Pergamon

Pattern Recognition, Vol. 31, No. 5, pp. 623–639, 1998

0031–3203/98 $19.00 + .00

PII: S0031-3203(97)00067-8

OPTIMAL ENCODING OF GRAPH HOMOMORPHISM ENERGY USING FUZZY INFORMATION AGGREGATION OPERATORS

P. N. SUGANTHAN,*,† H. YAN,‡ E. K. TEOH† and D. P. MITAL†

† School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 2263, Republic of Singapore
‡ Department of Electrical Engineering, University of Sydney, NSW 2006, Australia

(*Received* 8 *September* 1995; *accepted* 12 *June* 1997)

Abstract—The attributed relational graph matching (ARG) strategy is a well-known approach to object/pattern recognition. In the past for the parallel solution of ARG matching problem, an overall objective function was constructed using linearly weighted information aggregation function and one set of parameter values was chosen for all models by trial-and-error for the parameters in the function. In this paper, the compatibility between every pair of model and scene attributes is interpreted as a fuzzy value and subsequently the nonlinear fuzzy information aggregation operators are used to fuse the information captured in the chosen attributes. To learn the parameters in the fuzzy information aggregation operators, the "learning from samples" strategy is used. The learning of weight parameters is formulated as an optimisation problem and solved using the gradient projection algorithm based learning procedure. The learning procedure implicitly evaluates ambiguity, robustness and discriminatory power of the relational attributes chosen for graph matching and assigns weights appropriately to the chosen attributes. The learning procedure also enables us to compute a distinct set of optimal parameters for every model to reflect the characteristics of the model so that the homomorphic ARG matching problem can be optimally encoded in the energy function for the model. Experimental results are presented to illustrate effectiveness and necessity of the parameter estimation and learning procedures.

Parameter learning Pattern recognition Fuzzy information aggregation operators
Information fusion Constrained optimisation Potts MFT neural network
Attributed relational graph matching Inexact homomorphism

1. INTRODUCTION

The attributed relational graph matching (ARG) strategy is a well-known approach to object/pattern recognition. As ARG matching is an NP-complete problem, parallel ARG matching approaches§ are preferred to the sequential search methods. In our recent papers,(1, 2) a novel programming strategy was proposed to generate homomorphic graph matching using the optimising connectionist models. It was also emphasised that in the past the parallel graph matching problem had been primarily addressed from the viewpoint of subgraph isomorphism which implicitly requires the scene to have at most one occurrence of any object model. In all parallel ARG matching approaches, an overall cost/objective function is constructed and optimised. In order to construct such an objective function, an information aggregation formulation and numerical values for a number of parameters in the formulation have to be chosen to fuse the information captured in the chosen attributes. In the past, linearly weighted information aggregation functions have been used. In general, the weighting factors and other parameters in the formulation have been assigned with numerical values by trial-and-error approach. Further, a single set of parameters has been used irrespective of the variations between different models.

In this paper, the compatibility value between every pair of model and scene attributes is interpreted as a fuzzy value and the mapping of the ARG matching problem is regarded as information fusion. In order to fuse the information captured in the chosen attributes, fuzzy information aggregation operators are employed. The computation of the parameters in the fuzzy aggregation operators is formulated as an optimisation problem and solved using a gradient projection algorithm based learning scheme. The learning scheme implicitly evaluates various characteristics of every attribute such as ambiguity, robustness and discriminatory power and assigns weighting factors accordingly. The parameter estimation and learning schemes

* Author to whom correspondence should be addressed. Current address: Department of Computer Science and Electrical Engineering, University of Queensland, QLD 4072, Australia. E-mail: sugan@elec.uq.edu.au.

§ In parallel ARG matching, all scene vertices are simultaneously matched with all model vertices and the best match satisfying syntactic constraints is obtained. In sequential matching, one scene vertex is chosen and matched with model vertices one- by- one to determine a possible match. For more detail, refer to references (1)–(3).

enable the ARG matching problem to be optimally encoded in the energy function and subsequently solved by a parallel ARG matching strategy such as the Potts MFT networks,[1] the Hopfield network,[2] relaxation labelling[4] and simulated annealing.[5] In this paper, the Potts MFT neural network is employed to perform the recognition. Line and circle patterns recognition problems are used to illustrate the effectiveness and necessity of parameter learning scheme.

The rest of the paper is organised as follows. In Section 2, the ARG representation scheme, line and circle pattern models and scenes are introduced. The Potts MFT neural networks and energy formulation to generate homomorphic ARG mapping are presented in Section 3. The motivations, fuzzy information aggregation operators, related work in optimal utilisation of various attributes for efficient recognition and the parameter estimation and learning schemes are presented in Section 4. Experimental results and discussions are presented in Section 5. The paper is concluded in Section 6.

2. ARGs, PATTERN MODELS AND SCENES

2.1. *Graph representation and morphisms*

Since recognition by ARG matching does not require a segmentation of the scene feature primitives as belonging to different instances of object model(s) prior to recognition and allows geometric and photometric information to be captured in itself, it has been employed frequently[6–10] in object recognition, scene labelling and stereo correspondence and motion correspondence applications. The ARG representation is defined below.

An *attribute* is an ordered 2-tuple, (a^t, a^v), consisting of a name or type a^t and a value, a^v. An *attribute set* is a set of all 2-tuples belonging to a particular entity–namely a vertex or an edge. A *vertex* of an ARG has a unary attribute set associated with it, referred to as *vertex attribute set*. The attribute set that belongs to a vertex x with N_u attributes can be denoted as $\{\ldots, (a_{n_u}^{t,x}, a_{n_u}^{v,x}), \ldots\}$ where $n_u = 1, \ldots, N_u$.

An *attributed edge*, e_{xy} between vertices v_x and v_y has an associated binary attribute set called *edge attribute set*. If a rotation invariant representation is desired, the ARG edges are undirected, that is $e_{xy} = e_{yx}$, otherwise $e_{xy} = -e_{yx}$ or $e_{xy} \neq e_{yx}$. In this paper, we are concerned with rotation invariant recognition. An ARG of a scene s is denoted by $G_s = (V^s, E^s)$ where V^s is the set of attributed vertices, $\{v_1^s, v_2^s, \ldots, v_N^s\}$ and E^s is the set of attributed edges, $\{\ldots, e_{xy}^s, \ldots\}$. Likewise an ARG of an object model m can be defined. Object recognition by graph matching is a mapping, also referred to as a *morphism*, from a scene relational graph G_s to a model graph G_m. In this paper, we are interested in homomorphic ARG matching which is many (scene vertices)-to-one (model vertex) mapping.

2.2. *Models and scenes*

In our experiments, synthetic object models and scenes are used to test the performance of our algorithms. Similar synthetic images have been used widely[8,9] when the subject under investigation was exclusively high level vision. Further, these objects offer an elegant means of controlling the types and degrees of ambiguities, so that the attractive properties of our algorithms can be illustrated easily.

2.2.1. Line pattern models and scenes. The synthetic line pattern models consist of random line patterns. Figures 3(a)–(d) show four synthetic line pattern models. A scene can be created by choosing a number of subset of vertices (line segments) from each model, applying a rigid-body rotation and translation separately, deviating the end points randomly to introduce noise and arbitrarily overlapping them. A few extraneous line segments may also be introduced to the scenes. Two such line pattern scenes are shown in Figs 3(e) and (f).

In the ARG of line patterns, vertices represent lines and edges represent relationships between them. The line patterns are described using a unary (or vertex) attribute (r1 below) and four binary (or edge) attributes (r3–r6 below) in the ARG representation. The attributes are

r1. line segment length,
r3. angle between two line segments,
r4. distance between centre points of two line segments,
r5. maximum distance between end points of two line segments, and
r6. minimum distance between end points of two line segments.

To measure unary and binary compatibilities between relations, the following functions are used:[1,2]

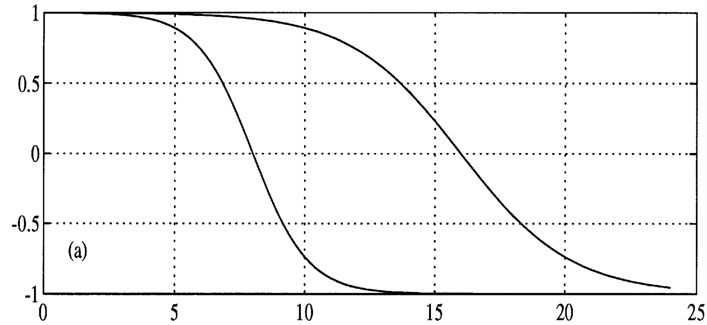
$$f^u(b_1, b_2, k, t) = -\tanh(k(|b_1 - b_2| - t)), \quad (1)$$

$$f^b(b_1, b_2, k, t) = \tfrac{1}{2}(1 - \tanh(k(|b_1 - b_2| - t))). \quad (2)$$

where b_1, b_2, k and t are model attribute value, scene attribute value, the steepness and the tolerance parameters corresponding to the model attribute, respectively. The functions are shown in Figs 1(a) and (b) for different values of steepness and the threshold parameters. In the figures, the X-axis and Y-axis represent $k(|b_1 - b_2| - t)$ and the value of $f^*(\cdot)$ respectively.

2.2.2. Circle pattern models and scenes. The synthetic circle patterns are shown in Figs 5(a)–(d). Two scenes were generated as explained in line patterns application, and are shown in Figs 5(e) and (f). The circle patterns are described using one unary (vertex) attribute (r1 below) and one binary (edge) attribute (r3 below). They are

r1. radius of circle,
r3. distance between centre points of two circles.



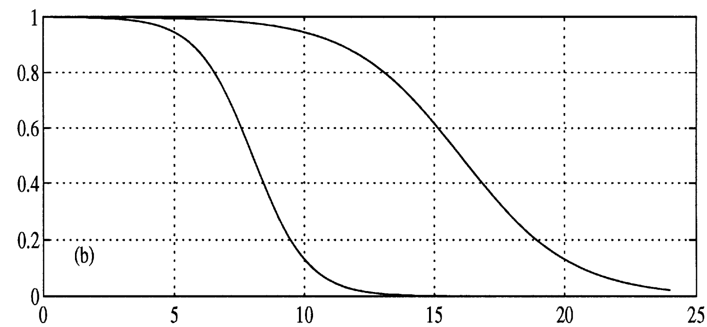


Fig. 1. Two types of compatibility functions: (a) function $f^u(\cdot)$, (b) function $f^b(\cdot)$.

To measure compatibilities between binary and unary relations, the functions defined in Section 2.2.1 are used.

3. HOMOMORPHIC ARG MATCHING BY POTTS MFT NETWORKS

In this paper, the Potts MFT neural network based homomorphic ARG matching scheme is employed to perform the recognition. The energy function and the constraint used in the MFT algorithm are:[11]

$$E = -\tfrac{1}{2}\mathbf{V}^{\mathrm{T}}\mathbf{T}\mathbf{V} - \mathbf{i}^b\mathbf{V}, \tag{3}$$

$$\sum_i V_{xi} = 1, \tag{4}$$

where the vectors $\mathbf{V}$ and $\mathbf{i}^b$ are the output potential and the external bias current, respectively, and $\mathbf{T}$ is the connectivity matrix. The constraint $\sum_i V_{xi} = 1$ is appropriate to solve problems such as the travelling salesman problem (TSP) and graph partitioning (GP).[11] The update equations are as follows.

$$u_{xi} = -\frac{1}{T}\frac{\partial E}{\partial V_{xi}}, \tag{5}$$

$$V_{xi} = \frac{\mathrm{e}^{u_{xi}}}{Z_x}, \tag{6}$$

where the vector $\mathbf{u}$, Z_x and T are the input state potential, the local partition function expressed by $\sum_j e^{u_{xj}}$ and the temperature parameter T, respectively. Since the constraint that one scene vertex should be matched to at most one model vertex, can be enforced by neuron normalisation, the energy function used to solve the recognition problem is identical to the actual cost of the recognition problem. The energy function for homomorphic ARG matching is:

$$E = -\frac{1}{2}\sum_{x,y,i,j} C_{xi,yj} V_{xi} V_{yj}. \tag{7}$$

The mapping of the matching problem onto the network is illustrated in Fig. 2. In the above cost function formulation, the compatibility measure $C_{xi,yj}$ $(= C_{yj,xi})$ encodes both unary compatibilities between the model ARG vertex v_x and the scene ARG vertex v_i and the model ARG vertex v_y and the scene ARG vertex v_j and binary compatibilities between the model edge e_{xy} and the scene edge e_{ij} collectively. As the energy function is minimised, the compatibility

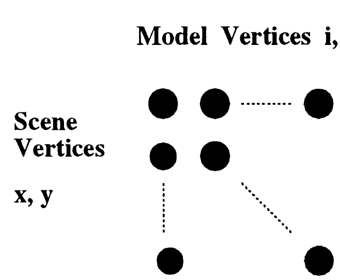


Fig. 2. The Potts MFT neural networks used to generate homomorphic ARG matching.

$C_{xi,yj}$ between two corresponding vertices pairs, i.e. $v_x \to v_i$ and $v_y \to v_j$, should be positive. This positive compatibility in turn encourages neurons V_{xi} and V_{yj} to "on" state. Conversely, negative compatibility between two non-corresponding vertices pairs, i.e. $v_x \nrightarrow v_i$ and $v_y \nrightarrow v_j$, encourages neurons V_{xi} and V_{yj} to "off" state. In Section 4.1.1, the computation of compatibility measure is cast as an information fusion problem and solved using the fuzzy information aggregation operators.

The constraint to be enforced strongly to solve the TSP and GP problems is $\sum_i V_{xi} \neq 1$ as in equation (4). To solve the relational homomorphism, the only constraint is $\sum_i V_{xi} \leq 1$. Therefore, the standard update equation (6) is modified as follows:

$$V_{xi} = \begin{cases} \frac{e^{u_{xi}/T}}{Z_x} & \text{if } Z_x > 1, \\ e^{u_{xi}/T} & \text{otherwise.} \end{cases} \tag{8}$$

Programming the network connections and bias currents for line and circle patterns recognition, network initialisation and hypothesis interpretation are explained in detail in references (1) and (2) and therefore not discussed in this paper.

4. LEARNING COMPATIBILITY FUNCTION PARAMETERS

In this section, procedures are presented to learn and estimate parameters used in the compatibility function for optimal encoding of homomorphic energy. The concept of learning from training samples has been widely employed. In the back propagation and the radial basis function networks, hetero-association of input and output pattern pairs are used to learn an internal representation of the problem. On the other hand, in CAM models, such as the Hopfield network, homo-association of input–output pairs are used to learn weight values. In this application, homom-association of attributes are used to learn and estimate parameters in the compatibility measure equations (11) (for line pattern recognition) and (12) (for circle pattern recognition). This learning is performed off-line to obtain values for a number of parameters. These parameters are then used during the recognition stage to optimally encode the ARG matching energy for parallel solution.

4.1. Fuzzy connectives-based optimal mapping

In computer vision, various sensing modalities are used to acquire data. Generally, the acquired data is unreliable, ambiguous and uncertain. Therefore, at various levels of information processing in computer vision, decisions have to be made by aggregating unreliable partial evidences captured in various attributes of the acquired data. Lee[12] studied a number of uncertainty management frameworks such as Bayesian probability approach,[13] MYCIN/EMYCIN calculi,[14,15] Dempster–Shafer belief theory[16] and fuzzy set theory based methods and concluded that the fuzzy set theory based approaches were the most appropriate framework for information fusion and multi-attribute decision making in computer vision due to its flexibility, completeness and efficiency. In ARG matching, the compatibility value between each pair of model and scene attributes can be regarded as unreliable partial confidence of match between the corresponding model and scene vertices, and fuzzy set connectives can be employed to aggregate the partial evidence captured by every pair of attributes.

The fuzzy connectives have been widely used in a number of domains to aggregate pieces of partial evidence in different attributes to obtain an overall confidence measure. Some examples of their applications are classification of remote sensing data,[17] colour image segmentation and recognition[18] and pattern recognition.[19] In any application, a proper aggregation operator should be chosen to reflect the inter-relationships between various attributes used to obtain the overall confidence measure. Although a large number of information aggregation operators are known, the following types of operators have been commonly used: (a) intersection operator;[20,21] (b) union operator;[22,21] (c) averaging operator;[23] (d) generalised mean operator[24] and (e) hybrid operators.[23,25]

Amongst the operators listed above, the union operator provides complete compensation, while the intersection operator provides no compensation at all.[24] In other words, in the case of ARG matching, for a pair of vertices to be non-corresponding, all pairs of attributes must be incompatible if the union operator is used, while at least one pair of attributes must be incompatible if the intersection operator is used. Apparently, these operators do not reflect the nature of the ARG matching problem. In ARG matching applications, some attributes may be ambiguous, while other attributes may be unique. Likewise, their robustness against sensing noise, feature extraction noise and occlusion noise also vary considerably. Therefore, the operator that aggregates partial evidences to generate the overall evidence, should also be able to compensate for ambiguity and noise in some attributes with other informative and robust attributes. Further, significance of each attribute should also be weighted to reflect the importance of each attribute with respect to others. A number of compensatory operators, such as the generalised mean operator and hybrid operator, have been proposed with weighting factors. In the following section, the generalised mean operator is introduced.

4.1.1. Generalised mean operator. The generalised mean of n-arguments $x_1, x_2, \ldots, x_n \in [0, 1]$ is a function g_m: $[0, 1]^n \to [0, 1]$ which is defined as[24]

$$g_m(x_1, x_2, \ldots, x_n; w_1, w_2, \ldots, w_n; p) = \left(\sum_{i=1}^{n} w_i x_i^p \right)^{1/p}, \tag{9}$$

where $p \in R$ and $p \neq 0$. The weighting factors should satisfy the following conditions:

$$\sum_{i=1}^{N} w_i = 1 \quad \text{and} \quad w_i \geq 0 \, \forall \, i. \tag{10}$$

The weighting factors may be expressed as a vector $W = [w_1, w_2, \ldots, w_n]$. Further, the generalised mean operator has the following properties[24] as $p \to \pm\infty$, and bounded within the respective limits:

1. $\lim_{p \to \infty} \quad g_m(x_1, x_2, \ldots, x_n; w_1, w_2, \ldots, w_n; p) = \max (x_1, x_2, \ldots, x_n)$,
2. $\lim_{p \to -\infty} \; g_m(x_1, x_2, \ldots, x_n; \; w_1, w_2, \ldots, w_n; p) = \min (x_1, x_2, \ldots, x_n)$,
3. $\min(x_1, x_2, \ldots, x_n) \leq g_m(x_1, x_2, \ldots, x_n; w_1, w_2, \ldots, w_n; p) \leq \max(x_1, x_2, \ldots, x_n)$.

The overall compatibility measure between model vertices v_i, v_j and scene vertices v_x, v_y can be computed using the generalised mean operator for line patterns matching as follows (by replacing the fuzzy variables x_i with appropriate compatibility values $f^*(\cdot)$ while retaining weights and parameter p):

$$C_{xi,yj} = g_m(f^u_{r1}(\cdot), f^u_{r1}(\cdot), f^b_{r3}(\cdot), f^b_{r4}(\cdot), f^b_{r5}(\cdot), f^b_{r6}(\cdot); \\ w_1, w_2, \ldots, w_6; p) \tag{11}$$

where $w_1 = w_2$ corresponds to the unary attribute $r1$ and $w_3, \ldots, w_6$ correspond to the binary attributes $r3, \ldots, r6$. In the above equation, compatibility function parameters [see equations (1) and (2)] are not shown for clarity. In this formulation, it is assumed that every type of attribute is assigned with a particular set of values for weighting factor, threshold parameter and steepness parameter. In line patterns matching, there are five sets of weighting factors, threshold parameters and steepness parameters corresponding to attributes r_1, r_3, r_4, r_5, r_6 and the parameter p to be obtained.

Likewise, the overall compatibility measure can be computed using the generalised mean operator for circle patterns matching as follows:

$$C_{xi,yj} = g_m(f^u_{r1}(\cdot), f^u_{r1}(\cdot), f^b_{r3}(\cdot); w_1, w_2, w_3; p) \tag{12}$$

where $w_1 = w_2$ and w_3 correspond to the unary and binary attributes, respectively. In circle patterns matching, there are two sets of weighting factors, threshold parameters and steepness parameters corresponding to attributes r_1 and r_3 and the parameter p to be obtained.

In theory, it is possible to assign distinct set of parameters for every model attributes and learn them using the learning scheme presented in Section 4.3. However, the complexity of the learning scheme will be immensely increased. Let us consider the number of weighting factors, threshold parameters or steepness parameters needed for a model that has N vertices of a particular type (Here, type means either a line segment, circular arc segment, circle or a corner point with N_u unary attributes and N_b binary attributes.) The total number of weighting factors is $NN_u + \frac{1}{2} - (N-1)N_b$. Apparently, as the number of vertices or attributes is increased, the number of weighting factors increases quadratically or linearly. Apparently, it is essential to reduce the number of parameters to be learned in order to increase the efficiency of the recognition system. Therefore, in this study, the simplified compatibility equations (11) and (12) are used to map the ARG matching problem for line patterns and circle patterns, respectively. These simplified equations have distinct weight, threshold and steepness parameters for every attribute for every model. For instance, for line patterns, there are five sets of threshold, steepness and weight parameters for every model.

4.2. Related work

In the past, research in optimal utilisation of multi-source–multiattribute, and single-source-multiattribute data was carried out in the context of object recognition,[26–28] pixel classification of outdoor natural scenes,[29] pattern recognition,[30] feature selection for stereo correspondence,[31] learning compatibility coefficients for relaxation labelling processes[32] and robotic vision.[33] Therefore, it is appropriate to review some of the related work prior to presenting our approach.

A recent work on optimal feature selection from CAGD model database for range image recognition was conducted by Hansen and Henderson.[27] The following criteria were identified to be important for efficient recognition of objects using an arbitrary recognition strategy.

(1) *Discriminatory power*: This is probably the most important characteristic of a representation scheme for object discrimination. The features with higher discriminatory power, in other words unique or rare features, should be assigned with higher priority or weight values.
(2) *Robustness*: This criterion represents how accurately a feature can be sensed and extracted from the sensed data. This factor collectively accounts for noise introduced during the sensing as well as feature extraction processes. Generally, binary relations are more robust against noise, while symbolic features such as surface types,[34] colour,[35] and angle-related binary relations are more robust against occlusion.
(3) *Completeness*: It is expected that the representation scheme completely describes each object models so that pose estimation can also be performed. However, it is an understatement in most cases. Generally, a redundant representation is preferred over just a complete one because redundancy may provide extra tolerance against noise and occlusion.

Hansen and Henderson[27] applied so-called *feature filters* sequentially to extract features that fulfill the above requirements. Although the sequence of application of feature filters was crucial, they stopped short of providing any guideline on this aspect. Further, extracted such features were sequentially ranked which is only suitable for sequential object matching strategies such as the interpretation tree search. In their experiments they used only one polyhedral object. Therefore, their experiments did not involve discriminating between different objects models.

Turney *et al.*[28] studied the recognition of partially occluded parts from their silhouette templates. They primarily considered the discriminatory power of the silhouette segments referred to as subtemplates. The subtemplates were assigned with weighting factors that reflected their discriminatory power between intra-model and inter-model subtemplates. Krishnapuram and Lee[29] also studied the fusion of information in computer vision using fuzzyset-based hierarchical neural networks. They employed gradient-descent-based learning algorithms to determine the significance or weight values of unary attributes and to detect redundancy. However, Turney *et al.* and Krishnapuram and Lee did not employ binary and higher-order relational attributes.

Lew *et al.*[31] investigated the issue of optimal feature selection for stereo correspondence. They used features such as intensity, X-gradient, Y-gradient, gradient direction, gradient magnitude, Laplacian and curvature of feature points. The left image was matched to itself to determine discriminatory power of features and to assign weights accordingly. They optimised a class separation distance criterion to obtain weights. Pellio and Refice[32] recently presented a methodology to learn the compatibility coefficients used in the relaxation labelling from sample training scenes. They used the quadratic cost function defined as the discrepancy between the final labelling and the labelling provided in the training scenes. The compatibility coefficients that minimised the cost were considered as the optimal values. To obtain the solution, active-set-based gradient projection algorithm was used.[36]

Apparently, most of the approaches discussed above deal with the classification of feature vectors which does not involve relational attributes. Further, the learning schemes of Pellio and Refice[32] and Li[26] did not assign distinct parameter sets for every model, while the Hansen and Henderson learning scheme is suitable only for sequential search method. Therefore, in our opinion, this is the first attempt to estimate/learn parameters used in the compatibility function for every model for optimal encoding of homomorphic ARG matching energy.

4.3. Learning methodology

As discussed above, the selected features should be complete if not redundant, possess sufficient discriminatory information and robust. However, we provide a slightly differing view on the concept of discriminatory property of attributes for parallel graph matching. Although Hansen and Henderson[27] and Turney *et al.*[28] identified the inter-model and also to some extent the intra-model discriminatory power of the features as the most crucial property in recognising patterns and objects, we choose to consider only the intra-model discriminatory power in the computation of the weighting factors. The following ambiguities may be recognised to be in existence in models.

(1) *Inter-model ambiguities*: Inter-model ambiguous attributes are attributes shared by many object models. Such features do not possess high information for discrimination between object models. Some examples of such ambiguities are geometric subparts and a symbolic feature *colour red* when many objects in the model database are red in colour. Some examples of such ambiguities are presented in references (1) and (37). In our opinion, ambiguities caused by subparts or common features cannot be resolved by any weighting factors assignment mechanism. For instance, if only the common subparts are visible, it will be impossible to discriminate between object models by any weighting factor assignment. Hence, we conclude that the presence of subparts should be resolved by some other mechanism as explained in the hypothesis interpretation section by the authors.[1]

(2) *Intra-model ambiguities*: When an object model has repetitive attributes, intra-model ambiguity is said to exist. Intra-model ambiguities have the potential to degrade the performance of a recognition system. Some examples are equal line segment lengths and/or parallel line segments in a line pattern model as in model Fig. 3(d), and many circles with the same radius as in circle pattern models Figs 5(c) and (d). This type of ambiguous attributes should be assigned with relatively lower weight values. For instance, if line segment length is assigned with a large weight value to extract model Fig. 3(d), a scene line segment may be matched to a wrong model line segment, as will be illustrated in the experimental results section. Hence, we conclude that the intra-model ambiguity should be dealt with care so that the graph matching problem can be mapped optimally and potential ambiguities in the mapping can be minimised. In fact, our investigations into the recognition performance of the optimising connectionist models have shown a degradation in performance due to a high degree of intra-model ambiguity.

The above arguments justify the consideration of intra-model discriminatory power in the computation of the weighting factors. Other advantages of considering only intra-model ambiguities are that the computation of weighting factors can be performed independently for each model, and addition of new models to the model database and deletion of models from the model database do not alter the weighting factors of existing models.

The training scenes have a single model under consideration as the whole. The purpose of using the training scenes is to discover the robustness of individual attributes against the sensing noise and feature extraction noise. Our approach deals accordingly with physical meanings of all attributes and does not require any modelling of any processes. The compatibility measures, $C_{xi,yj}$ expressed by equations (11) and (12) are functions of parameters t, k, p and w_i. First, compatibility function parameters t and k are estimated from sample scenes so that variations in units of measurement and noise robustness of individual attribute are taken into account. Then parameters w_i and p are learned in order to obtain the best overall separation between corresponding and noncorresponding vertices pairs. These two learning schemes in effect encode various compatibilities optimally in the energy function for efficient recognition by a parallel ARG matching procedure.

4.4. *Choice of threshold and steepness parameters*

In the training scene, there are a number of matched pairs. However, as noise is introduced in the scenes, they are not identical. Further, in these examples attributes may become larger or smaller. Since synthetic scenes were generated by rotating, translating, adding random noise to models and arbitrarily overlapping individual instances, the effect of noise is independent of the magnitude of the attributes. Thus, it is possible to use fixed threshold values for a particular type of attribute irrespective of its magnitude in the model. In this application, threshold parameters were set to be larger than the largest deviation between corresponding attributes. Having estimated t, a suitable value for k should be assigned. The role of the parameter k is to compensate for variations in the scale or units of measurement. In our experiments, k is chosen such that when scene and model relational attributes are identical the compatibility measure function $\tanh(kt)$ yields 0.95. If this value is set to approximately one instead of 0.95, the compatibility function would become a step-like function and lose its ability to discriminate between different degrees of match and mismatch.

4.4.1. Consequences of the choice of threshold and steepness parameter values. A number of criteria were identified to be crucial for accurate recognition in the preceding sections. One of them is robustness against noise. If a particular relational property is more prone to be corrupted by noise than others, it will be assigned with a large t value. As the parameter k is fixed so as to obtain a constant value of 0.95 when the scene and model attributes are identical, the compatibility function corresponding to noisy attribute has large transition region. Large transition region means that the compatibility function $f^u(\cdot)$ is more likely to take values around 0 than extreme values ± 1 and the function $f^b(\cdot)$ is more likely to take values around 0.5 than extreme values 1 or 0. Hence, the contribution of noisy attributes to the overall compatibility measure is not decisive and marginal. The *vice versa* holds for robust attributes. This concept is illustrated in Fig. 1. In these figures, sharper function has smaller threshold value and consequently smaller transition region.

4.5. *Learning parameters using gradient projection algorithm*

To learn parameters t and k only corresponding relational features are considered. To achieve optimal separation between corresponding and noncorresponding vertex pairs, corresponding as well as noncorresponding vertex pairs should be considered. However, there are a large number of noncorresponding vertex pairs. Hence, in order not to bias the optimal weighting factor computation strongly towards noncorresponding vertex pairs, the contribution of corresponding and noncorresponding vertex pairs should be balanced. By examining equations (11) and (12), weight constraints expressed by equation (10), energy function (7) and Fig. 1, it is clear that for accurate recognition, the compatibility measure $C_{xi,yj}$, corresponding to matching vertex pairs x, y in the scene and i, j in the model should be assigned to $a+1$. The squared error for the corresponding vertex pairs is

$$e_{CP}(W,p) = \sum_{\forall [xi,yj] \in S_{CP}} (C_{xi,yj} - 1)^2, \tag{13}$$

where the set $S_{CP} = \{[(v_x, v_i), (v_y, v_j)] \mid v_x \to v_i, v_y \to v_j, x \neq y, i \neq j\}$ and W and p are as defined in equations (9) and (10).

The compatibility measure $C_{xi,yj}$ corresponding to noncorresponding pairs should be assigned to -1 when a monomorphic mapping or subgraph isomorphic mapping is sought [1, 2]. However, when the homomorphic mapping is sought, the desired value should not be set to -1 because the compatibility function $f^b(\cdot)$ used to measure binary compatibility is constrained between 0 and $+1$ [see Fig. 1(b)]. Therefore, the desired value for noncorresponding vertex pairs d_{NP} should be treated as a variable. The squared error for the noncorresponding vertex pairs is

$$e_{NP}(W(d_{NP}), p(d_{NP})) = \sum_{\forall [xi,yj] \in S_{NP}} (C_{xi,yj} - d_{NP})^2, \tag{14}$$

where the set $S_{NP} = \{[(v_x, v_i), (v_y, v_j)] \mid v_x \not\to v_i, v_y \not\to v_j, x \neq y, i \neq j\}$. Let us defer the balancing process to a later stage and consider the overall cost which can be expressed as follows:

$$e(W(d_{NP}), p(d_{NP})) = \sum_{\forall [xi,yj] \in [S_{CP} \& S_{NP}]} (C_{xi,yj} - d)^2, \tag{15}$$

where d is $+1$ if $l \in S_{CP}$ and d is d_{NP} otherwise. If d_{NP} can be assumed to be a constant, the overall objective function in equation (15) can be solved by the gradient projection algorithm.[36] In this study, d_{NP} is decremented using a step of 0.01 starting from 0 towards -1. At each value for d_{NP}, regarding d_{NP} as a constant, the parameters and the cost $e(W(d_{NP}), p(d_{NP}))$ are computed using the gradient projection algorithm. The value of d_{NP} and the corresponding weighting factors that minimise the overall cost in equation (15) will be regarded as solutions.

Since the individual attribute compatibilities are regarded as confidence measures, they should be shifted by 1.0 and normalised by 2.0 so that all compatibility values and desired values fall between 0 and 1.0. Consequently, the desired values d_l in equation (15) for corresponding and noncorresponding (d_{NP}) vertex pairs are 1.0 and a variable in the range 0–0.5, respectively.

Prior to formulating the solution of this cost function as constrained optimisation problem, it is appropriate to establish relationship between attributes described in Section 2.2, constraints expressed by equation (10) and the cost function in equation (15). The compatibility function defined in equations (1) and (2) are used to compute individual model and sample scene attribute's compatibility. The chosen attributes are described in Section 2.2.1 for line pattern matching and in Section 2.2.2 for circle pattern matching. These individual compatibility values are adjusted and normalised to obtain a fuzzy confidence measure for every pair of attributes and are substituted into equations (11) and (12) to combine the evidence captured in individual compatibilities and to obtain an overall compatibility measure. These compatibility values are used to obtain the error expressed in equation (15) for learning the weights and parameter p. During the matching phase, these compatibilities are used to programme the network connections to solve the recognition problem.

The solution of weight parameters $\mathbf{W}$ and parameter p is a constrained optimisation problem. First, the overall cost in equation (15) and constraints in equation (10) can be expressed as follows:

$$\begin{aligned} &\text{minimise} \quad e(\mathbf{W}, p) \\ &\text{subject to} \quad \mathbf{h}(\mathbf{W}) \leq \mathbf{0}. \end{aligned} \tag{16}$$

It is a standard constrained optimisation problem. The necessary conditions for this problem are[36]

$$\begin{aligned} \nabla e(\mathbf{W}_{opt}, p_{opt}) + \lambda^{\mathrm{T}} \nabla \mathbf{h}(\mathbf{W}_{opt}) &= \mathbf{0}, \\ \mathbf{h}(\mathbf{W}_{opt}) &\leq \mathbf{0}, \\ \lambda^{\mathrm{T}} \mathbf{h}(\mathbf{W}_{opt}) &= \mathbf{0}, \\ \lambda &\geq \mathbf{0}. \end{aligned} \tag{17}$$

These conditions can be expressed in a simpler form in terms of the set of active constraints A^s.[36] The set of active constraints include all equality constraints and some inequality constraints which satisfy equality condition when the objective function and the variables are optimised. The other constraints which satisfy the inequality, are regarded nonactive:

$$\begin{aligned} \nabla e(\mathbf{W}_{opt}, p_{opt}) + \sum_{i \in A^s} \lambda_i \nabla h_i(\mathbf{W}_{opt}) &= \mathbf{0}, \\ h_i(\mathbf{W}_{opt}) &= 0, \; i \in A^s, \\ h_i(\mathbf{W}_{opt}) &> 0, \; i \notin A^s, \\ \lambda_i &> 0, \; i \in A^s, \\ \lambda_i &= 0, \; i \notin A^s. \end{aligned} \tag{18}$$

The constraints on weighting factors are that the summation should be equal to one and all weights should be nonnegative as expressed by equation (10). The set of active constraints include the summation constraint and constraints associated with weights which have reached the boundary value of zero. In other words, the active set constraints include the equality constraint which is always active and inequality constraints which have reached the inequality limit. The other weighting factors which have positive weight values are regarded as inactive. That is, those inequality constraints are naturally satisfied by the present weight values.

Therefore, if the set of active constraints corresponding to the optimal solution were known *a priori*, the original problem could be replaced by a problem having only equality constraints. Generally, the active constraints are not known *a priori*. Alternatively, if it is possible to make a guess for the set of active constraints and solve the problem for the particular active set, the solution will be the optimal solution of the original problem, provided all Lagrange multipliers are nonnegative and all inequality constraints are satisfied. Exploiting this idea, in the active set method, the parameters are initialised at a feasible point and moved on the surface of the active constraints so as to reduce the cost and to satisfy all active constraints.

This algorithm is a straightforward extension of the basic unconstrained gradient descent or steepest descent algorithm. In the gradient projection algorithm, at every iterations, a feasible direction is obtained which satisfies $\nabla e(\mathbf{W}(k), p(k))\ \mathbf{d}(k) < 0$ and $\mathbf{A}(k)\ \mathbf{d}(k) = 0$, where $\mathbf{d}(k)$ and $\mathbf{A}(k)$ are the constraint direction of descent and the matrix composed of rows of active constraints. In other words, the gradient projection algorithm is used to project the negative gradient of the cost function onto the surface formed by the set of active constraints so that at iteration k the cost is reduced and all active constraints remain active, respectively. Let us consider the derivation of constrained direction of descent. The negative gradient can be decomposed into two components: one in the direction of $\mathbf{d}(k)$ and the other perpendicular to

$\mathbf{d}(k)$ as follows:

$$-\nabla e(\mathbf{W}(k), p(k)) = \mathbf{d}(k) + \mathbf{A}^{\mathrm{T}}(k)\,\lambda(k), \qquad (19)$$

where $A(k)$ is the matrix of active constraints at iteration k. Since $\mathbf{A}(\mathbf{k})\mathbf{d}(k) = 0$

$$\lambda(k) = -(\mathbf{A}(k)\,\mathbf{A}^{\mathrm{T}}(k))^{-1}\,\mathbf{A}(k)\nabla e(\mathbf{W}(k), p(k)). \qquad (20)$$

Substituting for $\lambda(k)$ in equation (19) gives

$$\mathbf{d}(k) = -[\mathbf{I} - \mathbf{A}^{\mathrm{T}}(k)(\mathbf{A}(k)\mathbf{A}^{\mathrm{T}}(k))^{-1}\,\mathbf{A}(k)] \times \nabla e(\mathbf{W}(k), p(k)). \qquad (21)$$

The constrained direction of descent $\mathbf{d}(k)$ can be expressed in terms of the projection matrix $\mathbf{P}(k)$, defined as follows.

$$\mathbf{P}(k) = [\mathbf{I} - \mathbf{A}^{\mathrm{T}}(k)\,(\mathbf{A}(k)\,\mathbf{A}^{\mathrm{T}}(k))^{-1}\,\mathbf{A}(k)], \qquad (22)$$

$$\mathbf{d}(k) = -\mathbf{P}(k)\nabla e(\mathbf{W}(k), p(k)). \qquad (23)$$

Now the parameters can be updated as follows provided all $w_i(k+1)$ are feasible and $\mathbf{d}(k) \neq 0$:

$$[\mathbf{W}; p](k+1) = [\mathbf{W}; p](k) + \alpha(k)\,\mathbf{d}(k), \qquad (24)$$

where $\alpha(k)$ is the step-size parameter. As 'training by samples' approach is employed in this application, a sufficiently small fixed value, α is used for the step-size parameter. Further, the step-size parameter used with noncorresponding vertex pairs should be weighted using the fraction $\#(S_{CP})/\#(S_{NP})$ to balance the contribution of corresponding and noncorresponding vertex pairs to the optimisation, where $\#(X)$ and $\#(S_{CP})/\#(S_{NP})$ are the cardinality of the set X and the balancing factor, respectively.

If any one of the weighting factors is not feasible after the up-date, i.e. $w_i(k+1) < 0$, then the inequality constraint corresponding to the weight should be included as equality constraint into the active set and the new constrained direction of descent should be computed to up date the parameters.

If $\mathbf{d}(k) = 0$, the Lagrange multipliers should be computed using equation (20). If all Lagrange multipliers are nonnegative, the necessary conditions for optimal solution in equation (17) are satisfied. If some Lagrange multipliers are negative, then drop the equality constraint corresponding to the most negative Lagrange multiplier, and repeat the procedure until $\mathbf{d}(k)$ becomes 0 again.

Now let us consider the estimation of gradients in the above equations. The derivative of $e(\mathbf{W}, p)$ with respect to a weighting factor w_i is

$$\frac{\partial e(\mathbf{W}, p)}{\partial w_i} = 2 \sum_{\forall\, j \in [S_{CP} \& S_{NP}]} (g_m - d_j)\,\frac{\partial g_{m,j}}{\partial w_i}. \qquad (25)$$

The derivative of the generalised mean operator with respect to w_i is

$$\frac{\partial g_m}{\partial w_i} = \frac{(0.5 f_i^*(\cdot) + 0.5)^p}{p}\, g_m^{1-p}. \qquad (26)$$

In the above equation, the compatibility value is normalised to obtain fuzzy measure as follows $(0.5 f_i^*(\cdot) + 0.5)$. By substituting equation (26) into equation (25), the derivative of $e(\mathbf{W}, p)$ with respect to a weighting factor w_i can be obtained. Likewise, the derivative of e with respect to the parameter p at iteration k is

$$\frac{\partial e(\mathbf{W}, p)}{\partial w_i} = 2 \sum_{\forall\, j \in [S_{CP} \& S_{NP}]} (g_{m,j} - d_j)\,\frac{\partial g_m}{\partial p}. \qquad (27)$$

The derivative of the generalised mean operator with respect to p is

$$\frac{\partial g_m}{\partial p} = \frac{1}{p}\ln(0.5 f_i^*(\cdot) + 0.5)(w_i(0.5 f_i^*(\cdot) + 0.5)^p)\, g_m^{1-p} - \frac{1}{p^2}\ln\left(\sum_{i=1}^{N} w_i(0.5 f_i^*(\cdot) + 0.5)^p\right) g_m. \qquad (28)$$

By substituting equation (28) into equation (27), the derivative of $e(\mathbf{W}, p)$ with respect to the parameter p can be obtained. The active-set-based gradient projection algorithm used to obtain the optimal set of parameter values is summarised in Table 1.

4.5.1. Consequences of learning parameters. It is now possible to qualitatively analyse the effects of parameter learning on the following factors identified to be important in the preceding sections for the efficient recognition of objects and patterns.

(1) *Intra-class ambiguity*: If some attributes are repetitively present in the model, the term $f_*^*(\cdot)$ may be near positive one when noncorresponding but repetitive attributes are considered. As the desired value d_{NP} for noncorresponding vertex pairs should be between 0 and 0.5, the weighting factors associated with such attributes are assigned with low values. Hence, intra-class ambiguous attributes are assigned with low weight values, as evidenced by the experimental results.

Table 1. The active-set-based gradient projection algorithm

Input: $\mathbf{H}$, $\mathbf{W}_{ini}$, S_{CP}, S_{NP} and $\mathbf{g}(\mathbf{W}, p)$
Output: $\mathbf{W}_{opt}(d_{NP})$, $p_{opt}(d_{NP})$ and $e(\mathbf{W}_{opt}(d_{NP}), p_{opt}(d_{NP}))$

Step 1 Obtain the matrix $\mathbf{A}(k)$ consisting of active constraints

Step 2 Compute the projection matrix $\mathbf{P}(k)$ using equation (22) and the constrianed direction of descent $\mathbf{d}(k)$ using equation (23)

Step 3 If $\mathbf{d}(k) \neq \mathbf{0}$ and $\mathbf{W}(k+1)$ is feasible, set $\mathbf{W}(k+1) = \mathbf{W}(k) + \alpha_j \mathbf{d}(k)$, where $\alpha_j = \alpha$ if $j \in S_{CP}$ and $\alpha_j = (\#(S_{CP})/\#(S_{NP}))\ \alpha$ if $j \in S_{NP}$. If $\mathbf{W}_{k+1}$ is not feasible, include the newly active inequality constraint and go to step 2.

Step 4 Else if $\mathbf{d}(k) = \mathbf{0}$, compute the Lagrange multipliers expressed by equation (20). If all Langrange multipliers are nonnegative, the solution corresponds to p_{opt} and $\mathbf{W}_{opt}$. Otherwise, drop the constraint corresponds to the most negative Lagrange multiplier from the active set and go to step 2.

(2) *Robust features*: Attributes that are robust against noise and occlusion generate higher compatibility values because the estimation scheme for threshold parameter t assigns low threshold value. This will result in a sharp compatibility function which takes extreme values more frequently. Such attributes will be further emphasised by parameters learning scheme as well. Likewise, noisy attributes that generate compatibility values which are well distributed over the range of the compatibility function due to high threshold value, are further suppressed by these weighting factors estimation procedures for not being at the extremes.
(3) *Rare and discriminatory features*: Although rare features were not explicitly searched for to be included in the optimisation process, lower emphasis on intra-class ambiguous attributes in turn amplify the relative weights associated with the rare and unique attributes in every model.

5. EXPERIMENT RESULTS

Experiments were performed with the models and scenes defined in Section 2 and shown in Figs 3–5, to demonstrate the ability of the parameter learning schemes to discover inherent characteristics of the models and necessity of the weighting factor computation scheme to generate the desired mapping. The final matching results are also presented. Since the models and scenes are synthetic, the threshold parameter did not vary with the magnitude of attributes and models. Hence, it is sensible to use a fixed set of threshold values for all models.

5.1. Line patterns matching results

The following values were used for thresholds: [14, 0.2, 14, 14, 14]. The steepness parameter was estimated using the following relation: $\tanh(kt) = 0.95$. The learnt parameter values are shown in Table 2 for the line pattern models in Fig. 3. In model Fig. 3(a), many lines join smoothly at one point. Hence, the last binary relation (defined as r6 in Section 2.2.1) is ambiguous and weighted low. In model 3(b), all line segments have one of three segment lengths and they are either parallel or perpendicular to each other. Hence, the unary relation and the angle related binary relation (defined as r3 in Section 2.2.1) are assigned with low weight values. Models 3(c) and (d) have line segments of the same length. Further all lines are either parallel or perpendicular. Therefore, the unary relation and the angle related binary relation have 0 or near zero weighting.

Table 2. Learnt parameter values for fuzzy optimal mapping of line patterns

Model	$\mathbf{W}^{\mathrm{T}}$	p
Fig. 3(a)	[0.0538, 0.2362, 0.2722, 0.2457, 0.1382]	0.4997
Fig. 3(b)	[0.0297, 0.0000, 0.3275, 0.3004, 0.3127]	0.3277
Fig. 3(c)	[0.0000, 0.0038, 0.3263, 0.3064, 0.3634]	0.4270
Fig. 3(d)	[0.0000, 0.0000, 0.3038, 0.3270, 0.3693]	0.4634

The number of iterations needed for convergence did not vary significantly with the variations in the weighting factors, but remained in the range of 40–60 iterations. However, the recognition rate was strongly dependent on the weighting factors. The weighting factors corresponding to model (a) could not be used to generate the correct mapping for all other models. However, the correct mapping can be generated for model (a) using all four parameter sets, as model (a) does not have significant ambiguity. Therefore, it is apparent that if there is intra-model ambiguity, it is essential to employ the optimal set of parameters for every model. However, as the degree and the types of ambiguities may vary between models, it is essential to learn these weighting factors and map the graph matching problem optimally. The results reported below were obtained using the optimal set of weights for every model.

When the scenes were matched against each model, the MFT neural networks generated some spurious matches, in addition to the desired matches. These spurious matches were eliminated by the pose clustering algorithm.[1] The matched pairs are presented for the line patterns, after eliminating the spurious matches. The matched pairs between scene in 3(e) and model in 3(a) are {(1, 2), (2, 7), (3, 11), (4, 12), (5, 13), (6, 15)}. The matched lines in model Fig. 3(a) are overlaid onto the scene Fig. 3(e) and shown in Fig. 4(a) by solid lines. The original scene is shown by dotted lines. The matched pairs between scene 3(e) and model in 3(b) are {(7, 1), (8, 3), (9, 5), (10, 8), (11, 9), (12, 10), (13, 12), (14, 19)} and shown in Fig. 4(b). Matching the scene 3(e) with the model in 3(c) did not detect an occurrence. The matched pairs between scene in 3(e) and model in Fig. 3(d) are {(15, 3), (16, 4), (17, 5), (18, 6), (19, 7), (20, 10), (21, 11)} and {(22, 1), (23, 2), (25, 8), (26, 10), (27, 11), (28, 15)}. These two occurrences are shown in Fig. 4(c) by solid and dashed lines. The line segment 24 was not correctly matched. Line segments 29 and 30 were correctly classified as extraneous line segments.

The matched pairs between scene 3(f) and model 3(a) are {(1, 2), (2, 7), (3, 9), (4, 13), (5, 15)} and shown in Fig. 4(d). Matching the scene Fig. 3(f) with the model Fig. 3(b) did not detect an occurrence. The matching results between scene 3(f) and model 3(c) are {(6, 1), (7, 3), (8, 5), (9, 6), (10, 7), (11, 11), (12, 14)} and {(13, 1), (14, 2), (15, 3), (16, 5), (17, 11), (18, 13)} which are shown in Fig. 4(e) by solid lines and dashed lines. When model 3(d) was matched against scene 3(f) the following occurrence was detected {(19, 2), (20, 5), (21, 6), (22, 7), (23, 9), (24, 12), (25, 14)} which is shown in Fig. 4(f). The line segments 26–29 were identified to be extraneous and not classified with any model.

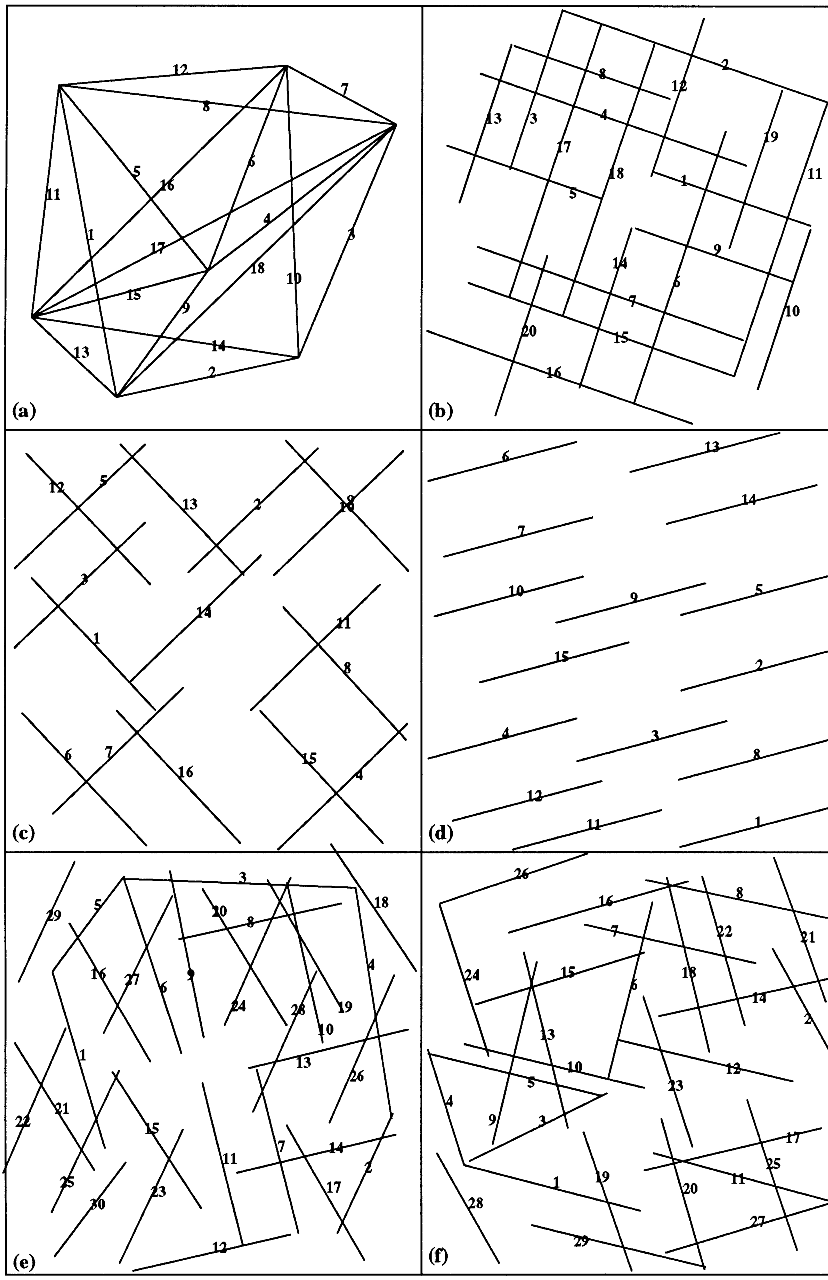


Fig. 3. Line pattern models and scenes: (a)–(d) four line pattern models, (e) and (f) two overlapped scenes.

5.2. Circle patterns matching results

The following value was used for the threshold of binary attribute: [0.5, 14]. The steepness parameter was estimated using the following relation: $\tanh(kt) = 0.95$. The learnt parameter values are shown in Table 3 for the circle pattern models in Fig. 5. The major ambiguity in circle pattern is radius of circles. Models Fig. 5(a), (b), (c) and (d) have 1–2 circles, 2 circles, 3–4 circles and 6 circles with the same radius, respectively. Accordingly, the weight value associated with the radius decreases from 0.2000 to 0.0222.

As far as the recognition accuracy is concerned, a similar observation is made for circle patterns recognition too. The weighting factors corresponding to model (a) could not be used to generate the correct mapping for all other models. However, the correct mapping can be generated for model (a) using all four parameter sets, as model (a) does not have significant ambiguity. Therefore, it is apparent that if there is intra-model ambiguity, it is essential to employ the optimal set of parameters for every model. The results reported below were obtained using the optimal set of weights for every model.

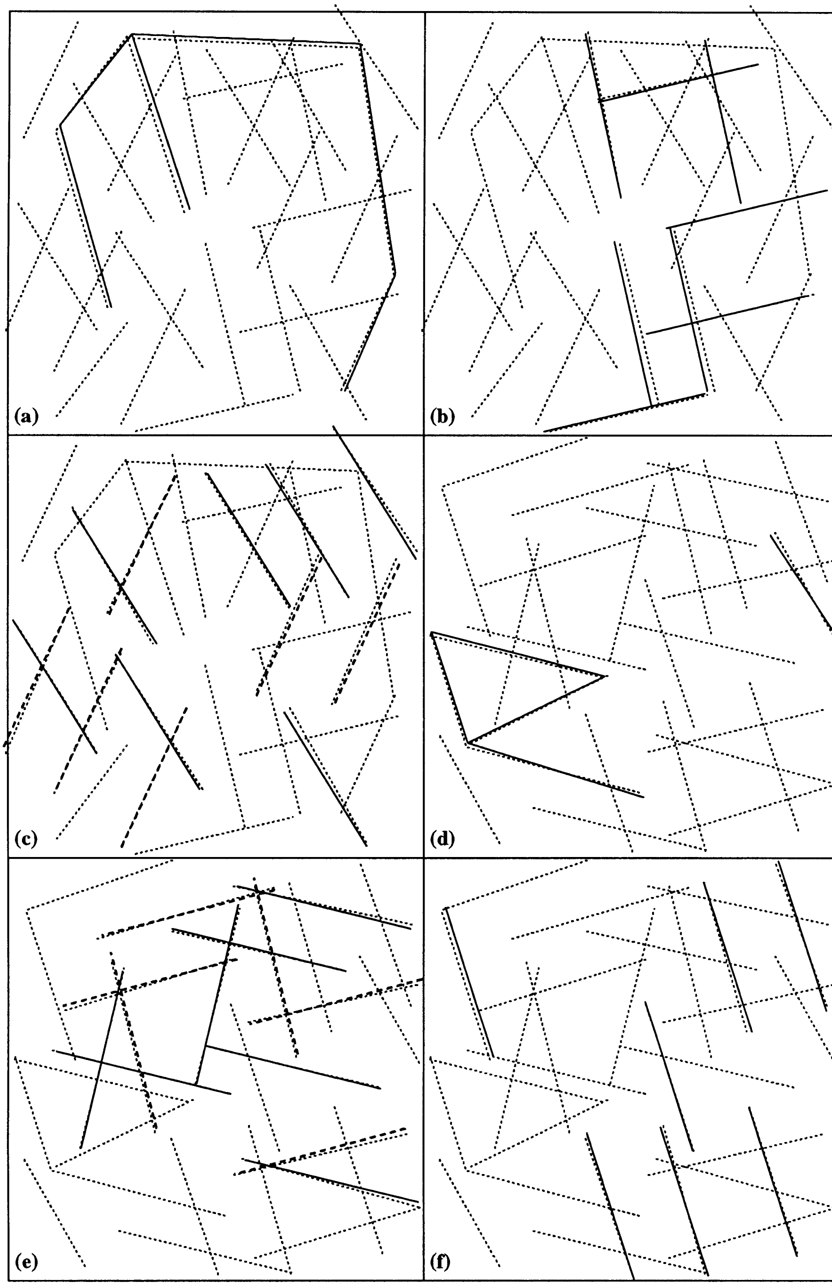


Fig. 4. Line patterns matching results. Matched lines in the models Figs 3(a)–(d) are overlaid onto the scenes in Figs 3(e)–(f).

Table 3. Learnt parameter values for fuzzy optimal mapping of circle patterns

Model	$\mathbf{W}^T$	p
Fig. 5(a)	[0.2000, 0.6000]	0.5000
Fig. 5(b)	[0.0612, 0.8770]	0.6229
Fig. 5(c)	[0.0361, 0.9277]	0.6284
Fig. 5(d)	[0.0222, 0.9556]	0.6051

The matched pairs between scene in 5(e) and model in 5(a) are {(1, 3), (2, 5), (3, 10), (4, 12), (5, 14), (6, 15), (7, 16)}. The matched circles in model Fig. 5(a) are overlaid onto the scene Fig. 5(e) and shown in Fig. 6(a) by filled circles. The original scene is shown by blank circles. The matched pairs between scene 5(e) and model in 5(b) are {(8, 2), (9, 3), (10, 10), (11, 11), (12, 15), (13, 17)} and shown in Fig. 6(b). Matching the scene 5(e) with the model in 5(c) did not detect an occurrence. The matched pairs between scene in 5(e) and model in Fig. 5(d) are {(14, 1), (15, 3), (16, 4), (17, 6), (18, 12), (19, 18)} and {(20, 5), (21, 6), (22, 8), (23, 10), (24, 12)}. These two occurrences are shown in Fig. 6(c) by different shades. Circles 25–28 were correctly classified as extraneous circles.

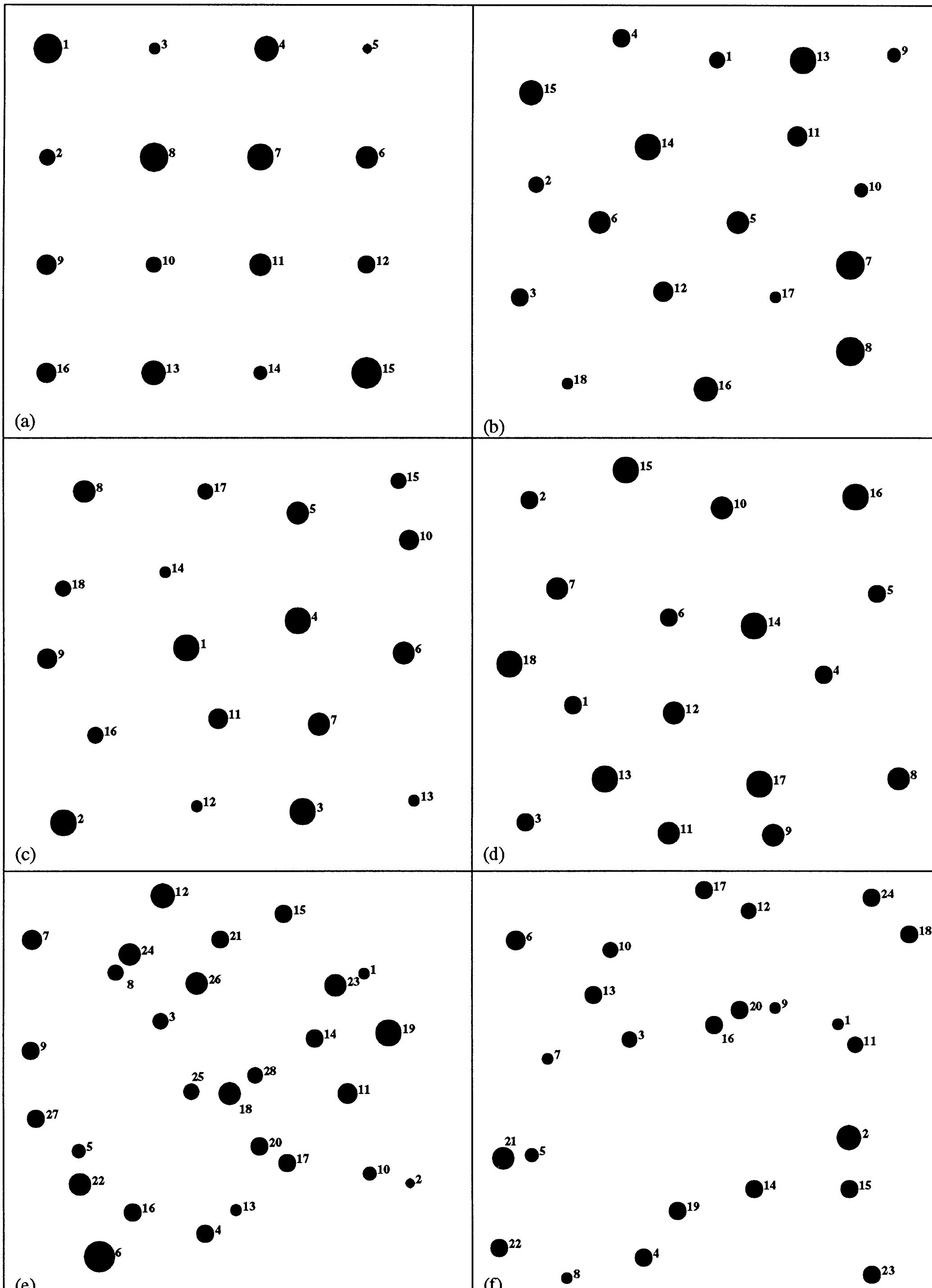


Fig. 5. Circle pattern models and scenes: (a)–(d) four circle pattern models, (e) and (f) two overlapped scenes.

The matched pairs between scene 5(f) and model 1(a) are {(1, 3), (2, 4), (3, 10), (4, 12), (5, 14), (6, 16)} and shown in Fig. 6(d). Matching the scene Fig. 5(f) with the model Fig. 5(b) did not detect an occurrence. The matching results between scene 5(f) and model 5(c) are {(7, 12), (8, 13), (9, 14), (10, 16), (11, 17), (12, 18)} which is shown in Fig. 6(e). When model 5(d) was matched against scene in 5(f) the following occurrence was detected {(13, 1), (14, 4), (15, 5), (16, 6)} and {(17, 1), (18, 2), (19, 5), (20, 6), (21, 8)} which are shown

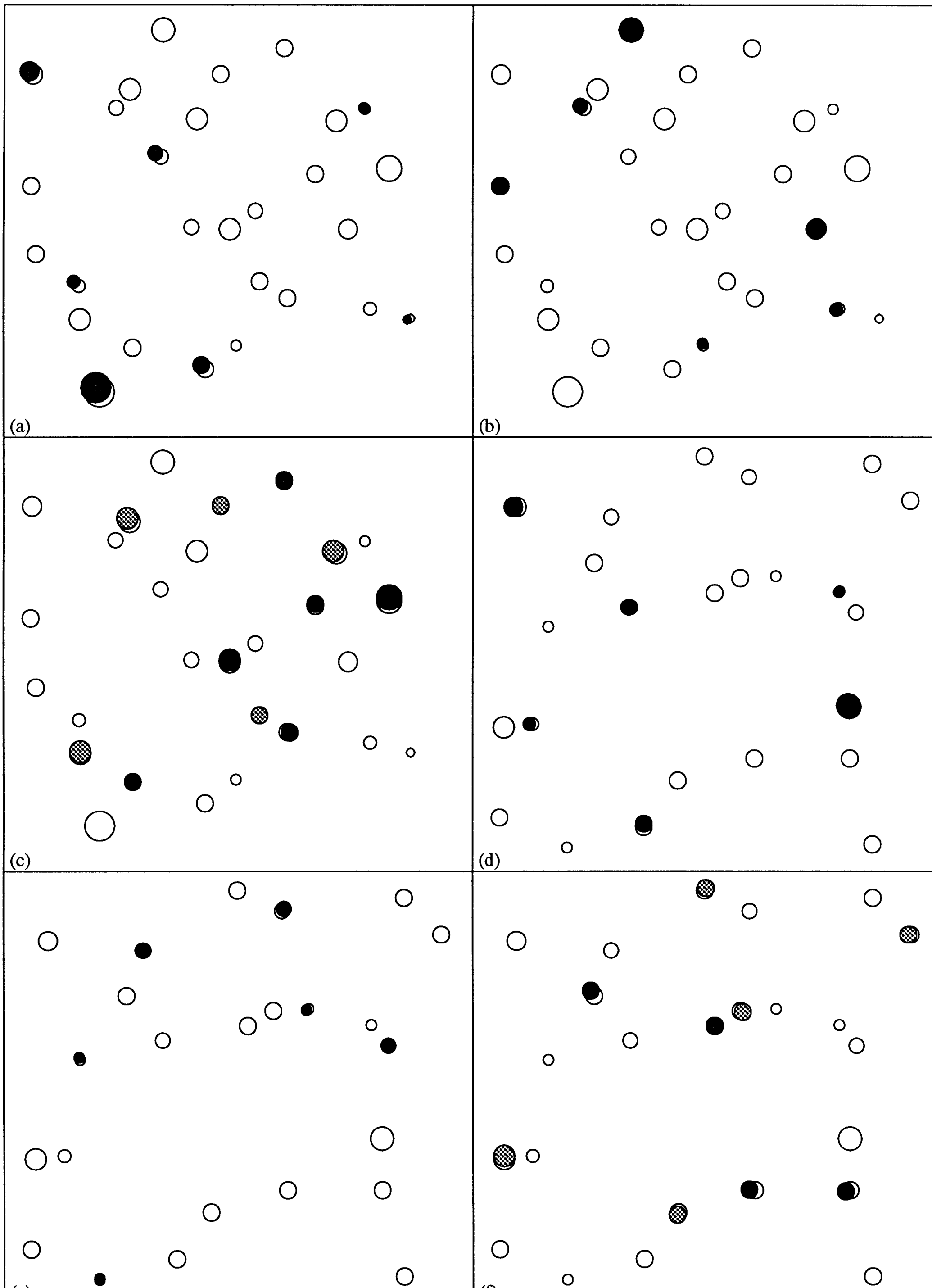


Fig. 6. Circle patterns matching results. Matched circles in the models Figs 5(a)–(d) are overlaid onto the scenes in Figs 3(e) and (f).

in Fig. 6(f). The circles 22–24 were identified to be extraneous and not classified with any model.

5.3. Discussion

Our experiments with the line and circle patterns showed clearly that the optimal set of parameters generated more accurate mapping. Further, extensive experiments performed using various images also showed clearly and consistently that the optimal set of parameters always generated more accurate mapping than a number of other parameter sets used in the simulations. The results also confirmed that the objective function constructed based on the principal of maximum separation between corresponding and noncorresponding vertices is suitable to learn the parameters for optimal mapping.

6. CONCLUSION

The overall homomorphic ARG matching strategy described in this paper can be summarised as follows. Given the models in the learning phase, (a) estimate threshold and steepness parameters as explained in Section 4.4, and (b) learn the weighting factors and the parameter p in the generalised mean compatibility function using the gradient projection algorithm (Section 4.5). These estimation and learning are performed off-line and independently for every model. Hence, addition of new models to database and deletion of models from database do not alter the parameters of the other models. During the application phase, to match a model ARG with the given test scene ARG, use the optimal parameter set corresponding to the model to obtain the optimally encoded energy (Section 3) and employ the Potts MFT neural networks (Section 3) or any other parallel ARG matching strategy to generate the homomorphic ARG mapping.

We have shown the ability of the algorithms to discover inherent characteristics of the model and assign parameters to reflect the characteristic of the model and the application environment so that the ARG matching problem can be optimally encoded into the energy function for better quality parallel solution. The proposed procedures were applied to solve line and circle patterns recognition problems and it was shown that the parameter learning schemes were essential if the models had strong intra-model ambiguity. Different models may have different types and degrees of ambiguities. Hence, a particular set of parameters may not be the best to compute the overall compatibility for all models. Therefore, it is essential to obtain the optimal set of parameters for every model. In addition, these learning procedures replace the commonly used trial-and-error approaches to estimating the compatibility function parameters and offer a theoretically sound alternative.

7. SUMMARY

The attributed relational graph matching (ARG) strategy is a well-known approach to object/pattern recognition [see Reference (1) for a detailed discussion on recognition strategies]. As ARG matching is an NP-complete problem, parallel ARG matching approaches are preferred to the sequential search methods. In all parallel ARG matching approaches, an overall cost/objective function is constructed and optimised. In order to construct such an objective function, an information aggregation formulation and numerical values for a number of parameters in the formulation have to be chosen to fuse the information captured in the chosen attributes. In the past, linearly weighted information aggregation functions have been used. In general, the weighting factors and other parameters in the formulation have been assigned with numerical values by trial-and-error methods. Further, a single set of parameters has been used irrespective of the variations between different models.

In this paper, the compatibility between every pair of model and scene attributes is interpreted as a fuzzy value and subsequently nonlinear information aggregation functions, namely the fuzzy information aggregation operators are used to fuse the information captured in the chosen attributes. To learn the parameters used in the fuzzy information aggregation operators, the "learning from samples" strategy is used. The computation of weight parameters is formulated as an optimisation problem and solved using the gradient projection algorithm based learning procedure. The learning procedure implicitly evaluates ambiguity, robustness and discriminatory power of the relational attributes chosen for graph matching and assigns weighting factors appropriately to them. Further, the proposed approach enables us to compute a distinct set of optimal parameters for every model to reflect the characteristics of the model. The parameter learning procedure enables the homomorphic ARG matching problem to be optimally encoded in the energy function for every model. In order to illustrate the necessity and effectiveness of the learning procedure, the Potts mean field theory (MFT) neural network was employed to perform the parallel ARG matching. Apparently, any other parallel ARG matching approach could be employed to generate such a mapping.

The overall homomorphic ARG matching strategy described in this paper can be summarised as follows. Given the models in the learning phase, (a) estimate threshold and steepness parameters (Section 4.4), and (b) learn the weighting factors and the parameter p using the gradient projection algorithm (Section 4.5). These estimation and learning are performed off-line and separately for every model. Hence, addition of new models to database and deletion of models from database do not alter the parameters of the other models. During the application phase, to match a model ARG with the given test scene ARG, use the optimal parameter set corresponding to the model to obtain the optimally encoded energy (Section 3) and employ the Potts MFT neural networks (Section 3) (or any other parallel ARG matching strategy) to generate the homomorphic ARG mapping. It could be now noted that there are two distinct cost functions. One is defined in Section 4.5 used to learn weights and the parameter p. The other is defined in Section 3 used to perform the actual recognition.

Experimental results are presented to illustrate that the parameter learning scheme is essential when the models have intra-model ambiguity and the optimal set of parameters always generates a better mapping. In addition, an extensive literature on information fusion for computer vision and pattern recognition is also presented to highlight the novelty of our proposed learning strategy.

REFERENCES

1. P. N. Suganthan, E. K. Teoh and D. P. Mital, Pattern recognition by graph matching using the Potts MFT neural networks, *Pattern Recognition* **28**(7), 997–1009 (1995).
2. P. N. Suganthan, E. K. Teoh and D. P. Mital, Pattern recognition by homomorphic graph matching using Hopfield network, *Image Vision Comput.* **13**(1), 45–60 (1995).
3. N. Ayache and O. D. Faugeras, HYPER: a new approach for the recognition and positioning of two-dimensional objects, *IEEE Trans. on Pattern Anal. Machine Intelligence* **8**(1), 44–54 (1986).
4. R. A. Hummel and S. W. Zucker, On the foundation of relaxation labeling processes, *IEEE Trans. Pattern Anal. Machine Intelligence* **5**(3), 267–287 (1983).
5. S. Kirkpatrick, C. D. Gellat and M. P. Vecchi, Optimization by simulated annealing, *Science* **220,** 671–680 (1983).
6. T. J. Fan, G. Medioni and R. Nevatia, Recognising 3-D objects using surface descriptions, *IEEE Trans. Pattern Anal. Machine Intelligence* **11**(11), 1140–1157 (1989).
7. S. Gold and A. Rangarajan, A graduated assignment algorithm for graph matching, *IEEE Trans. Pattern Anal. Machine Intelligence* **4**(18), 309–319 (1996).
8. W. E. L. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints*, The MIT Press, Cambridge, Massachusetts (1990).
9. S. Z. Li, Matching: invariant to translations, rotations and scale changes, *Pattern Recognition* **25**(6), 583–594 (1992).
10. B. Radig, Image sequence analysis using relational structures, *Pattern Recognition* **17**(1), 161–167 (1984).
11. C. Peterson and B. Soderberg, A new method for mapping optimization problems onto neural networks, *Int. J. Neural Systems* **1**(1), 3–22 (1989).
12. J. Lee, Fuzzy-connective-based information fusion networks and their application to computer vision, Ph.D. thesis, University of Missouri, Columbia (1990).
13. S. J. Henkind and M. C. Harrison, An analysis of four uncertainty calculi, *IEEE Trans. Systems Man Cybernet.* **18**(5), 921–932 (1988).
14. E. H. Shortliffe and Buchanan, A model of inexact reasoning in medicine, *Math. Biosci.* **23,** 351–379 (1975).
15. G. F. Luger and Stubblefield, *Artificial Intelligence and Design of Expert Systems.* The Benjamin Cummings Publishing Company Inc., Menlo Park, California (1989).
16. G. Shafer, *A Mathematical Theory of Evidence.* Princeton Press, New Jersey (1976).
17. H. Caillo, A. Hillion and W. Piecyynski, Fuzzy random fields and unsupervised image segmentation, *IEEE Trans. Geosci. Remote Sensing* **31**(4), 801–810 (1993).
18. Y. W. Lim and S. U. Lee, On the color image segmentation algorithm based on the thresholding and the fuzzy *c*-means technique, *Pattern Recognition* **23**(9), 935–952 (1990).
19. J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, New York (1981).
20. D. Dubois and H. Prade, A class of fuzzy measures based on triangular norms, *Int. J. General Systems* **8**(1), 43–61 (1982).
21. G. Klir and T. A. Folger, *Fuzzy Sets Uncertainty and Information.* Prentice-Hall, Englewood Cliffs, New Jersey (1988).
22. D. Dubois and H. Prade, A review of fuzzy set aggregation connectives, *Inform. Sci.* **36**(1 & 2), 85–121 (1985).
23. M. Mizumoto, Pictorial representations of fuzzy connectives: Parts I and II, *Fuzzy Sets and Systems* **31–32,** 217–242, 45–79 (1989).
24. H. Dyckhoff and W. Pedrycz, Generalised means as a model of compensation connectives, *Fuzzy Sets and Systems* **14**(2), 143–154 (1984).
25. H. J. Zimmerman and P. Zyso, Decisions and evaluations by hierarchical aggregation of information, *Fuzzy Sets and Systems* **10**(3), 243–260 (1983).
26. S. Z. Li, MRF parameter estimation for object recognition, *Int. J. Comput. Vision*, to appear.
27. C. Hansen and T. C. Henderson, CAGD-based computer vision, *IEEE Trans. Pattern Anal. Machine Intelligence* **11**(11), 1181–1193 (1989).
28. J. L. Turney, T. N. Mudge and R. A. Volz, Recognizing partially occluded parts, *IEEE Trans. Pattern Anal. Machine Intelligence* **7**(4), 410–421 (1985).
29. R. Krishnapuram and J. Lee, Fuzzy-set-based hierarchical networks for information fusion in computer vision, *Neural Networks* **5**(5), 335–350 (1992).
30. P. A. Devijer and J. Kittler, *Pattern Recognition: A Statistical Approach.* Prentice-Hall Int., London (1982).
31. M. S. Lew, T. S. Huang and K. Wong, Learning and feature selection in stereo, *IEEE Trans. Pattern Anal. Machine Intelligence* **16**(9), 869–881 (1994).
32. M. Pelillo and M. Refice, Learning compatibility coefficients for relaxation labeling processes, *IEEE Trans. Pattern Anal. Machine Intelligence* **16**(9), 933–945 (1994).
33. Y. Nakamura, *Advanced Robotics: Redundancy and Optimization.* Addison-Wesley, Reading, Massachusetts (1991).
34. P. J. Besl and R. C. Jain, Invariant surface characteristics for 3D object recognition in range images, *CVGIP* **33,** 33–80 (1986).
35. K. P. Ngoi, Colour recognition under illumination fluctuation, Master's thesis, Electrical and Electronic Engineering, Nanyang Technological University, Singapore (1993).
36. D. G. Luenberger, *Linear and Nonlinear Programming.* Addison-Wesley, Reading, Massachusetts (1984).
37. P. N. Suganthan, E. K. Teoh and D. P. Mital, A self-organising Hopfield network for attributed relational graph matching, *Image Vision Comput.* **13**(1), 61–73 (1995).

About the Author—P. N. SUGANTHAN received the B.A. degree, Postgraduate Certificate and M.A. degree in Electrical and Information Engineering from the University of Cambridge, U.K. in 1990, 1992 and 1994, respectively. In 1996 he completed his Ph.D. dissertation in the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He was a predoctoral Research Assistant in the Department of Electrical Engineering, University of Sydney. At present, he is a lecturer in the department of Computer Science and Electrical Engineering, University of Queensland, Australia. His research interests include Computer Vision, Neural Networks, Pattern Recognition and Fuzzy Systems. He is a member of the IEEE and IEE.

About the Author—H. YAN received his B.E. degree from Nanking Institute of Posts and Telecommunications in 1982, MSE degree from the University of Michigan in 1984, and Ph.D. degree from Yale University in 1989, all in electrical engineering. From 1987 to 1989, he was a research scientist at General Network Corporation, New Haven, CT, U.S.A. where he worked on developing a CAD system for optimising telecommunication systems. Since 1989 he has been with the University of Sydney where he is currently a Professor in Electrical Engineering. His research interests include medical imaging, signal and image processing, computer vision, neural networks and pattern recognition. He is an author or co-author of more than 100 technical papers in these areas. Dr Yan is a senior member of IEEE, and a member of SPIE, International Neural Networks Society (INNS), Pattern Recognition Society (PRS), and Society of Magnetic Resonance (SMR).

About the Author—E. K. TEOH received his B.E. and M.E. degrees in Electrical Enginnering from the University of Auckland, New Zealand in 1980 and 1982, respectively, and the Ph.D. degree in Electrical & Computer Enginnering from the University of Newcastle, New South Wales in 1986. Currently, he is the Vice-Dean (Research) of the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He is a co-inventor of an Australian patent on an Adaptive Thickness Controller for a Rolling Mill. His current research interests are in the field of Computer Vision and Pattern Recognition, Intelligent Systems, Robotics and Industrial Automation and Microprocessor-based Control System Design. He has published more than 120 journal and conference papers in these areas.

About the Author—D. P. MITAL received his B.Tech degree in Electrical Engineering from IIT Kanpur, India in 1968. Thereafter, he received his M.Sc. and Ph.D. degrees from SUNY, Stony Brook in 1970 and 1974, respectively. From 1974 to 1976 he worked in NCR Corporation, Dayton, OH, U.S.A., as an advanced Development Engineer. At present he is working as Associate Professor in the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His current areas of research include Advance Control Systems, Microprocessor Applications, Computer Control, Machine Vision, Robotics, Artificial Intelligence and Expert Systems. He has published over 100 technical papers in related areas.