

# A Generalized Prime-Number-Based Matrix Strategy for Efficient Iconic Indexing of Symbolic Pictures <sup>1</sup>

Ye-In Chang<sup>†</sup>, Bi-Yen Yang<sup>‡</sup> and Wei-Horng Yeh<sup>†</sup>

<sup>†</sup>Dept. of Computer Science and Engineering  
National Sun Yat-Sen University  
Kaohsiung, Taiwan  
Republic of China  
{E-mail: changyi@cse.nsysu.edu.tw}  
{Tel: 886-7-5252000 (ext. 4334)}  
{Fax: 886-7-5254301}

<sup>‡</sup>Dept. of Applied Mathematics  
National Sun Yat-Sen University  
Kaohsiung, Taiwan  
Republic of China

## Abstract

In this paper, we propose an efficient iconic indexing strategy called *Generalized Prime-Number-based Matrix* (GPN Matrix) for symbolic pictures, in which each spatial relationship between any two objects is represented as a product of some prime numbers from a set of 12 prime numbers and is recorded in a matrix. In the proposed strategy, we classify 169 spatial relationships between two objects in 2D space into five spatial categories, and define a *generalized category rule* (based on module operations) for each of those five spatial categories. As compared to the Prime-Number-based Matrix (PN Matrix) strategy [5], in which each spatial relationship between any two objects is represented as a product of some prime numbers from a set of 17 prime numbers, the GPN Matrix strategy has a smaller storage space requirement than the PN Matrix strategy, which also improves the query processing time.

(*Keywords:* 2D string, 2D C-string, image databases, pictorial query, pictorial databases, similarity retrieval, spatial reasoning)

---

<sup>1</sup>This research was supported in part by the National Science Council of Republic of China under Grant No. NSC-87-2213-E-110-014.

# 1 Introduction

One of the most important problems in the design of multimedia systems is how images (or symbolic pictures) are stored in the database [2, 12]. In traditional database systems, the use of indexing to allow database access has been well established. Analogously, pictorial indexing techniques are needed to make ease pictorial information retrieval from a pictorial database.

Over the last decade, many approaches to represent symbol data have been proposed, for example, *2D-string* [2, 3], *2D C-string* [8, 10, 11], and *9DLT matrix* [4]. Based on those representations, several algorithms in pictorial querying, spatial reasoning and similarity retrieval are proposed, where pictorial querying allows the users to query images with a specified spatial relationship, spatial reasoning means the inference of a consistent set of spatial relationships among the objects in an image, and the target of similar retrieval is to retrieve the images that are similar to the query image.

In [5], Chang and Yang proposed a prime-number-based strategy (denoted as the PN Matrix strategy), which combines the advantages of the 2D-C string and 9DLT Matrix strategies. In this PN Matrix strategy, each spatial relationship between any two objects is represented as a product of some prime numbers from a set of 17 prime numbers and is recorded in a matrix. Moreover, the PN Matrix strategy classifies those 169 spatial relationships between two objects in 2D space into five spatial categories, including *disjoin*, *join*, *contain*, *belong* and *part-overlap*, by *category rules* which are efficient modulus-based operations. For example,  $2 \times 47$  is used to denote the "disjoin" spatial relationship between objects *A* and *B* along the *x*-axis (or *y*-axis), and a "mod 2" operation can answer whether objects *A* and *B* have the "disjoin" spatial relationship.

In this paper, we propose an efficient iconic indexing strategy called *Generalized Prime-Number-based Matrix* (GPN Matrix) for symbolic pictures, which is a revised version of the PN Matrix strategy. In the proposed GPN Matrix strategy, each spatial relationship between any two objects is represented as a product of some prime numbers from a set of 12 prime numbers and is recorded in a matrix. Moreover, we present the related *generalized category rules*. Next, we carefully assign a product of some of the 12 real prime numbers to each of the 13 spatial operators that denote those 13 spatial relationships, which tries to

make the maximum value of those spatial operators as small as possible. As compared to Chang and Yang’s PN Matrix strategy [5], in which 20 bits are needed to record the product of some of those 17 prime numbers (due to the maximum value of 13 spatial operators =  $17 \times 29 \times 37 \times 43 = 784363$ ), the proposed GPN Matrix strategy needs 14 bits to achieve the same goal (due to the maximum value of 13 spatial operators =  $2 \times 3 \times 7 \times 11 \times 23 = 10626$ ). Furthermore, based on a different approach to handle the *join* category, we modify our GPN Matrix strategy to GPN\*, so that only 11 prime numbers and 11 bits are needed (due to the maximum value of 13 spatial operators =  $2 \times 5 \times 7 \times 19 = 1330$ ). Based on the current technology, a short integer is represented in 16 bits and an integer is represented in 32 bits, the proposed GPN (and GPN\*) strategy can reduce the storage space requirement (needed in the PN strategy) from  $32 \times N^2$  bits to  $16 \times N^2$  bits per picture, where  $N$  is the number of objects in the symbolic picture, which also improves the query processing time.

The rest of the paper is organized as follows. In Section 2, we give a brief description of the PN Matrix representation. In Section 3, we will present the proposed generalized prime number matrix representation for symbolic pictures. Finally, Section 4 gives a conclusion.

## 2 Background

Table 1 shows the definition of the set of spatial operators that are used to denote spatial relationships, where the notation "begin( $A$ )" denotes the value of begin-bound of object  $A$  and "end( $A$ )" denotes the value of end-bound of object  $A$  [8]. (Note that for an efficient storage and retrieval of the extended structure objects, in all of the previous approaches mentioned before, each object of a picture is abstracted as a minimum bounded rectangle (MBR).) According to the begin-bound and end-bound of the picture objects, spatial relationships between two enclosing rectangles along the  $x$ -axis (or  $y$ -axis) can be categorized into 13 types ignoring their length. Therefore, there are 169 types of spatial relationships between two rectangles in 2D space, as shown in Figure 1, where  $operator^*$  denotes the inverse operator of the related  $operator$ , for example,  $A < B$  implying  $B <^* A$ . They can be categorized in five types, *disjoin*, *join*, *contain*, *belong* and *partial overlap*. The five types of the spatial relations between objects are defined in Figure 2. The measure criteria for

Notation	Condition	Meaning
$A < B$	$\text{end}(A) < \text{begin}(B)$	A disjoins B
$A = B$	$\text{begin}(A) = \text{begin}(B)$ $\text{end}(A) = \text{end}(B)$	A is the same as B
$A   B$	$\text{end}(A) = \text{begin}(B)$	A is edge to edge with B
$A \% B$	$\text{begin}(A) < \text{begin}(B)$ $\text{end}(A) > \text{end}(B)$	A contains B and they have not the same bound
$A [ B$	$\text{begin}(A) = \text{begin}(B)$ $\text{end}(A) > \text{end}(B)$	A contains B and they have the same begin bound
$A ] B$	$\text{begin}(A) < \text{begin}(B)$ $\text{end}(A) = \text{end}(B)$	A contains B and they have the same end bound
$A / B$	$\text{begin}(A) < \text{begin}(B)$ $< \text{end}(A) < \text{end}(B)$	A is partly overlapping with B

Table 1: Definitions of Lee’s spatial operators

categorization is the area of the intersection of  $A$  and  $B$ .

Suppose  $A$  and  $B$  are two objects in a picture  $f$ , and the spatial relationship between them in terms of  $x$ -axis and  $y$ -axis is  $(Ar_{A,B}^x B, Ar_{A,B}^y B)$ , where  $r_{A,B}^x$  and  $r_{A,B}^y$  are the spatial operators in Table 1. By observing the five spatial categories among the 169 spatial relationships in Figure 1, Chang and Yang [5] discovered some interesting and useful results as shown in Figure 3. Based on the observation, the PN Matrix strategy can support efficient spatial reasoning by making use of the Prime-Number-based (PN) Matrix. The major step is to assign each spatial operator a unique number (= a product of some prime numbers) according to these five spatial categories.

The assignments of spatial-operator-values for these 13 spatial operators are shown in Figure 4, and the five category rules based on the module operation are shown in Figure 5. Note that in Figure 5, let  $sov(r)$  be the *spatial-operator-value* of  $r$  (i.e., the product of some prime numbers as shown in Figure 4) and  $sv(A, B) = sov(r_{A,B}^x) \times sov(r_{A,B}^y)$  as the *spatial-value* of the objects pair  $(A, B)$ .

For the symbolic picture shown in Figure 6, the corresponding spatial matrix  $S$  is shown as follows:

$$S = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \left[ \begin{array}{ccccc} 0 & | & \% & \%^* & /^* \\ < & 0 & <^* & \%^* & <^* \\ < & | & 0 & \%^* & /^* \\ <^* & <^* & <^* & 0 & [ \\ \%^* & /^* & <^* & < & 0 \end{array} \right] \end{matrix}$$

where the lower triangular matrix stores the spatial information along the  $x$ -axis, and

Disjoin (48)	Join (40)	Part_overlap (50)	Contain (16)	Belong (16)

$<$	: $2 \times 47$ (= 94)	$<^*$	: $2^2$ (= 4)
$ $	: $3 \times 53$ (= 159)	$ ^*$	: $3^2$ (= 9)
$\%$	: $5 \times 7 \times 37 \times 41$ (= 53095)	$\%^*$	: $17 \times 19 \times 37 \times 43$ (= 513893)
$[$	: $5 \times 13 \times 37 \times 41$ (= 98605)	$[^*$	: $17 \times 29 \times 37 \times 43$ (= 784363)
$]$	: $5 \times 11 \times 37 \times 41$ (= 83435)	$]^*$	: $17 \times 23 \times 37 \times 43$ (= 622081)
$/$	: $31 \times 37 \times 59$ (= 67673)	$/^*$	: $31^2 \times 37$ (= 35557)
$=$	: $5 \times 17 \times 37$ (= 3145)		

Figure 4: The assignments of those 13 spatial operators in the PN Matrix strategy by using 17 prime numbers

<b>Disjoin</b>	if $sv(A, B) \bmod 2 = 0$ .
<b>Join</b>	if $sv(A, B) \bmod 2 \neq 0$ , and $sv(A, B) \bmod 3 = 0$ .
<b>Contain</b>	if $sv(A, B) \bmod 5^2 = 0$ .
<b>Belong</b>	if $sv(A, B) \bmod 17^2 = 0$ .
<b>Part_overlap</b>	if $sv(A, B) \bmod (31 \times 37^2) = 0$ , or $sv(A, B) \bmod (41 \times 43) = 0$ .

Figure 5: Category rules in the PN Matrix strategy

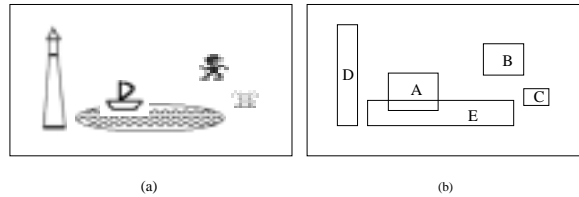


Figure 6: An image and its symbolic representation

the upper triangular matrix stores the spatial information along the  $y$ -axis. That is,  $S[v_i, v_j] = r_{j,i}^x$  if  $i > j$ ;  $S[v_i, v_j] = r_{i,j}^y$  if  $i < j$ ;  $S[v_i, v_j] = 0$  if  $i = j$ ,  $\forall v_i, v_j \in V$ ,  $\forall r_{j,i}^x, r_{i,j}^y \in A$ ,  $1 \leq i, j \leq m$ , where  $r_{j,i}^x$  is the spatial operator between objects  $v_i$  and  $v_j$  along the  $x$ -axis and  $r_{i,j}^y$  is the spatial operator between objects  $v_i$  and  $v_j$  along the  $y$ -axis. Note that in this representation, we always record the relationships between two objects  $v_i$  and  $v_j$  from the view point of object  $v_i$  no matter along the  $x$ -axis or the  $y$ -axis, where  $i < j$ . That is why  $S[v_i, v_j] = r_{j,i}^x$  when  $i > j$ . For instance, for the example shown in Figure 6, the spatial relationship between  $A$  and  $B$  along the  $x$ -axis is  $A < B$ .

According to the assignments of spatial-operator-values for those 13 spatial operators described before, the PN Matrix strategy can transform the spatial matrix  $S$  of  $f$  into a *PN Matrix*  $T$  by replacing each spatial operator  $r_{j,i}^x$  ( $r_{i,j}^y$ ) with its unique spatial-operator-value as follows:

$$T = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{bmatrix} 0 & 3 \times 53 & 5 \times 7 \times 37 \times 41 & 17 \times 19 \times 37 \times 43 & 31^2 \times 37 \\ 2 \times 47 & 0 & 2^2 & 17 \times 19 \times 37 \times 43 & 2^2 \\ 2 \times 47 & 3 \times 53 & 0 & 17 \times 19 \times 37 \times 43 & 31^2 \times 37 \\ 2^2 & 2^2 & 2^2 & 0 & 5 \times 13 \times 37 \times 41 \\ 17 \times 19 \times 37 \times 43 & 31^2 \times 37 & 2^2 & 2 \times 47 & 0 \end{bmatrix} \end{matrix}$$

In this example, since  $sv(A, B) \bmod 2 = ((2 \times 47) \times (3 \times 53)) \bmod 2 = 0$ , objects  $A$  and  $B$  have the "disjoin" spatial relationship.

### 3 The GPN Matrix Strategy

Suppose  $r$  is a spatial operator in the set  $\{<, <^*, |, |^*, [, [^*, ], ]^*, \%, \%^*, /, / ^*, =\}$  and  $A, B$  are two objects in the symbolic picture. We define  $gsov(r)$  as the *generalized spatial-operator-value* of  $r$  with a initial value 1, and  $gsv(A, B) = gsov(r_{A,B}^x) \times gsov(r_{A,B}^y)$  as the *generalized spatial-value* of the objects pair  $(A, B)$ . Based on the same observation of those five spatial categories as shown in Figure 3, we have a different way of assignments. Assume a prime number set  $P = \{p_1, p_2, p_3, \dots, p_{12}\}$ , where there is no relationship between  $p_i$  and  $p_j$ ,  $1 \leq i, j \leq 12$ . That is,  $P$  is not an ordered set.

To classify the *disjoin* category, the prime number  $p_1$  is applied. That is,

$$\begin{aligned} gsov(<) &:= gsov(<) \times p_1; \\ gsov(<^*) &:= gsov(<^*) \times p_1^2. \end{aligned}$$

Since only  $gs\text{ov}(<)$  and  $gs\text{ov}(<^*)$  are the multiples of  $p_1$ , one  $(gs\text{v}(A, B) \bmod p_1)$  operation can determine whether one or both the  $r_{A,B}^x, r_{A,B}^y$  spatial operators are in the set  $\{<, <^*\}$ . (Note that to distinguish spatial operators " $<$ " and " $<^*$ ", we let  $gs\text{ov}(<^*)$  multiply with one more  $p_1$ .) That is, one  $(gs\text{v}(A, B) \bmod p_1)$  operation can determine the *disjoin* category.

To classify the *join* category, the prime number  $p_2$  is applied. That is,

$$\begin{aligned} gs\text{ov}(|) &:= gs\text{ov}(|) \times p_2; \\ gs\text{ov}(|^*) &:= gs\text{ov}(|^*) \times p_2^2. \end{aligned}$$

Since only  $gs\text{ov}(|)$  and  $gs\text{ov}(|^*)$  are the multiples of  $p_2$ , one  $(gs\text{v}(A, B) \bmod p_2)$  operation can determine whether one or both the  $r_{A,B}^x, r_{A,B}^y$  spatial operators are in the set  $\{ |, |^* \}$ . (Note that to distinguish spatial operators " $|$ " and " $|^*$ ", we let  $gs\text{ov}(|^*)$  multiply with one more  $p_2$ .) Moreover, none of the  $r_{A,B}^x, r_{A,B}^y$  spatial operator should be in the set  $\{<, <^*\}$ , so a prime number  $p_3$  is applied. (In other words, both of the  $r_{A,B}^x, r_{A,B}^y$  spatial operators should be in the set  $\{=, \%, \%^*, ], ]^*, [, [*], |, |^*, /, /^*\}$ ). That is,

$$gs\text{ov}(r) := gs\text{ov}(r) \times p_3, \quad \forall r \in \{=, \%, \%^*, ], ]^*, [, [*], |, |^*, /, /^*\}.$$

Finally, one  $(gs\text{v}(A, B) \bmod (p_2 \times p_3^2))$  operation can determine the *join* category.

To classify the *contain* category, the prime number  $p_4$  is applied. We multiply  $p_4$  to the generalized spatial-operator-value of the spatial operators which are in the set  $\{=, \%, ], [ \}$ . That is,

$$gs\text{ov}(r) := gs\text{ov}(r) \times p_4, \quad \forall r \in \{=, \%, ], [ \}.$$

From the observation of the *contain* category, one  $(gs\text{v}(A, B) \bmod p_4^2)$  operation can determine whether both the  $r_{A,B}^x, r_{A,B}^y$  spatial operators are in the set  $\{=, \%, ], [ \}$ . That is, one  $(gs\text{v}(A, B) \bmod p_4^2)$  operation can determine the *contain* category. But to distinguish these four symbols, three more prime numbers must be applied. Therefore,

$$\begin{aligned} gs\text{ov}(\%) &:= gs\text{ov}(\%) \times p_5; \\ gs\text{ov}()] &:= gs\text{ov}()] \times p_6; \\ gs\text{ov}([) &:= gs\text{ov}([) \times p_7. \end{aligned}$$

To classify the *belong* category, the prime number  $p_8$  is applied. We multiply  $p_8$  to the generalized spatial-operator-value of the spatial operators in the set  $\{=, \%^*, ]^*, [* \}$ . That is,



$$gs\text{ov}(r) := gs\text{ov}(r) \times p_8, \quad \forall r \in \{=, \%*, ], [* \}.$$

From the observation of *contain* category, one  $(gs\text{v}(A, B) \bmod p_8^2)$  operation can determine whether both the  $r_{A,B}^x, r_{A,B}^y$  spatial operators are in the set  $\{=, \%*, ], [* \}$ . That is, one  $(gs\text{v}(A, B) \bmod p_8^2)$  operation can determine the *contain* category. But to distinguish these four symbols, three more prime numbers must be applied. At this time, we re-use those prime numbers  $p_5, p_6$  and  $p_7$ . Although they are used before, it will not cause any problem here. Therefore,

$$gs\text{ov}(\%*) := gs\text{ov}(\%*) \times p_5;$$

$$gs\text{ov}( [* ]) := gs\text{ov}( [* ]) \times p_6;$$

$$gs\text{ov}([*]) := gs\text{ov}([*]) \times p_7.$$

To classify the *part\_overlap* category, let's consider the following two cases stated before. First, to determine whether one of the  $r_{A,B}^x, r_{A,B}^y$  spatial operators is in the set  $\{/, /*\}$ , and the other is in the set  $\{\%, [, ], /, =, \%*, [, ], /*\}$ , two prime numbers  $p_9$  and  $p_{10}$  are applied. That is,

$$gs\text{ov}(/) := gs\text{ov}(/) \times p_9;$$

$$gs\text{ov}(/*) := gs\text{ov}(/*) \times p_9^2;$$

$$gs\text{ov}(r) := gs\text{ov}(r) \times p_{10}, \quad \forall r \in \{\%, [, ], /, =, \%*, [, ], /*\}.$$

Therefore, one  $(gs\text{v}(A, B) \bmod (p_9 \times p_{10}^2))$  operation can determine whether one of the  $r_{A,B}^x, r_{A,B}^y$  spatial operators is in the set  $\{/, /*\}$ , and the other is in the set  $\{\%, [, ], /, =, \%*, \%*, [, ], /*\}$ . That is, one  $(gs\text{v}(A, B) \bmod (p_9 \times p_{10}^2))$  operation can determine the first case of the *part\_overlap* category. (Note that since "/" and "/\*" also appear in the set  $\{\%, [, ], /, =, \%*, \%*, [, ], /*\}$ ,  $gs\text{ov}(/)$  (and  $gs\text{ov}(/*)$ )  $\bmod (p_9 \times p_{10}) = 0$ .) Second, to determine whether one of the  $r_{A,B}^x, r_{A,B}^y$  spatial operators is in the set  $\{\%, [, ],$ , and the other is in the set  $\{\%*, ], [* \}$ , two prime numbers  $p_{11}$  and  $p_{12}$  are applied. That is,

$$gs\text{ov}(r) := gs\text{ov}(r) \times p_{11}, \quad \forall r \in \{\%, [, ],$$

$$gs\text{ov}(r) := gs\text{ov}(r) \times p_{12}, \quad \forall r \in \{\%*, ], [* \}.$$

Therefore, one  $(gs\text{v}(A, B) \bmod (p_{11} \times p_{12}))$  operation can determine whether one of the  $r_{A,B}^x, r_{A,B}^y$  spatial operators is in the set  $\{\%, [, ],$ , and the other is in the set  $\{\%*, ], [* \}$ . That is, one  $(gs\text{v}(A, B) \bmod (p_{11} \times p_{12}))$  operation can determine the second case of the *part\_overlap* category. Consequently, one  $(gs\text{v}(A, B) \bmod (p_9 \times p_{10}^2))$  operation and one

$<$	: $p_1 \times p_5$	$<^*$	: $p_1^2$
$ $	: $p_2 \times p_3 \times p_6$	$ ^*$	: $p_2^2 \times p_3$
$\%$	: $p_3 \times p_4 \times p_5 \times p_{10} \times p_{11}$	$\%^*$	: $p_3 \times p_5 \times p_8 \times p_{10} \times p_{12}$
$[$	: $p_3 \times p_4 \times p_7 \times p_{10} \times p_{11}$	$[^*$	: $p_3 \times p_7 \times p_8 \times p_{10} \times p_{12}$
$]$	: $p_3 \times p_4 \times p_6 \times p_{10} \times p_{11}$	$]^*$	: $p_3 \times p_6 \times p_8 \times p_{10} \times p_{12}$
$/$	: $p_3 \times p_7 \times p_9 \times p_{10}$	$/^*$	: $p_3 \times p_9^2 \times p_{10}$
$=$	: $p_3 \times p_4 \times p_8 \times p_{10}$		

Figure 7: The assignments of those 13 spatial operators in the GPN Matrix strategy

<b>Disjoin</b>	if $gsv(A, B) \bmod p_1 = 0$ .
<b>Join</b>	if $gsv(A, B) \bmod (p_2 \times p_3^2) = 0$ .
<b>Contain</b>	if $gsv(A, B) \bmod p_4^2 = 0$ .
<b>Belong</b>	if $gsv(A, B) \bmod p_8^2 = 0$ .
<b>Part_overlap</b>	if $gsv(A, B) \bmod (p_9 \times p_{10}^2) = 0$ , or $gsv(A, B) \bmod (p_{11} \times p_{12}) = 0$ .

Figure 8: Generalized category rules

$(gsv(A, B) \bmod (p_{11} \times p_{12}))$  operation can determine the *part\_overlap* category.

According to the above descriptions, we have assigned each spatial operator a unique value which can be used to determine different spatial categories efficiently. However, in order to determine the spatial relationships between any two objects efficiently, we have to make each of the generalized spatial-operator-values of these spatial operators indivisible by the generalized spatial-operator-value of any other spatial operator. That is, no one generalized spatial-operator-value of a spatial operator is a multiple of the generalized spatial-operator-value of any other spatial operator. Therefore, we let  $gsov(<) := gsov(<) \times p_5$  to distinguish spatial operators  $<$  and  $<^*$ , let  $gsov(|) := gsov(|) \times p_6$  to distinguish spatial operators  $|$  and  $|^*$ , and let  $gsov(/) := gsov(/) \times p_7$  to distinguish spatial operators  $/$  and  $/^*$ . Finally, the assignments of generalized spatial-operator-values for these 13 spatial operators are shown in Figure 7, and the five generalized category rules based on the modulus operation are shown in Figure 8.

By carefully assign a different prime number to each  $p_i$  [14], we can assign 12 real prime numbers to those  $p_i$  as shown in Table 2. The complete assignment of those 13 spatial operators is shown in Figure 9, and the related category rules are shown in Figure

Old	$p_1$	$p_2$	$p_3$	$p_4$	$p_5$	$p_6$	$p_7$	$p_8$	$p_9$	$p_{10}$	$p_{11}$	$p_{12}$
New	37	31	2	5	17	19	23	7	29	3	13	11

Table 2: The relationships between the original assignments and the new assignments in the GPN Matrix strategy

$<$	: $17 \times 37$ ( = 629)	$<^*$	: $37^2$ ( = 1369)
$ $	: $2 \times 19 \times 31$ ( = 1178)	$ ^*$	: $2 \times 31^2$ ( = 1922)
$\%$	: $2 \times 3 \times 5 \times 13 \times 17$ ( = 6630)	$\%^*$	: $2 \times 3 \times 7 \times 11 \times 17$ ( = 7854)
$[$	: $2 \times 3 \times 5 \times 13 \times 23$ ( = 8970)	$[^*$	: $2 \times 3 \times 7 \times 11 \times 23$ ( = 10626)
$]$	: $2 \times 3 \times 5 \times 13 \times 19$ ( = 7410)	$]^*$	: $2 \times 3 \times 7 \times 11 \times 19$ ( = 8778)
$/$	: $2 \times 3 \times 23 \times 29$ ( = 4002)	$/^*$	: $2 \times 3 \times 29^2$ ( = 5046)
$=$	: $2 \times 3 \times 5 \times 7$ ( = 210)		

Figure 9: The assignments of those 13 spatial operators in the GPN Matrix strategy by using 12 prime numbers

10. (Note that in this assignment, we try to assign a smallest prime number to a  $p_i$  which occurs in the longest product of prime numbers (i.e., a product which has the largest number of prime numbers) and has the largest times of occurrence in all assignments. For example, we will assign 2 to  $p_3$ .) Based on this assignment, the maximum value is 10626 in the GPN Matrix strategy, which needs 14 bits to store it, as compared to 784363 in the assignment of the PN Matrix strategy as shown in Figure 4, which needs 20 bits to store it. Therefore, our GPN Matrix strategy requires a smaller storage space than the PN Matrix strategy.

If we apply the join category rule as described in the PN Matrix strategy [5] in which

<b>Disjoin</b>	if $gsv(A, B) \bmod 37 = 0$ .
<b>Join</b>	if $gsv(A, B) \bmod (2^2 \times 31) = 0$ .
<b>Contain</b>	if $gsv(A, B) \bmod 5^2 = 0$ .
<b>Belong</b>	if $gsv(A, B) \bmod 7^2 = 0$ .
<b>Part_overlap</b>	if $gsv(A, B) \bmod (3^2 \times 29) = 0$ , or $gsv(A, B) \bmod (11 \times 13) = 0$ .

Figure 10: Generalized category rules in the GPN Matrix strategy by using 12 prime numbers

two modulus (and comparison) operations are used (as shown in Figure 5), instead of one modulus (and comparison) operation used in the GPN Matrix strategy, we can use only 11 prime numbers to distinguish all spatial operators as shown in Figure 11. We call it the GPN\* Matrix strategy. The resulting category rules are shown in Figure 12. In the GPN\* Matrix strategy, the maximum value is 1330, which needs only 11 bits to record it, resulting in a smaller storage space requirement than the PN Matrix strategy. Based on the current technology, a short integer is represented in 16 bits and an integer is represented in 32 bits, the proposed GPN (and GPN\*) strategy can reduce the storage space requirement (needed in the PN strategy) from  $32 \times N^2$  bits to  $16 \times N^2$  bits per picture, where  $N$  is the number of objects in the symbolic pictures. Therefore, the storage space requirement both in the GPN Matrix and GPN\* Matrix strategies is smaller than that of the PN Matrix strategy.

To study the effect of the storage space requirement to the query processing time, we also do a simulation study. Our experiments were performed on a Pentium II with one CPU clock rate of 450 MHz, 256 MB of main memory, running under Window 2000 Professional with the Microsoft Visual C++ 6.0 compiler in a released mode. To simplify our simulation, we let the maximum number of different objects appearing in the database be 20. For each object, it can appear in a picture with  $100000 * 100000$  points [1]. We prepare 2000 pictures with each of them 10 different objects appearing in each picture, which are created randomly with a uniform distribution. (That is, for each object shown in a picture, we generate its left-top and right-bottom coordinates randomly.) Those 2000 pictures then represented in the GPN matrix and the PN matrix representation in the database in advance. For this case, we consider the query processing time of *type-0* similarity retrieval, where a picture is of *type-0* similarity if all the spatial category relationships of object pairs are the same to the query picture. We compare one input query picture represented in the GPN matrix (or PN matrix) with each of those prepared 2000 pictures in the database, respectively. We then compute the average cost for comparing the query picture with each of those 2000 pictures. (Note that this is the same case as we have 200 pictures in the database, and then we prepare 10 query pictures to compare with each of those 200 pictures stored in the database and compute the average cost of those  $200 * 10 = 2000$  tests. Therefore, the simulation result is the average cost of 2000 tests.) From the simulation result, the query

$<$	: $13 \times 29$ ( = 377)	$<^*$	: $29^2$ ( = 841)
$ $	: $13 \times 31$ ( = 403)	$ ^*$	: $31^2$ ( = 961)
$\%$	: $2 \times 3 \times 11 \times 13$ ( = 858)	$\%^*$	: $2 \times 5 \times 7 \times 13$ ( = 910)
$[$	: $2 \times 3 \times 11 \times 17$ ( = 1122)	$[^*$	: $2 \times 5 \times 7 \times 17$ ( = 1190)
$]$	: $2 \times 3 \times 11 \times 19$ ( = 1254)	$]^*$	: $2 \times 5 \times 7 \times 19$ ( = 1330)
$/$	: $2 \times 13 \times 23$ ( = 598)	$/^*$	: $2 \times 23^2$ ( = 1058)
$=$	: $2 \times 3 \times 5$ ( = 30)		

Figure 11: The assignments of those 13 spatial operators in the GPN\* Matrix strategy by using 11 prime numbers

<b>Disjoin</b>	if $sv(A, B) \bmod 29 = 0$ .
<b>Join</b>	if $sv(A, B) \bmod 29 \neq 0$ , and $sv(A, B) \bmod 31 = 0$ .
<b>Contain</b>	if $sv(A, B) \bmod 3^2 = 0$ .
<b>Belong</b>	if $sv(A, B) \bmod 5^2 = 0$ .
<b>Part_overlap</b>	if $sv(A, B) \bmod (2^2 \times 23) = 0$ , or $sv(A, B) \bmod (7 \times 11) = 0$ .

Figure 12: Category rules in the GPN\* Matrix strategy by using 11 prime numbers

processing time in the GPN strategy is 0.005 seconds per picture, while the query processing time in the PN strategy is 0.01 seconds per picture. Therefore, we show that the proposed GPN strategy also has better performance than the PN strategy in terms of time. (Note that although the PN and GPN (or GPN\*) strategies have the same total performance complexity ( $O(K * N^2)$ ) for query processing, where  $K$  is the number of pictures and  $N$  is the number of objects in the database, the difference of the storage space requirement between them has really affected the query processing time as shown in our simulation study.)

## 4 Conclusion

In this paper, we have proposed an efficient iconic indexing strategy called *Generalized Prime-Number-based Matrix* for symbolic pictures. In the proposed strategy, we have assigned each spatial operator a unique value which is a product of some prime numbers from a set of 12 prime numbers, and derived five generalized category rules. As we have shown that the GPN Matrix strategy needs a smaller storage space requirement than the PN

Matrix strategy, which also improves the query processing time. To handle large amounts of image databases, several access methods [6, 9] have been proposed by using the concept of *superimposed coding* [7] and *two-level* signature files [13]. However, such a bit-pattern-based signature approach can cause a false match, where a false match is that a record signature matches a query signature but the corresponding record does not satisfy the query. How to efficiently handle large amounts of image databases with a low false match rate is an important future research direction.

## References

- [1] C.C. Chang and J. C. Lin, "A Fast Spatial Match Accessing Scheme for Symbolic Pictures," *Journal of Electrical Engineering*, Vol. 33, No. 3, pp. 129-137, June 1990.
- [2] S. K. Chang, Q. Y. Shi and C. W. Yan, "Iconic Indexing by 2-D Strings," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No. 3, pp. 413-428, May 1987.
- [3] S. K. Chang, C. W. Yan, D. C. Dimitroff and T. Arndt, "An Intelligent Image Database System," *IEEE Trans. on Software Eng.*, Vol. 14, No. 5, pp. 681-688, May 1988.
- [4] C. C. Chang, "Spatial Match Retrieval of Symbolic Pictures," *Journal of Information Science and Engineering*, Vol. 7, pp. 405-422, 1991.
- [5] Y. I. Chang and B. Y. Yang, "A Prime-Number-Based Matrix Strategy for Efficient Iconic Indexing of Symbolic Pictures," *Pattern Recognition*, Vol. 30, No. 10, pp. 1-13, Oct. 1997.
- [6] Y. I. Chang and B. Y. Yang, "Efficient Access Methods for Image Databases," *Information Processing Letters*, Vol. 64, NO. 10, pp. 95-105, Oct. 1997.
- [7] C. Faloutsos, "Signature Files: Design and Performance Comparison of Some Signature Extraction Methods," *ACM SIGMOD*, pp. 63-82, May 1985.
- [8] S. Y. Lee and F. J. Hsu, "2D C-String: A New Spatial Knowledge Representation for Image Database Systems," *Pattern Recognition*, Vol. 23, No. 10, pp. 1077-1087, 1990.
- [9] S. Y. Lee and M. K. Shan, "Access Methods of Image Database," *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 4, No. 1, pp. 27-44, 1990.
- [10] S. Y. Lee and F. J. Hsu, "Picture Algebra For Spatial Reasoning of Iconic Images Represented in 2D C-string," *Pattern Recognition Letters*, Vol. 12, No. 7, pp. 425-435, 1991.
- [11] S. Y. Lee and F. J. Hsu, "Spatial Reasoning and Similarity Retrieval of Images Using 2D C-String Knowledge Representation," *Pattern Recognition*, Vol. 25, No. 3, pp. 305-318, 1992.
- [12] C. L. Sabharwal and S. K. Bhatia, "Image Databases and Near-Perfect Hash Table," *Pattern Recognition*, Vol. 30, No. 11, pp. 1867-1976, Nov. 1997.
- [13] R. Sacks-Davis and A. Kent, "Multikey Access Methods Based on Superimposed Coding Techniques," *ACM Trans. on Database Systems*, Vol. 12, No. 4, pp. 655-696, 1987.
- [14] B. Y. Yang, "A Matrix Representation for Efficient Access of Symbolic Pictures," Master Thesis, Dept. of Applied Math., National Sun Yat-Sen Univ., June 1996.