Computational Complexity of Similarity Retrieval in Iconic Image Database

D. J. Guan Chun-yen Chou Chiou-Wei Chen
Department of Applied Mathematics
National Sun Yat-Sen University
Kaohsiung 80424, Taiwan, R. O. C.
E-mail: {guan,chou}@math.nsysu.edu.tw

Abstract

Iconic indexing is an important tool for image database. For similarity retrieval of iconic image database, there are 3 types of similarity. Only one of them is isometric, hence transitive. We present polynomial time algorithms to find maximum similar subpicture of two given pictures when the similarity is isometric. For nonisometric types, we show that the problems are NP-complete, even when all symbols are distinct, by reducing the 3CNF satisfiability problem to them.

Keywords. Image database, Spatial relationship, Similarity retrieval.

1 Introduction

Tanimoto introduced iconic indexing in [1]. In short, we use symbols to represent objects in image files. This allows the use of spatial relationship of symbols for retrieval of images. Spatial relationship is a fuzzy concept and is thus often dependent on human interpretation. Thus, similarity retrieval of images, which is one of the distinguishing functions different from a conventional database, is a necessity.

Chang et al [2] introduced 2D strings for iconic indexing. In this approach, a picture query can also be specified as a 2D string. The problem of pictorial information retrieval then becomes a problem of 2D subsequence matching. Chang et al [2] then defined three types of 2D subsequence matching and, accordingly, similarity relation. The three types of similarity are equivalent to the following coordinatewise definitions.

Definitions. Let A and B be two icons in P_1, and A' and B' be two icons in P_2 such that A and A' represent the same object, and B and B' represent the same object. Suppose icons A and B have the coordinates $(x_1, y_1), (x_2, y_2)$ in picture P_1 and the corresponding icons A' and B' have the coordinates $(a_1, b_1), (a_2, b_2)$ in P_2. Let $\Delta_x = x_2 - x_1, \Delta_y = y_2 - y_1, \Delta_a = a_2 - a_1$, and $\Delta_b = b_2 - b_1$. Between P_1 and P_2, we say A and B in P_1 and A' and B' in P_2 have the relation of type-i, $i = 0, 1, 2$, according to the corresponding criterion.

type-0 $\Delta_x\Delta_a \geq 0$ and $\Delta_y\Delta_b \geq 0$

type-1 "$\Delta_x\Delta_a > 0$ or $\Delta_x = \Delta_a = 0$" and "$\Delta_y\Delta_b > 0$ or $\Delta_y = \Delta_b = 0$"

type-2 $\Delta_x = \Delta_a$ and $\Delta_y = \Delta_b$

For brevity, when the corresponding icons are clear, we simply say A and B have the relation of type-i.

Definition. Let S be a subset of icons in picture P_1, and T be a subset of icons in P_2. We say S and T are similar subpictures of type-i, $i = 0, 1, 2$, respectively, if there is a function f from S to T such that

1. f is bijective,
2. For every A in S, A and $f(A)$ represent the same object,
3. For any pair of icons A and B in S, A and B in P_1 and $f(A)$ and $f(B)$ in P_2 have the spatial relation of type-i, $i = 0, 1, 2$, respectively.

Intuitively, in finding maximum similar subpictures, the case when the symbols are all distinct is easier than the case when the symbols are not all distinct. Given two iconic pictures P_1 and P_2. When symbols are all distinct in each picture, Chang et al reduce the problem of maximum similar subpictures of type-0 (respectively, type-1, type-2) to the clique problem as follows. Let V be the set of symbols appearing in both P_1 and P_2. A graph $G_i(P_1, P_2)$, $i = 0, 1, 2$, is constructed as follows. There is an edge between s and t if and only if they have the relation of type-i, $i = 0, 1, 2$. Therefore we have an undirected graph $G_i(P_1, P_2)$, and maximum complete subgraph of $G_i(P_1, P_2)$ corresponds to maximum similar subpictures of type-i, $i = 0, 1, 2$.

Note that similarity of type-2 is isometric, that is, the difference in both coordinates are preserved, hence transitive. For maximum similar subpictures of type-2, we present an $O(n^2)$ algorithm when symbols are all distinct and an $O(n^3)$ algorithm when symbols are not all distinct.

A related problem is called the *picture pattern matching problem*: Given an iconic picture P and an iconic pattern Q, determine whether the pattern Q appears in the picture P or not. Tucci et al [5] showed that, when the symbols are not all distinct, the type-0 (respectively, type-1) picture matching problem is NP-complete. Note that when all the symbols are distinct, we can construct the graph $G_i(P, Q)$ and check if the induced subgraph $G_i(P, Q)[U]$ is a complete graph or not, where U is the subset of vertices corresponding to the pattern Q. This shows that the picture pattern matching problem can be solved in polynomial time when all the symbols are distinct.

We show in this paper that, for picture similarity problem of type-0 and type-1, the problem is NP-complete even when the symbols are all distinct. Our method is to reduce the 3CNF satisfiability problem to them.

2 Maximum Similar Subpictures of Type-2

Assume that we are given two iconic pictures P_1 and P_2 each containing a set of n distinct icons. Recall that, to find a maximum similar subpictures of type-2, we can construct the corresponding undirected graph $G_2(P_1, P_2)$. We show that $G_2(P_1, P_2)$ is a union of complete subgraphs. Therefore, there is an efficient algorithm for finding maximum similar subpictures of type-2.

Note that similarity of type-2 is transitive. Let s, t, and u be three distinct icons in P_1 and P_2. Assume that (1) s and t has the relation of type-2, and (2) t and u has the relation of type-2. We can conclude that s and u must have the relation of type-2. In the graph $G_2(P_1, P_2)$, if there is an edges between s and t and an edge between t and u, then there must be an edge between s and u. Therefore, the graph $G_2(P_1, P_2)$ is a union of complete subgraphs. Hence, by checking the the adjacency lists, it only takes $O(n)$ time to identify each maximal complete subgraphs. Since it only takes $O(n^2)$ time to find all relations of type-2, we have the following theorem.

Theorem 1 *It takes $O(n^2)$ time to solve the problem of maximum similar subpictures of type-2 when all symbols are all distinct.*

When symbols are not all distinct, we have the following algorithm, taking $O(n^3)$ time, to solve the problem of maximum similar subpictures of type-2.

The algorithm first sets the lower left corner of each iconic picture to be the origin and record the coordinates of all symbols s_i's in P_1 and t_j's in P_2. We get two lists $S = \{(x_i, y_i; s_i)\}_{1\leq i\leq n}$ and $T = \{(p_j, q_j; t_j)\}_{1\leq j\leq n}$. We require that the indexing is in the order of the sum of both coordinates and, when there is a tie, in the order of horizontal coordinates.

The algorithm then finds a maximum subset of icons Q, such that each pair of icons in Q have the relation of type-2. Suppose that $(x_i, y_i; s_i)$ in S is matched with $(p_l, q_l; t_j)$ in T in the resulting similarity subpicture defined by Q. Then any icon s_k satisfying the following two conditions must be in Q: (1) $(x_k, y_k; s_k)$ is in S, and $(p_l, q_l; t_l)$ is in T, (2) $p_l - p_j = x_k - x_i$ and $q_l - q_j = y_k - y_i$, (3) s_k and t_l represent the same object.

The algorithm computes Q by superimposition. The method is described as follows. For each $(x_i, y_i; s_i)$ in S, let T_i be the subset of T whose third entry t_j represent the same object as s_i. Then, for each $(p_j, q_j; t_j)$ in T_i, the algorithm counts the number of matches by superimposing P_1 and P_2. The superimposing is done by identifying the coordinates (x_i, y_i) of P_1 and (p_j, q_j) of P_2.

To make the counting of matches efficient, the algorithm creates another two lists from the original lists S and T by shifting their coordinates so that the new origins are located at (x_i, y_i) and (p_j, q_j), respectively. It is easy to see that, for each $(x_i, y_i; s_i)$ in S and ecah $(p_j, q_j; t_j)$ in T_i, it takes $O(n)$ time to count the matches. Since the size of S is at most n, and the size of T_i is at most n, the counting of matches is performed at most n^2 times. Therefore the above algorithm is of $O(n^3)$ time.

Theorem 2 *It takes $O(n^3)$ time to solve the problem of maximum similar subpictures of type-2 when symbols are not all distinct.*

3 Maximum Similar Subpicture Problem of Type-0 or Type-1

In this section, we show that the maximum similar subpicture problem of type-0 or type-1 is NP-complete. We reduce the 3CNF satisfiability problem to them.

Let the set of Boolean variables be $\{x_1, x_2, \cdots, x_k\}$. Given a Boolean formula Φ in 3CNF, say,

$$\Phi = c_1 \wedge c_2 \wedge \cdots \wedge c_n,$$

where for each c_r, $r = 1, 2, \cdots, n$, $c_r = (l_1^r \vee l_2^r \vee l_3^r)$ for some distinct literals l_1^r, l_2^r, l_3^r. We can construct two iconic images P_1 and P_2 from Φ as follows.

P_1 and P_2 are both of size $3n$ in width and $2k$ in height. Thus we can identify their background as a rectangle of $3n$ columns and $2k$ rows. The corresponding areas for clauses, literals and variables are as follows.

- For each c_r, we associate the $3r-2$-th to $3r$-th columns to it in both P_1, P_2.
- Define g_1 and g_2 as follows.

$$g_1(i, r) = \begin{cases} 3r-2 & \text{if } i = 1 \\ 3r-1 & \text{if } i = 2 \\ 3r & \text{if } i = 3 \end{cases}$$

$$g_2(i, r) = \begin{cases} 3r & \text{if } i = 1 \\ 3r-1 & \text{if } i = 2 \\ 3r-2 & \text{if } i = 3 \end{cases}$$

For each l_i^r, we associate the $g_j(i,r)$-th column to it in P_j, $j = 1, 2$.

- For each x_i, we associate the $(2i-1)$-th row to it in P_1 and the $2i$-th row to it in P_2. For each $\bar{x}_i$, we associate the $2i$-th row to it in P_1 and the $(2i-1)$-th row to it in P_2.

Then P_1 and P_2 are iconic pictures with the above background and $3n$ symbols filled in it. The symbols are l_i^r's, hence all distinct. For each l_i^r, in both P_1 and P_2, it is at the corresponding column to l_i^r and and at the corresponding row to the variable, possibly a negation, of l_i^r.

Note that it takes polynomial time to construct P_1 and P_2.

Example. If $\Phi = (x_1 \vee \bar{x}_3 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2 \vee x_3)$ with the specified order of clauses and literals, then the corresponding P_1 is

	l_2^1				
					l_3^2
		l_3^1			
				l_2^2	
			l_1^2		
l_1^1					

The corresponding P_2 is

			l_3^2		
	l_2^1				
				l_2^2	
l_3^1					
		l_1^1			
					l_1^2

Between P_1 and P_2, the only differences are (i) within each clause, the order of columns for the three literals are opposite; (ii) for two complementary literals, the order of rows are opposite. Therefore we have the following properties.

Property I For the same r and $i \neq j$, l_i^r and l_j^r don't have the relationship of type-0 or type-1.

Property II For $r \neq s$ and any i, j, if l_i^r and l_j^s are complementary then they don't have the relationship of type-0 or type-1.

Property III For $r \neq s$ and any i, j, if l_i^r and l_j^s are not complementary then they have the relationship of type-0 and type-1.

Clearly, by Property I, any set with the relation of type-0 or type-1 between P_1 and P_2 has at most one literal from each clause. Therefore its size is at most n. For brevity, we use $\max_i\{P_1, P_2\}$, $i = 0, 1$, to denote the maximum size of sets having the relation of type-i between P_1 and P_2. Clearly, $\max_i\{P_1, P_2\} \leq n$, $i = 0, 1$.

Lemma 1 *Φ is satisfiable if and only if* $\max_0\{P_1, P_2\} = n$

Proof: If Φ is satisfiable, then there is a satisfying assignment and each c_r contains at least one literal l_i^r assigned true. Pick one such literal from each clause, we get a set S of size n. Note that for any l_i^r, l_j^s in S they are not complementary. Therefore they have the relationship of type-0 by Property III. Hence S has the relationship of type-0. Therefore $\max_0\{P_1, P_2\} = n$.

If there is a set S of size n with the relation of type-0 then S has a literal from each c_r by Property I and pigeonhole principle. Any two of them are not complementary by Property II. Therefore, by assinging true to each literal in S, Φ is satisfiable.

∎

Similarly, we have the following lemma, too.

Lemma 2 *Φ is satisfiable if and only if* $\max_1\{P_1, P_2\} = n$

Therefore, we have the following two lemmas.

Lemma 3 *3CNF-SAT $\leq_p$ the problem of maximum similar subpictures of type-0.*

Lemma 4 *3CNF-SAT $\leq_p$ the problem of maximum similar subpictures of type-1.*

Recall that, when symbols are all distinct, we can reduce the problem of maximum similar subpictures of type-0 (respectively, type-1) to the clique problem. Therefore, both problems are in NP. We then have the following conclusions.

Theorem 3 *The problem of maximum similar subpictures of type-0 is NP-complete even when all symbols are distinct.*

Theorem 4 *The problem of maximum similar subpictures of type-1 is NP-complete even when all symbols are distinct.*

References

[1] S. L. Tanimoto, An Iconic Symbolic Data Structuring Scheme *Pattern Recognition and Artificial Intelligence*, pp. 452-471, Academic Press, New York, 1976.

[2] S. K. Chang, Q. Y. Shi, and C. W. Yan, Iconic Indexing by 2D Strings *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 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 Transactions on Software Engineering*, vol. 14 no. 5, pp. 681-688, May 1988.

[4] S. Y. Lee, M. K. Shan, and W. P. Yang Similarity Retrieval of Iconic Image Database *Pattern Recognition*, vol. 22, pp. 675-682, June 1988.

[5] M. Tucci, G. Costagliola, and S. K. Chang, A Remark on NP-completeness of Picture Matching *Information Processing Letters*, vol. 39, pp. 241-243, 1991.

[6] G. Costagliola, F. Ferrucci, G. Tortora, and M. Tucci, Correspondences - Non-Redundant 2D Strings *IEEE Transactions on Knowledge and Data Engineering*, vol. 7, no. 2, pp. 347-350, May 1995.