Pergamon

Pattern Recognition, Vol. 30, No. 6, pp. 953–970, 1997

0031-3203/97 $17.00+.00

PII: S0031-3203(96)00123-9

INEXACT GRAPH MATCHING USING GENETIC SEARCH

ANDREW D. J. CROSS, RICHARD C. WILSON and EDWIN R. HANCOCK*

Department of Computer Science, University of York, York YO1 5DD, U.K.

(*Received* 10 *April* 1996; *in revised form* 11 *July* 1996; *received for publication* 2 *August* 1996)

Abstract—This paper describes a framework for performing relational graph matching using genetic search. There are three novel ingredients to the work. Firstly, we cast the optimisation process into a Bayesian framework by exploiting the recently reported global consistency measure of Wilson and Hancock as a fitness measure. The second novel idea is to realise the crossover process at the level of subgraphs, rather than employing string-based or random crossover. Finally, we accelerate convergence by employing a deterministic hill-climbing process prior to selection. Since we adopt the Bayesian consistency measure as a fitness function, the basic measure of relational distance underpinning the technique is Hamming distance. Our standpoint is that genetic search provides a more attractive means of performing stochastic discrete optimisation on the global consistency measure than alternatives such as simulated annealing. Moreover, the action of the optimisation process is easily understood in terms of its action in the Hamming distance domain. We demonstrate empirically not only that the method possesses polynomial convergence time but also that the convergence rate is more rapid than simulated annealing. We provide some experimental evaluation of the method in the matching of aerial stereograms and evaluate its sensitivity on synthetically generated graphs. © 1997 Pattern Recognition Society. Published by Elsevier Science Ltd.

Genetic search Discrete relaxation Graph matching Aerial stereograms

1. INTRODUCTION

Discrete optimisation problems[1–8] are of pivotal importance in high and intermediate level vision where symbolic interpretations must be assigned to relational descriptions of image entities.[9–13] Although the theory of continuous optimisation is mature,[1] discrete or configurational optimisation is still in its infancy.[2,4,6,7] The main difficulties stem from the fact that objective functions defined over a set of discontinuous states are prone to develop local optima.[3] It is for this reason that techniques such as simulated annealing,[4,3,14,15] mean-field annealing[8] and most recently genetic search[2,6,7] have been developed to overcome some of the local convergence problems.

Broadly speaking, discrete or configurational optimisation problems can be approached using two distinct methodologies. The first of these effectively corresponds to transforming the discrete optimisation problem into a continuous one.[1,5,8] In other words, the discrete symbolic assignments which are the goal of computation are replaced by a continuous representation. Algorithms falling into this category include mean-field annealing[8] and probabilistic relaxation.[1] One of the advantages of this form of optimisation process is that the continuous state-variables can in principle convey information concerning the degree of ambiguity associated with the final solution.[1] Whereas the mean-field technique[5,8] is capable of tracking the global optimum of the associated cost function under rescalings induced by a change in annealing temperature, probabilistic relaxation is only guaranteed to converge to a local optimum.[1] The second class of algorithm retains the discrete assignment representation, but avoids local optima by incorporating a stochastic element into the update process.[2–4,6,7,16] Perhaps the most popular algorithm falling into this category is the simulated annealing idea of Kirkpatrick,[4] which has been exploited with seminal impact by Geman and Geman[3] in the context of low-level vision. Gidas has addressed the problem of slow convergence by developing a multi-resolution Markov model that efficiently tracks the optima of the cost function from coarse to fine detail in a resolution pyramid.[14] A more recent addition to the family of stochastic optimisation methods is genetic search.[2,6,7] Rather than being motivated by the heat-bath analogy of simulated annealing,[3,4,16] genetic search appeals to ideas concerning chromosomal evolution.[2,17]

Although genetic search is a new and imperfectly understood optimisation method, it offers certain attractive computational features. Basic to genetic search is the idea of maintaining a population of alternative global solutions to the discrete optimisation problem in hand. The initial population may be generated in a number of different ways, but should in some sense uniformly sample the feasible solution space. Associated with each of the different solutions is a cost function which in keeping with the evolutionary analogy is termed the "fitness".[2] Genetic updates involve three distinct stages. Crossover involves selecting pairs of solutions randomly from the current population and interchanging the symbols at corresponding configuration sites with a

* Author to whom correspondence should be addressed.

uniform probability.[6] Mutation aims to introduce new information into the population by randomly updating the component symbols for individual solutions.[7] The mutation process usually takes place with a uniform probability. The net effect of modifying the population in this way is to randomly sample the landscape of the fitness function. Configurations generated by crossover and mutation are subjected to a stochastic selection process in order to avoid convergence to a local optimum.[6] If roulette-wheel selection is used, then the probability that a modified configuration enters the population is computed on the basis of the relative fitness measure.[17]

In many ways, genetic search provides an interesting compromise between the continuous transformation of the discrete optimisation problem[1,5,8] and its realisation by simulated annealing.[3,4,16] In the first instance, the roulette-wheel selection process is analogous to the Metropolis[16] algorithm employed in the sampling of Gibbs distributions in simulated annealing. Moreover, maintaining a population of solutions each with an associated fitness measure, naturally bridges the gap between the idea of having continuous optimisation variables and discrete ones. Provided diversity is sustained, the frequency of different solutions in the genetic population provides a discrete way of assessing ambiguity. One of the unique features of genetic search is the possibility of crossover. This effectively provides a means of mixing existing solutions to produce new ones. In this way locally consistent subsolutions may be combined to generate a globally consistent solution. If effectively controlled, this feature can provide convergence advantages over simulated annealing. Finally, although the use of local optimisation with multiple random-starts[18,19] shares the feature of employing a population of alternative solutions, it is the crossover, mutation and selection operations that ensure global convergence.

It is for these reasons that we would like to exploit genetic search in this paper. Our principal interest is in the symbolic matching of relational graphs.[10,11,13,20] Here we aim for find a discrete matching configuration that optimises a measure of relational consistency. Symbolic approaches to the relational matching problem have proved of perennial popularity since Barrow and Popplestone's pioneering work which located consistent matches by searching for cliques of the association graph.[9] Difficulties associated with matching inexact relational structures representing imperfectly segmented or cluttered scenes soon became evident.[10,11,20] One way of circumventing these difficulties is to pose the matching process as one of minimising a relational distance measure.[10,11,20] This measure should be capable of gauging both matching inconsistencies and structural errors. The idea of quantifying relational inexactness in this way has been pursued by Shapiro and Haralick[10] who attempted to minimise the number of inconsistently matched cliques. Structural errors are accommodated by inserting dummy nodes into the relational graphs without penalty. The structural edit operations of Sanfeliu and Fu[11] are more complex. Separate heuristic costs are associated with the operations of node relabelling, node deletion and node reinsertions; there are additional costs for the analogous operations on edges. In each of these methods consistent matches are located by deterministic search and there is little scope for exploring the ambiguity structure of the interpretation. One of the few discrete relaxation algorithms that allows for the exploration of ambiguities is Waltz's[21] classical constraint filtering algorithm which has been exploited in the context of line labelling.

Viewed from the perspective of relational matching, our genetic search process also has a number of novel features. Firstly, we gauge consistency using a matching probability defined over connected subgraphs of the relational structures. The development of this consistency measure commences from an objective Bayesian model of matching errors and has been extensively reported elsewhere by Wilson and Hancock.[12,13] Compared with this earlier work, the novelty of the work reported in this paper resides in the use of relational matching probability as a fitness measure in genetic search. By maintaining a population of matches, we potentially have a natural mechanism for simultaneously enumerating different ambiguous solutions. The second novel contribution of our work resides in the fact that rather than performing randomised crossover we realise the process at the level of subgraphs. In this way crossover allows us to mix solutions from the population so as to combine consistent subgraphs to rapidly form a more globally consistent solution. This can offer accelerated convergence, since standard discrete relaxation algorithms only propagate constraints over a distance of one neighbourhood with each iteration. Prior to performing selection we employ a hill-climbing process to locate the nearest local optimum of the fitness function for each solution residing in the genetic population. This further accelerates the convergence process and offers a third novel departure from standard genetic search. Integral to this hill-climbing process is the removal of structural errors by graph-edit operations[11] aimed at increasing the fitness of match. Selection operations stochastically refine the population on the basis of probabilities derived from the Bayesian consistency measure.[12,13] Finally, it is worth mentioning that since probabilistic selection is a critical stage in genetic optimisation, the availability of an objective Bayesian measure greatly simplifies the search procedure. Moreover, the population of solutions can be regarded as sampling the probability distribution for consistent relational matches.

In other words, our genetic algorithm represents a variant of the canonical version described by Goldberg.[17] The literature abounds with many algorithm variants which are in reality tailored to different aspects of the global optimisation problem. In fact, our ideas concerning hill climbing, subgraph crossover and the use of a probabilistic measure for selection all have identifiable progenitors elsewhere in the literature. For instance, the hill-climbing idea is also central to Davis's[22] hybrid genetic search technique. Both Eshelman's[23] CHC algorithm, and Louis and Rawlins[24] use Hamming

distance to control a non-uniform crossover process with the aim of sustaining population diversity. Although roulette-wheel selection is the most common means of converting an *ad hoc* fitness measure into a survival probability, there are several alternatives in the literature. For instance, Goldberg's "tournament selection" has many features in common with the use of our Bayesian consistency measure as a fitness function in the selection phase.[17] This idea of using genetic algorithms in conjunction with a Bayesian inference process has also been explored by Gelsema.[25] Finally, Pelillo *et al.*[26] have explored the symbiosis of relaxation labelling and genetic search in developing a strategy for learning compatibility coefficients. None the less, the idea of using genetic search to realise relational graph matching using the Bayesian consistency measure[12,13] is not only novel, but also represents a natural and powerful extension of the existing methodology.

The outline of this paper is as follows. In Section 2 we review the basic ingredients of the Wilson and Hancock relational consistency measure.[12,13] Section 3 describes how the optimisation of this measure may be mapped onto a genetic search procedure. Section 4 provides some performance evaluation on synthetic data. Section 5 shows the utility of the genetic matching procedure in the registration of aerial stereograms. Finally, Section 6 offers some conclusions.

2. FITNESS

The first step in the development of our genetic search procedure is to formally review the Bayesian consistency measure recently reported by Wilson and Hancock.[12,13] The consistency measure generalises the label-error process originally developed by Hancock and Kittler[27] from regular lattice arrangements to the more demanding application of labelling relational graphs. Our aim in this paper is to exploit this consistency measure as a fitness function for performing relational matching by genetic search. Hitherto, the optimisation of the consistency measure has been confined to the use of the deterministic iterative discrete relaxation process developed by Hancock and Kittler.[27] This hill-climbing algorithm is only guaranteed to locate the local MAP estimate and is in many ways analogous to Besag's[28] iterative conditional modes algorithm. Genetic search not only provides a global optimisation strategy, but as we shall demonstrate in this paper, its architecture is also naturally suited to the Wilson and Hancock consistency measure.[12,13]

Central to this paper is the aim of matching relational graphs represented in terms of configurations of symbolic labels. We represent such a graph by $G = (V, E)$, where V is the symbolic label-set assigned to the set of nodes and $E \subset V \times V$ is the set of edges between the nodes. In our genetic realisation of the matching process, we maintain a population of matches between a set of edited versions of a graph representing the data and a single model graph. These editing operations involve deleting nodes from the original data and recomputing the edge set. Suppose that α is an index which runs over the set of solutions in the population. Formally, we represent the matching of the nodes in the data graph indexed α in the population, i.e. $G_\alpha = (V_\alpha, E_\alpha)$ against those in the model graph $G_m = (V_m, E_m)$ by the function $f_\alpha : V_\alpha \rightarrow V_m$. In other words, the current state of match is denoted by the set of Cartesian pairs constituting the function $f_\alpha = \{(i, f_\alpha(i)),\ \forall i \in V_\alpha\} \subseteq V_\alpha \times V_m$.

In order to describe local interactions between the nodes at a manageable level, we will represent the graphs in terms of their clique structure. The clique associated with the node indexed j consists of those nodes that are connected by an edge of the graph, i.e. $C_j = \{i \in V_\alpha | (i,j) \in E_\alpha\}$. The labelling or mapping of this clique onto the nodes of the graph G_m is denoted by $\Gamma_j = \{f(i) \in V_m,\ \forall i \in C_j\}$. Suppose that we have access to a set of patterns that represent feasible relational mappings between the cliques of graph G_α and those of graph G_m. Typically, these relational mappings would be configurations of consistent clique labellings which we want to recover from an initial inconsistent state of the matched graph G_α. Assume that there are Z_j relational mappings for the clique C_j which we denote by $\Lambda^\mu = \{\lambda_i^\mu \in V_m,\ \forall i \in C_j\}$, where $\mu \in \{1, 2, \ldots, Z_j\}$ is a pattern index. According to this notation $\lambda_i^\mu \in V_m$ is the match onto graph G_m assigned to the node $i \in V_\alpha$ of graph G_α by the μth relational mapping. The complete set of legal relational mappings for the clique C_j is stored in a *dictionary* which we denote by $\Theta_j = \{\Lambda^\mu | \mu = 1, Z_j\}$.

The discrete relaxation procedure is based on maximising the joint probability of the matched label configuration, i.e. $P(\Gamma_j)$. It is therefore necessary to find a way of enumerating $P(\Gamma_j)$ when the label configuration is highly inconsistent, i.e. when there are no dictionary items for which the Hamming distance is zero. The approach is to adopt a Bayesian viewpoint in which it is assumed that only consistent labellings in the dictionary are legal and have uniform non-zero *a priori* probabilities of occurrence, i.e. $P(\Lambda^\mu) = Z_j^{-1}$. Other configurations do not occur *a priori* but are the corrupted realisations of the dictionary items. This idea is realised by applying the axiomatic property of joint probability to expand $P(\Gamma_j)$ over the space of consistent configurations

$$P(\Gamma_j) = \sum_{\mu=1}^{Z_j} P(\Gamma_j | \Lambda^\mu) P(\Lambda^\mu). \tag{1}$$

Further development of a useful objective function for discrete relaxation requires a model of the label corruption process, that is of the conditional probabilities of the potentially inconsistent configurations given each of the Z_j feasible relational mappings $P(\Gamma_j | \Lambda^\mu)$. We adopt a very simple viewpoint; matching errors are assumed to be memoryless and to occur with uniform probability p.

The first consequence of the assumed absence of memory is that the errors are independent. As a result we can factorise the conditional probabilities over the individual nodes in the graph, i.e.

$$P(\Gamma_j | \Lambda^\mu) = \prod_{i \in C_j} P(f_\alpha(i) | \lambda_i^\mu). \tag{2}$$

Our next step is to propose a model for the label corruption mechanism at each node in the graph. Again, taking recourse to the memoryless assumption, the probability of label errors on individual objects is independent of the class of label. This leads us to the following assignment of probability

$$P(f(i)|\lambda_i^\mu) = \begin{cases} 1-p & \text{if } f_\alpha(i) = \lambda_i^\mu, \\ p & \text{otherwise.} \end{cases} \tag{3}$$

As a result of this distribution rule, the conditional matching probabilities depend on the Hamming distance H_μ between the matched configuration Γ_j and the individual dictionary items Λ^μ, i.e.

$$P(\Gamma_j|\Lambda^\mu) = (1-p)^{|C_j|-H_\mu} p^{H_\mu}, \tag{4}$$

where the Hamming distance H_μ is defined to be $H_\mu = \sum_{i \in C_j} \left(1 - \delta_{f_\alpha(i),\lambda_i^\mu}\right)$. The model components given in equations (2)–(4) naturally lead to the following expression for $P(\Gamma_j)$ in terms of the set of Hamming distances to the consistent labellings residing in the dictionary:

$$P(\Gamma_j) = \frac{b_j}{Z_j} \sum_{\mu=1}^{Z_j} \exp(-kH_\mu), \tag{5}$$

where $b_j = (1-p)^{|C_j|}$ and $k = \ln[(1-p)/p]$. According to our picture of discrete relaxation, Hamming distance is the basic measure of consistency. Systematic softening of the constraints residing in the dictionary is controlled by the parameter p. It is tempting to draw analogies between the exponentials appearing in equation (5) and the Boltzmann distribution. The quantity k clearly plays the role of inverse temperature while H_μ is related to the Gibbsian potential.

The configurational probability $P(\Gamma_j)$ is the basic ingredient of our genetic search procedure. It represents the probability of a particular matching configuration evaluated over the state-space of feasible possibilities (i.e. the dictionary). We use as our global measure of consistency the sum of clique configurational probabilities, i.e.

$$P_G^\alpha = \frac{1}{|V_\alpha|} \sum_{j \in V_\alpha} P(\Gamma_j). \tag{6}$$

In Section 3 we will describe how this average consistency measure can be utilised as a fitness measure in the genetic search for relational matches. However, it is worth commenting on the relationship between our idea of using a compound exponential function of Hamming distance as a fitness measure and other attempts at exploiting Hamming distance reported elsewhere in the literature on genetic search. There are two contributions which merit mention. Eshelman[23] has used Hamming distance to control crossover and mutation operations to reintroduce diversity into an otherwise degenerate population. Louis and Rawlins[24] realised the same goal by adding a proportion of bit-complement patterns to the genetic pool. However, in neither case is the Hamming distance used to construct a fitness measure for use in selection.

3. GENETIC SEARCH

Genetic search[2,6,7] provides a very natural way of locating the global optimum of the global consistency measure described in Section 2. In essence, the approach relies on generating a population of random global matching configurations. These undergo crossover, mutation and selection to locate the match that optimises a fitness measure. Mutation operations ensure that the fitness landscape is uniformly sampled. Crossover introduces diversity by mixing partially consistent solutions; if effectively controlled this can accelerate the merging of consistent subgraphs. Selection stochastically selects from the population so as to locate the solution of optimum fitness in a manner analogous to the Metropolis algorithm.[16] However, since the algorithm commences from a set of random matches, accurate initialisation is not an issue of critical importance. One of the novel features of our genetic search process is the incorporation of a deterministic hill-climbing stage. This additional step is applied to the fitness measure once mutations have occurred and is used to accelerate convergence to the nearest optimum of the average consistency measure. In this way suboptimal solutions may be rapidly rejected by the selection process. It is interesting to compare the idea of maintaining a population of alternative solutions which is central to genetic search using multiple starting configurations employed by Beveridge *et al.*[18,19,29] Whereas the final solutions located with multiple starts effectively sample the local optima of the fitness landscape, it is the mutation, crossover and selection steps which ensure that genetic search converges to a global optimum.

Genetic search was originally inspired by an analogy with chromosomal evolution.[17] In consequence the algorithm is of a largely heuristic nature. Because of its effectiveness and its attractive computational properties, the technique has recently attracted attempts to provide a more rigorous theoretical understanding. For instance Qi and Palmieri[6,7] have provided a statistical analysis of the various components of genetic search. Moreover, each of the algorithmic components have been subject to various refinements and extensions.[2] In this section we therefore describe the main components of our genetic relational matching algorithm. We also provide some rationale for the search process from the viewpoint of our relational consistency measure described in Section 2.

3.1. *Initial population generation*

Key to genetic search is the idea of maintaining a population of alternative solutions, each with a computed fitness value. The initial choice of the trial solutions which undergo genetic refinement may be made in a number of ways. Here we choose the initial matching configurations so as to uniformly sample the feasible

search space. From the perspective of structural matching, this has a number of advantages. Firstly, it means that we obviate the need for accurate initialisation, which has proved to be a perennial problem in the application of iterative labelling schemes. Secondly, it means that our matching scheme is purely symbolic. An alternative to selecting a uniformly distributed initial population is to adopt a bias towards the matches suggested by unary measurements.

The choice of population size determines the rate of convergence for genetic search. There is a trade-off between the sampling, the fitness landscape with a fine granularity and the computational overheads associated with maintaining a large population of solutions. Since Hamming distance is the basic ingredient of our consistency model, we have appealed to a simple pattern space-model[30] to select the population size. We have demanded that the spacing of the initial solutions is less than the average Hamming distance between random graph pairs. For a uniformly distributed population of initial configurations, the distribution of inter-pattern Hamming distance is binomial with mean $|V_\alpha|[1-(1/|V_m|)]$. This means that the population size is approximately equal to the number of nodes in the data graph.

3.2. *Crossover*

Crossover is the process which mixes the pool of solutions to produce new ones. If effectively controlled, the process can be used to combine pairs of suboptimal or partially consistent matches to produce one of improved global consistency. Typically, deterministic updating of the match will propagate constraints only over the distance of one neighbourhood with each iteration. Crossover can accelerate this process by combining disconnected yet internally consistent subgraphs from the individual solutions in the pool.

The standard crossover procedure involves selecting at random pairs of global matching configurations from the current population. Random matches at corresponding sites in the match are then interchanged with uniform probability $\frac{1}{2}$; we term this as probabilistic crossover. However, this crossover mechanism will not necessarily facilitate the merging of locally consistent subgraphs. Moreover, the process also ignores the underlying structure of the graphs. A better strategy is to combine the solutions by physically dividing the graphs into two disjoint subgraphs. In this way internally consistent portions of the individual solutions may be exchanged at the structural level. As we will demonstrate in Section 4, this idea of geometric crossover offers certain advantages over the canonical process in terms of convergence speed.

In our experimental evaluation of the genetic search procedure we will confine our attention to Delaunay graphs. Here the nodes of the graphs are points on the image plane. Each node is used to seed a Voronoi cell. Edges in the Delaunay graph indicate region adjacency of the Voronoi polygons. In this case, the subgraph crossover process is easily implemented by dividing the original image plane with a random line. This has the effect of partitioning the set of data graph nodes into two subsets $V_\alpha^{(a)}$ and $V_\alpha^{(b)}$, where $V_\alpha = V_\alpha^{(a)} \cup V_\alpha^{(b)}$. New solutions may be constructed by interchanging the matches of the two partitions between pairs of matched graphs in the genetic population. Suppose that the two solutions undergoing crossover are denoted by the matches $f_\alpha = \{(i,f_\alpha(i)),\ \forall i \in V_\alpha\}$ and $f_\beta = \{(i,f_\beta(i)),\ \forall i \in V_\beta\}$. The new solutions produced by the crossover process are $f_\gamma = \{(i,f_\alpha(i)),\ \forall i \in V_\alpha^{(a)}\} \cup \{(i,f_\beta(i)),\ \forall i \in V_\beta^{(b)}\}$ and $f_\delta = \{(i,f_\beta(i)),\ \forall i \in V_\beta^{(a)}\} \cup \{(i,f_\alpha(i)),\ \forall i \in V_\alpha^{(b)}\}$.

If the two solutions are defined over the same sets of nodes, i.e. $V_\alpha \equiv V_\beta$, then the partitioning of the data graph by randomly dividing the image plane has no effect on the triangulation, i.e. the edge-set of the graph. In Section 3.4.2 we will describe our strategy for controlling extraneous clutter in the matching process. This involves a graph-editing process which centres around removing nodes from the point-set and recomputing the Delaunay graph. This means that different solutions in the genetic population may in principle be defined over dissimilar sets of data graph nodes. When this is the case, the crossover process not only involves exchanging matches, it must also include retriangulation. For Delaunay graphs this is a straightforward process. The Voronoi seeds falling into the two partitions of the image plane are exchanged in the usual way and two new Delaunay triangulations are computed. This process is illustrated in Fig. 1.

It is a straightforward matter to show that crossover operations result in no net change in the average Hamming distance. It is only the selection process that results in a modification in the Hamming distance distribution by removing members of the population on the basis of their fitness value. In the case of our fitness function given in equation (6), the net effect of selection operations will be to reduce both the mean and average Hamming distance, by incrementally removing the high Hamming distance tail of the distribution. In other words, at later epochs, when the genetic population is dominated by a few ambiguous solutions, the principal mode of the step-size distribution will be at zero Hamming distance with a number of submodes corresponding to the distance between different solutions. In reality, however, the population becomes swamped by a single solution and the Hamming distance variance approaches zero. For instance Lois and Rawlins[24] overcome this collapse in diversity towards a degenerate population by complementing the solutions in such a way as to maintain a non-zero Hamming distance variance. This is also the motivation behind both Eshelman's[23] CNC algorithm and the multi-niche crowding algorithm of Cedeno *et al.*[31] However, since our objective in this paper is the rapid location of a maximally fit solution, rather than exhaustively tracking ambiguities, maintaining population diversity is not an issue of central importance.

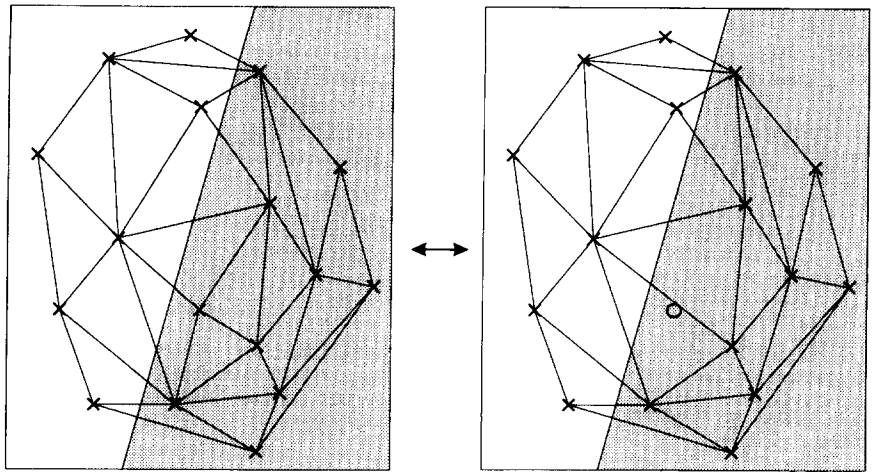

Fig. 1. Subgraph crossover (the figure illustrates how new graphs are generated in the crossover process through subgraph exchange).

3.3. Mutation

A further randomisation stage is applied to the individual matches to introduce new information into the population of global matches through a process of mutation. This is effected by randomly swapping the matches on individual sites with a uniform probability. In other words we randomly reassign a fixed-fraction the matches $f_\alpha(i)$ with random labels selected from the set V_m. In order to sample the probability distribution for relational matches in equation (5), we perform mutations with probability p, i.e. the prevailing value of the label-error probability. In this way mutation operations effectively sample probability distribution specified in equation (3). In other words, random label updates are generated according to our imposed model of the label-error process.

The effect of mutation operations can again be understood by reference to our simplistic pattern-space model. The distribution of Hamming distance associated with uniform mutation will again be binomial. The mean Hamming distance step size is $|V_\alpha|p$ while the variance is $|V_\alpha|p(1-p)$. In other words, $1/p$ is the average number of mutation steps required to transform the different initial solutions into one another.

3.4. Population refinement

The crossover and mutation stages of genetic search take place without reference to the value of the fitness measure; they simply mix and diversify a population of matches for further refinement. In our matching process the refinement process is effected using both hill-climbing and selection operations. Both stages of the algorithm are aimed at optimising the global configurational probability measure P_G^α.

3.4.1. Hill climbing. The aim in performing hill-climbing operations is to restore consistency to graphs modified by the crossover and mutation operations. Although this can be effected by stochastic means, it is time consuming. The hill-climbing stage involves iteratively reconfiguring the graphs modified by crossover or mutation to maximise the value of P_G^α. Formally, this corresponds to a parallel iterative application of the following decision rule:

$$f_\alpha(i) = \arg\max_{V_m} P_G^\alpha. \tag{7}$$

This application of this rule has the effect of locating the nearest local optima of the global consistency measure. It therefore redistributes the population of solutions to reside at the modes of this fitness measure. Suboptimal modes become increasingly unlikely as they are removed from the population by the stochastic selection operations. This process not only accelerates convergence, it also diminishes the requirements for a large population of graphs.

It is worth commenting that the idea of employing hill-climbing to accelerate convergence is also central to Davis's hybrid genetic search algorithm.[22]

3.4.2. Unmatchable nodes. One of the critical ingredients in effective relational matching is the way in which unmatchable entities or clutter are accommodated. Conventionally, there are two principal ways in which the effect of clutter can be neutralised. The first of these is to retain clutter nodes as an integral part of the graphs, but to explicitly label them as null-matched.[12] The alternative is to follow a graph-edit philosophy and to remove the clutter nodes, recomputing the edge-set of the graph if necessary.[11] The main advantage of graph-edit operations is that if effectively controlled, they can overcome relational fragmentation due to severe levels of clutter. In an extensive comparative sensitivity analysis Wilson and Hancock[32] have demonstrated that although effective when subgraph matching is being attempted, the null-labelling technique has a greater susceptibility to noise.

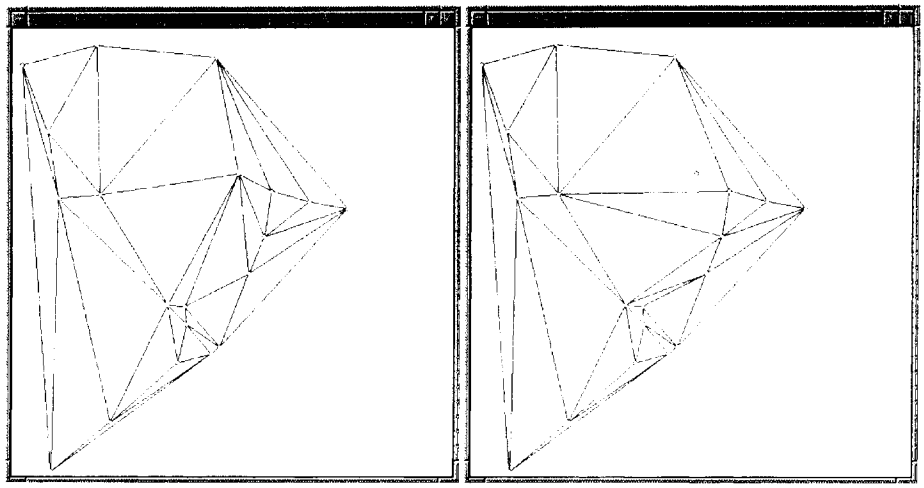

Fig. 2. An example of a graph-edit process in a Delaunay triangulation (the figure illustrates the effect of deleting a single node on the edge-set of a Delaunay graph).

We therefore choose to control clutter using a graph-edit process which allows nodes to be deleted and reinstated. This process is incorporated into the hill-climbing stage in the following way. Each node in turn is deleted from the graph and the edge-set recomputed. For our experimental evaluation of the method, we have chosen to use Delaunay graphs representing Voronoi tessellations of the image plane. Here the node deletion process corresponds to removing a particular Voronoi cell and growing adjacent cells to fill the vacated space. This process effectively modifies the edge-set of the associated Delaunay graph. By adopting the Delaunay representation we simplify the graph-editing process by lifting the requirement for an explicit set of edge-edit operations of the sort employed by Sanfeliu and Fu.[11] Our decision concerning node deletion or reinsertion is based on the value of P_G^{α}. If the value of P_G^{α} increases due to the deletion process, then the node is edited from the graph. If, on the other hand, the value of P_G^{α} increases as a result of node reinsertion at a later stage, then it is reinstated. This process is illustrated in Fig. 2.

In a recent paper, Wilson and Hancock[33] have shown how this operation of active graph reconfiguration can be realised as an optimisation process. Details of the derivation are outside the scope of this paper. The basic idea is to gauge the net effect of deleting a node by examining those contributions to the consistency measure that arise from modification of the super-cliques containing the node in question. Suppose we are considering deleting node $u \in V_{\alpha}$ from the data graph G_{α} to produce a new graph $G_{\beta} = (V_{\beta}, E_{\beta})$. Here the two node-sets are related by $V_{\beta} = V_{\alpha} - \{u\}$ and the edge set E_{β} is computed by retriangulating the modified node-set V_{β}. At the clique level the change in the consistency measure is monitored identifying those nodes that form a clique with node u in graph G_{α}, i.e. $C_u - \{u\}$, and determining their counterpart cliques in the reconfigured graph G_{β}. Let χ_u^+ denote the clique set of object u in graph G_{α} and χ_u^- denote the corresponding clique set in the reconfigured graph G_{β}. With this notation the change in the consistency criterion caused by the deletion of the node u is proportional to

$$\Delta_u^- = \sum_{j \in \chi_u} \frac{b_j}{Z_j} \sum_{\mu=1}^{Z_j} \exp(-kH_{\mu}). \tag{8}$$

By contrast, when considering the change in the MAP criterion caused by reinsertion of the node u it is the super-clique set χ_u^+ to which we turn our attention. The corresponding change to the consistency criterion is proportional to

$$\Delta_u^+ = \sum_{j \in \chi_u^+} \frac{b_j}{Z_j} \sum_{\mu=1}^{Z_j} \exp(-kH_{\mu}). \tag{9}$$

The decision criteria for node deletion or reinsertion are as follows. We delete node u provided $\Delta_u^+ < \Delta_u^-$ and reinstate it provided $\Delta_u^+ > \Delta_u^-$. This graph-editing procedure is applied at each of the solutions in turn as part of the hill-climbing process. The set of nodes constituting the reconfigured graph is therefore $V_{\beta} = \{u \in V_{\alpha} | \Delta_u^+ > \Delta_u^-\}$.

3.4.3. Selection. The hill-climbing and node deletion operations are purely deterministic processes which effectively bring about local improvements in matching consistency. These operations would otherwise prove time consuming if pursued by stochastic means. The final stochastic element of genetic search is the selection process. The aim here is to randomly admit the configurations refined by the hill-climbing process to the population on the basis of their fitness measure.

The probability distribution defined in equation (5) lends itself naturally to the definition of a population membership probability. By normalising the sum of clique configuration probabilities over the population of matches, we arrive at the following probability for randomly admitting the solution indexed α to the pool of graphs $\mathscr{P}$:

$$P_s = \frac{P_G^{\alpha}}{\sum_{\beta \in \mathscr{P}} P_G^{\beta}}. \tag{10}$$

With this survival probability in hand, population selection can be facilitated by a straightforward application of the roulette-wheel algorithm.[17] The final optimal match is located by selecting the graph for which P_G^{α} is maximum. Provided that population diversity can be maintained,[23,24,31] the idea of maintaining a population of alternative weighted matching configurations effectively bridges the conceptual gap between classical discrete relaxation methods[27] and continuous labelling algorithms such as probabilistic relaxation[1] or mean-field annealing.[8]

4. ALGORITHM EVALUATION

Our aim in this section is to evaluate the behaviour of the genetic search procedure on synthetic data sets with known ground truth. The synthetic graphs used in this study have been constructed by generating random-dot patterns and constructing the associated Delaunay graphs. Structural corruption has been generated by adding random noise dots and retriangulating. The main issues here in our algorithm evaluation are convergence properties, computational efficiency and parameter sensitivity.

4.1. Convergence

As we indicated in Section 3, there are a number of ways in which the crossover process can be realised. One of the anticipated shortcomings of randomly selecting nodes in the crossover process is the slow convergence. Figure 3 illustrates this point. Here we show the maximum value of the global fitness function P_G^{α} for the pool of graphs as a function of iteration number for both geometric crossover and probabilistic crossover. Although the probabilistic crossover has more rapid initial convergence, its subsequent behaviour is much slower. In other words, there appear to be distinct advantages in using disjoint subgraphs as structural units in the crossover process.

In addition to the maximum fitness, it is also interesting to study how the distribution of the fitness value over the pool of graphs evolves with iteration number. This is illustrated in Fig. 4(a). Initially P_G^{α} is concentrated close to the origin. As the genetic search proceeds, the mode of the histogram slowly moves to larger value of P_G^{α}. However, of greater importance to the genetic search procedure, the largest fitness value increases at a more rapid rate. In fact, the rightmost column of Fig. 3 illustrates how the frequency of the fittest match increases with iteration number. The ground-truth solution is in fact first encountered in the third iteration and has saturated the genetic population by the 10th iteration. Finally, it is interesting to note that the mode of the histogram corresponds in a statistical sense to the typical results achievable by virtue of deterministic search. Figure 3 therefore underlines some of the convergence advantages of genetic search over its deterministic counterpart.

To underline some of the points made in Section 3, we have plotted the distribution of Hamming distance between the ground-truth solution and the individual members of the population of graphs. Figure 4(b) shows the Hamming distance distribution as a function of iteration number for the 40-node problem studied in Figs 3 and 4(a). Initially, the minimum Hamming distance is 36, while the mode occurs at 39. As the genetic search process iterates, the modal Hamming distance moves towards zero as the correct solution saturates the population.

Finally, we demonstrate that the mutation, crossover and selection stages of our genetic search procedure offer advantages in terms of the global quality of match. We meet this goal by offering some comparison with multiple random starts. Commencing from a population of

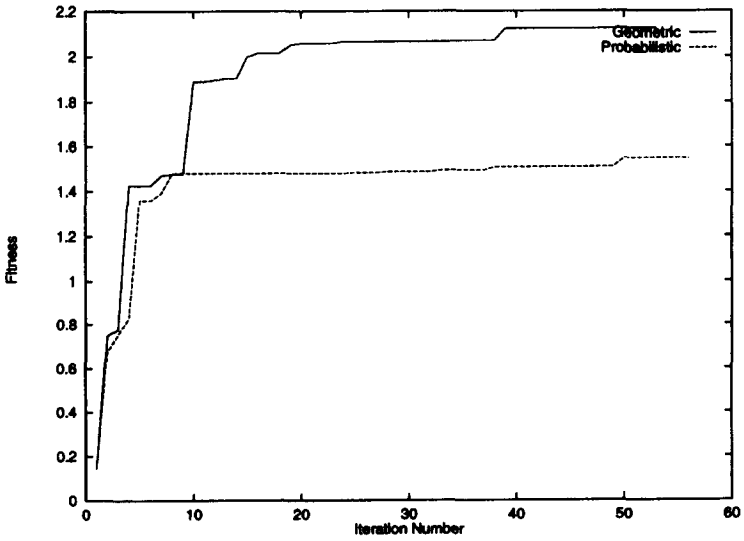

Fig. 3. A comparison of different crossover schemes. The maximum fitness in the population is plotted as a function of iteration number; the solid curve shows the result of geometric or subgraph crossover, while the dashed curve is the result of applying probabilistic crossover.

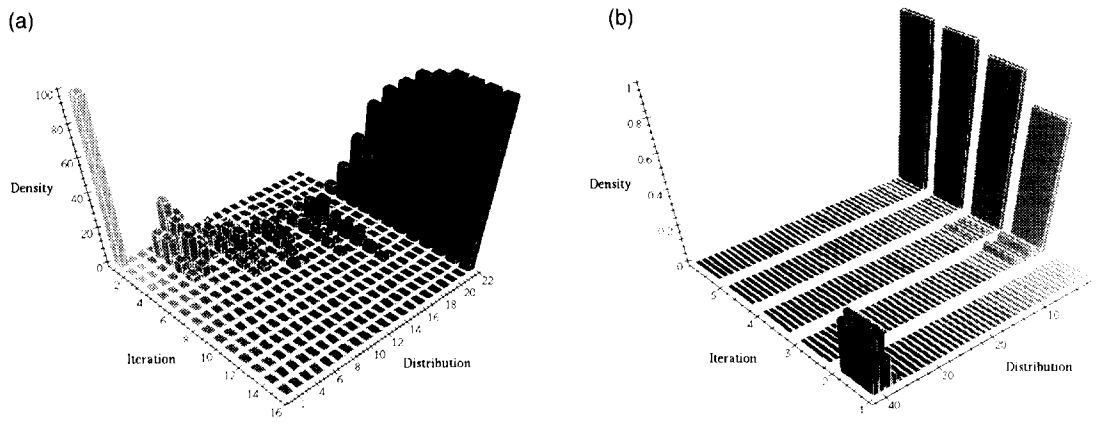

Fig. 4. (a) Cost function distribution as a function of iteration number. (b) Hamming distance distribution as a function of iteration number.

random initial matching configurations we apply only the hill-climbing and graph-editing steps described in Sections 3.4.1 and 3.4.2. In other words, we omit the mutation, crossover and selection stages of our genetic search algorithm. Figure 5 shows a histogram of P_G for multiple random starts and genetic search at various iterative epochs commencing from identical initial conditions. The solid line shows the initial P_G distribution that is common to both algorithms. The dashed curve is the result of multiple random starts at convergence. The short-dashed and dotted curves are the corresponding distributions from two and four iterations of genetic search. Initially, the distribution is concentrated at the origin. After two iterations of genetic search the distribution is almost identical to that for multiple random starts. However, in the case of multiple random starts the maximum fitness value is only 70% of the maximum value obtained by genetic search. After four iterations of genetic search the mean value of the fitness is more than twice that obtained with multiple random starts. Moreover, some 20% of the solutions are in the maximum fitness bin. Given identical population size and initial conditions, genetic search is therefore capable of locating better quality solutions.

4.2. *Comparison with alternative optimisation strategies*

To provide some indication as to the relative convergence performance of our genetic search procedure, we have undertaken some comparison with deterministic hill climbing and simulated annealing. The deterministic process is based on locating the optimum value of P_G^0 by parallel iterative updates along the lines suggested by Wilson and Hancock.[12,13] The simulated annealing method is more involved and requires the specification

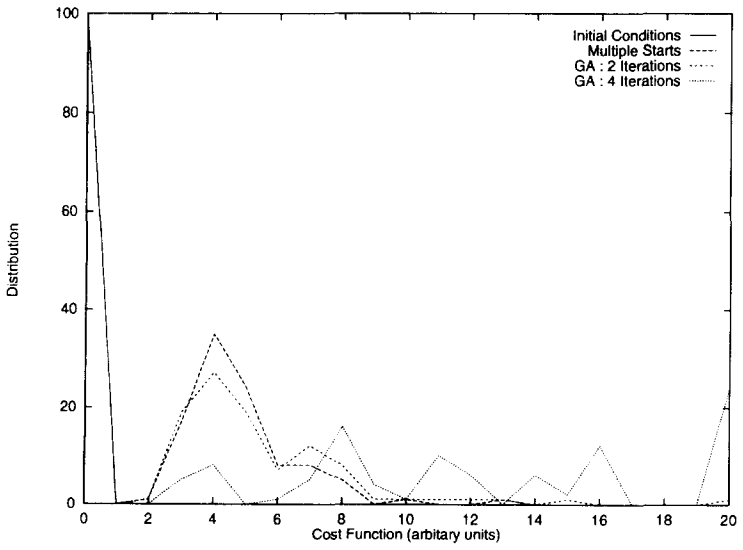

Fig. 5. Result of using multiple starts. The plot shows the distribution of fitness in the population for both genetic search and multiple starts. The solid curve is the initial fitness distribution which is common to both algorithms. The dashed curve is the result of multiple random starts at convergence. The short-dashed and dotted curves are the corresponding distributions from two and four iterations of genetic search.

of Gibbs potentials for the super-clique matching configurations. Details are beyond the scope of this paper. Suffice to say that the apparatus of statistical physics[5,8] can be applied to $P(\Gamma_j)$ to compute a corresponding Gibbs potential $U(\Gamma_j)$. The starting point is the following relationship which draws on the assumption that $P(\Gamma_j)$ can be regarded as a local partition function for the dictionary.[34]

$$U(\Gamma_j) = -\frac{1}{P(\Gamma_j)}\frac{\partial P(\Gamma_j)}{\partial k}. \tag{11}$$

Upon substituting for $P(\Gamma_j)$ from equation (5),

$$U(\Gamma_j) = \frac{\sum_{\mu=1}^{Z_j} H_\mu \exp(-kH_\mu)}{\sum_{\mu=1}^{Z_j} \exp(-kH_\mu)}. \tag{12}$$

In other words, the effective potential for a particular label configuration is just a weighted sum of Hamming distances over the different dictionary items. We perform simulated annealing on the sum of the configurational clique potentials, i.e.

$$U_G = \sum_{j \in V_a} U(\Gamma_j). \tag{13}$$

Using the Metropolis algorithm,[3,16] we randomly select both nodes and potential updates. The update is accepted if it leads to a reduction in the global configurational potential U_G. If this is not the case, then the random update is accepted with probability

$$P_r = \frac{\exp\left[-k\sum_{i\in C_j} U(\Gamma_i)\right]}{\sum_{\Gamma_i} \exp\left[-k\sum_{i\in C_j} U(\Gamma_i)\right]}. \tag{14}$$

To illustrate the relative convergence performance for the three optimisation strategies, Fig. 6 shows a plot of P_G^α as a function of iteration number. In the case of the genetic algorithm the value plotted is the maximum configurational probability evaluated over the population of matches. Although deterministic hill climbing has rapid and uniform convergence, the final value of P_G^α is suboptimal when compared to the alternatives. Simulated annealing, on the other hand, converges to a larger value of P_G^α, but is slow and non-uniform. Genetic search converges rapidly and uniformly to the maximum value of P_G^α. By contrast, the deterministic update process remains trapped at the mode of the P_G^α histogram.

4.3. Algorithm efficiency

One of the attractive features of genetic search is that much of the computation concerned with optimising the different solutions and evaluating their associated fitness can take place in parallel. Although no real theoretical bounds exist for convergence, we can provide some empirical measurements which illustrate how the computational overheads grow with the increasing size of the graphs. To evaluate the computational cost associated with our genetic search procedure, we will count the number of Hamming distances needed to locate the correct solution. We will first concern ourselves with the problem of locating graph isomorphisms. The normalisation point in this study is the number of Hamming distances needed to locate a consistent graph isomorphism for a 10-node matching problem. Figure 7(a) shows the cost as a function of the number of nodes in the graphs under match. Fitting a polynomial to this graph, the number of Hamming distance computations increases as $0.00711*N^{2.13}$. In other words, there is empirical evidence that the genetic search process can locate graph isomorphisms in polynomial time. In fact it is interesting to compare this polynomial growth with the results reported by Beveridge[19] and by Grimson.[35] Grimson's

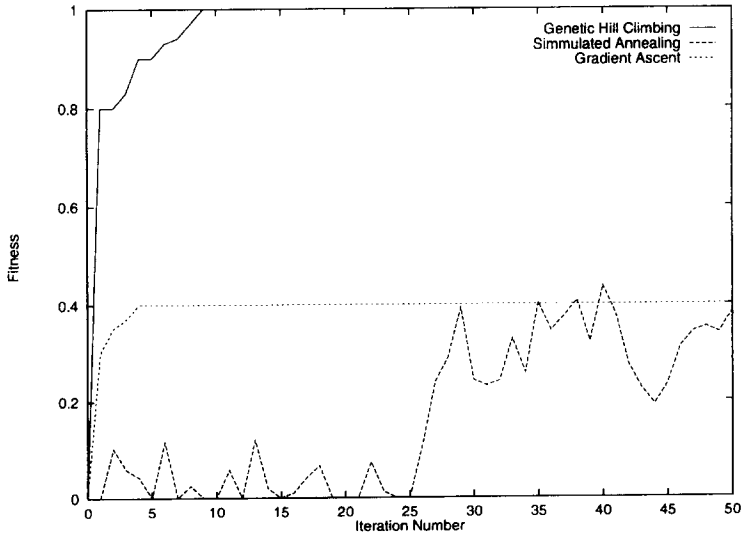

Fig. 6. Comparison of convergence rates for different optimisation schemes. The plot shows a plot of P_G^α as a function of iteration number. The solid curve is the result obtained with genetic hill climbing. The dotted curve is the result of applying deterministic hill climbing. The dashed curve is the result of applying simulated annealing.

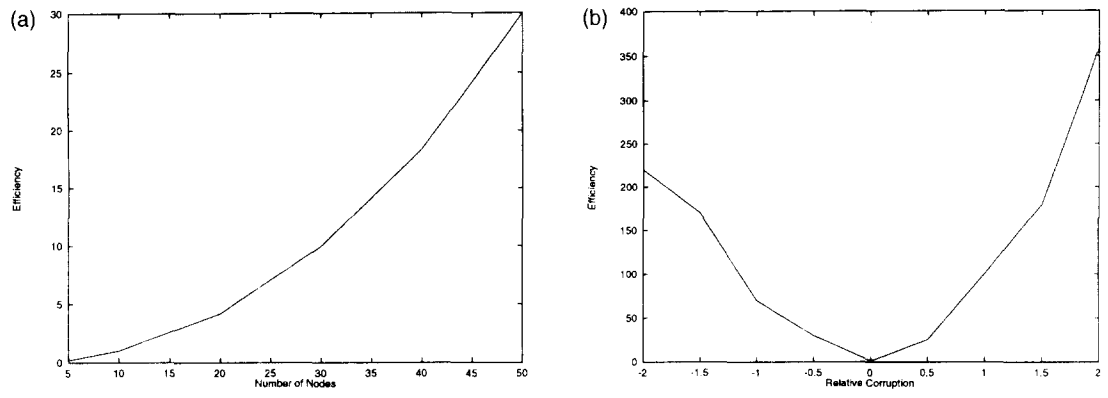

Fig. 7. (a) Number of computations to convergence as a function of the number of nodes in the graphs. (b) Number of computations to convergence as a function of graph corruption.

theoretical analysis suggests that the complexity of object recognition should rise as N^2 while Beveridge shows empirically that for random starts the complexity increases to $N^{2.5}$. In other words, the empirical bounds on the complexity of our genetic search procedure are within both the theoretical and empirical limits for deterministic search algorithms. It is also interesting to note that there exist deterministic heuristic search algorithms that return subgraph isomorphisms in polynomial time.[36]

Obviously, the task becomes more difficult if we are dealing with inexact matches. Figure 7(b) shows the number of Hamming distance computations required when controlled number of nodes are added to and deleted from the graphs. A positive number of corrupt nodes represents the addition of noise to G_m while a negative number represents noise addition to G_α. It is interesting to note that the values are not symmetric about the zero corruption case; the number of Hamming distance computations to convergence increases more rapidly when noise is added to the model graph. In other words, although the empirical complexity increases as we move from the case of graph isomorphism to subgraph isomorphism, the increase is at worst polynomial.

4.4. *Parameter sensitivity*

There are two parameters that control the genetic search procedure. The first of these is the population size, while the second is the mutation probability. Our aim in this subsection is to investigate the sensitivity of the convergence rate to systematic variation of these two parameters. Figure 8(a) shows the number of iterations to convergence as a function of population size for a graph of 40 nodes. The main feature of this graph is the existence of a critical population size. Once the population size exceeds 20 graphs, convergence takes place within five iterations. These convergence results are many orders of magnitude faster than those obtained if we omit the hill-climbing stage or adopt a probabilistic rather than geometric crossover strategy.

The second parameter of the optimisation scheme is the mutation probability. Normally, this is taken to be the label-error probability p used in the computation of global fitness P_G^α. Figure 8(b) shows the number of iterations to convergence when this constraint on the mutation probability is relaxed. Provided that $p<0.6$, the algorithm appears to be completely insensitive to the mutation probability. In fact, we find that mutation is a relatively insignificant component of the algorithm. Most

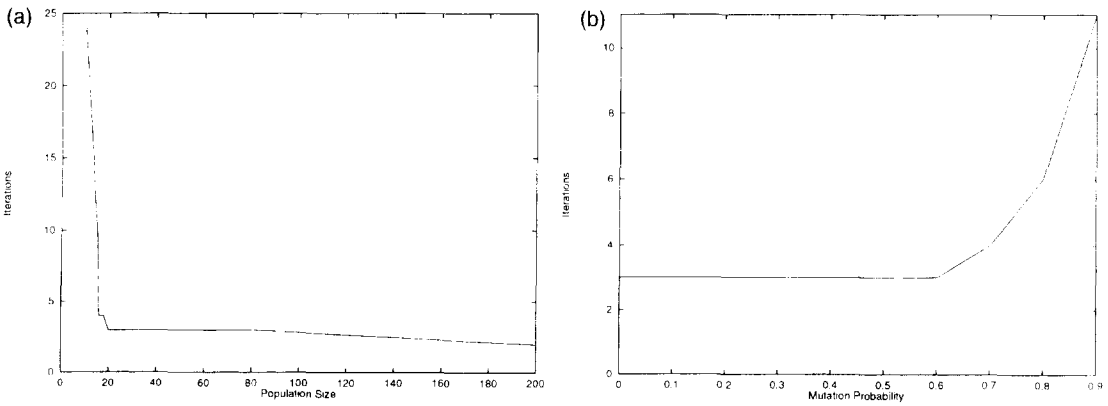

Fig. 8. (a) Number of iterations to converge for a given population size. (b) Iterations to converge for a given mutation probability.

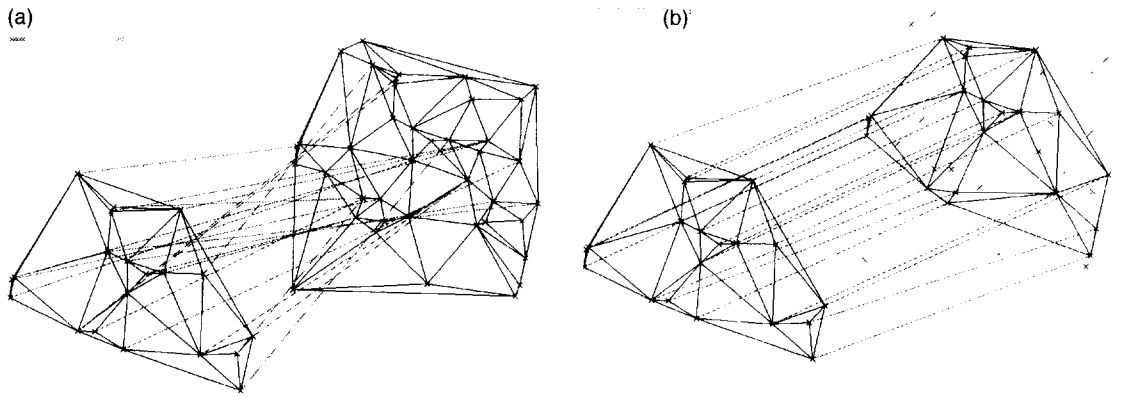

Fig. 9. (a) An initial guess. (b) The recovered solution.

of the effectiveness of the algorithm is derived from the crossover, hill-climbing and selection stages.

4.5. *Matching examples*

Figure 9(a) and (b) illustrates some typical matching results on synthetic graphs. Figure 9(a) shows a representative solution from the initial population. The left-hand graph is the model while the right-hand graph is the data; lines between the two graphs indicate matches. The data graph has been obtained by adding random clutter to the model and perturbing the nodes with Gaussian position errors. The original model graph contains 20 nodes while the corrupted data graph contains 40 nodes. Figure 9(b) shows the fittest match from the genetic population after three iterations. There are two features worth noting. Firstly, the overall consistency of match has improved. The lines connecting the nodes in the data and model graphs are no longer randomly distributed. Secondly, the added clutter nodes have all been correctly identified and deleted from the data graph; they appear as disjoint points on the right-hand image of Fig. 9(b). The overall accuracy of match in this example is 100%.

The example described above is typical of the problem of matching a relational description that is subsumed in noise or clutter. Another common problem in computer vision is to match scenes containing multiple objects. Under particularly severe imaging conditions these objects may be significantly overlapped. The following two examples illustrate the capacity of our genetic search procedure to match under these two sets of conditions. It should be noted that these results have been obtained using a more complex fitness measure than that described in Section 2 which draws more heavily on attribute information. Full details can be found in the recent paper of Cross and Hancock.[37]

The simplest example involves the matching of multiple non-overlapping models. Figure 10(a) shows the fittest initial match while Fig. 10(b) shows the final match. The data graph, on the left-hand side of Fig. 10(a) and (b), is a non-overlapping union of the three models on the right-hand side of the figures. Here the genetic search algorithm correctly partitions the data graph into three disjoint subgraphs. As indicated by the lines between the data and model, each of the subgraphs is correctly matched.

The more complex case in which the three graphs are overlapped is illustrated in Fig. 11(a) and (b). Here our genetic matching technique is again capable not only of correctly partitioning the nodes of the data graph into the three disjoint subgraphs but also of locating the consistent matches. In fact these results indicate that our

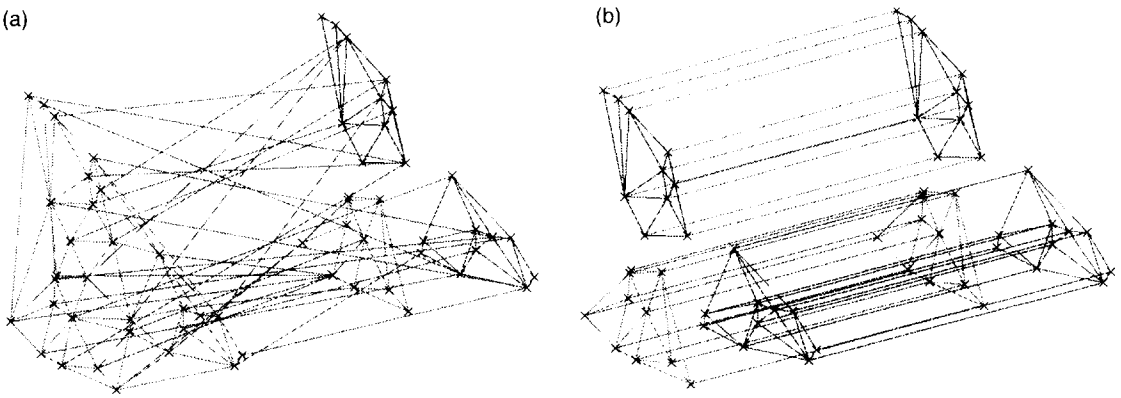

Fig. 10. (a) An initial guess. (b) Recovered solution.

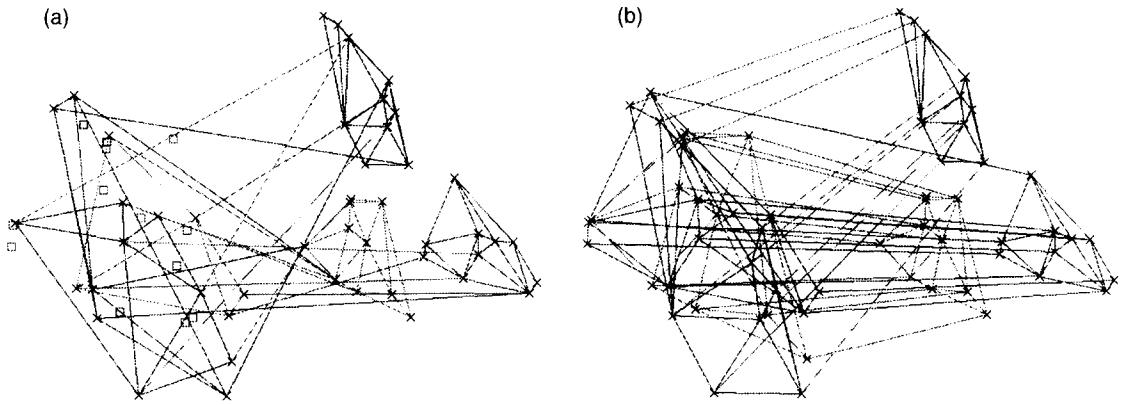

Fig. 11. (a) An initial guess. (b) Recovered solution.

matching technique has considerable potential as a tool for extracting relational clusters from highly overlapped data.

4.6. *Performance comparison*

In order to illustrate the effectiveness of the genetic search technique, we have compared its performance with deterministic hill climbing. The deterministic algorithm aims to optimise the global cost function given in equation (6) by gradient ascent; in other words, the label update that results in the greatest increase in P_G^{α} is always accepted at a particular node. The comparison has been performed under conditions of controlled structural corruption. We have generated random graphs and added a controlled fraction of spurious noise. Figure 12 shows the fraction of the graph correctly recovered and matched as a function of the fraction of added noise nodes. The lower curve is the result obtained by iterating the deterministic method to convergence. The intermediate curve is the result after performing one iteration of genetic search with a population size of 100 graphs. After two iterations of genetic search the upper curve is obtained. The main conclusion from this study is that once the corruption level exceeds 20%, the gradient ascent technique is likely to become trapped in a local minimum. By exploring a much greater fraction of the search-space, genetic search is capable of finding good results even at very severe corruption levels. In other words, when combined with the probabilistic cost function, genetic search can recover significantly better results than its deterministic counterpart.

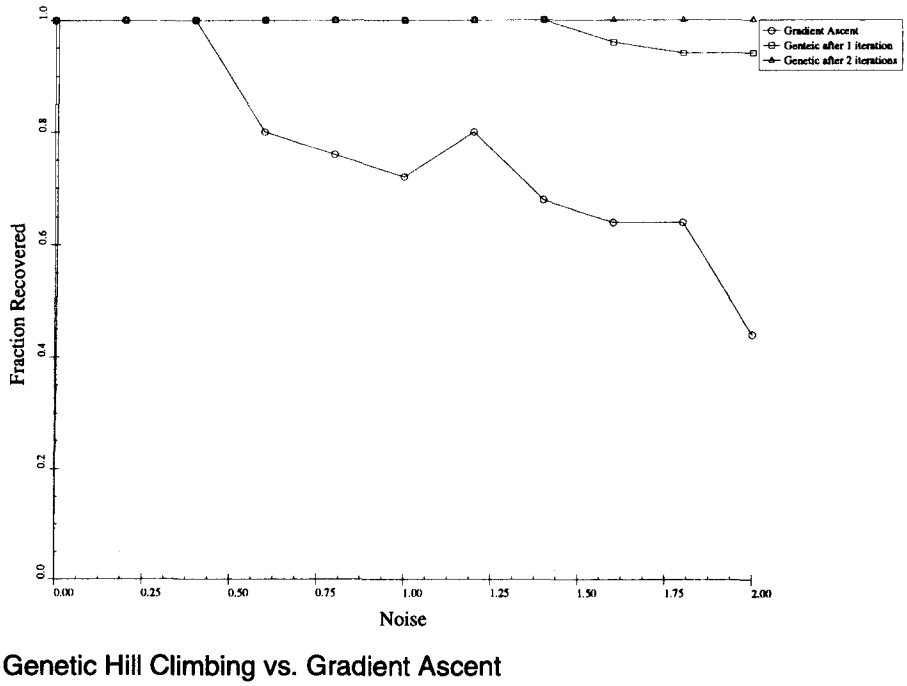

Fig. 12. The effect of controlled structural corruption.

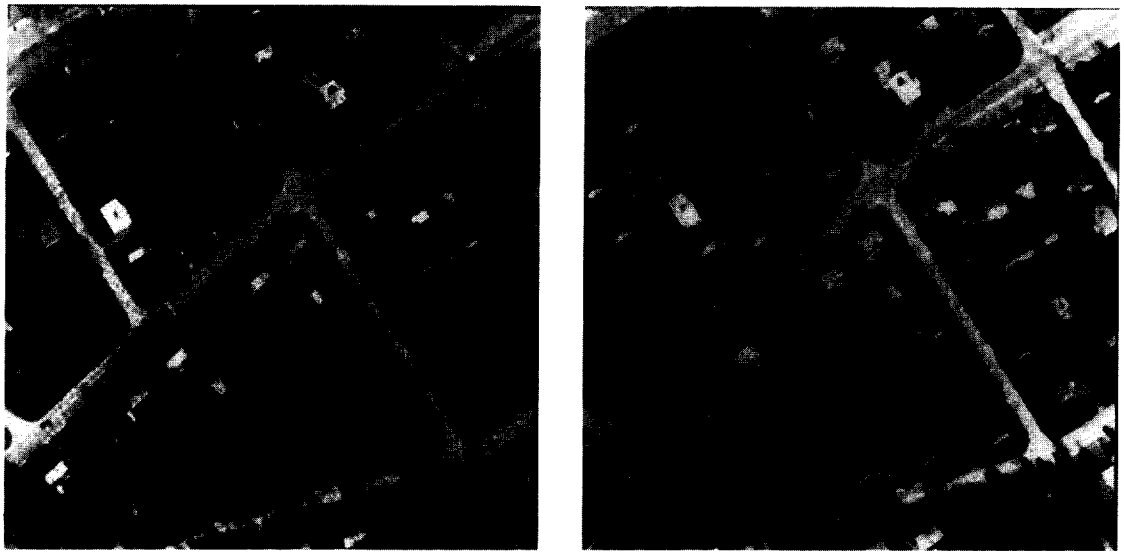

Fig. 13. Aerial stereograms.

5. MATCHING EXPERIMENTS WITH AERIAL STEREOGRAMS

Our experimental vehicle is provided by the matching of aerial stereograms of suburban areas. A typical image pair is shown in Fig. 13. The image registration problem is posed as one of matching rooftops. In this section we briefly review the image processing operations required so that we can abstract the registration process in terms of genetic graph matching, before proceeding to detail algorithm performance.

5.1. Extracting relational graphs from aerial stereograms

To commence we must localise the rooftop features to be matched. Here we adopt a matched filtering technique. Details of the algorithm are outside the scope of this paper. Suffice to say that we use a Wiener-filtering technique, which exploits the Fourier duality between convolution in the spatial domain and multiplication in the frequency domain to determine the coefficients of the matched filter over a set of training examples. The training examples are chosen on the basis of their orientation, size and shape. Once the matched filter is in hand, we can attempt to identify typical rooftop structures by locating maxima of the image convolution. We localise significant rooftop responses by applying a simple thresholding technique to the convolution output and locating the centroids of the associated connected components. A neighbourhood structure is established by computing the Voronoi tessellation of the centroids. The Delaunay triangulation of the Voronoi regions provides us with a relational structure for matching.

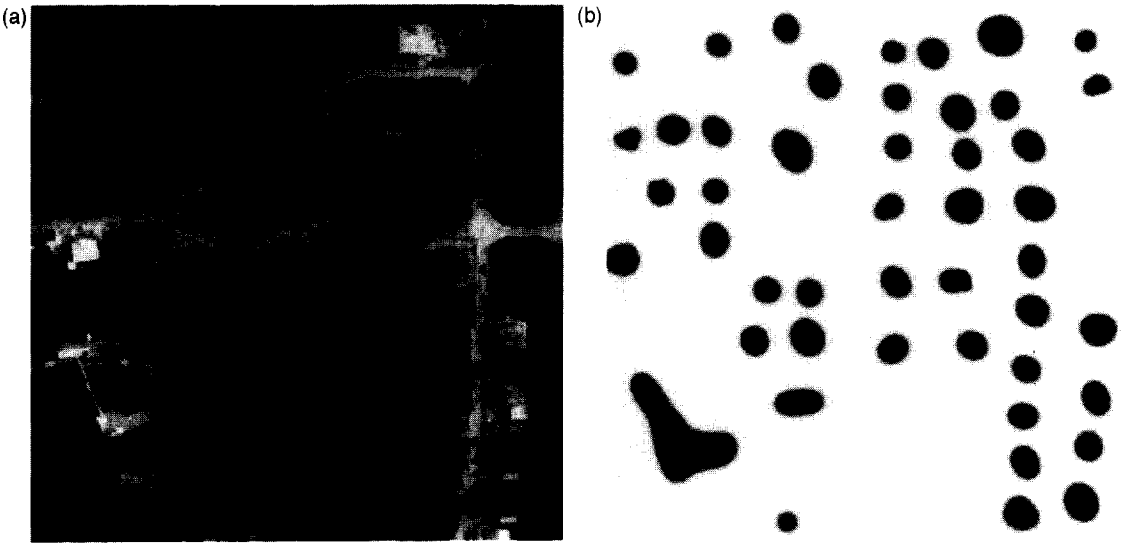

Fig. 14. (a) Training image. (b) Convolution response for the training data.

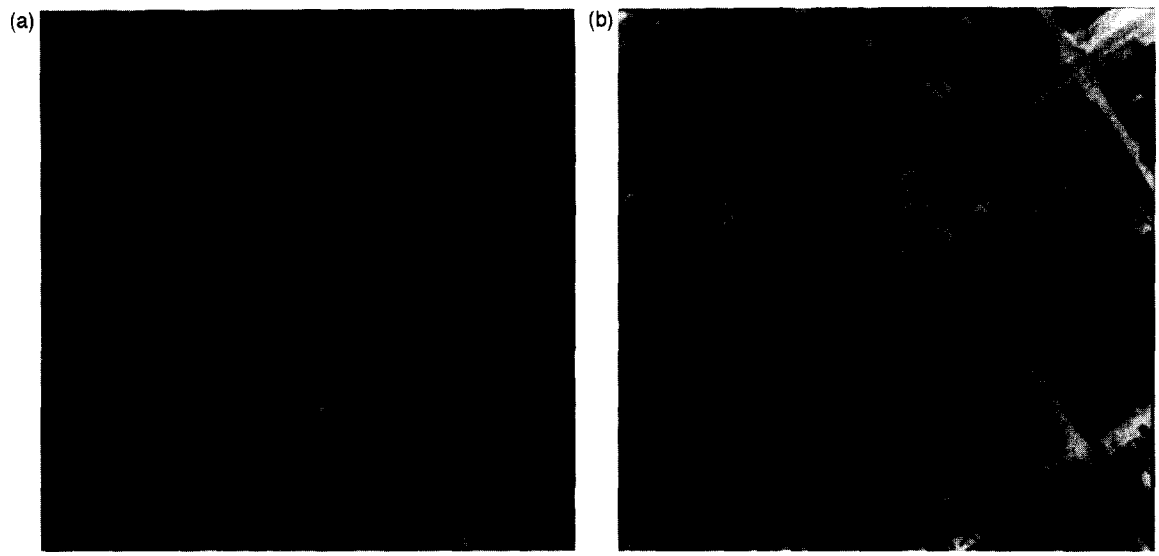

Fig. 15. (a) Matched filter applied to the right stereo image. (b) Houses detected.

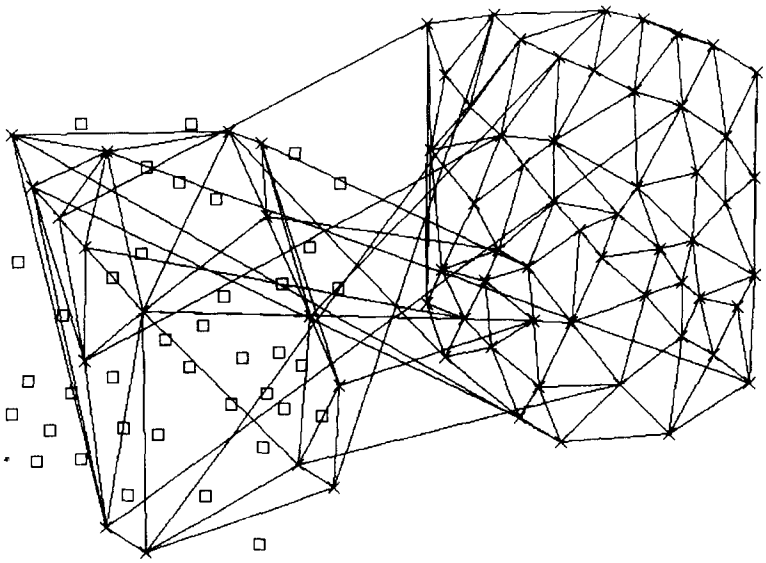

Fig. 16. Initial matching results (popsize=100).

This simple processing chain is illustrated in Figs 14 and 15. Figure 14(a) shows the training image used to compute the matched filter coefficients. Figure 14(b) shows the result of reapplying the convolution filters to the training data. Figure 13 shows the aerial stereograms under match. The result of applying the convolution filters to one of these images is shown in Fig. 15(a). Figure 15(b) shows the original data with the centroids of the thresholded connected components superimposed. Finally, Fig. 16 shows the extracted Delaunay triangulations for the left and right stereograms.

5.2. *Results*

The lines between the two graphs in Figs 16 and 17 and 18 represent the current state of matches between the two graphs. This sequence of figures shows various stages in the iterative matching process. Figure 16 is the fittest match from the initial population, while Fig. 19 is the final match. Figures 17 and 18 show intermediate solutions which are partially consistent. We present them to show how the subgraph crossover can combine partially consistent solutions to form more globally consistent solutions of the type shown in Fig. 19. It is important to stress that only one hill-climbing and crossover epoch has elapsed between the results shown in Figs 17 and 18 and those shown in Fig. 19.

This example represents a less demanding test than the synthetic matching experiment in Figs 9–11. There is only one spurious node, which appears in the left-hand image, due to thresholding errors in the centroid location process. This node is correctly identified as

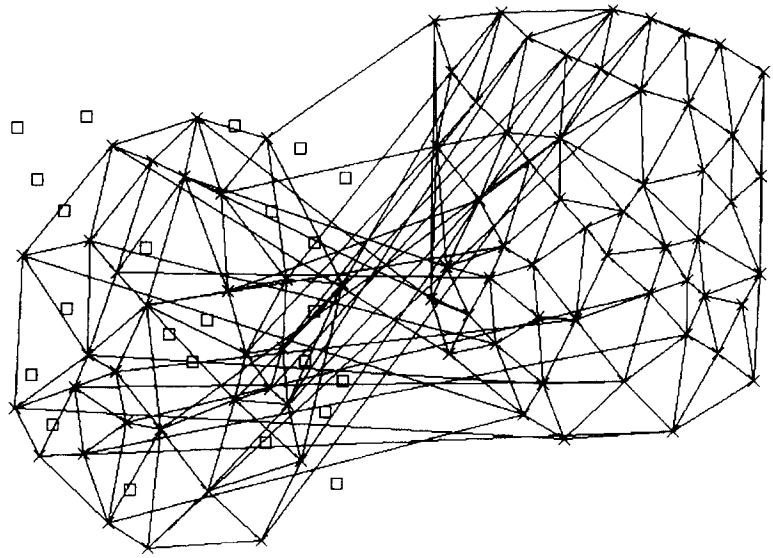

Fig. 17. Intermediate matching results (popsize=100).

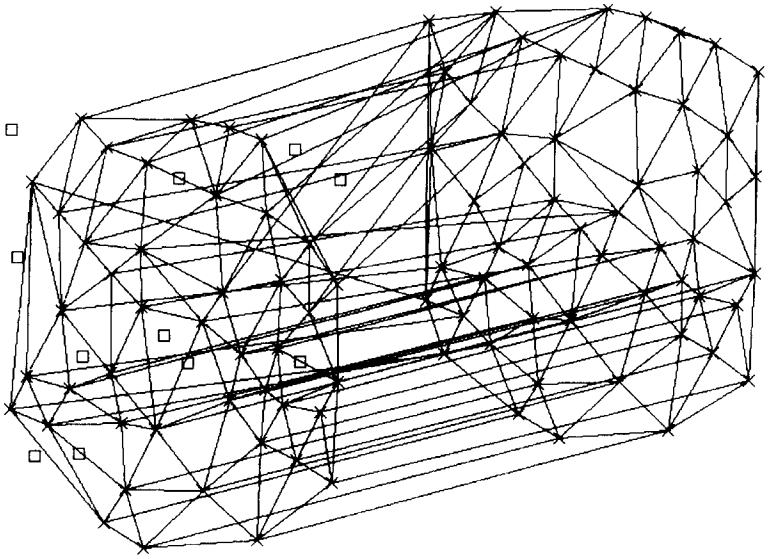

Fig. 18. Intermediate matching results (popsize=100).

clutter and is deleted. The remaining nodes are all correctly matched. Although the segmental input to this matching problem is good, the regularity of the graphs poses potential problems of ambiguity. The genetic search process is effective in overcoming these and locates a solution of maximum global consistency.

6. CONCLUSIONS

To conclude, we have shown how the optimisation of the relational consistency measure of Wilson and Hancock[12,13] naturally maps onto genetic search. Moreover, the physical variable underpinning this consistency measure, namely Hamming distance, allows us to picture the action of the various stages of the search process in an intuitive way. Based on an extensive simulation study, we evaluate the performance of the resulting optimisation process. This yields a number of interesting results. Empirical results suggest that the number of computations required to locate a consistent solution rises in a polynomial manner with graph size. Moreover, despite being stochastic in nature the convergence is more rapid than simulated annealing. It is also interesting to note that the method requires a population size which is approximately equal to the number of nodes in the graphs under match. In addition to the population size the only parameter of the method is the mutation probability. Again, convergence is not critically dependent upon the choice of parameter.

The genetic search procedure therefore represents a relatively effective stochastic optimisation process that is not sensitive to the choice of parameters. One intriguing

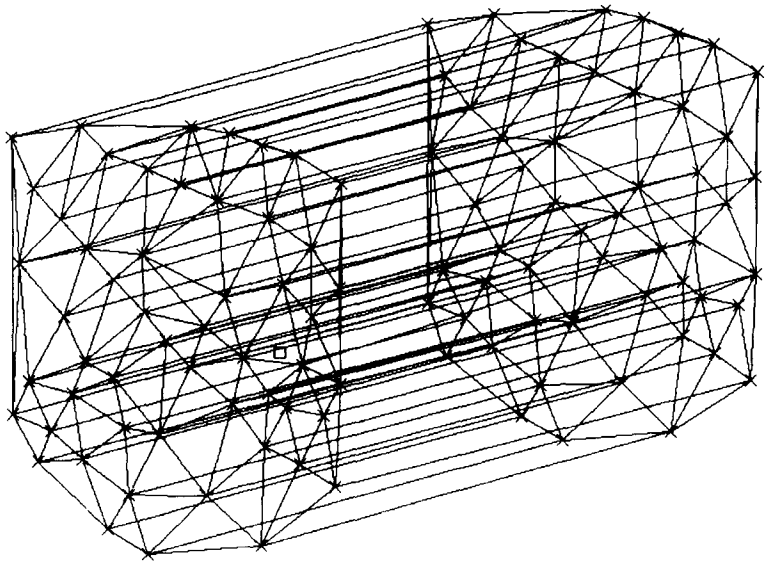

Fig. 19. Final matching results (popsize=100).

possibility not explored in this paper is to exploit the genetic population to maintain equivalent ambiguous interpretations of the same graphs. This topic has not been extensively studied in the relaxation literature since the work of Faugeras and Berthod.[1] We are currently extending the work reported in this paper by conducting a study of the classical toy triangle and other ambiguous labelling problems. The results will be reported in due course.[30,38–40]

REFERENCES

1. O. D. Faugeras and M. Berthod, Improving consistency and resolving ambiguity in relaxation labelling, *IEEE PAMI* **3**, 412–424 (1981).
2. D. B. Fogel, An introduction to simulated evolutionary optimisation, *IEEE Trans. Neural Networks* **5**, 3–14 (1994).
3. S. Geman and D. Geman, Stochastic relaxation, Gibbs distributions and Bayesian restoration of images, *IEEE PAMI* **6**, 721–741 (1984).
4. S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, Optimisation by simulated annealing, *Science* **220**, 671–680 (1983).
5. C. Peterson and B. Soderberg, A new method for mapping optimisation problems, *Int. J. Neural Systems* **1**, 2–33 (1989).
6. X. F. Qi and F. Palmieri, Theoretical analysis of evolutionary algorithms with an infinite population in continuous space: basic properties of selection and mutation, *IEEE Trans. Neural Networks* **5**, 102–119 (1994).
7. X. F. Qi and F. Palmieri, Theoretical analysis of evolutionary algorithms with an infinite population in continuous space: analysis of the diversification role of crossover, *IEEE Trans. Neural Networks* **5**, 120–129 (1994).
8. A. Yuille, Generalised deformable models, statistical physics and matching problems, *Neural Comput.* **2**, 1–24 (1990).
9. H. G. Barrow and R. J. Popplestone, Relational descriptions in picture processing, *Mach. Intell.* **6**, 377–396 (1971).
10. L. G. Shapiro and R. M. Haralick, A metric for comparing relational descriptions, *IEEE PAMI* **7**, 90–94 (1985).
11. A. Sanfeliu and K. S. Fu, A distance measure between attributed relational graphs for pattern recognition, *IEEE SMC* **13**, 353–362 (1983).
12. R. C. Wilson and E. R. Hancock, Graph matching by discrete relaxation, in *Pattern Recognition in Practice IV*, E. Gelsema and L. Kanal, eds, pp. 165–176 (1994).
13. R. C. Wilson, A. N. Evans and E. R. Hancock, Relational matching by discrete relaxation, *Image Vision Comput.* **13**, 411–421 (1995).
14. B. Gidas, A re-normalisation-group approach to image processing problems, *IEEE PAMI* **11**, 164–180 (1989).
15. L. Herault, R. Horaud, F. Veillon and J-J. Niez, Symbolic image matching by simulated annealing, *Proc BMVC90*, 319–324 (1990).
16. E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines*. Wiley, New York (1989).
17. D. Goldberg, *Genetic Algorithms in Search, Optimisation and Learning*. Addison-Wesley, Reading, Massachusetts (1989).
18. J. R. Beveridge and E. M. Riseman, Optimal geometric model matching under full 3D perspective, *Comput. Vision Image Understanding* **61**, 351–364 (1995).
19. J. R. Beveridge, E. R. Riseman and C. R. Graves, Demonstrating polynomial run-time growth for local search, *Proc. IEEE Comput. Soc. Int. Symp. on Computer Vision*, Coral Gables, Florida, 533–538 (1995).
20. A. K. C. Wong and M. You, Entropy and distance of random graphs with application to structural pattern recognition, *IEEE PAMI* **7**, 599–609 (1985).
21. D. Waltz, Understanding line drawings of scenes with shadows, in *The Psychology of Computer Vision*, P.H. Winton, ed. McGraw-Hill, New York (1975).
22. L. Davis, Bit-climbing, representational bias and test suite design, *Proc. 4th Int. Conf. on Genetic Algorithms*, 18–23 (1991).
23. L. Eshelman, The CHC adaptive search algorithm: how to have safe search when engaging in non-traditional genetic recombination, *Found. Genetic Algorithms* **1**, 265–283 (1991).
24. S. J. Louis and G. J. E. Rawlins, Syntactic analysis of convergence in genetic algorithms, *Found. Genetic Algorithms* **2**, 141–151 (1993).
25. E. S. Gelsema, Abductive reasoning in Bayesian belief networks using a genetic algorithm, *Pattern Recognition Lett.* **16**, 865–871 (1995).

26. M. Pelillo, F. Abbattista and A. Maffione, An evolutionary approach to training relaxation labelling processes, *Pattern Recognition Lett.* **16**, 1069–1078 (1995).
27. E. R. Hancock and J. Kittler, Discrete relaxation, *Pattern Recognition* **23**, 711–733 (1990).
28. J. Besag, On the statistical analysis of dirty pictures, *Proc. Roy. Statist. Soc. Ser. B* **48**, 259–302 (1986).
29. D. Whitley, J. R. Beveridge, C. Graves and K. Mathias, Test driving three 1995 genetic algorithms: new test functions and geometric matching, *Heuristics* **1**, 77–104 (1995).
30. E. R. Hancock and J. Kittler, A Bayesian interpretation for the Hopfield network, *IEEE Int. Conf. Neural Networks* **1**, 341–346 (1993).
31. W. Cedeno, V. R. Vemuri and T. Slezak, Multi-niche crowding in genetic algorithms and its application to the assembly of DNA restriction fragments, *Evolutionary Comput.* **2**, 321–345 (1995).
32. R. C. Wilson and E. R. Hancock, Gauging consistency and controlling structural errors, *Proc. IEEE Comput. Soc. Comput. Vision and Pattern Recognition Conf.*, 47–54 (1996).
33. R. C. Wilson and E. R. Hancock, Relational matching with dynamic graph structures, *Proc. 5th Int. Conf. on Computer Vision*, 450–456 (1995).
34. A. D. J. Cross and E. R. Hancock, Relational matching with stochastic optimisation, *IEEE Int. Symp. on Computer Vision*, 365–370 (1995).
35. W. E. L. Grimson, The combinatorics of object recognition in cluttered environments using constrained search, *Artificial Intell.* **44**, 121–165 (1990).
36. B. T. Messmer and H. Bunke, Efficient error-tolerant subgraph isomorphism detection, in *Shape, Structure and Pattern Recognition*, D. Dori and A. Bruckstein, eds, pp. 231–240 (1995).
37. A. D. J. Cross and E. R. Hancock, Inexact matching with genetic search, Advances in Structural and Syntactical Pattern Recognition, P. Perner, P. Wang and A. Rosenfeld, eds, Lecture Notes in Computer Science, Vol. 1121, pp. 150–159. Springer, Berlin (1996).
38. A. D. J. Cross, R. C. Wilson and E. R. Hancock, Genetic search for structural matching, *Proc. ECCV96*, B. Buxton and R. Cipolla, eds, *Lecture Notes in Computer Science*, Vol. 1064, 514–525. Springer, Berlin (1996).
39. A.M. Finch, R.C. Wilson and E.R. Hancock, , *Matching Delaunay graphs, Pattern Recognition* **30**, 123–146 (1997).
40. R. C. Wilson and E. R. Hancock, A Bayesian compatibility model for graph matching, *Pattern Recognition Lett.* **17**, 263–276 (1996).

About the Author — ANDREW CROSS is currently undertaking research towards a D.Phil. degree in Computer Vision and Pattern Recognition in the Department of Computer Science at the University of York. Prior to this he gained a first class honours degree in Computational Physics at the University of Manchester Institute of Science and Technology, graduating in 1994. His research interests are in novel optimisation techniques, computer vision and graphics. He has published some 10 papers on these topics.

About the Author — RICHARD WILSON is currently a research associate in the Department of Computer Science at the University of York. Dr Wilson was awarded an open scholarship to study Physics at St John's College, University of Oxford, and graduated with first class honours in 1992. Between 1992 and 1995 he undertook research at the University of York on the topic of relational graph matching for which he was awarded the D.Phil. degree. He has published some 30 papers in journals, edited books and refereed conferences. His research interests are in high-level vision, scene understanding and structural pattern recognition.

About the Author — EDWIN HANCOCK gained his B.Sc. in Physics in 1977 and Ph.D. in High Energy Nuclear Physics in 1981, both from the University of Durham, U.K. After a period of postdoctoral research working on charm-photo-production experiments at the Stanford Linear Accelerator Centre, he moved into the fields of computer vision and pattern recognition in 1985. Between 1981 and 1991, he held posts at the Rutherford–Appleton Laboratory, the Open University and the University of Surrey. Since 1991, Dr Hancock has been a Lecturer in Computer Science at the University of York where he now leads a group of some 10 researchers in the areas of computer vision and pattern recognition. He has published over 100 refereed papers in the areas of high energy nuclear physics, computer vision, image processing and pattern recognition. He was awarded the 1990 Pattern Recognition Society Medal. Dr Hancock serves as an Associate Editor of the journal *Pattern Recognition* and has been a guest editor for the *Image and Vision Computing Journal*. He chaired the 1994 British Machine Vision Conference and has been a programme committee member for several national and international conferences.