A GENERAL APPROACH TO INDEXING AND RETRIEVAL OF IMAGES IN IMAGE DATABASES[*]

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

Abstract. In this paper we present a very general approach to indexing and retrieval of images in image database systems. We allow different ways of indexing and matching for objects in an image, and integrate them with our existing approach of the generalized extended pseudo-2D string and generalized combined 2D string representation of the image to formulate a more general framework for iconic indexing and retrieval of images. Image retrieval is then discussed, and the query processing of image retrieval consists of both matching of objects and matching of spatial relationships. This suggestion for general-purpose image database systems is particularly suitable for building huge, heterogeneous and distributed image databases.

1. Introduction. Image database systems have been very active over the past 20 years. With the explosive interest for the last 10 years in multimedia systems, content-based image retrieval has attracted the attention of researchers across several disciplines [7], including computer vision, pattern recognition, human-computer interaction, and image databases. Applications that use image databases include office automation, computer-aided design, robotics, art museums, geographic information systems, remote sensing and management of earth resources, law enforcement and criminal investigation, large medical image databases, digital library, 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 much developed for facilitating pictorial information retrieval from a pictorial database.

Tanimoto [10] suggested the use of picture icons as picture indexes, thus introducing the concept of iconic indexing. Subsequently, Chang et al. [3] developed the concept of iconic indexing by introducing the 2D string representation of the image. With the 2D string approach, the problem of pictorial information retrieval for 2D pictures becomes a problem of 2D string subsequence matching [3, 9]. Since then, the 2D string approach has been studied further in the literature (see, e.g., [2]). Other methods on image representation and retrieval can also be found in the literature (see, e.g., [1, 4, 5, 6, 8]).

Zhang [12, Chapter 3] defined two efficiently decidable classes of two-dimensional scenes, pseudo-symbolic pictures and extended pseudo-symbolic pictures, and introduced the extended pseudo-2D string representation for them. 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. Zhang et al. [13] then extended his work on the extended pseudo-symbolic pictures to work for the whole images. In that paper [13], they proposed a unified iconic indexing, the generalized combined 2D string representation, for images in image databases. Each 2D image is modelled as a generalized extended pseudo-symbolic picture, which has the GEP-2D string representation. Their proposed iconic indexing (GC-2D string representation) combines both the GEP-2D string representation and the usual 2D string representation to capture absolute and relative spatial relationships in the image. The result is better representation of spatial relationships in image databases. Spatial relationships may be classified into directional and topological relationships. The 2D string approach developed by Chang et al. [3] is based on (relative) directional spatial relationships: *left-of, right-of, above,* and *below.* Spatial relationships proposed in our work [12, 13, 15] can be (absolute) directional, (relative) directional, or (absolute) topological.

In their most recent paper [14], Zhang et al. proposed the important consistency problem about content-based image indexing and retrieval, and presented a model for Content-based Image Database Systems (CIDBS). They described major components of Image Database Systems, and demonstrated how these components work together to facilitate content-based image indexing and retrieval.

2. Image Indexing. In this section we propose a very general approach to iconic indexing of images in image database systems.

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.

Recently, Zhang et al. [14] formulated a model for 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.

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.

We assume that an image database is associated with a finite set of object indexing schemes $\mathcal{I} = \{ \boldsymbol{I}_\alpha \mid \alpha \in I \}$ and another finite set of object matching schemes $\mathcal{M} = \{ \boldsymbol{M}_\beta \mid \beta \in M \}$. Observe that, there must be an object matching scheme $\boldsymbol{M}_\beta$ in $\mathcal{M}$ corresponding to a particular object indexing scheme $\boldsymbol{I}_\alpha$ in $\mathcal{I}$. Given an image, we allow different image-processing algorithms to better process their specialized objects in the image. That is, we allow different ways of indexing (and thus matching) for objects in an image. Let p be an image containing totally n significant objects O_k, where $1 \le k \le n$. Now this means that each object O_k $(1 \le k \le n)$ in the image p is specifically processed by an object indexing scheme $\boldsymbol{I}_{\alpha_k} \in \mathcal{I}$, where $\alpha_1, \alpha_2, \ldots, \alpha_n$ in I are not necessarily distinct. Then we can integrate them with our existing approach of the generalized extended pseudo-2D (GEP-2D) string and generalized combined 2D (GC-2D) string representation of the image [12, 13, 15] to formulate a more general framework for iconic indexing and retrieval of images.

There are three advantages on allowing different ways of indexing for objects in a particular image. First, it would be better to utilize different image-processing algorithms to process their specialized objects and (generalized) local scenes in the image. Second, it is natural to use this method to handle images with embedded iconic subimages. For example, it is often the case for screen images shown in the TV news, where one screen image may contain another embedded subimage. Note that an iconic subimage embedded in an image may be quite different from the rest of the image. An embedded iconic subimage should then be separately processed by image-processing algorithms from the rest of the image. Third, the integration of this method and the GEP-2D string and GC-2D string approach provides a unified framework for iconic indexing and retrieval of images. This suggestion for general-purpose image database systems is particularly suitable for building huge, heterogeneous and distributed image databases.

Figure 1 shows the pseudo-symbolic picture g and the reduced symbolic picture g_s of g, representing an image containing totally eight significant objects. This image has four distinct objects $A, B, C,$ and D, processed by, respectively, object indexing schemes $\boldsymbol{I}_{\alpha_1}, \boldsymbol{I}_{\alpha_2}, \boldsymbol{I}_{\alpha_3},$ and $\boldsymbol{I}_{\alpha_4}$ in $\mathcal{I}$, where $\alpha_1, \alpha_2, \alpha_3,$ and α_4 in I are not necessar-

C inside B		
	C_1 outside D	B_1
A overlaps A_1		A_2

e_1		
	e_2	e_4
e_0		e_3

(a) (b)
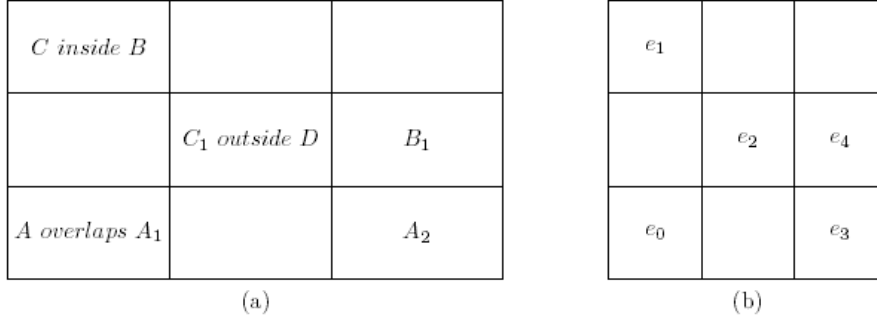
FIG. 1. *An image: (a). its pseudo-symbolic and (b). reduced symbolic representations.*

		A_1 overlaps A_2	
C inside B		D	
	C_1		
A			

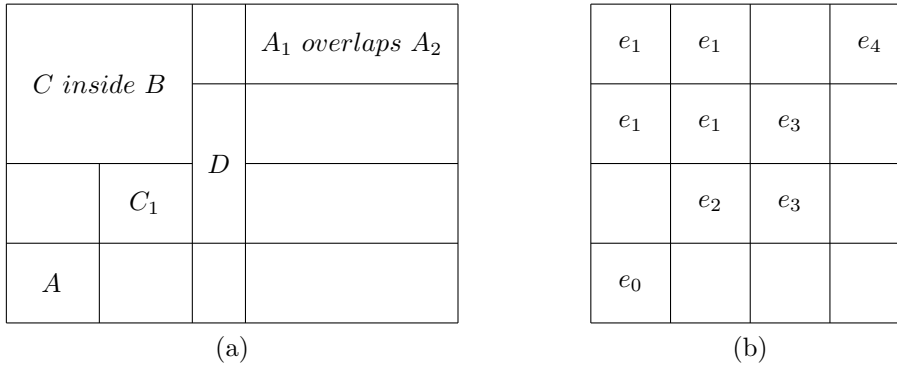|			
e_1	e_1		e_4
e_1	e_1	e_3	
	e_2	e_3	
e_0			

(a) (b)

FIG. 2. *An image: (a). its extended pseudo-symbolic and (b). reduced symbolic representations.*

ily distinct. Three occurrences of the same object A are represented by A_0 (simply A), A_1, and A_2, respectively. Two occurrences of the same object B are represented by B_0 (simply B) and B_1, respectively. Two occurrences of the same object C are represented by C_0 (simply C) and C_1, respectively. The object D has a single occurrence in the given image. Then, the pseudo-2D string representation of g is $(u, v) = (e_0 e_1 \; < \; e_2 \; < \; e_3 e_4, e_0 e_3 \; < \; e_2 e_4 \; < \; e_1)$, where e_0, e_1, e_2, e_3, and e_4, respectively, point to the local scenes $\{A$ overlaps $A_1\}$, $\{C$ inside $B\}$, $\{C_1$ outside $D\}$, "A_2", and "B_1".

Figure 2 shows the extended pseudo-symbolic picture h and the reduced symbolic picture h_s of h, representing an image containing totally seven significant objects. This image has four distinct objects A, B, C, and D. The objects A, B, C, and D are the same as those in Figure 1. Three occurrences of the same object A are represented by A_0 (simply A), A_1, and A_2, respectively. Two occurrences of the same object C are represented by C_0 (simply C) and C_1, respectively. The objects B and D, respectively, have a single occurrence in the given image. Then, the extended pseudo-2D string representation of h is $(u, v) = (e_0 e_1 \; < \; e_1 e_2 \; < \; e_3 \; < \; e_4, e_0 \; < \; e_2 e_3 \; < \; e_1 e_3 \; < \; e_1 e_4)$,

where e_0, e_1, e_2, e_3, and e_4, respectively, point to the local scenes "A", $\{C \ inside \ B\}$, "C_1", "D", and $\{A_1 \ overlaps \ A_2\}$.

To construct examples of generalized pseudo-symbolic pictures and generalized extended pseudo-symbolic pictures, the reader may refer to [13].

3. Image Retrieval. In this section we discuss image retrieval in image database systems, based on our general approach to iconic indexing of images proposed in Section 2.

Image Matching is an important component that retrieves images by content. After an image query is sent to the Image Matching Component, the query-processing mechanism will 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 consists of both matching of objects and matching of spatial relationships. Finally, a finite set (possibly null) of images matched by the query image will be sent to the Human-Computer Interface.

In many applications, such as geographic information systems, spatial relationships between objects are the criteria for image retrieval. Example queries are "find images containing a highway which is to the left of a forest" and "find images containing a lake in front of a mountain." For these types of applications, objects and their spatial relationships in a real picture have to be determined. With our proposed generalized extended pseudo-2D (GEP-2D) string approach, a database is assumed to consist of images associated with their (generalized/extended) pseudo-2D strings.

The picture-matching algorithm consists of both matching of objects and matching of spatial relationships. Given a query picture Q and a picture f' stored in the database, the matching step between objects in Q and objects in f' will check all possible pairs of an object in Q and an object in f'. Let Q have totally m significant objects O_k^Q $(1 \leq k \leq m)$ and f' have totally n significant objects $O_l^{f'}$ $(1 \leq l \leq n)$. Then the step for matching of objects will yield that: each object O_k^Q $(1 \leq k \leq m)$ in Q matches a possibly empty subset of objects $O_l^{f'}$ $(1 \leq l \leq n)$ in f'. An object O^Q in the query image Q matches an object $O^{f'}$ in a stored image f', denoted by $O^Q \longrightarrow O^{f'}$, if the picture-matching process determines the matching between O^Q and $O^{f'}$ using an object matching scheme $\boldsymbol{M}_\beta$ in $\mathcal{M}$. If there exists an object O^Q in Q matching none of objects $O^{f'}$ in f', then the stored image f' is clearly not matched by the given query image Q, and the picture-matching process will continue on the next available image stored in the database. Otherwise, that is, if every object O^Q in Q matches at least one object $O^{f'}$ in f', then the step for matching of spatial relationships will follow for further picture-matching between the query image Q and the stored image f'.

There are two general ways for content-based image retrieval: exact picture matching and approximate picture matching.

3.1. The Exact Picture Matching. The 2D string approach transforms the image retrieval into a 2D string matching problem. Chang et al. [3] defined type-0, type-1, and type-2 picture matchings. Similarly, with generalized (extended) pseudo-2D strings, we can define type-0, type-1 and type-2 picture matchings for generalized (extended) pseudo-symbolic pictures; and with (extended) pseudo-2D strings, we can define type-1 and type-2 picture matchings for (extended) pseudo-symbolic pictures, however, type-0 picture matching is not applicable because of the "absolute" nature of spatial relationships. Informally, a generalized pseudo-symbolic (generalized extended pseudo-symbolic, respectively) picture f is a *type-2* subpicture of a generalized pseudo-symbolic (generalized 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'; f is a *type-0* subpicture of f' if f is a type-1 subpicture of f_1, where f_1 is the picture derived from f' by gathering some adjacent rows and some adjacent columns in f'. Accordingly, 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 ($i = 0, 1, 2$) 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 (generalized) local scene in Q is part (i.e., a subscene) of the corresponding (generalized) 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 (generalized) local scene in Q is part (i.e., a subscene) of the corresponding (generalized) local scene in f.

- *Type-0 picture matching*: Given a query picture Q, Q matches a picture f' stored in the database if there exists a type-0 subpicture f of f' such that both Q and f have the same matrix or regular partition configuration and each (generalized) local scene in Q is part (i.e., a subscene) of the corresponding (generalized) local scene in f.

Observe that the type-2 picture matching is a special part of the type-1 picture matching and the type-0 picture matching is more general than the type-1 picture matching. We also call, both the type-1 and type-0 picture matchings, *the exact picture matching*. Notice that the picture matching problem can be considered as a two-level subpicture matching problem, with level-1 subpicture matching for the (generalized) local scenes, and level-2 subpicture matching for the reduced symbolic pictures, where each (gen-
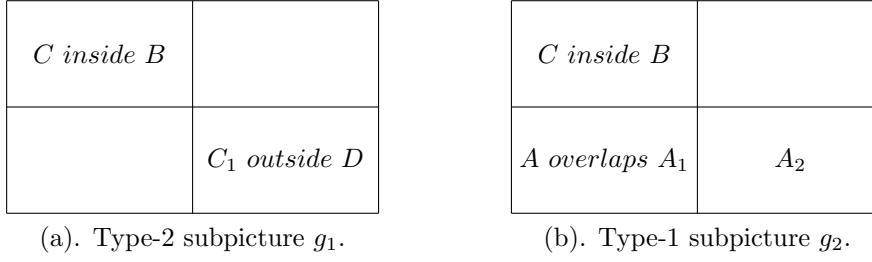
| C inside B | |
| | C_1 outside D |

(a). Type-2 subpicture g_1.

| C inside B | |
| A overlaps A_1 | A_2 |

(b). Type-1 subpicture g_2.

FIG. 3. *Subpicture examples for a pseudo-symbolic picture.*

| C inside B | |
| C_1 | D |

(a). Type-2 subpicture h_1.

| C inside B | D |
| A | |

(b). Type-1 subpicture h_2.

| C inside B | |
| | D |

(c). Type-1 subpicture h_3.

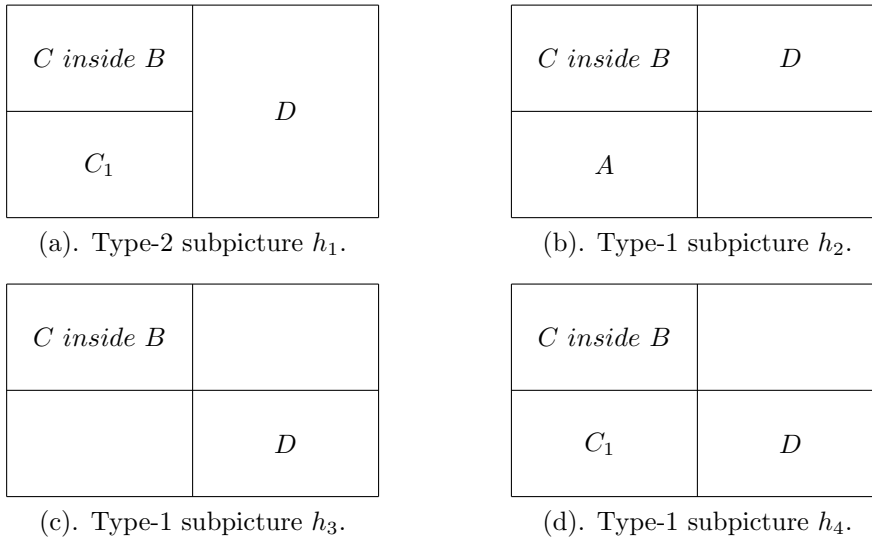| C inside B | |
| C_1 | D |

(d). Type-1 subpicture h_4.

FIG. 4. *Subpicture examples for an extended pseudo-symbolic picture.*

eralized) 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.

Consider, for example, the subpicture instances in Figure 3 for the pseudo-symbolic picture g in Figure 1. Figures (a) and (b) in Figure 3 show, respectively, a type-2 subpicture g_1 and a type-1 subpicture g_2 of the pseudo-symbolic picture g.

Figure 4 shows the subpicture instances for the extended pseudo-symbolic picture h in Figure 2. Figures (a) and (b) in Figure 4 show, respectively, a type-2 subpicture h_1 and a type-1 subpicture h_2 of the extended pseudo-symbolic picture h.

Because of the absolute spatial relationships introduced in the extended pseudo-symbolic pictures, sometimes the type-1 subpictures may capture some absolute spatial relationships which are not satisfied by the original extended pseudo-symbolic picture. Consider, for example, the type-1 subpicture h_3 in Figure 4 (c), and the type-1 subpicture h_4 in Figure 4 (d), of the extended pseudo-symbolic picture h in Figure 2. Both two subpictures h_3 and h_4 capture the *below* relationship between

the two local scenes "C $inside$ B" and "D", which is not satisfied by h. Note that, with the extended pseudo-2D (EP-2D) string approach, this kind of anomaly is not applicable to any type-2 subpictures of an extended pseudo-symbolic picture and the case for subpictures of a pseudo-symbolic picture; that is, this kind of anomaly can only happen to the case for type-1 subpictures of an extended pseudo-symbolic picture. However, this kind of anomaly is still fine for image retrieval by EP-2D strings, as long as we do not lose any image which is matched by the user query, since it only increases the instances matched by the user query. In fact, we can automatically detect and discard those extra "unwanted" instances caused by this type of anomaly, by adding one checking step into the picture-matching process. This checking step is done as follows: We can invoke the picture-matching process to perform the type-1 picture-matching between a query image Q and an image instance f' stored in the image database (assume f' is the extended pseudo-symbolic picture modelling this stored image instance). If a type-1 subpicture f of f' is matched by the given query image Q (note that the picture-matching process without this checking step would then take f' as a matched instance against the query image Q), then we will check whether there exists any $left$-of or $below$ relationship between two local scenes in the query image Q, whose corresponding relationship in f is not satisfied by f'. If there doesn't exist such a "bad" relationship, then f' is a matched instance against the query image Q. Otherwise, the type-1 subpicture f of f' could not become a witness for qualifying f' as a matched instance, and we need to continue searching another possible type-1 subpicture of f' being matched by the given query image Q.

Because of the close connection between the (generalized/extended) pseudo-2D string and the usual augmented 2D string, we can adapt existing picture-matching algorithms (see, e.g., [3]) originally developed for the augmented 2D string to work on the (generalized/extended) pseudo-2D string.

For the rest of this Section 3.1, we present an example to demonstrate the retrieval process using the exact picture matching (i.e., the type-1 picture matching) with (extended) pseudo-2D strings.

With the extended pseudo-2D string approach, a database is assumed to consist of images associated with their (extended) pseudo-2D strings, and user queries are limited to retrieve images which have (extended) pseudo-symbolic picture representations.

For simplicity, consider an example database consisting of only two images, respectively, in Figure 1 and Figure 2. Both two images are modelled by the pseudo-symbolic picture g in Figure 1 and the extended pseudo-symbolic picture h in Figure 2 respectively.

A user query can be described by presenting the content-based information about objects and their spatial relationships in a query image. Then the content-based information can be efficiently transformed into the (extended) pseudo-2D string rep-
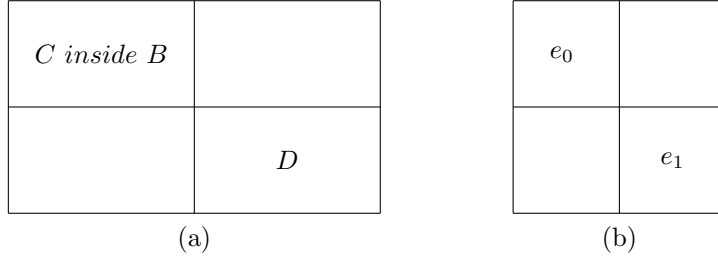
C inside B	
	D

e_0	
	e_1

(a)   (b)

FIG. 5. *A user query: (a). its pseudo-symbolic and (b). reduced symbolic representations for the "minimal" query image.*

resentation using both algorithms for deciding (extended) pseudo-symbolic pictures in [12, Chapter 3].

Consider, for example, a user query Q: "find images containing an object D which is in the below-right position of an object B containing an object C." Figure 5 shows the pseudo-symbolic picture representing the *minimal* query image, and its reduced symbolic picture for the given query Q. Here we use "minimal" to indicate that a minimal query image contains exactly what a given user query asks for, so any stored image containing the minimal query image will be matched by the given user query. The objects B, C, and D in Figure 5 are the same as those in Section 2. Then, the pseudo-2D string representation of Q is $(u, v) = (e_0 < e_1, e_1 < e_0)$, where e_0 and e_1 point to the local scenes $\{C \text{ inside } B\}$ and "D" respectively.

After a user query enters, the query processing mechanism will perform matching between the query image and a stored image in the database, based on (extended) pseudo-2D strings using the exact picture matching (i.e., type-1 picture matching). Notice that, a pseudo-symbolic (an extended pseudo-symbolic, respectively) picture can be uniquely reconstructed from a pseudo-2D (an extended pseudo-2D, respectively) string, mentioned in [12, Chapter 3]. In our example, the pseudo-symbolic picture representation for the user query Q in Figure 5, and g in Figure 1 and h in Figure 2 can be uniquely reconstructed from their (extended) pseudo-2D strings during the query processing. Then a two-level subpicture matching will be executed, with level-1 subpicture matching for the local scenes and level-2 subpicture matching for the reduced symbolic pictures. For the level-1 subpicture matching, all possible pairs of a local scene in the query image and a local scene in a stored image are checked for matching, where a local scene e_1 in the query image matches a local scene e_2 in a stored image, denoted by $e_1 \longrightarrow e_2$, if e_1 is part (i.e., a subscene) of e_2.

In our example, the step for matching of objects will yield that:

- Q and g: $B^Q \longrightarrow B^g$; $C^Q \longrightarrow C^g$; and $D^Q \longrightarrow D^g$.
- Q and h: $B^Q \longrightarrow B^h$; $C^Q \longrightarrow C^h$; and $D^Q \longrightarrow D^h$.

Note that the matching, $O^Q \longrightarrow O^{f'}$, between an object O^Q in the query image Q and an object $O^{f'}$ in a stored image f' is determined by the picture-matching process using an object matching scheme $\boldsymbol{M}_\beta$ in $\mathcal{M}$.

We present below the matching results of the level-1 subpicture matching for our example, where e_i^f denotes a local scene e_i in a given (extended) pseudo-symbolic picture f.

- Q and g: $e_0^Q \longrightarrow e_1^g$; and $e_1^Q \longrightarrow e_2^g$.
- Q and h: $e_0^Q \longrightarrow e_1^h$; and $e_1^Q \longrightarrow e_3^h$.

Figure 6 shows the level-1 subpicture matching between the query Q and the stored image g.
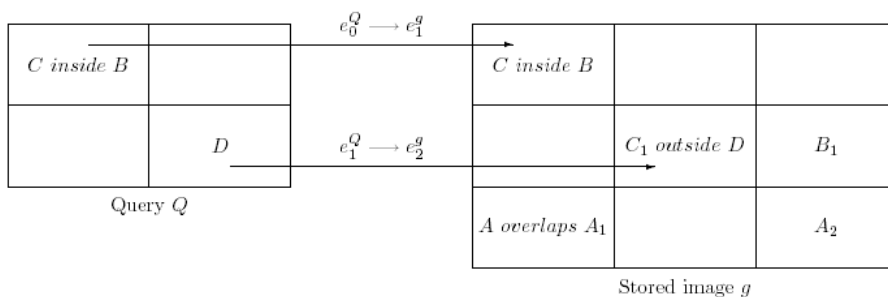

FIG. 6. *An example for the level-1 subpicture matching.*

A local scene in a query image may match several local scenes in a given (extended) pseudo-symbolic picture. Consider, for example, a query image having a local scene $e =$ "A" (i.e., the local scene with an object A only), where the object A is the same as that in Section 2. Then e can match two local scenes e_0 and e_3 in g, and two local scenes e_0 and e_4 in h.

A local scene in a query image may not match any local scenes in a given (extended) pseudo-symbolic picture. If this happens, the given (extended) pseudo-symbolic picture will not be a candidate matched by the query image and no further query processing for this stored image will be needed. Otherwise, that is, if every local scene in the query image matches at least one local scene in a given stored image, then the query processing mechanism continues on the level-2 subpicture matching for the reduced symbolic pictures.

For the level-2 subpicture matching, we can adapt existing picture-matching algorithms (see, e.g., [3]) originally developed for the symbolic pictures to work for the reduced symbolic pictures. Observe that a reduced symbolic picture is much simpler than a symbolic picture, in the sense that each slot of the matrix in a reduced symbolic picture can have at most one symbol (i.e., one super-symbol in our approach); but a symbolic picture in [3] can generally have several symbols in one slot of its matrix.
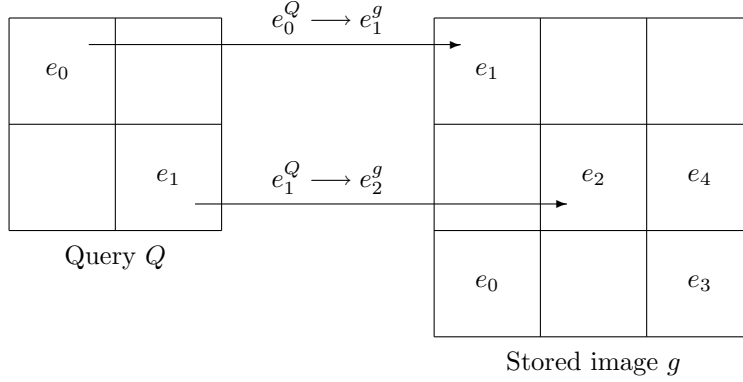
FIG. 7. *An example for the level-2 subpicture matching.*

The type-2 (type-1 automatically) subpicture g_1 in Figure 3 (a) will be identified during the level-2 subpicture matching to be a witness for the query image of Q matching the pseudo-symbolic picture g. This is illustrated in Figure 7. The level-2 subpicture matching will also yield that, both type-1 subpictures h_3 and h_4, respectively, in Figure 4 (c) and (d), will be identified in certain order to be two possible witnesses for the query image of Q matching the extended pseudo-symbolic picture h. As mentioned earlier in this Section, however, both h_3 and h_4 are the type-1 subpicture examples of the certain anomaly, and will be further checked out by the query processing mechanism to be false witnesses. Then the query processing mechanism will try to search another possible type-1 witness and no more will be found, yielding that the extended pseudo-symbolic picture h is not a candidate matched by the query image.

Therefore, the query processing mechanism will output the image in Figure 1, which is the only image, stored in the example database, matched by the user query Q.

There is a possibility that a local scene in a query image may match the combination of several local scenes in a stored image. Consider, for example, a user query: "find images containing an object A which is outside an object C." The objects A and C here are the same as those in Section 2. This user query obviously matches both images in Figure 1 and Figure 2, but either one of type-1 and type-2 picture matchings cannot be applicable here. This case can be handled by the type-0 picture matching in [13], for the generalized extended pseudo-symbolic pictures modelling the whole images.

3.2. The Approximate Picture Matching. Tucci et al. [11] proved that the type-1 symbolic picture matching is *NP*-complete. It can also be proved that the exact picture matching problem with (generalized/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.

4. Conclusion. In this paper we have proposed to allow different ways of indexing and matching for objects in an image, and integrate them with our existing approach of the generalized extended pseudo-2D string and generalized combined 2D string representation of the image to formulate a more general framework for iconic indexing and retrieval of images in image database systems. The query processing of image retrieval consists of both matching of objects and matching of spatial relationships. This suggestion for general-purpose image database systems is particularly suitable for building huge, heterogeneous and distributed image databases.

Because of *NP*-completeness of the exact picture matching, further research is required to address the issue of using approximate picture matching to handle an image query in image databases. Future research is also required to further investigate the Content-based Image Database Systems (CIDBS) model, recently presented in [14], for facilitating fast image indexing and retrieval.

REFERENCES

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

[2] S.-K. Chang and E. Jungert, *Symbolic Projection for Image Information Retrieval and Spatial Reasoning*, Academic Press, 1996.

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] V. N. Gudivada and V. V. Raghavan, *Content-based Image Retrieval Systems*, IEEE Computer, pp. 18–22, September 1995.

[8] 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.

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

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

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

[12] 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.

[13] 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.

[14] 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.

[15] 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.