nonempty T_α (namely, T_α is assigned a nonempty local scene) does not have the below-bound i. Then every nonempty T_α across the ith row must have the below-bound l, where $1 \le l < i$. An $(m-1) \times n$ matrix can be formed from the original $m \times n$ matrix by removing the ith row and still preserves the same *left-of* and *below* relationships among nonempty T_α's as the original matrix does. Similar arguments can be applied for the other three cases.

The minimality condition of rows and columns automatically satisfies the condition of nonempty rows and columns. That is, for $1 \le i \le m$, there exist one integer $1 \le j \le n$ and one $T_\alpha \in \mathcal{T}$ such that $(i,j) \in T_\alpha$ and T_α is assigned a nonempty local scene; similarly, this is true for columns. Furthermore, the minimality condition of rows and columns guarantees that there are no redundant rows or columns, that is, the numbers of rows and columns are minimal, respectively.

As with the minimality condition of rows and columns in Definition 4.2, minimality of rows and columns in Definition 4.1 is necessary for compactness of a regular partition $\mathcal{T}$ without being assigned local scenes, and it guarantees that there are no redundant rows or columns. It is easy to see that the minimality condition of rows and columns in Definition 4.2 implies minimality of rows and columns in Definition 4.1, but not reversely (Observe that any regular partition $\mathcal{T} = \{T_\alpha \mid \alpha \in I\}$, in which every T_α is assigned an empty local scene, does not satisfy the minimality condition of rows and columns in Definition 4.2).

Now, we introduce the extended pseudo-2D string to represent the extended pseudo-symbolic pictures. Let f be an extended pseudo-symbolic picture and $\mathcal{T} = \{T_\alpha \mid \alpha \in I\}$ be its regular partition on an $m \times n$ matrix. We first represent each T_α assigned with a nonempty local scene by a super-symbol e_i ($i \geq 0$). Then f will become a symbolic picture f_s over the set of super-symbols under the absolute spatial relationships "$<_a$" if, for each $T_\alpha \in \mathcal{T}$ associated with a super-symbol e (possibly blank if T_α is assigned an empty local scene), every slot $(i, j) \in T_\alpha$ is assigned the super-symbol e (possibly blank). Each slot in f_s contains at most one super-symbol. We call f_s the *reduced symbolic picture* of f. The extended pseudo-2D string representation of f is just the non-redundant 2D string of f_s.

DEFINITION 4.3. *Let f be an extended pseudo-symbolic picture over a set V of symbols, f_s be the reduced symbolic picture of f, and (u, v) be the non-redundant 2D string representation of f_s under the absolute spatial relationships "$<_a$". Then the extended pseudo-2D string* (*abbr. EP-2D string*) *representation of f is just the pair (u, v), where each super-symbol in u or v points to a local scene over V.*

Although the extended pseudo-2D string representation (u, v) of f looks like a 2D string, it is not really a 2D string in the sense that if one super-symbol e takes across the ith and jth columns, then u contains the substring $e < e$, but $e < e$ does not mean that e *left-of* e holds as in the usual 2D string representation. Similarly, this is true for v when one super-symbol takes across different rows. However, for distinct super-symbols d and e, $d < e$ in u (v, respectively) means that d *left-of* e (d *below* e, respectively) holds as in the usual 2D string representation. Certainly, $<$ here represents absolute spatial relationship, while it represents relative spatial relationship in the usual 2D string representation, as mentioned in Section 1. Therefore, the extended pseudo-2D string can be considered as an "almost" 2D string.

Given an extended pseudo-2D string (u, v), we can, easily and uniquely, reconstruct an extended pseudo-symbolic picture f from it. In fact, for each super-symbol e, we can determine its left-bound and right-bound, and its below-bound and above-bound by checking the continuous occurrences of e in u and v, respectively. Then the regular partition $\mathcal{T} = \{T_\alpha \mid \alpha \in I\}$ on an $m \times n$ matrix can be formed, where m and n are one more than the numbers of $<$ in u and v, respectively. The extended

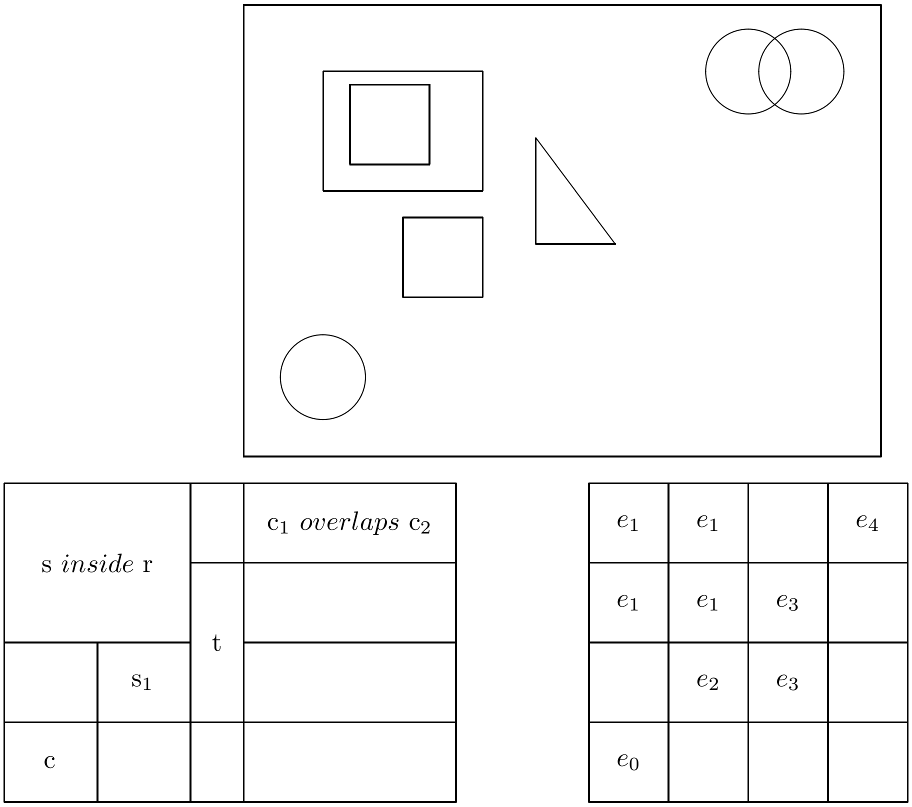


FIG. 3. *An image, its extended pseudo-symbolic and reduced symbolic representations.*

pseudo-symbolic picture f is obtained from $\mathcal{T}$ by replacing each super-symbol by its corresponding local scene.

Figure 3 shows an image, the extended pseudo-symbolic picture h representing it, and the reduced symbolic picture h_s of h. The symbols c, r, s, and t and the set of symbols V are the same as those in Section 2. Then, the extended pseudo-2D string representation of h is $(u, v) = (e_0e_1 < e_1e_2 < e_3 < e_4, e_0 < e_2e_3 < e_1e_3 < e_1e_4)$, where e_0, e_1, e_2, e_3, and e_4, respectively, point to the local scenes "c", {s *inside* r}, "s_1", "t", and {c_1 *overlaps* c_2}.

We can derive a consistent set $\boldsymbol{F}$ of spatial relationships from an extended pseudo-symbolic picture f as follows: every relationship in each local scene is in $\boldsymbol{F}$, a relationship A *left-of* B (A *below* B, respectively) is in $\boldsymbol{F}$ if and only if $j < l$ ($i < k$, respectively), where A and B are in the local scenes T_p and T_q respectively, and T_p has the right-bound j and above-bound i, and T_q has the left-bound l and below-bound k. This set $\boldsymbol{F}$ is called the *derived set* of spatial relationships associated with f. We say that the set of rules $\mathcal{R}$ is *complete* for extended pseudo-symbolic pictures if it satisfies the following property for every derived set $\boldsymbol{F}$ associated with extended pseudo-symbolic pictures: every relationship implied by $\boldsymbol{F}$ is deducible from $\boldsymbol{F}$ by using the rules in $\mathcal{R}$.

Now we can use similar argument as in Theorem 3.4 to prove the following theorem.

THEOREM 4.4. *The set of rules $\mathcal{R}$ is complete for planar extended pseudo-symbolic pictures.*

Proof. Let $\boldsymbol{F}$ be the derived set of spatial relationships associated with an extended pseudo-symbolic picture f, and let r be a relationship implied by $\boldsymbol{F}$. By the definition of "implied by $\boldsymbol{F}$," the extended pseudo-symbolic picture f particularly satisfies the spatial relationship r, since f satisfies all the spatial relationships in $\boldsymbol{F}$ by the definition of "the derived set." Now we only need to verify that r is deducible from $\boldsymbol{F}$ by using the rules in $\mathcal{R}$ for each case $r = AxB$, where A and B are two involved objects in f and $x \in \{$*left-of, below, inside, outside, overlaps*$\}$.

Case I. Both A and B are in the same local scene.

By the definition of "the derived set", $\boldsymbol{F}$ contains every spatial relationship in each local scene of f. So $r \in \boldsymbol{F}$ and obviously r $(= AxB)$ is deducible from $\boldsymbol{F}$ by using the rules in $\mathcal{R}$, where $x \in \{$*inside, outside, overlaps*$\}$ (note that a local scene could not have *left-of* or *below* relationships).

Case II. A and B are in different local scenes.

Observe that both A *inside* B and A *overlaps* B could not be satisfied in f, since A and B are in different local scenes. Thus, the spatial relationship symbol x could not be *inside* or *overlaps*.

Let A and B are in the local scenes T_p and T_q respectively, where T_p has the below-bound i_p, above-bound j_p, left-bound k_p, and right-bound l_p respectively, and T_q has the below-bound i_q, above-bound j_q, left-bound k_q, and right-bound l_q respectively.

For the case $x =$ *left-of* ($x =$ *below*, respectively), $l_p < k_q$ ($j_p < i_q$, respectively) holds, since A *left-of* B (A *below* B, respectively) is satisfied in f. So, A *left-of* B (A *below* B, respectively) must be in $\boldsymbol{F}$ by the definition of "the derived set" and then is obviously deducible from $\boldsymbol{F}$ by using the rules in $\mathcal{R}$.

For the last case $x =$ *outside*, obviously A *outside* B is always satisfied in f (i.e., the assumption is always true), since A and B are in two distinct local scenes. Observe that at least one of the following four inequalities $j_p < i_q$, $i_p > j_q$, $l_p < k_q$, or $k_p > l_q$ holds; that is, at least one of the following four spatial relationships A *below* B, B *below* A, A *left-of* B, or B *left-of* A is satisfied in f and is in $\boldsymbol{F}$. So, $r = A$ *outside* B can be deduced from the above satisfied spatial relationship (i.e., $\boldsymbol{F}$) by using Rule V and possibly Rule IV in $\mathcal{R}$.

This completes the proof of this theorem. □

5. Decidability of (Extended) Pseudo-Symbolic Pictures. In this section we consider the *decidability* problem for (extended) pseudo-symbolic pictures. That is, *is there a decidable procedure to determine whether a given planar picture is* (*extended*) *pseudo-symbolic?* For a planar picture f, we assume all objects and spatial relationships in f are given. We first give an efficient algorithm for the decidability problem of pseudo-symbolic pictures, then modify it to an efficient algorithm for the decidability problem of extended pseudo-symbolic pictures.

5.1. Algorithm for Deciding Pseudo-Symbolic Pictures. For a planar picture f, we will use $\boldsymbol{O}_f$ and $\boldsymbol{R}_f$ to denote the sets of all objects and spatial relationships in f, respectively.

1. Encode Local Scenes by Super-Symbols

For two distinct objects x, y in $\boldsymbol{O}_f$, we call x and y *spatial-comparable* if one of the four spatial relationships, x *left-of* y, y *left-of* x, x *below* y, and y *below* x, holds, and *spatial-incomparable*, otherwise. For each object $x \in \boldsymbol{O}_f$, let $\boldsymbol{E}_x$ be the set of all objects in $\boldsymbol{O}_f$ that are spatial-incomparable with x. Given a symbol $x \in \boldsymbol{O}_f$, if $\{y\} \cup \boldsymbol{E}_y = \{x\} \cup \boldsymbol{E}_x$ for each $y \in \boldsymbol{E}_x$, then $\{x\} \cup \boldsymbol{E}_x$ forms a local scene of f and can be encoded by a super-symbol e_i (i is a nonnegative integer beginning at 0). But if $\{y\} \cup \boldsymbol{E}_y \neq \{x\} \cup \boldsymbol{E}_x$ for some $y \in \boldsymbol{E}_x$, then either $\{x\} \cup \boldsymbol{E}_x$ or $\{y\} \cup \boldsymbol{E}_y$ can never form a local scene of f, and it can be easily seen that f is not (extended) pseudo-symbolic.

Since spatial-incomparability of any two objects in $\boldsymbol{O}_f$ needs to be checked at most once, and it takes only constant time to check whether two objects are spatial-incomparable. Thus, the total time in Step (1) is at most $O(n(n-1)/2)$ (i.e., $O(n^2)$), where n is the cardinality of $\boldsymbol{O}_f$.

2. Define *left-of* and *below* Relationships among Super-Symbols

Let $\boldsymbol{O}_e$ be the set of all objects in a local scene represented by a super-symbol e. For two super-symbols e_i and e_j, we define e_i *left-of* e_j (resp. e_i *below* e_j) if O_i *left-of* O_j (resp. O_i *below* O_j) for each $O_i \in \boldsymbol{O}_{e_i}$ and each $O_j \in \boldsymbol{O}_{e_j}$. However, if there exist O_1, O_2 in $\boldsymbol{O}_{e_i}$, and O_3, O_4 in $\boldsymbol{O}_{e_j}$ such that either "O_1 *left-of* O_3 holds and O_2 *left-of* O_4 does not hold" or "O_1 *below* O_3 holds and O_2 *below* O_4 does not hold" (called *the bad condition* between the super-symbols e_i and e_j), occurs, then f can not be (extended) pseudo-symbolic.

Let $e_0, e_1, \ldots, e_{l-1}$ be all super-symbols of f, and let n_i be the cardinality of $\boldsymbol{O}_{e_i}$, $0 \leq i \leq l-1$. For every pair e_i and e_j, to determine their *left-of* and *below* relationships, we need to check *left-of* and *below* relationships between one object in $\boldsymbol{O}_{e_i}$ and another object in $\boldsymbol{O}_{e_j}$. The number of comparisons needed for this pair is at most $2n_in_j$. Hence, the total number of comparisons needed for Step (2) is at most

$$\sum\nolimits_{0\leq i<j\leq l-1} 2n_in_j \leq (n_0 + n_1 + \cdots + n_{l-1})^2 = n^2.$$

Therefore, the total time needed for Step (2) is at most $O(n^2)$.

Notice that, at the end of Step (2), for every pair of super-symbols e_i and e_j, one of the four relationships, e_i *left-of* e_j, e_j *left-of* e_i, e_i *below* e_j, and e_j *below* e_i, holds, since both one object in $\boldsymbol{O}_{e_i}$ and another object in $\boldsymbol{O}_{e_j}$ are spatial-comparable.

3. Find Local Substrings of Super-Symbols

For two distinct super-symbols e_i and e_j, $0 \leq i, j \leq l-1$, we call e_i and e_j *left-of-comparable* (resp. *below-comparable*) if either e_i *left-of* e_j (resp. e_i *below* e_j) or e_j *left-of* e_i (resp. e_j *below* e_i) holds, and *left-of-incomparable* (resp. *below-incomparable*), otherwise. For each super-symbol e, let $\boldsymbol{L}_e$ (resp. $\boldsymbol{B}_e$) be the set of all super-symbols e_i, $0 \leq i \leq l-1$, that are *left-of*-incomparable (resp. *below*-incomparable) with e. Given a super-symbol e, if $\{x\} \cup \boldsymbol{L}_x = \{e\} \cup \boldsymbol{L}_e$ (resp. $\{x\} \cup \boldsymbol{B}_x = \{e\} \cup \boldsymbol{B}_e$) for each $x \in \boldsymbol{L}_e$ (resp. $x \in \boldsymbol{B}_e$), then we call $\{e\} \cup \boldsymbol{L}_e$ (resp. $\{e\} \cup \boldsymbol{B}_e$) *a local set* (or *substring*) of super-symbols with respect to *left-of* (resp. *below*) relationships. It is easy to see that if f is pseudo-symbolic, both $\{e\} \cup \boldsymbol{L}_e$ and $\{e\} \cup \boldsymbol{B}_e$ form local substrings for each super-symbol e; and if $\{x\} \cup \boldsymbol{L}_x \neq \{e\} \cup \boldsymbol{L}_e$ for some $x \in \boldsymbol{L}_e$ (resp. $\{x\} \cup \boldsymbol{B}_x \neq \{e\} \cup \boldsymbol{B}_e$ for some $x \in \boldsymbol{B}_e$), then $\{e\} \cup \boldsymbol{L}_e$ (resp. $\{e\} \cup \boldsymbol{B}_e$) cannot form a local substring and f can never be pseudo-symbolic.

As in Step (1), the total time in Step (3) is at most $O(l^2)$ and thus is bounded by $O(n^2)$ (note that $l \leq n$).

4. Produce the Pseudo-2D String Representation

Let $\boldsymbol{L}_1, \boldsymbol{L}_2, \ldots, \boldsymbol{L}_p$ and $\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_q$ be all local substrings with respect to *left-of* and *below* relationships, respectively. Observe that one super-symbol in $\boldsymbol{L}_i$ and another super-symbol in $\boldsymbol{L}_j$ are always *left-of*-comparable, and preserve the same *left-of* relationships, that is, either "x_i *left-of* x_j holds for all $x_i \in \boldsymbol{L}_i$ and $x_j \in \boldsymbol{L}_j$" or "x_j *left-of* x_i holds for all $x_i \in \boldsymbol{L}_i$ and $x_j \in \boldsymbol{L}_j$" occurs. Hence, we can define the *left-of* relationships among the $\boldsymbol{L}_1, \boldsymbol{L}_2, \ldots, \boldsymbol{L}_p$, and call $\boldsymbol{L}_i$ *left-of* $\boldsymbol{L}_j$ if there exist $x_i \in \boldsymbol{L}_i$ and $x_j \in \boldsymbol{L}_j$ such that x_i *left-of* x_j. Clearly, $\boldsymbol{L}_i$ *left-of* $\boldsymbol{L}_j$ if and only if x_i *left-of* x_j for all $x_i \in \boldsymbol{L}_i$ and $x_j \in \boldsymbol{L}_j$. Thus, there is a natural *left-of* relationship between any two distinct local substrings $\boldsymbol{L}_i$ and $\boldsymbol{L}_j$; that is, $\{\boldsymbol{L}_1, \boldsymbol{L}_2, \ldots, \boldsymbol{L}_p\}$ forms a linear order with respect to the *left-of* relationship. Similarly, we can define the natural *below* relationships among the $\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_q$, and verify that $\{\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_q\}$ forms a linear order with respect to the *below* relationship. Now, we may apply any existing order-sorting algorithms, such as Quicksort (see, e.g., [1, 3]), to sort $\{\boldsymbol{L}_1, \boldsymbol{L}_2, \ldots, \boldsymbol{L}_p\}$ and $\{\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_q\}$ with respect to *left-of* and *below* relationships, respectively. Let u and v represent the sorted orders of the two sets $\{\boldsymbol{L}_1, \boldsymbol{L}_2, \ldots, \boldsymbol{L}_p\}$ and $\{\boldsymbol{B}_1, \boldsymbol{B}_2, \ldots, \boldsymbol{B}_q\}$, respectively. Then the pseudo-2D string representation of f can be obtained from (u, v) by replacing the $\boldsymbol{L}_i$'s and $\boldsymbol{B}_j$'s by their represented local substrings of super-symbols.

Since any order-sorting algorithm has the time complexity of at most order 2 in the size of the sequence in the worst case, the total time spent in the last step (4) is at most $O(p^2 + q^2)$, bounded by $O(n^2)$.

Therefore, the presented algorithm has the time complexity $O(n^2)$, where n is the number of all involved objects in a picture. The correctness of the algorithm can be easily verified.

THEOREM 5.1. *There exists an efficient algorithm with time complexity of* $O(n^2)$, *where* n *is the number of all involved objects in a picture, to determine whether a given planar picture is pseudo-symbolic, and if it is, the algorithm also returns its pseudo-2D string representation.*

Now, we present the algorithm.

Algorithm. *Decide whether a given planar picture* f *is pseudo-symbolic.*

Input: the set of objects $\boldsymbol{O_f}$ and the set of relationships $\boldsymbol{R_f}$ representing f.

Output: YES if f is pseudo-symbolic; NO, otherwise.
And if YES, the pseudo-2D string representation of f is also produced.

Step 1. Encode local scenes by super-symbols
Set $\boldsymbol{O} = \boldsymbol{O_f}$;
While $\boldsymbol{O} \neq \emptyset$ do
begin
Choose one symbol $x \in \boldsymbol{O}$, and calculate the set $\boldsymbol{E_x}$, then
check whether $\{x\} \cup \boldsymbol{E_x}$ forms a local scene of f.
If yes, use a super-symbol e_i to represent it and continue;
otherwise, output "NO" and exit the procedure.
Reset $\boldsymbol{O} = \boldsymbol{O} - \{x\} \cup \boldsymbol{E_x}$;
end; /* while */

Step 2. Define *left-of* and *below* relationships among super-symbols
For every pair of super-symbols e_i and e_j, check whether
the bad condition between e_i and e_j occurs.
If yes, then output "NO" and exit the procedure;
otherwise, define either *left-of* or *below* relationships between them.

Step 3. Find local substrings of super-symbols
/* Form local substrings of super-symbols w.r.t. *left-of* relationships. */
Set $S = \{e_0, e_1, \ldots, e_{l-1}\}$; /* the set of super-symbols */
While $S \neq \emptyset$ do
begin
Choose one super-symbol $e \in S$, and calculate the set $\boldsymbol{L_e}$, then check
whether $\{e\} \cup \boldsymbol{L_e}$ forms a local substring w.r.t. *left-of* relationships.
If yes, represent it by $\boldsymbol{L_i}$ (i is some integer) and continue;
otherwise, output "NO" and exit the procedure.
Reset $S = S - \{e\} \cup \boldsymbol{L_e}$;
end; /* while */
/* Form local substrings of super-symbols w.r.t. *below* relationships.
The following code is the same as the above one except
all *left-of*'s are replaced by *below*'s. */
Set $S = \{e_0, e_1, \ldots, e_{l-1}\}$; /* the set of super-symbols */
While $S \neq \emptyset$ do
begin

Choose one super-symbol $e \in S$, and calculate the set $\boldsymbol{B_e}$, then check whether $\{e\} \cup \boldsymbol{B_e}$ forms a local substring w.r.t. *below* relationships.
If yes, represent it by $\boldsymbol{B_j}$ (j is some integer) and continue;
otherwise, output "NO" and exit the procedure.
Reset $S = S - \{e\} \cup \boldsymbol{B_e}$;
end; /* while */

Step 4. Produce the pseudo-2D string representation (u, v) of f
Sort $\{\boldsymbol{L_1}, \boldsymbol{L_2}, \ldots, \boldsymbol{L_p}\}$ w.r.t. the natural *left-of* relationship, and use u to store the sorted order;
Sort $\{\boldsymbol{B_1}, \boldsymbol{B_2}, \ldots, \boldsymbol{B_q}\}$ w.r.t. the natural *below* relationship, and use v to store the sorted order;
Replace all $\boldsymbol{L_i}$'s and $\boldsymbol{B_j}$'s in u and v by their represented local substrings of super-symbols, respectively, and then output "YES" and (u, v).

/* End of the algorithm. */

Note that the above presented algorithm needs only the first three steps to verify whether a given planar picture is pseudo-symbolic. And if it is, then the algorithm executes Step (4) to output its pseudo-2D string at the extra cost of time complexity of at most $O(n^2)$.

5.2. Algorithm for Deciding Extended Pseudo-Symbolic Pictures. Note that, in our algorithm for deciding extended pseudo-symbolic pictures, the first two steps are the same as those in the algorithm for deciding pseudo-symbolic pictures.

1 and 2. Same as 1 and 2 in the case of pseudo-symbolic pictures.

3. Produce the Extended Pseudo-2D String Representation

Since one super-symbol may take across several rows and columns in the extended pseudo-symbolic picture f, finding local substrings of super-symbols here is much harder than that in the case of pseudo-symbolic pictures.

Let S be the set of super-symbols. A super-symbol $x \in S$ is called *left-most* in S if, for any other super-symbol $y \in S$, y *left-of* x cannot hold. Let L be the set of all left-most super-symbols in S and set $S_1 = S - L$. Then we have the following three claims.

CLAIM 5.2. $L \neq \emptyset$.

Proof. Choose one super-symbol x_1 from S. If x_1 is left-most in S, then $x_1 \in L$. Otherwise, there exists $x_2 \in S$ such that $x_2 \neq x_1$ and x_2 *left-of* x_1 holds. Now if x_2 is left-most in S, then $x_2 \in L$. Otherwise, there exists $x_3 \in S$ such that x_3 is different from x_1 and x_2, and x_3 *left-of* x_2 holds. Continue this process. Because of the finiteness of $\mid S \mid$, there exist a positive integer k and one super-symbol $x_k \in S$ such that $x_k \neq x_i$ and x_{i+1} *left-of* x_i, where $1 \leq i < k$, and x_k is left-most in S. Thus, $x_k \in L$ and $L \neq \emptyset$. □

CLAIM 5.3. *For any $y \in S_1$, there exists $x \in L$ such that x left-of y holds.*

Proof. Let $y \in S_1$. Note that $S_1 = S - L$, so y is not in L, namely, y is not left-most in S. Then there exists $x_1 \in S$ such that x_1 *left-of* y holds. By using the argument of Claim 5.2, there exists $x_k \in S$ such that x_k *left-of* x_1 holds and $x_k \in L$. The relationship x_k *left-of* y can be deduced from relationships x_k *left-of* x_1 and x_1 *left-of* y by using the transitive rule (i.e., Rule I) in the system of rules $\mathcal{R}$. □

CLAIM 5.4. *There exists $e \in L$ such that for any $y \in S_1$, e left-of y holds.*

Proof. Let $x_r(O)$ be the x-coordinate of the right-most point (i.e., the supreme x-coordinate of points) in an object O, and let $x_r(e)$ be the supreme of the set $\{x_r(O) \mid O$ is an object in the local scene encoded by $e\}$ for a super-symbol e. Then choose one super-symbol $e \in L$ such that $x_r(e) \leq x_r(e')$ for all $e' \in L$. For any $y \in S_1$, by Claim 5.3, there exists $x \in L$ such that x *left-of* y holds. By the semantics definition of spatial relationship *left-of*, we conclude that e *left-of* y holds. □

Now, by Claims 5.2 and 5.3, L forms first local set (or substring) of super-symbols. Note that, a super-symbol $x \in L$ ends at the first column (i.e., x does not take across the second column) if and only if x *left-of* y holds for every $y \in S_1$. Let L_1 be the set of all x's ending at the first column and set $L_2 = L - L_1$. Then, by Claim 5.4, $L_1 \neq \emptyset$. And every super-symbol in L_2 cannot end at the first column and must take across the second column. Note that every super-symbol in S_1 must start at the second or later columns. Let $S' = S_1 \cup L_2$, then $\mid S' \mid \leq \mid S \mid -1$. Repeat this process by replacing S by using S' until $S_1 = \emptyset$. Finally, we can have the first component u of the extended pseudo-2D string representation (u, v) of f.

Similarly, the above process can be applied to get the second component v of the extended pseudo-2D string representation (u, v) of f by simply replacing all *left-of*'s by using *below*'s.

For each loop, the time needed for computing L and S_1 is $O(l^2)$, and the comparisons needed for computing L_1 is at most $\mid L \mid \times \mid S_1 \mid \leq l^2$. Hence, the total time needed for one loop is $O(l^2)$. Since $\mid S \mid$ decreases at least one after each loop, the total time spent in Step (3) is $O(l^3)$ and thus is bounded by $O(n^3)$.

Therefore, our algorithm has the time complexity $O(n^3)$, where n is the number of all involved objects in a picture. The correctness of the algorithm can be easily verified.

THEOREM 5.5. *There exists an efficient algorithm with time complexity of $O(n^3)$, where n is the number of all involved objects in a picture, to determine whether a given planar picture is extended pseudo-symbolic, and if it is, the algorithm also returns its extended pseudo-2D string representation.*

Now, we present the algorithm.

Algorithm. *Decide whether a given planar picture f is extended pseudo-symbolic.*

Input: the set of objects $\boldsymbol{O_f}$ and the set of relationships $\boldsymbol{R_f}$ representing f.

Output: YES if f is extended pseudo-symbolic; NO, otherwise.
And if YES, the extended pseudo-2D string representation of f is also produced.

Step 1. Encode local scenes by super-symbols
Same as Step 1 in the algorithm for deciding pseudo-symbolic pictures.

Step 2. Define *left-of* and *below* relationships among super-symbols
Same as Step 2 in the algorithm for deciding pseudo-symbolic pictures.

Step 3. Produce the extended pseudo-2D string representation (u, v) of f
/∗ Find the first component u of the pair (u, v). ∗/
Set $u = \emptyset$ and $S = \{e_0, e_1, \ldots, e_{l-1}\}$;
/∗ Initially S is the set of super-symbols ∗/
While $S \neq \emptyset$ do
    begin
    Calculate the set u_1 of all left-most super-symbols in S
    and set $S_1 = S - u_1$;
    /∗ $\circ$ means the concatenation operation of two strings. ∗/
    If $S_1 = \emptyset$ then $u = u \circ u_1$ and reset $S = \emptyset$;
    else begin
        $u = u \circ u_1 \circ <$;
        Calculate the set u_{11} of all x's ending at the current column
        and set $u_{12} = u_1 - u_{11}$;
        Reset $S = S_1 \cup u_{12}$;
        end;
    end; /∗ end of while ∗/
/∗ Find the second component v of the pair (u, v). The following code is
  the same as the above one used to find u except
  replacing all *left-of*'s using *below*'s. ∗/
Set $v = \emptyset$ and $S = \{e_0, e_1, \ldots, e_{l-1}\}$;
/∗ Initially S is the set of super-symbols ∗/
While $S \neq \emptyset$ do
    begin
    Calculate the set v_1 of all below-most super-symbols in S
    and set $S_1 = S - v_1$;
    If $S_1 = \emptyset$ then $v = v \circ v_1$ and reset $S = \emptyset$;
    else begin
        $v = v \circ v_1 \circ <$;
        Calculate the set v_{11} of all x's ending at the current row and set
        $v_{12} = v_1 - v_{11}$;
        Reset $S = S_1 \cup v_{12}$;
        end;
    end; /∗ end of while ∗/
Output "YES" and (u, v).

/∗ End of the algorithm. ∗/

Note that our above algorithm needs only the first two steps to verify whether a given planar picture is extended pseudo-symbolic. Hence, it takes only time complexity of $O(n^2)$ to verify the decidability problem for extended pseudo-symbolic pictures. And if it is, then the algorithm executes Step (3) to output its extended pseudo-2D string at the extra cost of time complexity of $O(n^3)$.

6. Picture Retrieval by (Extended) Pseudo-2D Strings. The 2D string approach transforms the image retrieval into a 2D string matching problem. Chang et al. [11] defined type-0, type-1, and type-2 picture matchings. Similarly, with (extended) pseudo-2D strings, we can define type-1 and type-2 picture matchings for (extended) pseudo-symbolic pictures. Informally, a pseudo-symbolic (an extended pseudo-symbolic, respectively) picture f is a *type-2* subpicture of a pseudo-symbolic (an extended pseudo-symbolic, respectively) picture f' if f occurs somewhere in f', in its native configuration; f is a *type-1* subpicture of f' if f is equal to the intersection of some rows and some columns from f'. Now we define type-i ($i = 1, 2$) picture matching as follows:

- *Type-2 picture matching*: Given a query picture Q, Q matches a picture f' stored in the database if there exists a type-2 subpicture f of f' such that both Q and f have the same matrix or regular partition configuration and each local scene in Q is part (i.e., a subscene) of the corresponding local scene in f.
- *Type-1 picture matching*: Given a query picture Q, Q matches a picture f' stored in the database if there exists a type-1 subpicture f of f' such that both Q and f have the same matrix or regular partition configuration and each local scene in Q is part (i.e., a subscene) of the corresponding local scene in f.

Observe that the type-2 picture matching is a special part of the type-1 picture matching. We also call the type-1 picture matching *the exact picture matching*. However, type-0 picture matching is not applicable here because of the "absolute" nature of spatial relationships. Notice that the picture matching problem can be considered as a two-level subpicture matching problem, with level-1 subpicture matching for the local scenes, and level-2 subpicture matching for the reduced symbolic pictures, where each local scene is considered as a super-symbol, and super-symbol e_1 matches super-symbol e_2 if e_1 is a subscene of e_2.

Because of the close connection between the (extended) pseudo-2D string and the usual augmented 2D string, we can adapt existing picture-matching algorithms (see, e.g., [11]) originally developed for the augmented 2D string to work on the (extended) pseudo-2D string.

For a description of query processing of picture retrieval, the interested reader may refer to [34].

Tucci et al. [27] proved that the type-1 symbolic picture matching is *NP*-complete. It can also be proved that the exact picture matching problem with (extended) pseudo-

2D strings is NP-complete.

While the exact picture matching yields the query outcome consisting of only those stored images matched exactly by a user query, it might take much long time to perform the query processing for certain irregular stored images because of NP-completeness of the exact picture matching. To address this type of inefficiency, approximate or heuristic picture matching algorithms need to be developed to help improve the performance of picture retrieval.

7. Conclusion. In this paper we have defined a large class of two-dimensional scenes, the extended pseudo-symbolic pictures. The existing rule system $\mathcal{R}$ is proved to be complete for (extended) pseudo-symbolic pictures. We have proposed efficient (i.e., polynomial-time) algorithms to determine whether a given planar picture is (extended) pseudo-symbolic or not and, if it is, these algorithms also return its (extended) pseudo-2D string representation. The detailed algorithms, presented in this paper, can be directly programmed into executable computer codes. We have developed a new iconic indexing, the (extended) pseudo-2D string representation, for the (extended) pseudo-symbolic pictures. The (extended) pseudo-2D string has the good properties of the 2D string. It is unambiguous, like the augmented 2D string, and has a compact form suitable for image retrieval.

In this paper we have partially but not completely attacked the completeness problem, proposed in [23], of reasoning about planar spatial relationships in pictorial retrieval systems. Theoretically this completeness problem still remains *open* because of its nature of difficulty. However, practically we [30] have proposed an alternative to maintain the complete information about the absolute spatial relationships in the image, which is sufficient for our generalized extended pseudo-2D string (GEP-2D string) approach of image retrieval.

As mentioned in the Introduction, the 2D string approach considers only relative spatial relationships among objects and thus overlooks the possible absolute spatial relationships that are more accurate, while the approach in [23] considers only absolute spatial relationships among objects and thus overlooks the possible relative spatial relationships that are less accurate. In [33], we proposed a new iconic indexing, called *the combined 2D string representation*, for 2D and 3D scenes. This new iconic indexing takes advantage of both approaches, eliminates their deficiencies, and thus gives a better representation of the spatial relationships in pictorial database systems. Later, Zhang et al. [30] extended our work on the extended pseudo-symbolic pictures, presented in this paper and [33], to work for the whole images.

The interested reader may refer to [28, 29, 30, 31, 32, 33, 34, 35, 36] for our further developments in content-based image database systems.

REFERENCES

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *Data Structures and Algorithms,* Addison-Wesley, Reading, Massachusetts, 1983.

[2] T. Arndt and S.-K. Chang, *Image Sequence Compression by Iconic Indexing*, in: Proc. IEEE Workshop Visual Languages, Rome, Italy, pp. 177–182, 1989.

[3] S. Baase, *Computer Algorithms: Introduction to Design and Analysis,* second edition, Addison-Wesley, Reading, Massachusetts, 1988.

[4] C. C. Chang and S. Y. Lee, *Retrieval of Similar Pictures on Pictorial Databases*, Pattern Recognition, 24:7(1991), pp. 675–680.

[5] S.-K. Chang, *Pictorial Information Systems,* Prentice-Hall, Englewood Cliffs, N.J., 1989.

[6] S.-K. Chang and A. Hsu, *Image Information Systems: Where Do We Go from Here?*, IEEE Transactions on Knowledge and Data Engineering, 4:5(1992), pp. 431–442.

[7] S.-K. Chang, E. Jungert, and Y. Li, *Representation and Retrieval of Symbolic Pictures Using Generalized 2D Strings*, in: SPIE Proc. Visual Commun. and Image Processing, Philadelphia, pp. 1360–1372, Nov. 5-10, 1989.

[8] S.-K. Chang, E. Jungert, and Y. Li, *The Design of Pictorial Databases Based upon the Theory of Symbolic Projections*, in: Proc. Int. Symposium on Large Spatial Databases (SSD), Springer-Verlag, 1989.

[9] S.-K. Chang and T. Kunii, *Pictorial Database Systems*, IEEE Comput. Mag., Special Issue on Pictorial Information Systems, 14(Nov. 1981), pp. 13–21.

[10] S.-K. Chang and Y. Li, *Representation of Multi-resolution Symbolic and Binary Pictures Using 2DH Strings*, in: Proc. IEEE Workshop Languages for Automation, pp. 190–195, 1988.

[11] S.-K. Chang, Q. Y. Shi, and C. W. Yan, *Iconic Indexing by 2-D Strings*, IEEE Trans. Pattern Analysis and Machine Intelligence, 9:3(1987), pp. 413–428.

[12] S.-K. Chang, C. W. Yan, D. C. Dimitroff, and T. Arndt, *An Intelligent Image Database System*, IEEE Trans. Software Engineering, 14:5(1988), pp. 681–688.

[13] G. Costagliola, F. Ferrucci, G. Tortora, and M. Tucci, *Non-Redundant 2D Strings*, IEEE Transactions on Knowledge and Data Engineering, 7:2(1995), pp. 347–350.

[14] G. Costagliola, G. Tortora, and T. Arndt, *A Unifying Approach to Iconic Indexing for 2D and 3D Scenes*, IEEE Transactions on Knowledge and Data Engineering, 4:3(1992), pp. 205–222.

[15] G. Costagliola, M. Tucci, and S.-K. Chang, *Representing and Retrieving Symbolic Pictures by Spatial Relations*, in: Visual Database Systems, North-Holland, pp. 49–59, 1992.

[16] 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.

[17] 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.

[18] C. L. Jackins and S. L. Tanimoto, *Oct-trees and Their Use in Representing Three-dimensional Objects*, Comput. Graphics and Image Processing, 14:3(1980), pp. 249–270.

[19] E. Jungert, *Extended Symbolic Projection Used in a Knowledge Structure for Spatial Reasoning*, in: Proc. 4th BPRA Conf. Pattern Recognition, Springer-Verlag, Cambridge, March 28–30, 1988.

[20] E. Jungert and S.-K. Chang, *An Algebra for Symbolic Image Manipulation and Transformation*, in: Visual Database Systems, North-Holland, pp. 301–317, 1989.

[21] S.-Y. Lee, M.-K. Shan, and W.-P. Yang, *Similarity Retrieval of Iconic Image Database,* Pattern Recognition, 22:6(1989), pp. 675–682.

[22] D. J. Meagher, *Geometric Modeling Using Octree Encoding*, Comput. Graph. and Image Processing, 19:2(1981), pp. 129–147.

[23] 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.

[24] H. Tamura and N. Yokoya, *Image Database Systems: A Survey*, Pattern Recognition, 17(1984), pp. 29–43.

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

[26] G. Tortora, G. Costagliola, T. Arndt, and S.-K. Chang, *Pyramidal Algorithms for Iconic Indexing*, Computer Vision, Graphics, and Image Processing, 53(1990), pp. 26–56.

[27] M. Tucci, G. Costagliola, and S.-K. Chang, *A Remark on NP-Completeness of Picture Matching*, Information Processing Letters, 39(1991), pp. 241–243.

[28] 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, University of Illinois, Chicago, 1996.

[29] 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, 99–106, August 22–23, 1996.

[30] 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.

[31] 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., 127–146, 1997.

[32] Q.-L. Zhang, S.-K. Chang, and S. S.-T. Yau, *The Consistency Problem on Content-based Pictorial Description in Pictorial Database Systems*, Communications in Information and Systems, International Press, 1:2(2001), pp. 225–240.

[33] 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, 1667–1672, June 23–27, 1997.

[34] Q.-L. Zhang and S. S.-T. Yau, *A General Approach to Indexing and Retrieval of Images in Image Databases*, Communications in Information and Systems, International Press, 3:1(2003), pp. 61–73.

[35] Q.-L. Zhang and S. S.-T. Yau, *A Stepwise Approximation of Intractable Spatial Constraints in Image Queries*, Communications in Information and Systems, International Press, 3:3(2003), pp. 203–221.

[36] Q.-L. Zhang and S. S.-T. Yau, *On Intractability of Spatial Relationships in Content-based Image Database Systems*, Communications in Information and Systems, International Press, 4:2(2004), pp. 181–190.