Pattern Recognition Letters 19 (1998) 287–297

A reinforced random algorithm for a partial contour perceptual similarity problem

Konstantin Y. Kupeev *, Victor L. Brailovsky

Computer Science Department, School of Mathematical Sciences, Tel Aviv University, Tel Aviv 69 978, Israel

Received 8 August 1997; revised 14 January 1998

Abstract

The goal of our paper is to suggest an algorithm for the detection of contour subparts which ''look similar'', for example, two sufficiently large spirals twirling clockwise. We proceed in three successive stages, where in each stage we suggest some algorithm derived from the previous one and closer to the partial contour similarity (PCS) problem. We start with a simplest reinforced random algorithm, proposed for maximizing an objective function F whose arguments belong to families of G-graphs. This is achieved by a series of descents in these families, such that, as a result of a sufficiently large number of descents, attractors arise in the families. These attractors impel the following descents. The second algorithm suggested gives a rough approximation to the PCS problem. The third algorithm is obtained from the second by replacing the subsets processed by weight distributions on contours; this allows an ''inexact matching'' between the perceptually similar subparts sought. Experimental results are presented. © 1998 Published by Elsevier Science B.V. All rights reserved.

Keywords: Contour; Similarity; Matching; G-graph

1. Introduction

The detection of similar subparts among two given shapes is an important pattern recognition problem, tackled in many works. The methods reportedly used for this problem vary with both the object representations exploited and with the recognition schemes applied to these representations; in most cases the representations are based on the local features of the objects represented. Approaches that make use of the boundary points of high curvature include relaxation labeling (Ogawa, 1994), the cluster structure approach (Bhanu and Ming, 1987), the hypothesis generation algorithm (Ray and Majumder, 1989), and the use of the chamfer 3/4 distance transformation (Liu and Srinath, 1990), the polygon moments (Koch and Kashyap, 1987), and the sphericity similarity measure between landmark points (Ansari and Delp, 1990). An approach to object recognition by affine invariant matching, proposed by Lamdan et al. (1988), allows the use different local features such as points, line segments, and curve segments. Some reported methods for partial shape similarity are not based on the object's local features. See, for example, (Bruckstein and Netravali, 1995).

* Corresponding author. Email: kup@math.tau.ac.il.

0167-8655/98/$19.00
PII S0167-8655(98)00011-7

In these works, the existence of a well-defined geometric transformation (e.g., a translation, or a rigid or affine transformation) is assumed between the subparts sought – superimposing the subparts via the transformation should make them congruent to each other or very close to it (a difference may, for example, occur as a result of noise).

The goal of our paper is to suggest an algorithm that, given two contours K_1 and K_2, outputs a pair of *perceptually* similar parts, S_1 and S_2 having close size and orientation. We call this problem *the partial contour similarity problem* (PCS). Also, we call such subparts $S_1 \subseteq K_1$ and $S_2 \subseteq K_2$ the *most similar parts* of K_1 and K_2. As is quite clear, these parts do not necessarily coincide, neither exactly nor approximately, under some (well-defined) superimposing transformation. Although ours is a vague definition, humans usually have few problems in making such a detection if such parts really exist in the contours. Nevertheless, the PCS problem has been found sufficiently intractable by algorithmic solutions, and the above approaches cannot be directly applied to the problem.

Recently, G-graphs have been applied to the following decision problems: the detection of the perceptual convexity of a 2-D contour (Kupeev and Wolfson, 1995), the detection of perceptual significance among the local maxima of a 1-D signal (Kupeev, 1996), and for the following problem: given input contours K_1 and K_2, decide whether they are perceptually similar (Kupeev and Wolfson, 1996).

In our paper, we use G-graphs to solve the more difficult PCS problem. The paper is organized as follows. Section 2 presents the simplest maximization model of the PCS problem – the so-called G-graph pair resolution problem – and the First Reinforced Random Algorithm (RRA) for the latter problem. In Section 3 we treat the PCS problem as the problem obtained from the G-graph pair resolution problem by incorporating into the latter a variable α representing contour rotation relative to the Y-axis of the Cartesian frame. Furthermore, a Second RRA that gives an (approximate) solution of the PCS problem is obtained from the First RRA by incorporating into the latter this variable α. Then, the Second RRA is transformed into the Third RRA suggested for the PCS problem. The latter algorithm allows us to detect most similar parts that do not necessarily have isomorphic structural descriptions via G-graphs. Section 4 presents experimental results for the PCS problem, and Section 5 contains a discussion of our method. Finally, Section 6 presents a summary of our work.

2. The simplest reinforced random algorithm

Let K be a contour in the Cartesian XY plane. The notions of the *segmentation* $E(K)$, the family $\mathscr{M}(K)$ of *simplified contours*, the *G-graph* $G(K)$, the *elementary morphism* $(G \to G/z)$, the *morphism lattice* $\mathscr{L}(G(K))$, and the *G-graph isomorphism* (‘‘ $\simeq$ ’’) were all defined by Kupeev and Wolfson (1996). For the purposes of our current research we will not include the horizontal segments participating in the construction of simplified contours from $\mathscr{M}(K)$ among these contours. Let q be an arbitrary lump in $K' \in \mathscr{M}(K)$. We define the weight $l(q)$ as the sum of the lengths of the pieces of the border of K belonging to q, and call this the *length weight assignment*. In this section, we describe an algorithm for solving the following problem.

Given G-graphs G_1 and G_2, and a cost function $F: \mathscr{L}(G_1) \times \mathscr{L}(G_2) \to \mathbb{R}$ assigning a real number to each pair of graphs $G'_1 \in \mathscr{L}(G_1)$ and $G'_2 \in \mathscr{L}(G_2)$ such that $G'_1 \simeq G'_2$, find an optimum pair $G_1^* \in \mathscr{L}(G_1)$, $G_2^* \in \mathscr{L}(G_2)$ maximizing $F(G'_1, G'_2)$.

We call this the G-graph pair resolution (GPR) problem. The GPR problem appeared in (Kupeev and Wolfson, 1996) as the problem of minimizing some nonnegative cost function $\mathscr{F}(G'_1, G'_2)$. It is easily shown that the slowest-random algorithm proposed there is not applicable to the GPR problem stated above.

Let us consider another problem and an algorithm for its solution; this will be an imaginary ‘‘physical’’ analogy illustrating our approach to the GPR problem. Let B be a sphere in 3-D Euclidean space, X be the set of vertices of a grid spanning B, and F be an objective function assigning a real number to some points $x \in X$. Let us restrict the possibilities of evaluating the values of F; for instance, we cannot, for a given $x_0 \in X$, enumerate the values of $F(x)$ for x belonging to a small neighborhood of x_0. The allowed measurements of the values of F are as follows. We can throw down from the top point of B a particle τ, so that τ descends in B over points of X and exits from the bottom point of B. As τ descends, we obtain the points x in its path and the values $F(x)$. In addition, suppose that we may create in B a "force field" ϕ by charging the points x obtained during the descent with a "potential" $\phi(x)$ such that τ, moving in the descent $\mathscr{D}$, will be attracted by the field ϕ created by the descents preceding $\mathscr{D}$. Let the attraction of τ to a point x, under the field created by charging x, increase as the distance between τ and x decreases.

The proposed algorithm for maximizing F proceeds as follows. We execute the descents $\mathscr{D}$, choose for each $\mathscr{D}$ a point $x^{\mathscr{D}}$ such that $F(x^{\mathscr{D}}) = \max_{x' \in \mathscr{D}} F(x')$, and, if $F(x)$ is sufficiently large, consider $x^{\mathscr{D}}$ a "good" point and charge $x^{\mathscr{D}}$ with a potential $\phi(x)$. In each descent $\mathscr{D}$ we execute leaps of τ among the points of X, each consisting of a step down of a small length and a small horizontal shift. This shift is random, occurring with probability λ, and, with probability $1-\lambda$, follows the potential of τ in the force field ϕ. This ensures that τ tends to be attracted to the points $x^{\mathscr{D}}$ of X where large values of F were reached during the descents preceding $\mathscr{D}$, while the "random part" of $\mathscr{D}$ tends to maximize F in these fields. The points x found in these fields that give bigger values of F impel the following descents to be attracted by these x to a greater degree.

At the beginning of the process, when the attractors are not stably detected, the search for "good" points $x^{\mathscr{D}}$ is provided by the "random part" of the descents. On the other hand, it is easily seen that to detect the deep good points we need a small probability of random steps in later descents. Thus, a good strategy is to decrease the probability λ as the number of descents increases.

In the GPR problem, X is $\mathscr{L}(G_1) \times \mathscr{L}(G_2)$, and the "points" x on which the function F is to be maximized are the pairs (G_1', G_2') with $G_1' \in \mathscr{L}(G_1)$ and $G_2' \in \mathscr{L}(G_2)$ such that $G_1' \simeq G_2'$. An algorithm for the GPR problem, implementing the above approach, is called the First Reinforced Random algorithm (RRA). See (Kupeev and Brailovsky, 1996) for the details.

3. An algorithm for the PCS problem

In this section, we describe the sequence of algorithms leading from the First RRA, for the GPR problem to the Third RRA, for the PCS problem. Due to lack of space, we cannot present here all the algorithms and refer the reader to (Kupeev and Brailovsky, 1996), where the material of this section is described in detail. We start with the construction of G-graph representations of contour subsets; then we reduce the PCS problem to the problem of maximizing an objective function F whose arguments are subsets of K_1, K_2 having isomorphic G-graph representations. The Second RRA for the latter problem is obtained from the First RRA by including an angle variable in the construction of the enumerated objects. Further, we modify the Second RRA and obtain the Third RRA, which enables us to detect most similar parts whose G-graph representations are not necessarily isomorphic.

From here on, for a vertex u in a graph $G' \in \mathscr{L}(G(K))$, $set(u)$ denotes the subset of the border of K corresponding to u. Let us suggest a construction of a G-graph representation relative to the Y direction of an arbitrary subset S of K. This representation is provided by the following path $\mathscr{V}$ in $\mathscr{L}(G(K))$: the first node of $\mathscr{V}$ is the root G, and each successive node in $\mathscr{V}$ is the result of the following procedure applied to the previous node G' (see Fig. 1):

1. choose an arbitrary leaf $u \in V(G')$ such that $set(u) \cap S = \emptyset$;
2. execute an elementary morphism $G' := G'/u$.

We call each G-graph terminating such a path a *cover element* of S in K, and denote them by $cov(S)$. It is easily shown that this construction generalizes the G-graph $G(K')$, which is actually the (unique) cover element of $S = K' \in \mathscr{M}(K)$. One may show (Kupeev, 1997) that if G_1, G_2 are arbitrary cover elements of S in K, then G_1 and G_2 are isomorphic as G-graphs, and for each vertex u of G_1 we have $set(u) \cap S = set(\lambda(u)) \cap S$, where λ is the isomorphism between G_1 and G_2. See Fig. 1. Thus all cover elements $cov(s)$ of S in K are

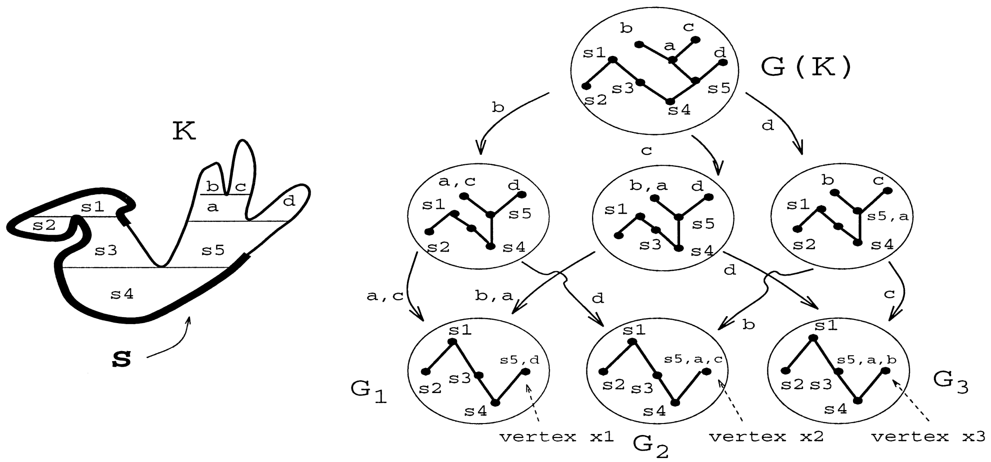

Fig. 1. G_1, G_2, G_3 are three isomorphic cover elements of S; $set(x1) \cap S = set(x2) \cap S = set(x3) \cap S$.

equivalent to a *unique* G-graph with subsets of S ''attached'' to its vertices; we call this G-graph (which provides a structural and quantitative representation of S) the *G-graph representation of S*.

Definition 1. Let XY_α denote the Cartesian frame XY rotated anti-clockwise through an angle α, and let Y_α denote the ordinate axis of this frame. Let $K \angle \alpha$ denote the contour obtained from a contour K as a result of rotating K clockwise through the angle α. We denote all cover elements of S constructed in the XY_α frame by $cov_\alpha(S)$.

Let

$$A_N = \left\{ \alpha_n \mid \alpha_n = (n-1)\frac{2\pi}{N}, 1 \leqslant n \leqslant N \right\} \tag{1}$$

be N equidistant angles of rotation. In (Kupeev and Brailovsky, 1996) the notion of regular subsets of a contour K is introduced, generalizing the notion of simplified contours. It has been shown that the PCS problem may be approximately treated as the problem of seeking the most similar regular subsets among the contours.

One may assume that the most similar parts $S_1^* \subseteq K_1$, $S_2^* \subseteq K_2$ sought satisfy the following properties to the greatest possible extent:

(P1) S_1^* and S_2^* are ''perceptually similar'';
(P2) S_1^* and S_2^* are large.

It follows from (P1) that the structural organization of S_1^* and S_2^* must be similar. Since the cover elements well represent this organization relative to the angles of rotation, one may assume that

(Q1) For the most similar regular subsets $S_1^* \subseteq K_1$ and $S_2^* \subseteq K_2$ we have $\forall \alpha \in A_N \; cov_\alpha(S_1^*) \simeq cov_\alpha(S_2^*)$.

Property (P1) also states that the sizes of the matched subparts composing S_1^* and S_2^* cannot differ much. Treating this ''relative to α'', one may expect that the total difference of the lengths of the parts of S_1^* and S_2^* related to the matched vertices of $cov_\alpha(S_1^*)$ and $cov_\alpha(S_2^*)$ must be small. Suppose that for $S_1 \subseteq K_1$, $S_2 \subseteq K_2$ we have $cov(S_1) \simeq cov(S_2)$; then we call

$$D(cov(S_1), cov(S_2)) = \sum_{v \in V(cov(S_1))} |l(set(v) \cap S_1) - l(set(\psi(v)) \cap S_2)|,$$

where ψ is the isomorphism function between $cov(S_1)$ and $cov(S_2)$ and $l(S')$ denotes the length of a subset S' of a contour, *the simple distance* between S_1 and S_2. In such a way, one may expect that

(Q2) A pair of regular sets $S_1^* \subseteq K_1$ and $S_2^* \subseteq K_2$ for which $D(cov_\alpha(S_1^*),cov_\alpha(S_2^*))$ gets smaller values would rather be treated as the pair of the most similar parts of K_1, K_2 than a similar pair $S_1' \subseteq K_1$ and $S_2' \subseteq K_2$ for which $D(cov_\alpha(S_1'),cov_\alpha(S_2'))$ gets bigger values.

The property (P2) can be reasonably interpreted as follows:

(Q3) A pair of regular sets $S_1^* \subseteq K_1$ and $S_2^* \subseteq K_2$ for which $l(S_1^*) + l(S_2^*)$ gets a bigger value would rather be treated as the pair of the most similar parts of K_1, K_2 than a similar pair $S_1' \subseteq K_1$ and $S_2' \subseteq K_2$ for which $l(S_1') + l(S_2')$ gets a smaller value.

The stipulations (Q1), (Q2), and (Q3) allow us to treat the PCS problem as follows:

For two given contours K_1 and K_2, one may search for their most similar parts by seeking regular subsets $S_1^* \subseteq K_1$ and $S_2^* \subseteq K_2$ maximizing $F(S_1,S_2)$, where $F(S_1,S_2)$ is the function defined on regular subsets $S_1 \subseteq K_1$, $S_2 \subseteq K_2$, such that $\forall \alpha \in A_N$ $cov_\alpha(S_1) \simeq cov_\alpha(S_2)$

$$F(S_1,S_2) = \sum_{\alpha \in A_N} F_\alpha(S_1,S_2), \tag{2}$$

where

$$F_\alpha(S_1,S_2) = |V(cov_\alpha(S_1))| \left(\gamma - \frac{D(cov_\alpha(S_1),cov_\alpha(S_2))}{l(S_1) + l(S_2)} \right), \tag{3}$$

and $\gamma > 0$ is an empirical constant.

The multiplication in Eq. (3) by the number of vertices in $cov_\alpha(S_1)$ reflects the fact that structurally more complicated S_1, S_2 must be treated as more perceptually similar. An algorithm for the latter problem, called Second RRA, is obtained from the First RRA by including in the construction of the enumerated objects a new ‘‘coordinate’’: the angle of rotation α. (Algorithms 5 and 7 in (Kupeev and Brailovsky, 1996).)

The firm definition of the similarity measure provided by Eq. (2) operates with the subsets whose G-graph representations are isomorphic for $\alpha \in A_N$. As we detected in our experiments the introduced measure well correlates with the human perception. It can be easily shown that small structural distortions could break the above isomorphism for some angles α but keep the subparts perceptually similar. This demonstrates the limited nature of the above measure. Introduction of the similarity measure among the contour subparts that do not necessarily have isomorphic structural representations seems to be a difficult problem – it is hard to suggest a well-justified model on how humans estimate the similarity among arbitrary contour subparts. Nevertheless, the Second RRA tolerates an empiric correction allowing us to detect the most similar parts that do not necessarily have isomorphic representations. The key to the correction is the replacement of the contour subsets processed in the Second RRA by weight functions defined on the points of the contours.

The *distributions* w considered below denote the nonnegative weight functions defined on the equidistantly distributed points of K_1 and K_2. Let S be a subset of a contour K. The weight distribution w_S is defined as follows:

$$\text{For any point } x \in K, \quad w_S(x) = \begin{cases} 1 & \text{if } x \in S, \\ 0 & \text{otherwise.} \end{cases}$$

Let w be a distribution on K. We define $set(w)=\{x\in K \mid w(x)>0\}$. The weighted average $(k_1,w_1)\times(k_2,w_2)$, for $0<k_1,k_2<1$, $k_1+k_2=1$, is defined as the following distribution on K (''soft intersection''):

$$\text{For all points } x\in K,\quad ((k_1,w_1)\times(k_2,w_2))(x)=\begin{cases}0 & \text{if } w_1(x)=0 \text{ and } w_2(x)=0,\\ \dfrac{w_1(x)}{c} & \text{if } w_1(x)\neq 0 \text{ and } w_2(x)=0,\\ \dfrac{w_2(x)}{c} & \text{if } w_1(x)=0 \text{ and } w_2(x)\neq 0,\\ w_1(x)^{k_1}w_2(x)^{k_2} & \text{if } w_1(x)\neq 0 \text{ and } w_2(x)\neq 0,\end{cases} \tag{4}$$

where $c\geqslant 1$ is an empirical constant.

Algorithm 1 (the Third Reinforced Random algorithm).

1. *Initialization*:
   Construct the G-graphs $G(K_1\angle\alpha)$, $G(K_2\angle\alpha)$, $\alpha\in A_N$;
   $F^*:=-\infty$; $i:=0$; $w(K_1)\equiv 0$; $w(K_2)\equiv 0$.
2. Repeat /* execution of i-runs */
2.1. $i:=i+1$;
2.2. Initialization of run $\mathscr{R}$; $w_{1,\alpha_0}\equiv 1$; $w_{2,\alpha_0}\equiv 1$.
2.3. /* Execution of run $\mathscr{R}$; $\mathscr{R}$ produces the distributions $w_1^{\mathscr{R}}$, $w_2^{\mathscr{R}}$ */
   for $j=1$ to $j=L$ do begin
2.3.1. Randomly choose an angle $\alpha_j\in A_N$.
2.3.2. Calculate the distributions $w^{sl}=(q_1,w_{1,\alpha_{j-1}})\times(q_2,w)$ on K_1 and
   $w^{sl}=(q_1,w_{2,\alpha_{j-1}})\times(q_2,w)$ on K_2. /* q_1, q_2 are empirical constants */
2.3.3. Execute a reinforced random descent $\mathscr{D}$ in the pair $(\mathscr{L}(G(K_1\angle\alpha_j)),\mathscr{L}(G(K_2\angle\alpha_j)))$
   relative to the w^{sl} weight assignment using Algorithm 2;
   obtain a pair $(G_1^{\mathscr{D}},G_2^{\mathscr{D}})$ providing the maximum $F^{\mathscr{D}}=\max_{\mathscr{D}}F_{\alpha_j}(G_1',G_2')$;
2.3.4. Calculate the distributions $w_{1,\alpha_j}:=(k_1,w_{1,\alpha_{j-1}})\times(k_2,w_{set(G_1^{\mathscr{D}})})$;
   $w_{2,\alpha_j}:=(k_1,w_{2,\alpha_{j-1}})\times(k_2,w_{set(G_2^{\mathscr{D}})})$. /* k_1, k_2 are empirical constants */
2.3.5. $F_{\alpha_j}^{\mathscr{D}}:=F^{\mathscr{D}}$.
   end.
2.4. $F^{\mathscr{R}}:=\sum_{j=m}^{L}F_{\alpha_j}^{\mathscr{D}}$; $w_1^{\mathscr{R}}=w_{1,\alpha_L}$; $w_2^{\mathscr{R}}=w_{2,\alpha_L}$.
2.5. if ($F^{\mathscr{R}}$ is ''good'') then
2.5.1. For each point x in K_r assign $w(x)=w_r^{\mathscr{R}}(x)$, $r=1,2$.
2.6. if $(F^{\mathscr{R}}>F^*)$ then
   $F^*:=F^{\mathscr{R}}$; $w_1^*:=w_1^{\mathscr{R}}$, $w_2^*:=w_2^{\mathscr{R}}$.
   Until *Stop*.

Algorithm 2 (reinforced random descent in the pair $(\mathscr{L}(G(K_1\angle\alpha_j)),\mathscr{L}(G(K_2\angle\alpha_j)))$ relative to the w^{sl} weight assignment).

1. *Initialization*:
   $F^{\mathscr{D}}:=-\infty$; $G_1':=G(K_1\angle\alpha_j)$; $G_2':=G(K_2\angle\alpha_j)$. /* graph copy */
   Assign to each vertex u of $G(K_1\angle\alpha_j)$ (resp. $G(K_2\angle\alpha_j)$)
   the weights $w^{sl}(u)=\sum_{x\in set(u)}w^{sl}(x)$ and $l(u)=\sum_{x\in set(u)}w_{1,\alpha_{j-1}}(x)$
   (resp. $l(u)=\sum_{x\in set(u)}w_{2,\alpha_{j-1}}(x)$).

2. *Descent step*:
2.1. If $|V(G'_1)| > |V(G'_2)|$ do the following:
2.1.1. Choose the leaf $z \in G'_1$ as in Algorithm 3 and execute $G'_1 := G'_1/z$.
2.1.2 goto 2.
2.2. If $|V(G'_1)| < |V(G'_2)|$ or $|V(G'_1)| = |V(G'_2)|$ and G'_1, G'_2 are not isomorphic as G-graphs, do the following:
2.2.1. Choose the leaf $z \in G'_2$ as in Algorithm 3 and execute $G'_2 := G'_2/z$.
2.2.2. goto 2.
2.3. /* here $G'_1 \simeq G'_2$ */ Evaluate $F_{\alpha_j}(G'_1, G'_2)$.
2.4. If $F_{\alpha_j}(G'_1, G'_2)$ do begin
2.4.1. $G_1^{\mathscr{D}} := G'_1$; $G_2^{\mathscr{D}} := G'_2$.
2.4.2. $F^{\mathscr{D}} := F_{\alpha_j}(G'_1, G'_2)$.
end.
2.5. If $|V(G'_1)| = 1$
Output $(F^{\mathscr{D}}, G_1^{\mathscr{D}}, G_2^{\mathscr{D}})$; *Stop*.
2.6. goto 2.

Algorithm 3 (choose the leaf z for a reinforced random descent step).

1. choose a random number t from a uniform distribution in $[0,1)$;
2. if $(t < \lambda)$ then /* ''random'' step */
2.1. randomly choose a leaf z in G;
else /* ''reinforced'' step – attraction to the ''charged points'' */
2.2. choose a leaf $z \in G'$ with the minimum weight $w^{sl}(z)$

Each run $\mathscr{R}$ in step 2.3 of Algorithm 1 is a sequence of j-descents producing the distributions w_{r,α_j}, $r = 1,2$, $1 \leqslant j \leqslant L$. For each j, $set(w_{1,\alpha_j}) \subseteq K_1$ and $set(w_{2,\alpha_j}) \subseteq K_2$ are the subsets of K_1, K_2 found in the run to be similar ''from the point of view'' of the axes $Y_{\alpha_1}, Y_{\alpha_2}, \ldots, Y_{\alpha_j}$. Thus, if L is sufficiently large, $set(w_1, \alpha_L)$ and $set(w_2, \alpha_L)$ can be treated as the most similar parts detected by the run. The values $w_{r,\alpha_j}(x) < 1$ mean that for some j', $1 \leqslant j' \leqslant j$, the point x violates the ''similarity from the point of view'' of $Y_{\alpha_{j'}}$ between $set(w_{1,\alpha_j})$ and $set(w_{2,\alpha_j})$. The basis of these properties of w_{r,α_j} is the following. In the jth descent subsets $U_1 \subseteq set(w_{1,\alpha_j})$ and $U_2 \subseteq set(w_{2,\alpha_j})$ similar ''from the point of view'' of Y_{α_j} are sought. After the descent, $w_{r,\alpha_{j-1}}$ are changed in such a way that the points of U_r increase their weight, while the points of $set(w_{r,\alpha_{j-1}}) \setminus U_r$ decrease their weight.

This proceeds as follows. In the descent, there the graphs $G_r^{\mathscr{D}}$ are sought, such that the subsets $U_1 = set(w_{1,\alpha_{j-1}}) \cap set(G_1^{\mathscr{D}})$ and $U_2 = set(w_{2,\alpha_{j-1}}) \cap set(G_2^{\mathscr{D}})$ are perceptually similar from the point of view of Y_{α_j}. This is provided by maximizing the function

$$F_{\alpha_j}(G'_1, G'_2) = |V(G'_1)| \left(\gamma - \frac{D(G'_1, G'_2)}{l(G'_1) + l(G'_2)} \right), \tag{5}$$

where $l(G'_r) = \sum_{v \in V(G'_1)} l(v)$,

$$D(G'_1, G'_2) = \sum_{v \in V(cov(G'_1))} |l(v) - l(\psi(v))|,$$

ψ is the isomorphism function between G'_1 and G'_2, and $r = 1,2$. This function is similar to that defined by Eq. (3) and has been introduced based on similar considerations. The assignment $l(u)$ before the descent ensures that the points $x \in set(w_{r,\alpha_{j-1}})$ violating the similarity of these sets for some angles $\alpha_{j'}$, play a lesser role in the estimation of similarity as reflected by $F_{\alpha_j}(G'_1,G'_2)$. After the descent, the values of $w_{r,\alpha}(x)$ in the points $x \in U_r$ (respectively $x \in set(w_{r,\alpha_{j-1}}) \setminus set(G_r^{\mathscr{D}})$) increase (respectively decrease), as is provided by step 2.3.4 of Algorithm 1. Also, the points of $set(G_r^{\mathscr{D}}) \setminus set(w_{r,\alpha_{j-1}})$ get a positive weight and are included in $set(w_{r,\alpha_j})$. This reflects the fact that these points participate in the similarity at angle α_j although they do not participate in the similarity detected in previous descents of the run.

The distributions $w^{\mathscr{R}} = w_{r,\alpha_L}$ that have been detected as good in the run, determine in step 2.5.1 of Algorithm 1 the distribution w, which, as in previous RRAs, forces w_{r,α_j} constructed in consequent runs to be closer to $w^{\mathscr{R}}$. (The assignment in step 2.5.1 was found in our experiments to provide the fastest convergence of Third RRA.) This is stipulated by the ''incorporation'' of w into the distribution w^{sl} in step 2.3.2. This compels the sets $set(G'_1)$, $set(G'_2)$ of the graphs enumerated in Algorithm 2 to be nearer the fields of the contours where $w^{\mathscr{R}}$ has big values. The distributions $w_{r,\alpha_{j-1}}$ also participate in the formation of w^{sl} in step 2.3.2. This provides that $set(G'_1)$ and $set(G'_2)$ enumerated in the jth descent tend to be nearer the contour parts where $w_{r,\alpha_{j-1}}$ has big values. It can be shown that this allows us to avoid a too rapid reduction of the constructed $set(w_{r,\alpha_j})$ during the run.

4. Experimental results

We implemented the Third RRA in a program written in C. Figs. 2–7 present some results of our experiments, where the boundaries of the processed contours have been perturbed under small random

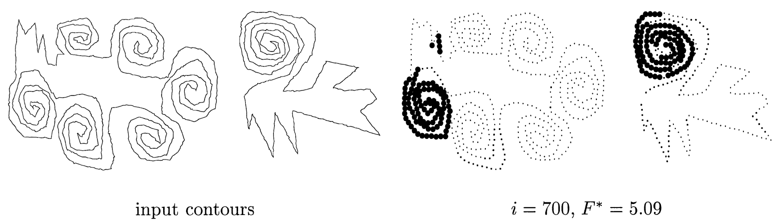

Fig. 2. Experimental results: Pattern 1 (two spirals twirled anti-clockwise).

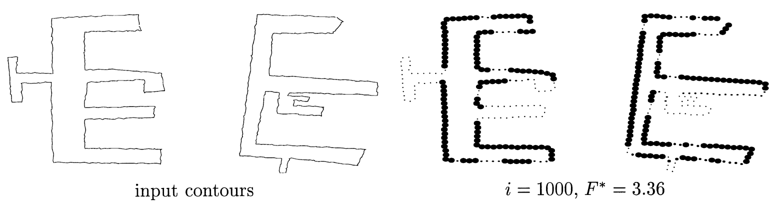

Fig. 3. Experimental results: Pattern 2.

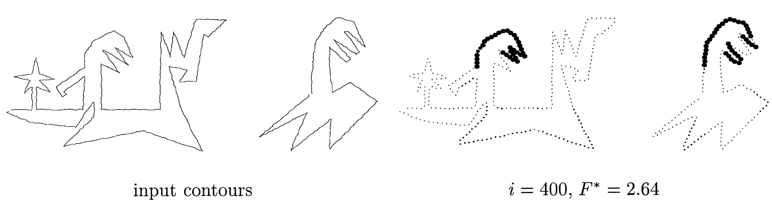

Fig. 4. Experimental results: Pattern 3.

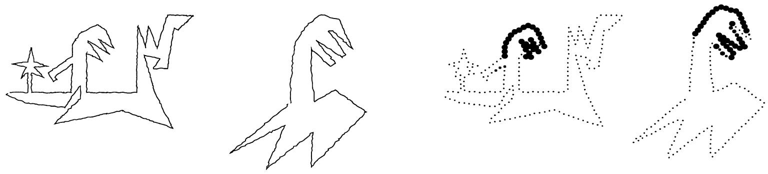

Fig. 5. Experimental results: Pattern 4.

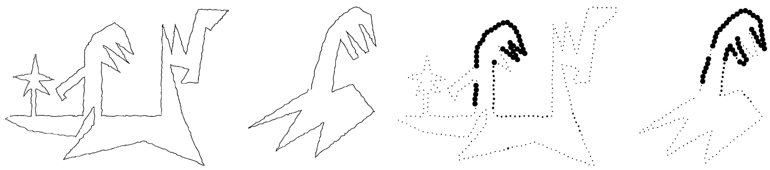

Fig. 6. Experimental results: Pattern 5.

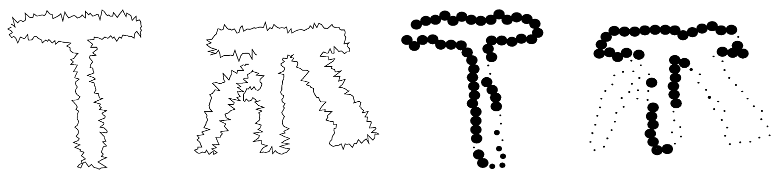

Fig. 7. Experimental results: Pattern 6.

fluctuations. The optimal weight distributions w_1^*, w_2^* obtained on the contours and the corresponding values of the objective function F^* are shown for different final i-iterations. The size of the circle representing the point x of the contour K_j is linearly proportional to the value of $w_j^*(x)$, $j = 1,2$. Pattern 6 (Fig. 7) is obtained by adding to the contours the maximal (in some sense) amount of noise such that applying the algorithm still gives good results. The results shown in Figs. 4–6 demonstrate the recognition of the most similar parts having some difference in size and orientation.

All the patterns were processed with the same values of the parameters (Kupeev and Brailovsky, 1996) used in the algorithm. The running time of the reinforced random process in this unoptimized implementation is not significant; for example, 400 iterations for Pattern 3 (Fig. 4) took approximately 31 s on a 100 MHz Hyper Sparc. It can be seen from our experiments that our method does indeed provide good results for partial contour similarity detection.

5. Discussion

The G-graph contour representations used are translation but not rotation-invariant. The major disadvantage of this is the inconvenience of applying the algorithm to detect of the most perceptually similar parts among shapes (i.e., contours considered up to rotations). In many works on global and partial shape matching, rotation-invariant representations based on various contour features are used. Some of these features, e.g., the directions of the principal moment axes of a contour (Hu, 1962), moment invariants (Hu, 1962), and the proper contour directions (Kupeev and Wolfson, 1996), are global contour features and would not help in recognizing contour subparts. Other features, e.g., points of curvature extrema (see Section 1) or bitangent points (Reiss, 1993), being local features that work well for a well-defined geometric transformation, could be essentially different for the perceptually similar parts sought. We therefore believe that local features can hardly be used as a basis for contour representation in detecting perceptually similar parts. Our second argument for the use of G-graphs in the PCS problem is that, considering the dearth of reported methods for this problem, the use of a representation that is not rotation-invariant seems justified.

Similarly to the First RRA, the probability λ of a random step (see Algorithm 3) is exponentially reduced in the Third RRA from 1 to 0 as a function of the number of runs. The algorithm's execution is stopped when new maximum values F^* of the function $F^{\mathscr{R}}$ cease to appear during long stretches of iterations. The experiments conducted verify that this occurs without fail when λ approaches 0. On the one hand, accelerating the convergence of λ accelerates the algorithm's convergence; on the other hand, too rapid a convergence of λ may cause the algorithm to converge to a distribution that does not yet correspond to the most similar parts sought. As we detected in our experiments, an optimal value of the coefficient of the exponential convergence of λ (and, therefore, the optimal number of of iterations required to detect the most similar parts among the contours K_1 and K_2) depends on properties of K_1 and K_2 such as their structural complexity, the degree of similarity between the most similar parts, the existence of parts possessing slightly lesser similarity than the most similar ones, and noise fluctuations on the border of the contours.

6. Conclusion

In this paper the algorithm for the partial contour similarity problem has been presented. Whereas current algorithms for partial shape matching deal with detection of contour subparts that are matched under a well-defined geometric transformation, the presented algorithm detects the subparts that are ''perceptually'' similar.

We presented three reinforced random algorithms such that each successive algorithm is obtained from the previous one and is closer to the PCS problem. The essence of each algorithm is the execution of a sequence of iterations such that at each iteration the construction of the resulted object is executed in a ''reinforced random'' fashion of descent. The reinforced steps of the descents ensures that these objects tend to be close to the objects that gave big values of the objective function F in previous iterations, while the random steps facilitate the search for objects giving the bigger values of F. The First RRA maximizes an objective function F whose arguments are pairs of isomorphic G-graphs. The Second RRA gives an approximation to the PCS problem and is obtained from the First RRA by incorporation of angle of rotation into the construction of enumerated objects.

Finally, the Third RRA, that is the algorithm for the PCS problem, is obtained from the Second RRA by substituting weight distributions for the contour subsets processed. This allows us to detect most similar parts that do not necessarily have isomorphic G-graph representations for all rotation angles. The results of experiments conducted show that the algorithm has good stability under small fluctuations of the processed contours.

Acknowledgements

Thanks go to Noam Shomron for his advice during the preparation of this paper.

This work forms part of the Ph.D. thesis of K.Y. Kupeev, Tel Aviv University.

References

Ansari, N., Delp, E.J., 1990. Partial shape recognition: A landmark-based approach. IEEE Trans. Pattern Anal. Machine Intell. 12 (5), 470–483.

Bhanu, B., Ming, J., 1987. Recognition of occluded objects: A cluster-structure algorithm. Pattern Recognition 20 (2), 199–211.

Bruckstein, A.M., Netravali, A.N., 1995. ORE differential invariants of planar curves and recognizing partially occluded planar shapes. Ann. Math. Artif. Intell. 13 (3,4), 227–250.

Hu, M.-K., 1962. Visual pattern recognition by moment invariants. IRE Trans. Inform. Theory 8, 179–187.

Koch, M.W., Kashyap, R.L., 1987. Using polygons to recognize and locate partially occluded objects. IEEE Trans. Pattern Anal. Machine Intell. 9 (4), 483–494.

Kupeev, K.Y., 1996. On significant maxima detection: A fine-to-coarse algorithm. In: Proc. 13th Internat. Conf. on Pattern Recognition, Vienna, vol. 2, pp. 270–274.

Kupeev, K.Y., 1997. A structural method for shape recognition, Ph.D. Thesis, Tel-Aviv University.

Kupeev, K.Y., Brailovsky, V.L., 1996. A slowest-random algorithm for a partial contour similarity problem. Tech. Rept. No. 310/96. Computer Science Dept., Tel Aviv University. (Also available via ftp://ftp.math.tau.ac.il/pub/TAU-PAPERS/.html/abstracts.html.)

Kupeev, K.Y., Wolfson, H.J., 1995. Perceptual convexity. In: Proc. SPIE Conf. on Vision Geometry IV, vol. 2573, San Diego, CA, pp. 125–136.

Kupeev, K.Y., Wolfson, H.J., 1996. A new method of estimating shape similarity. Pattern Recognition Lett. 17 (8), 873–887.

Lamdan, Y., Schwartz, J.T., Wolfson, H.J., 1988. Object recognition by affine invariant matching. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Ann Arbor, MI, pp. 335–344.

Liu, H.C., Srinath, M.D., 1990. Partial shape classification using contour matching in distance transformation. IEEE Trans. Pattern Anal. Machine Intell. 12 (11), 1072–1079.

Ogawa, H., 1994. A fuzzy relaxation technique for partial shape matching. Pattern Recognition Lett. 15, 349–355.

Ray, K.S., Majumder, D.D., 1989. Application of differential geometry to recognize and locate partially occluded objects. Pattern Recognition Lett. 9, 351–360.

Reiss, T.H., 1993. Recognition planar objects using invariant image features. In: Lecture Notes in Computer Science, vol. 676. Springer, Berlin.