International Journal of Systems Science, 1999, volume 30, number 8, pages 823–848

Connectionist object recognition: methods and methodologies

Jayanta Basak† and Sankar K. Pal†

An overview of different connectionist approaches for object recognition is presented. These approaches are first of all classified on the basis of functional characteristics and structural characteristics of the networks. At the next level, these are partitioned on the basis of their underlying assumptions and the types of recognition problem that they handle. Their principles and key features are highlighted. The issue of mixed category perception (in the context of multiple object recognition) is addressed. A comparison of the characteristics of various application-specific systems is provided in a tabular form. Some of the basic issues to be considered while building a connectionist system for object recognition are finally listed.

1. Introduction

Recognition of objects in an image, according to Suetens *et al.* (1992) refers to the task of finding and labelling parts of a two-dimensional (2D) image of a scene that correspond to the real objects in that scene. Object recognition is necessary in a variety of domains such as robot navigation, aerial imagery analysis and industrial inspection. Normally, different stategies for object recognition (Besl and Jain 1985, Chin and Dyer 1986, Wallace 1988, Zhao 1991, Suetens *et al.* 1992) involve establishing some general description of each object, and then labelling different parts of the scene according to the knowledge about the objects.

Objects can have 2D or three-dimensional (3D) descriptions. 2D descriptions are generated from viewer-centred representation where each view is represented using shape features derived from grey-level or binary images. On the other hand, 3D descriptions require viewpoint-independent volumetric representations that permit computation at an arbitrary viewpoint. Generation of 3D descriptions from the captured 2D images is a computationally difficult problem (Trivedi and Rosenfeld 1989) (one approach is to generate a $2\frac{1}{2}$D sketch from the 2D image (Marr 1982)), whereas the 2D descriptions of the objects can be constructed more easily. Construction of 2D descriptions is straightforward for flat objects such as a hammer or a spanner, and such descriptions are often used in inspection problems of flat industrial objects. Moreover, in several other tasks such as character recognition, analysis of remotely sensed imagery, etc., 3D information is neither necessary nor available, and information processing is to be performed only on the basis of 2D descriptions.

The present article is concerned only with 2D descriptions. 2D object recognition involves mainly two stages: firstly, extraction of features from the captured image and the associated preprocessing tasks and, secondly, interpretation of the extracted feature set. These are shown in figure 1.

The different steps involved in these stages are explained below.

(*A*) *Feature extraction and related pre-processing.* In the image acquisition process, the light coming from the scene is procjected on a plane (image plane), and the image plane content is digitized into a 2D array. Each location in the array specifies a position in the image plane, and the location contains an integer value specifying the intensity of the image at that positon, that is the amount of light received from the scene after projection (at that location). In the case of a colour image, the colour information is also stored in each location of the array. To recognize the objects present in the image of the scene, the image is pre-processed in several stages, and consequently some characterizing features are derived from the image. These features are used for the further interpretation. Different stages involved in the feature extraction and allied pre-processing task are (*a*) enhancement and noise removal, (*b*)

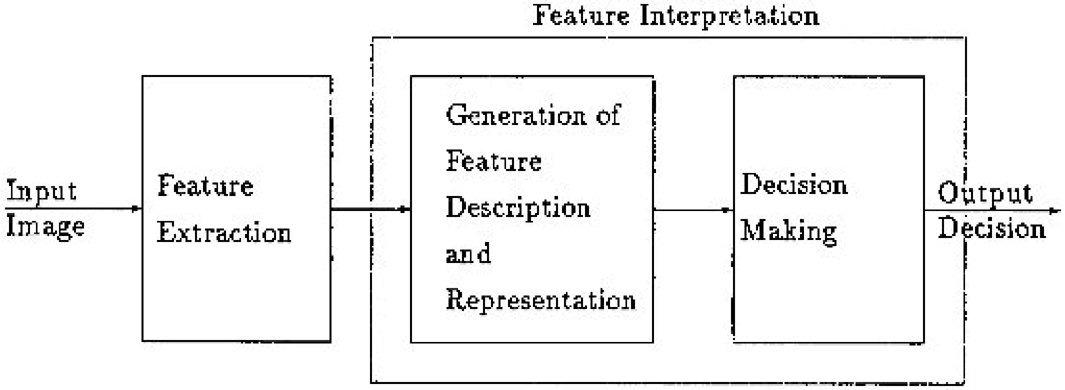

Figure 1. Block diagram showing the three stages involved in object recognition.

segmentation and edge or line detection and linking, (*c*) region- and edge-based feature extraction and (*d*) secondary-feature extraction. Although only four broadly classified tasks are mentioned, each task involves a number of subtasks which are normally performed in the feature extraction and related preprocessing stage.

(*B*) *Interpretation of features.* After extraction, the set of features is used to identify the objects present in the scene. To perform this task some descriptions of each object are stored in the knowledge base. The derived set of features are then matched against the selected attributes stored in the knowledge base utilizing various techniques. In the matching process, all features may not have equal importance (i.e. some objects may have some distinct features), and many times features are ordered according to some pre-assigned priority (feature ranking).

Note that the feature extraction and interpretation (matching) processes are dependent on each other. In other words, extraction of the features is dependent on the type of object to be found in the scene and consequently the interpretation process to be followed. Similarly, the matching strategy depends on the set of extracted features. For example, in the case of region-based features producing some global characteristics of the objects, statistical or distance-based decision rules perform better. This is because generally an explicit numerical feature vector characterizing the objects can be formed in this case. On the other hand, for structural and relational features characterizing the local properties, computationally expensive algorithms based on relaxation-labelling techniques, association and relational graph-matching techniques, generalized Hough transform (HT) techniques, heuristic search techniques, etc. (Wallace 1988), are necessary. Algorithms using the local and relational features have several advantages over those using only global features in many cases, such as interpreting more than one object simultaneously and dealing with occlusions.

In the task of interpreting the feature set, sometimes twofold processing is performed. One is hypothesis generation which is essentially a bottom-up process to generate the hypotheses about the presence of objects from the extracted feature set. The other is the hypothesis verification, which is a top-down process where the stored attributes are matched against those of the image-derived features. However, depending on the type of the description, the algorithms can be widely different.

In the present article, we shall mainly concentrate on the task of feature set interpretation. Various classical algorithms based on the theory of statistical–syntactic pattern recognition, artificial intelligence (AI)-based techniques (e.g. heuristic search, relational homomorphism, association graph matching, boundary correlation and dynamic programming) or massively parallel computational algorithms (e.g. generalized Hough transform (GHT) and relaxation labelling) have been used to deal with the various tasks of object recognition (interpretation of features). Connectionist modelling (based on artificial neural networks (ANNs)) which provides an efficient computational framework has also been extensively employed in these tasks. Recently, this paradigm has drawn attention of researchers from various disciplines.

There exist several review articles (Besl and Jain 1985, Chin and Dyer 1986, Wallace 1988, Chaudhury 1989, Zhao 1991, Suetens *et al.* 1992) concerning the classical techniques for object recognition, but connectionist approaches have not been discussed. During the past 15 years, a number of attempts has been made towards neural modelling of various aspects of object recognition. This resulted in various promising methods and methodologies for dealing with these tasks efficiently.

Therefore, it seems that categorical classification of these different connectionist approaches is necessary, at present, for better understanding of the state of the art and furtherance of research in this discipline. In this article, we present different methods and methodologies for object recognition with special emphasis on connectionist approaches.

The rest of this article is organized as follows. A brief discussion of different classical algorithms is presented in section 2. This will help the readers to understand the connectionist approaches. The relevance of neural networks is also briefly discussed in this section. Section 3 presents the different stages of connectionist approaches for object recognition together with their hierarchical classification based on their approach (characteristics of the networks), types of recognition problem being handled and their underlying assumptions. Section 4 reviews the different methods under this classification. A comparison of characteristics of different connectionist systems is made in section 5. Section 6 concludes this review. Some models for mixed category perception which are already being used or can possibly be used for object recognition (particularly in images with occlusion or overlapping) are presented in the appendix for the convenience of readers.

2. Classical approaches

The literature on various classical approaches for feature set interpretation is rich. There are many survey articles (Besl and Jain 1985, Chin and Dyer 1986, Wallace 1988, Chaudhury 1989, Zhao 1991, Suetens *et al.* 1992) on this. Wallace (1988) has classified different existing techniques according to their approaches. For the sake of the readers, we present a brief overview of different classical algorithms for object recognition along similar lines to those described by Wallace.

As mentioned before, the selection of algorithms for the interpretation of features depends on the types of feature. If the features reflect some global characteristics of the objects, statistical or distance-based decision rules perform better. On the other hand, for structural and relational features characterizing the local properties, computationally expensive AI-based techniques (Wallace 1988) are necessary.

The global feature-based methods essentially derive some global properties such as grey level, colour, area, perimeter, compactness, number of holes and Fourier descriptors from the image, and based on these properties the feature vector is classified using statistical or distance based classification rules (Duda and Hart 1978). Since these kinds of feature reflect the global characteristics of objects, it is difficult to deal with the problems of occlusion with this approach. The local (structural–relational) feature-based techniques include syntactic classification, graph matching, GHT-based methods, relaxation labelling, heuristic search and boundary correlation.

The syntactic approach to pattern recognition involves the representation of a pattern by a string of concatenated subpatterns called primitives. These primitives are considered to be the terminal alphabets of a formal grammar whose language is the set of patterns belonging to the same class. Recognition, therefore, involves a parsing of the string. Fu (1982) has presented a nice introduction to a variety of techniques based on this approach. Various applications of this approach include character recognition, chromosome analysis, identification of skeletal maturity from X-ray images, machine part recognition, shape analysis and recognition. The concept of fuzzy sets has also been incorporated in the syntactic approach to increase the generating power of a grammar (Pal and Dutta Majumder 1986). Currently, this approach is not being well studied by researchers compared with the other approaches to be discussed.

Let us now discuss the other local feature based techniques.

2.1. *Association-graph- and relational-graph-based techniques*

In the association-graph-based technique, a graph is formed by representing the acceptable matches between the scene and desired features as the vertices, and connecting the compatible associations by edges. Compatibility can be determined on various constraints. After formation of the association graphs, cliques are found in order to obtain the most compatible matches between the descriptions in the scene and the actual objects. Various methods under this category are available (Bolles and Cain 1982, Kashyap and Koch 1985, Han and Jang 1990, Wen and Lozzi 1992).

In relational-graph-matching techniques (Shapiro and Haralick 1982, 1985, Eshera and Fu 1984, Grimson and Losano-Perez 1985 Wallace 1985, 1987), the structural description of the objects consists of a set of primitives corresponding to various parts of the objects and a set of *n*-ary relations defined over the set of primitives. Matching rules between the objects are defined in terms of relational homomorphisms, monomorphisms and isomorphisms. To tolerate noisy and erroneous environments, a concept of ϵ homomorphism has been formulated. This is a mapping such that sum of the weights of the relations between the primitives in the candidate which are not satisfied in the corresponding subset of the prototype primitives is less than the threshold ϵ. Hence, matching is a problem of finding a relational homomorphism between the shapes. This can be solved by general constraint satisfaction tree search.

The searching operation can be speeded up by using look-ahead, forward checking and/or a relaxation operator.

2.2. *Generalized-Hough-transform-based techniques*

The HT can be used to extract simple structures such as lines, curves of known form, and circles (Illingworth and Kittler 1988). The HT concept was extended to the GHT (Ballard 1981) to match arbitrary contours. In the GHT, each edge point in the image is aligned with the edge points of the object and accordingly, the position (x, y) and orientation θ of the object are calculated. Thus for each constituent edge point in each object, (x, y, θ) values are computed which indicate the plausible locations of objects determined locally by the edge points. A four-dimensional accumulator array $A[N][X][Y][\Theta]$ is maintained where N is the number of objects, and X, Y and Θ are the quantized values of x, y and θ respectively. Corresponding to each (x, y, θ) value computed for each object, the accumulator value is incremented. After considering all edge points in the image, the peaks in accumulator space are found, which essentially represent the objects' identities and locations. Different variations of GHT have been devised to recognize the objects from a scene. Instead of considering all edge points, some characteristic features in the objects can also be considered. Some algorithms also try to reduce the space requirement to maintain the accumulator array at the cost of inherent parallelism embedded into GHT. Techniques employing GHT for 2D object recognition are available (Stockmann and Agarwala 1977, Segen 1983, Arbuschi *et al*. 1984, Turney *et al*. 1985, Bhanu and Ming 1986, Ullmann 1993).

2.3. *Heuristic search techniques*

In these techniques (Rummel and Beutel 1984, Ayache and Faugeras 1986, Knoll and Jain 1986, Chaudhury *et al*. 1990), some estimated positions and orientations are found from the matched set of features. Then a tree search technique based on heuristic reasoning is employed to find the match for other features. The nodes in the tree normally represent matches between object primitives and scene primitives. Each node is associated with some weight, indicating a measure of similarity of the scene primitives with that of the object. Different variations of A^* search algorithms are usually employed in the heuristic search process.

2.4. *Relaxation-labelling-based techniques*

Various algorithms for object recognition have been developed on the basis of the concept of relaxation labelling (Bhanu and Faugeras 1984, Medioni and Nevatia 1984, Henderson and Samal 1986, Grimson 1989, Umeyama 1993). In these techniques, the features derived from the scene are initially matched (and then associated) with the object features according to some similarity measures. Then the labellings (associations) are updated on the basis of some compatibility function which takes care of the labellings of other scene features. The updating process continues until a suboptimal quality of match (which may be quantified on the basis of a compatibility function) is reached.

2.5. *Other techniques*

Apart from using the techniques mentioned before, several other algorithms have been developed for 2D object recognition. Price (1984) used a boundary correlation approach where the orientation differences between compatible segments in the objects and the scene are stored in a disparity array. The orientation difference in the longest sequence of match is used to compute the transformation from object reference frame to scene. This transformation is then used to find the final set of matched segments.

In the method developed by Gorman *et al*. (1988), an inter-segment (between objects and scene descriptions) distance table is formed (distance is measured in terms of Fourier coefficients). A minimum-distance path in the table that has resulted from a diagonal transition is considered to be the desired match between the object and scene descriptions. Ansari and Delp (1990) used a similar technique where a new local shape measure, namely sphericity, was proposed as the similarity measure.

2.6. *General comments on the techniques*

The relative merits (and demerits) of the abovementioned techniques have been given by Wallace (1988). Although the algorithms have been classified according to the techniques that they have adopted, some of these algorithms use more than one technique to enhance the recognition capability of the system. For example, in the scheme developed by Grimson (1989), the initial choice of match between the object and scene features was guided by the GHT technique. The match quality was then further improved by using constraint satisfaction tree search.

2.7. *Why neural networks?*

From the discussion, it becomes apparent that the task of object recognition is a kind of optimization (constraint satisfaction) problem. A suitable solution can be approached by heuristic search, relaxation labelling, finding cliques from association graphs or finding the homomorphism between relational graphs.

Note that the GHT may be used to obtain some initial guess about the presence of objects but, in order to get the desired matching performance, the most prominent peak in the Hough space needs to be detected. This, in turn, is a non-trivial task. Again, in the implementation of the GHT, the space requirements are very high. However, one definite advantage of the GHT is that it can be directly implemented on a parallel machine. Also, the saliencies (degrees of importance) of different feature–object pairs can be effectively used in the GHT.

Whatever methodologies are used for feature extraction or interpretation of the feature set, several requirements must be satisfied for their applicability in real-life tasks. First, the methodologies should be robust and fast. Preferably, the algorithms should be implementable on parallel hardware. Second, in the task of interpreting the feature set, sometimes it is necessary to associate degree of importance (or weights) with the features. If the association of importance or weights with the features can be performed automatically depending on the environment, then the methodology may prove to be more versatile.

The performance of the classical algorithms, in general, is not comparable with the real-time performance of biological systems which are capable of adapting to the environment and seem to be more robust in its behaviour. Good comparisons between animate visual systems and machine recognition systems have been given by Bullock (1978) and Hochberg (l987). The principles of animate vision have also been elaborately discussed by Ballard and Brown (1992). Although the objective of machine vision is not necessary to emulate animate vision, its performance may plausibly be improved if some findings in the fields of neurobiology and psychology regarding visual cognition can be taken into account in the development of artificial systems. As discussed by Skryzpek (1989), the findings in neurobiology may provide a bottom-up guideline, while the findings in psychology may provide a top-down guideline for such improvement. Since ANNs attempt to provide a related computational framework to biological nervous systems, the neurobiological and psychological findings may possibly be incorporated into artificial recognition systems in a better way using ANN models. Moreover, ANNs sometimes provide an alternative framework for dealing with the optimization problems. ANNs are often referred to as *connectionist models*† or *parallel distributed models* or *computational neural networks* or simply *neural networks*.

ANN models are massively parallel interconnected networks of simple processing elements (neurons), intended to interact with the real world in the same way as biological nervous systems do. This does not necessarily mean that ANNs are able to emulate the behaviour of biological systems; rather they sometimes resemble biological systems in a very naive manner. However, neural networks (or connectionist models) having several basic characteristics such as robustness, scope for massive parallelism and capability of learning from examples (adaptivity and generalization capability), provide a tempting paradigm for dealing with real-life recognition tasks.

Neural networks have been applied to different real-life tasks such as optimization (Hopfield and Tank 1985, Lillo *et al*. 1993), image segmentation (Blanz and Gish 1991, Ghosh *et al*. 1991, 1993), enhancement and restoration (Lu and Szeto 1993), (Bedini and Tonazzini 1990), edge–line linking (Basak *et al.* 1994), scene labelling (Jamison and Schalkoff 1988), speech perception (Waibel *et al.* 1989), natural language processing (Miller and Gorin 1993), motion analysis (Marshall 1990 a, b, c, Tsao and Chen 1994), expert systems (Gallant 1988) and rule generation (Mitra and Pal 1992). In the following sections, we shall present the different connectionist approaches developed for object recognition.

3. Connectionist approaches: different stages and hierarchical classification

Keeping an analogy with conventional object recognition systems (figure 1), the different stages involved in connectionist object recognition are shown in figure 2. The task of representation of features in the classical approaches corresponds to the feature mapping on to neural networks in the connectionist approaches. Similarly, the decision-making part in the classical approaches (by either statistical or decision theoretic or AI-based techniques) corresponds to the neural-network-based classification or clustering or model-matching task.

As mentioned before, ANNs provide a paradigm for incorporating the neurobiological and psychological findings for the efficient design of an object recognition system. However, many of these connectionist approaches also borrow some concepts from classical techniques for suitable representation and decision making. For example, the GHT and relaxation labelling have been used for decision making, and association (or relational) graphs have been used for feature representation in the connectionist framework. The overlapping nature of these approaches is demonstrated in figure 3 where the existing classical methods are represented by the upper circle and those utilized by the connectionist

† Sometimes connectionist models refer to more general kinds of network which have the capability of distributed knowledge representation.

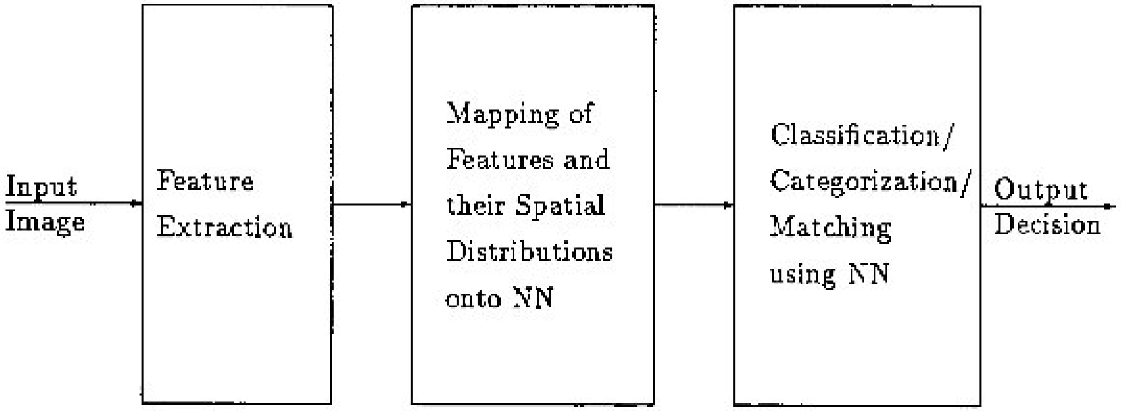
Input Image
Feature Extraction
Mapping of Features and their Spatial Distributions onto NN
Classification/ Categorization/ Matching using NN
Output Decision

Figure 2. Block diagram showing the three stages in connectionist object recognition: NN, neural network.

approach are represented by the lower circle. The overlapping zone of these two circles represents those classical techniques which are also used in connectionist methods. Note that those techniques have different forms of implementation, depending on the types of approach (classical or connectionist) that they are used in.

As stated in section 1, our discussion on object recognition will be restricted mainly to the tasks of representation and decision making (i.e. second and third stages of figure 2). Considering these tasks, the existing connectionist methods can be classified from two different points of view (figure 4) or methodologies. In methodology (category) I, a basic network model (e.g. multilayer perception (MLP), the Hopfield model or the Kohonen model) is considered and the recognition problem is mapped accordingly on to the network. Therefore, in this methodology, the challenge is how a given basic network can be utilized to solve a recognition problem.

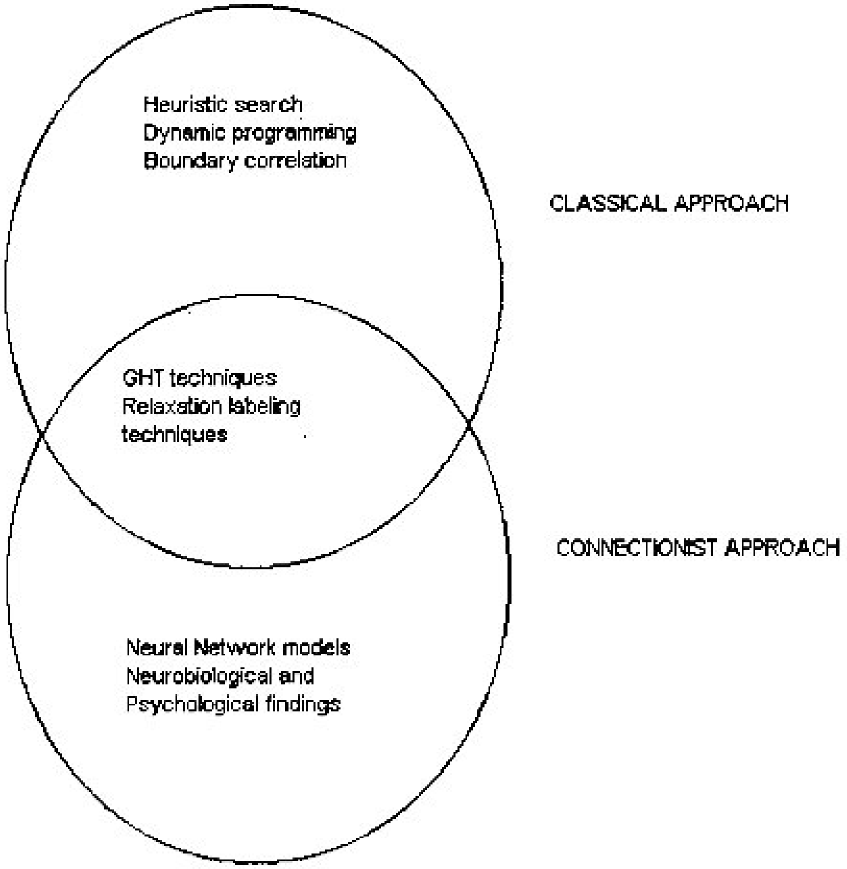

Figure 3. A schematic diagram showing different techniques involved in classical and connectionist object recognition.

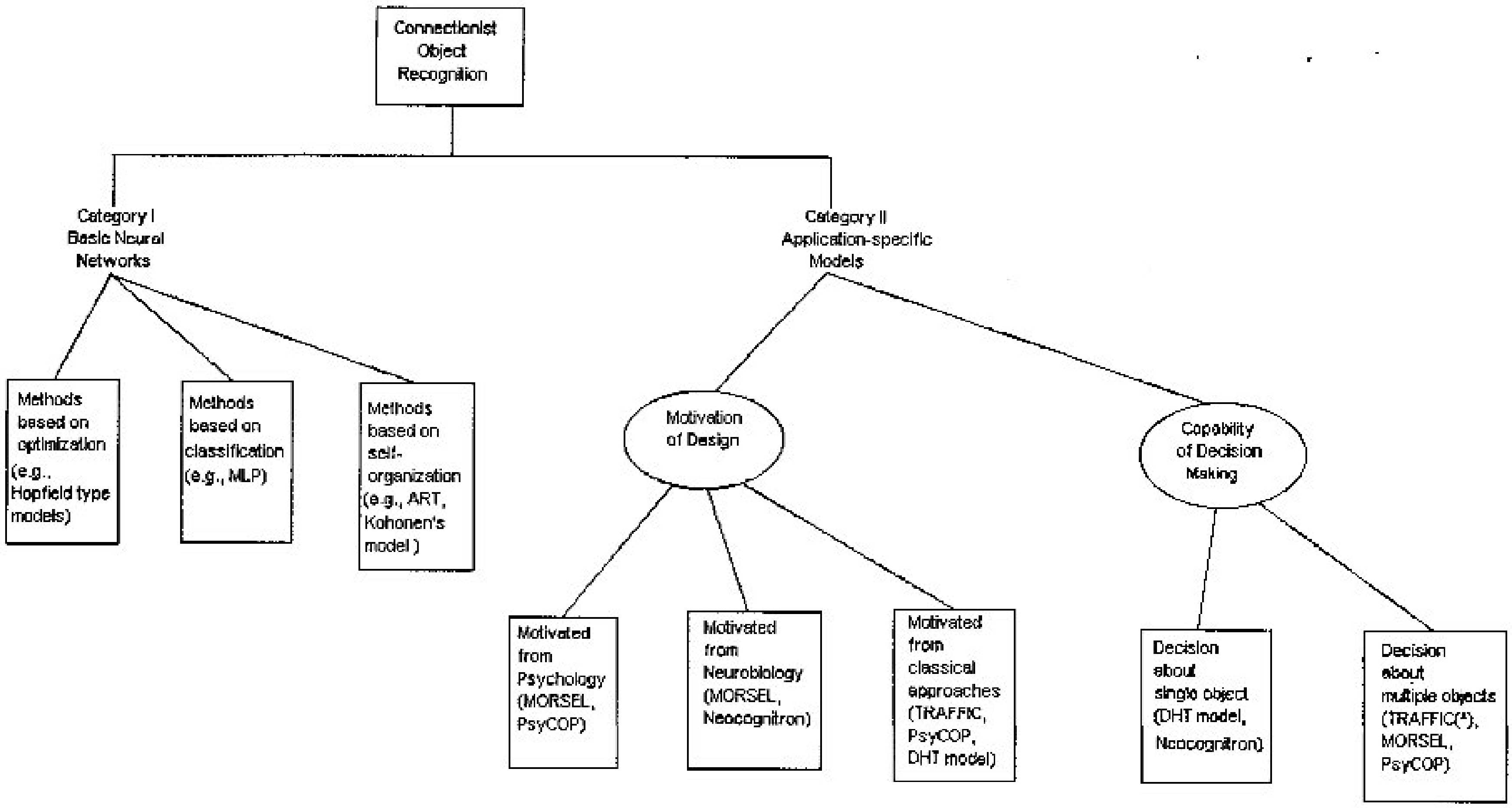
Connectionist Object Recognition
Category I Basic Neural Networks
Category II Application-specific Models
Methods based on optimization (e.g., Hopfield type models)
Methods based on classification (e.g., MLP)
Methods based on self-organization (e.g., ART, Kohonen's model)
Motivation of Design
Capability of Decision Making
Motivated from Psychology (MORSEL, PsyCOP)
Motivated from Neurobiology (MORSEL, Neocognitron)
Motivated from classical approaches (TRAFFIC, PsyCOP, DHT model)
Decision about single object (DHT model, Neocognitron)
Decision about multiple objects (TRAFFIC(*), MORSEL, PsyCOP)

Figure 4. A hierarchical classification of the connectionist approaches for object recognition.

On the other hand, in methodology (category) II, indigenous network architectures are designed for a given recognition problem. Here the challenge is how to design a special-purpose network which will be dedicated to solving a particular recognition problem.

In other words, it is the novelty in the mapping of the problem with which we are mainly concerned in category I, whereas it is the design of an application-specific network in the case of category II. It may be noted here that a particular method (algorithm) may have novelty both in the mapping of the problem and in the design of the architecture, that is the method can fall under both category I and category II. Therefore, categories I and II, in that sense, are not disjoint. For example, an MLP-based approach may also be classified under category II depending on the novelty in the design of the layered network. However, in figure 4, we identify the methods based on basic network models as category I and those with application-specific models as category II.

The principle of the methods in category I has a close resemblance to that of classical pattern recognition, where the task is mainly viewed as classification after feature selection or extraction, performed sequentially. On the other hand, in category II, the tasks of feature extraction and their interpretation are intermingled and performed simultaneously. Therefore, unlike category I or classical pattern recognition, the line of separation between these two operations cannot be drawn.

In category I, the stress is on the suitable functional characterization (e.g. classification or optimization) of the recognition problem and correspondingly its mapping on to a basic network model with its existing framework of learning. In category II, the issue of architectural design, state dynamics and formulation of learning rules becomes more important. Often some psychological and neurological findings guide the task of architectural design.

As an example, let us consider an MLP-based technique in category I (Tsang *et al.* 1992) and a similar layered-network-based system in category II, for example MORSEL (Mozer 1991). Although they may look alike, their approaches to recognition, as explained below, are entirely different. In the MLP-based technique of category I, suitable features are extracted with the aim that they can be classified by the MLP with a maximum possible recognition score. An appropriate MLP architecture is therefore selected in order to obtain the desired output. On the other hand, in the layered subnetwork of MORSEL, called BLIRNET, an architecture is designed on the basis of the retinotopic map of the biological systems. Here the feature extraction and recognition processes are simultaneously distributed over all the layers. The lower layers of the network essentially extract low-level structural information. As we go up in the hierarchy, more and more invariant and informative features are extracted. At the topmost layer of the hierarchy, the most useful and informative entity, which is essentially the object to be recognized, is extracted.

The models that we discuss under category II, in general, do not need any external expert for extracting the significant features and primitives. The architectural design of these models is dependent on the particular application (i.e. the recognition problem) at hand. For this reason, we refer to the models under category II as application-specific systems in our subsequent discussion.

The approaches under the headings of category I and category II are further classified on the basis of their underlying assumptions and the types of recognition problem that they deal with, within the respective category. For example, in category I, if the task is to classify objects from their representative feature vectors, then MLP-based techniques are preferred. The assumption in such classification techniques is that numerical features representative of an object remain invariant under different environments. Under occlusion and overlapping, such an assumption is not valid and in that case the MLP-based techniques do not perform well. If the objects are rich in structural information, that is they can be described in terms of structural relations and attributes of the primitives, then an optimization process or a relaxation-labelling process needs to be performed for their recognition. In that case, often the Hopfield types of model are employed. It is assumed here that the fixed point or attractor in the state space of the network provides a viable solution to the problem. If it is necessary to categorize the objects without the presence of any external teacher then self-organizing models are employed. Here, the objective, in the true sense, is not recognition but rather categorization of the given input which may be helpful in the later stage of recognition.

In category II, as mentioned before, it is difficult to draw a physical line of demarcation, unlike category I, between feature extraction and decision-making tasks. Depending on the types of object (i.e. whether structured or deformable), the design criteria of the network models for integrating the tasks of feature representation and decision making are motivated by knowledge of the neurology or the psychology or the principles of structured object recognition. The way that this design is made determines the partition. Such a partition under category II is shown in figure 4 in order to distinguish the application-specific models which are motivated from psychology, neurobiology and classical structured recognition principles. Another partition under category II is shown on the basis of the capability of decision making (i.e. whether one is able to recognize a single object only, or multiple objects at a time).

We have mentioned in the figure some of the application-specific systems, namely Neocognitron (Fukushima *et al.* 1983, Fukushima 1984), Dynamic Hough Transform (DHT) model (Hinton 1981a, b), MORSEL (Mozer 1991), TRAFFIC (Zemel 1989) and PsyCOP (Basak and Pal 1995a) under this classification. Note that some of these systems are placed in more than one block because they satisfy the respective characteristics.

4. Object recognition: techniques and systems

In section 3 (and figure 4) we mentioned two categories of connectionist approaches for object recognition. In this section, we detail them.

4.1. *Category I*

4.1.1. *Methods based on optimization*

4.1.1.1. Principle. Here, in general, the Hopfield types of model are used. The Hopfield (1982, 1984) model was proposed for retrieving output patterns from noisy input patterns. It is a recurrent network with all neurons connected to each other via weighted links. With a given input pattern, the activation value of each neuron is iteratively updated depending on the given input bias and the weighted signals coming from other neurons. The output of the neurons reaches a stable state if the weights are symmetric (i.e. the weight w_{ij} of the link between any two neurons i and j is the same as that of the link between j and i (i.e. w_{ji})). The Hopfield network is able to function as an associative memory where for a given partial information, the corresponding stored pattern is retrieved (auto-associative memory). The Hopfield model with continuous state dynamics has also been employed to obtain suboptimal solutions to constraint satisfaction problems (Hopfield and Tank 1985, 1986). Kosko (1988, 1991) developed a network for heteroassociative memories where associations between pairs of binary patterns are formed. Simpson (1990) designed a higher-order intraconnected associative memory network where relationships between autocorrelators and heterocorrelators have been studied. The higher-order associative memory incorporates multiplicative connections between nodes. Stochastic network models were also developed in the trend of research on neurocomputing. This includes the learning algorithm developed by Ackley *et al.* (1985) for the Boltzmann machines (which is a conceptual amalgamation of simulated annealing with the Hopfield-type models) for retrieving patterns from partial information.

The Hopfield model has been used in many object recognition algorithms by posing the task as an optimization problem. The fixed points in the state space of the network correspond to the suboptimal matches between the image features and the model features. Despite the fact that Hopfield nets are not guaranteed to provide optimal solutions, the rapid convergence of the network provides an important mechanism for tackling the computational complexity involved in dealing with object recognition problems. However, a limitation of the Hopfield model is that the relative importance of different feature–object pairs cannot be automatically associated, that is the weights cannot be learned. Rather, these weights are pre-assigned before any matching task is performed. Moreover, the problem of mixed category perception (see appendix A) has not been dealt with this model.

4.1.1.2. Techniques. Li and Nasrabadi (1989) and Nasrabadi and Li (1991) developed a scheme for object recognition with the help of a Hopfield model where the polygonal approximations of 2D objects are represented as graphs. A Hopfield network with a 2D array of neurons (where the number of rows represents the number of scene features, and the number of columns represents the number of object features) is used to match the object graphs (one at a time) with the scene graph. The best-matching subgraphs correspond to the objects present in the scene. However, this particular method deals with only symmetric relations between features, that is matching between undirected graphs only is performed. Features using the 'sphericity' property of objects (Ansari and Li 1993) have also been used to form the graphs in this scheme.

A Hopfield-net-based technique has also been developed for matching the structural descriptions of objects (descriptions of parts and spatial relations between them) (Basak *et al.* 1993a). Here, a transformation of the shape descriptions has been suggested with which shape descriptions containing asymmetric spatial constraints between the parts can be matched using symmetric interconnection weights for the Hopfield net, unlike the other attempts (Lin *et al.* 1991, Nasrabadi and Li 1991, Ansari and Li 1993).

The task of 3D object recognition using a Hopfield network was performed by Lin *et al.* (1991). Here, the network is used for matching prototype objects with the scene descriptions in two stages: feature-wise in the first stage and surface-wise in the second stage. Here also, only the symmetric relations between features have been considered.

4.1.2. *Methods based on classification*

4.1.2.1. Principle. The most widely used neural network for classification is the MLP which is essentially a layered feedforward model for generating complex nonlinear decision boundaries (Minsky and Papert 1969, Rumelhart and McClelland 1986). A learning rule, namely back propagation, was designed to train MLP in a supervised mode. However, for a given application, the selection of an appropriate number of hidden layers and nodes for producing a desired performance is still a research problem. Several attempts for network growing (Fahlman and Lebiere 1990, Sietsma and Dow 1991, Romaniuk and Hall 1993) and pruning (Reed 1993) are made for obtaining the optimum network architecture.

In a multilayer perceptron, the nodes generally have some sigmoidal transfer functions or step transfer function. Another class of feedforward networks (Moody and Darken 1989, Hertz *et al*. 1991, Hush and Horne 1993) has emerged, employing generalized radial basis functions in the hidden nodes.

The MLP has been used for object recognition by posing the task as a classification (supervised) problem. Here, feature vectors are derived for each object and an MLP is trained with these feature vectors under the supervised mode. The trained network then accepts the features extracted from the image and classifies these features accordingly. Note that the problem of simultaneously recognizing more than one object cannot be handled with this approach. This is because the feature set extracted from an image consisting of more than one object (occluding each other) is essentially an overlap of the feature sets corresponding to different objects. As a result, the input feature vector falls widely apart from the true decision regions formed by the learned parameters.

4.1.2.2. Techniques. In the method proposed by Tsang *et al.* (1992) for object recognition using an MLP, the features were extracted as follows. The entire range of angles (0–360°) was divided into a number of angular slots of equal size, and each has been treated as a feature. The feature value is determined by the number of corners whose angles fall within the corresponding slot. Besides this, another feature vector was formed in a similar way, where a feature value is equal to the number of arc changes (between two successive corners) by an angle within the corresponding slot. The union of these two feature vectors was then presented to the network for learning and classification.

There is another system developed by Tsang and Yuen (1993) for the recognition of partially occluded objects. The system consists of three stages. In the first stage, object boundaries are detected and then some salient features are extracted from a feature codebook. In the second stage, the presence of some possible objects is hypothesized using a nonlinear elastic matching technique. Finally, the presence of each possible object is verified using the corresponding salient features with the help of an MLP. However, in this system, the MLP is used only in the final stage, and hypotheses about the presence of the objects are made in the nonlinear elastic matching process itself.

Bebis and Papadourakis (1992) developed an MLP-based recognition system where some invariant features were extracted by using the cumulative angular and curvature representations of the object boundaries.

4.1.3. *Methods based on self-organization*

4.1.3.1. Principle. The MLP and other similar models are very good in performing the task of classification. However, biological systems exhibit self-organizing capability, where the entities are grouped into categories automatically without the help of any external teacher. Several investigations have been made on the self-organizing behaviour of connectionist models.

A basic module in most of the categorizing networks is the winner-take-all (WTA) network, which essentially determines the maximally activated neuron in a completely connected network through competitive processes. Grossberg (1982, 1987) has provided a mathematical theory of such WTA networks.

There are several self organizing networks employing the competitive learning process, such as Kohonen's (1988) self-organizing feature map, and the adaptive resonance theory (ART) (Carpenter and Grossberg 1987a). The Kohonen model can automatically produce topologically correct maps of the features of observable events. ART was developed by Carpenter and Grossberg for classifying input patterns into different categories. Both supervised and unsupervised modes of learning are possible in the ART of network models with fast and slow learning processes. ART was extended to categorize analogue input patterns to form stable category codes (Carpenter and Grossberg 1987b). The characteristics of such networks for additive and subtractive noise have also been studied (Carpenter and Grossberg 1990, Carpenter *et al*. 1991).

Amari (1972, 1977) developed a self-organizing model for concept formation and orthogonal and covariance learning techniques for the model. Amari and Maginu (1988) formulated the statistical neurodynamics of the associative memory. Anderson *et al.* (1977) have developed the brain-state-in-a-box model to produce the association between the input and the output patterns and applied it for categorical perception. They also discussed probability learning techniques in this model.

The principle used in object recognition based on self-organization is analogous to the methods based on classification, except that the decisions are made in an unsupervised manner (i.e. without, the help of any external teacher) whereas in MLP-based methods, supervised training is used.

4.1.3.2. Techniques. Bebis and Papadourakis (1992) also investigated the effectiveness of the Kohonen model with the same features as that used in the MLP-based techiuque, for performing the task of object recognition. Although the Kohonen model has been found to self-organize the feature sets derived from single objects, it would face the same difficulty as the MLP in the case of mixed category perception (MCP).

ART has also been used for shift- and orientation-invariant visual pattern recognition. Srinivasa and Jouaneh (1993) used an invariance network in conjunction with an ART1 module for this purpose. Here, different rotations (0, 90, 180 and 270°) and shifts (in discrete steps) are explicitly coded in the invariance network, which cooperatively interacts with the output layer of the ART1 module. Although shift and rotation invariance were achieved for simple form of visual patterns, real-life objects were not considered as input. Moreover, the ART1 module also has the same difficulty as the MLP in the perception of mixed categories.

4.2. *Category II (application-specific systems)*

4.2.1. *Principles.* Application-specific systems generally accept the input image directly or sometimes the spatially distributed features (e.g. edge map, or skeletal version of the image). The general difficulty of designing such systems lies in the incorporation of the characteristics of translation, rotation and scale invariance into the systems. Again, the incorporation of these invariance properties depends on the way of mapping the features automatically onto the network. This is also dependent on the types of object to be recognized by these systems. We specify three different strategies that are generally followed for this purpose.

Strategy 1. Layers of analysers (a collection of processing elements or neurons) are employed to achieve invariance

Strategy 2. Rigid transformations (i.e. shift, orientation and scaling transformations) are computed explicitly within neural architecture

Strategy 3. A selective attention mechanism is used.

In strategy 1, the analysers hierarchically extract and group the features from an image. The lower-layer analysers extract simpler features while the higher-layer analysers extract more complex features by grouping the simpler features extracted in the lower layer. The analysers in each layer are able to respond to the patterns with a small amount of invariance, because the neurons in each layer accept activations from a group of neurons in the lower layer. Thus, in this technique, the invariance is achieved incrementally within the layers of analysers. This particular strategy is motivated by the theory of perceptual organization and also by the organization of optical nervous systems of animals. This technique also enables the systems to tolerate a good amount of deformation besides shift, orientation and scale change. This kind of strategy is useful for the recognition of alphabetic characters, numerals, etc., and their deformable versions.

In strategy 2, the transformations from the feature reference frame to the object reference frame are computed within the connectionist system. In this process, the procedure of GHT is often incorporated within the links of the connectionist architecture so that the neuron corresponding to the object (together with its position, orientation and scale) gets maximum activation. Sometimes, the activation values are also updated in a way similar to relaxation algorithms. The incorporation of rigid transformations is performed by using different connectionist learning algorithms including back propagation. Sometimes, learning rules are used to generate the internal representations together with their reference frames in the neural architecture. This kind of strategy for computing transformations is particularly useful in the recognition of rigid objects (e.g. industrial objects).

The term 'selective attention' itself explains the fact that a region is selectively attended to at one time. This psychological phenomenon is sometimes incorporated into the neural architecture (strategy 3) in order to map certain portions of the image selectively on to the input layer of the recognition system. An additional mapping circuitry (attention controller) is often employed for this purpose. Whenever a particular zone is attended to, the attention controller has the relevant information about the location of that zone. Thus, if some object is present in the zone of attention, the positional information in the attention controller can aid the recognition system to specify the position of this object. The selective attention mechanism (apparently, a sequential process) is helpful in building connectionist systems which are used for reading texts or scripts or even for recognizing overlapped objects.

4.2.2. *Description of the systems.* Here, we discuss mainly five different models, namely the neocognitron, the dynamic Hough transform (DHT) model, MORSEL, TRAFFIC and PsyCOP. These models had been developed with different types of object as input.

4.2.2.1. Neocognitron. The cognitron (Fukushima 1975) was developed to categorize input patterns by employing *competitive learning* techniques, but it fails to recognize patterns suffering from positional shifts. To incorporate the properties of position and scale invariance, a multilayered model, namely the neocognitron (Fukushima *et al.* 1983, Fukushima 1984, 1987) was developed. The neocognitron employs strategy 1, as discussed in section 4.2.1, for incorporating shift and scale invariance.

The model uses two kinds of cell, namely S and C cells arranged in alternate layers. S cells extract the features at various stages, while the C cells ensure position and scale invariance. The shift invariance is achieved by tolerating some positional shift in each layer of C cells at a time. The network was designed to possess a self-organizing capability where the connections between two maximally activated cells (within a predefined vicinity) are reinforced. The main contribution of this model is that it introduces the concept of adjusting the positional shift or deformations incrementally within a group of cells in each layer.

The model was extended to incorporate the property of selective attention (Fukushima 1988 a, b) by using feedback pathways from the output layer to the input layer. With the help of feedback signals, the network is able to recognize one pattern (the most prominent) even when a mixture of more than one pattern is presented to the network. It was tested on a numeral recognition problem. The effectiveness of the neocognitron has also been tested for the character recognition (Fukushima *et al*. 1991, Fukushima 1992). Recently, the model has been extended to segment and recognize cursive scripts (Fukushima and Imagawa 1993) with the help of a 'search controller' which assists selection of a particular search area.

The power of the neocognitron and its variations lie in the fact that the models are capable of tolerating error due to positional shift, scale change or deformation. However, the model is not capable of recognizing more than one object simultaneously. Whenever a mixture of patterns is provided to the network, it always recognizes one (the most prominent) of them. Moreover, the model does not consider the structural relationships between the features and the objects. The model is also incapable of tolerating the rotational variance. Recently, a variation of the model has been used to achieve rotation invariant object recognition (Himes and Inigo 1992). Minnix *et al*. (1992) used the underlying concept of the neocognitron and developed a more elegant self-organizing model for translation-invariant object recognition.

4.2.2.2. Dynamic Hough transform. Hinton (1981 a, b) extended the idea of the GHT to the DHT. The DHT model employs strategy 2, as discussed in section 4.2.1, for the incorporation of shift, scale and orientation invariance.

The concept is to choose an appropriate object reference frame with respect to which the retinotopic features are described. Here, each object is associated with all possible frames of reference, and with respect to each frame of reference the objects' features are described. The objects' features in conjunction with the retinotopic features give votes to the units of a network which essentially maps the object level features to the retinotopic level. The appropriate frame of reference is then determined in a cooperative and competitive process. Hinton and Lang (1985) suggested a simulation method for this network.

The network has four different kinds of neuron. The retinotopic neurons represent the features present in the image together with their locations. The object-based units represent the object features together with their locations. The object features have more coarse-coded representation compared with the retinotopic features. Another set of units represent the object entities (object or letter units) which are connected to the object-based units. The retinotopic units and the object feature units are linked by the mapping units. The mapping units are connected by special three-way links to the retinotopic units and the object-based units.

The network has been tested with a set of selected English letters. Initially, a large number of mapping units is activated which in turn activate a number of object-based units. The three-way links between retinotopic units, mapping units and object-based units store the information of possible locations of activations of mapping units and object-based units. Once the object-based units are activated, they receive top-down support from the letter units, and a few of them are selectively enhanced. The corresponding mapping units are selectively enhanced because the mapping units are activated by the product of activations of the retinotopic units and the object-based units. The activation in the output layer dynamically determines the appropriate rigid transformation (HT) from the object reference frame to the image plane (retinotopic plane).

In the simulation, it has been found that the network has a tendency to perceive one shape in the position of some other shape when several shapes are presented to the network. Hinton and Lang refer to a psychological study (Triesman and Schmidt 1982) where human beings are found to make same sort of mistakes.

The model had been simulated for only six letters and the object-based units explicitly store all possible object features at each location. This indicates that, with an increase in the number of object features (necessary to describe more complex objects) and higher spatial resolution, the required number of neurons would drastically increase, and consequently the number of three-way links between mapping units, retinotopic units and the object-based units would also increase. A two stage method (Hinton and Lang 1985) was suggested for dealing with translation, rotation and scaling to reduce the number of gated connections at the cost of time.

However, in the DHT model, the relative importance of the features is not considered. A concept of part–whole hierarchy was provided (Hinton 1990) for efficient storage of and effective computation on the objects.

Although the concept has been illustrated with the examples of letters, it is also applicable to the recognition of other objects.

4.2.2.3. MORSEL. Mozer (1991) developed a word perception model (MORSEL) incorporating the selective attention mechanism. The word MORSEL stands for 'multiple object recognition and attention selection'. The model is able to learn and recognize multiple letters together with their relative positions. This particular system uses strategy 3 (section 4.2.1) for making decisions about the relative positions of different letter clusters.

The network consists of a shape detection module whose output is fed to another network called a pull-out (PO) network. The output of the PO network and the network for attentional mechanism are fed to a visual short-term memory which makes the decision about the location and the identity of an object. Note that, apart from the shape detection module, there may be other modules such as the colour detection module and motion detection module whose outputs are also fed to the PO network.

MORSEL was designed mainly for multiple-word recognition. Each word has been looked upon as a conjunction of different letter clusters, where the letter clusters were described by one or two known letters and one or two unknown letters. With several activated letter clusters, different words can be formed with different types of binding. The bindings in turn depend on the spatial positions of the letter clusters. The information about the spatial positions is extracted by the attentional mechanism and the binding process is performed in the visual short-term memory.

A hierarchical network called BLIRNET (Mozer 1991) has been designed as a part of MORSEL for detecting various letter clusters from a selected zone (controlled by attentional mechanism) in the image. The structure of BLIRNET is similar to an MLP, but the connection from a node to the lower layer is restricted to a zone. A hierarchical feature extraction process has been used in BLIRNET where the back-propagation learning rule has been used to learn the different letter clusters.

The output of BLIRNET is fed to another PO network which essentially enhances the output of BLIRNET in conjunction with semantic knowledge. Both bottom-up and top-down processes are activated in the PO net and the letter clusters receiving support from the higher region of the PO net are enhanced. The attentional mechanism (AM) sequentially scans the input image and its output is integrated in the visual short-term memory with the output of the PO network. AM selectively chooses one zone and the features present in that zone are mapped into BLIRNET's input. Thus BLIRNET is able to detect the letter clusters independent of their locations.

Different psychological phenomenan including neglect dyslexia and attention dyslexia in psychological patients have also been explained with the help of MORSEL (Mozer and Behrmann 1989). However, the performance of the model is dependent on the orientation of the objects and therefore it may be difficult to apply it for industrial object recognition. Again, like the neocognitron, no structural relationship between the features and the objects was considered and the performance of the network was also not tested on industrial objects.

4.2.2.4. TRAFFIC. The structural relationship between features and objects was considered in the connectionist system called TRAFFIC (Zemel 1989) where the word TRAFFIC is a loose acronym for 'transforming feature instances'. TRAFFIC employs strategy 2 (section 4.2.1) for the incorporation of rigid transformations.

The viewpoint-independent transformations from the feature reference frame to the object reference frame for rigid objects are learned within a hierarchical network architecture using the back-propagation learning rule. The transformations from feature level to object level are embedded into the links between successive layers. The features are grouped hierarchically where the intermediate layers in the hierarchy represent the subparts or macrofeatures.

A network architecture similar to that of the MLP has been designed where the connections of a neuron to its lower-layer neurons are restricted to a zone, called the receptive field of the corresponding entity. Each node has five parts representing position (x, y), orientation θ, scale s and confidence c about the presence of the corresponding entity. The highest layer in the hierarchy contains only one neuron for each object and represents its position, orientation, scale and confidence about its presence.

The values of x, y, θ and s are computed depending on the transformations stored in the links and the confidence values of the lower layer cells, for example

$$x_{o[q]} = \frac{\sum_{f \in F} M_{fo} c_{f[p]} x_{o[q]f[p]}}{\sum_{f \in F} M_{fo} c_{f[p]}}, \qquad (1)$$

where $c_{f[p]}$ is the confidence of the feature f in cell p which is in the receptive field of object o in cell q, M_{fo} is the degree of membership of f in o and $x_{o[q]f[p]}$ is the value of x of o predicted by f. Once the x, y, θ and s values are computed for some object o, the variances of these values are computed, and the sum of these variances (i.e. σ_x^2, σ_y^2, σ_θ^2 and σ_s^2) are normalized. The confidence value of the object is computed according to

$$c_{o[g]} = \frac{\sum_{f\in F} M_{fo} c_{f[p]}}{\sum_{f\in F} M_{fo}} \left(1 - \sigma^2_{o[g]}\right). \qquad (2)$$

The transformations x_{fo}, y_{fo}, θ_{fo}, s_{fo} and the degree of membership M_{fo} are computed using the back-propagation learning rule.

The system has been tested on different images of astral constellations. The network provides a structured framework for object recognition considering the rigid transformations from the feature reference frame to the object reference frame. However, one restriction has been used where, in each cell, at most one object or feature can be activated. This restricts the network to recognize multiple instances of the same object.

4.2.2.5. PsyCOP. The structural relationships between features and objects are also considered in PsyCOP (Basak and Pal 1995a). The word PsyCOP stands for psychologically motivated connectionist system for object perception. The system is designed by integrating the GHT technique with the psychological finding that identification and localization occur in two separate regions in the visual cortex (Kosslyn 1975, Kosslyn *et al.* 1985). The system also uses the mechanism of selective attention for initial hypotheses generation, that is both strategy 2 and strategy 3 (section 4.2.1) are incorporated in this system.

The system has two separate networks; one is a decision-making network and the other is an attention control network. The decision-making network has two seperate channels: one for entity (A cells) and the other for location (B cells). It has three layers: input, hidden and output. The input layer represents the features, the output layer the objects, and the hidden layer represents the feature–object associations. Activations of the B cells represent the confidence levels of the objects present in the corresponding locations. Activated A cells represent the objects present in the scene. A cells and B cells are selectively coupled by the links which are selectively stimulated from the lower layer and coordinated by the attention control mechanism.

PsyCOP has been designed for the recognition of structured objects such as flat industrial parts (hammer, spanner, plier, etc.). The transformations from the feature reference frame to the object reference frame (HT) are stored in the links between the input and hidden layers. The degrees of importance of different features with respect to the objects are stored in the links between the hidden and output layers. Initially, the output nodes are activated from the input layer through the hidden layer and these output nodes feed back their activations down to the hidden layer. The hidden nodes compete between themselves for support from the output layer and, in turn the output layer also gets support from the hidden layer. After settling of this cooperative and competitive process, the output layer represents the objects together with their locations.

Since PsyCOP uses two separate channels for identification and localization, the number of nodes is reduced. It is able to recognize not only multiple objects but also multiple instances of the same object simultaneously. However, the problem of scale invariance has not been dealt with in PsyCOP, although rotation and translation invariance have been taken care of.

Note that PsyCOP is designed for simultaneous recognition of multiple objects from an image based on the principle of MCP (appendix A). The problem of MCP addresses the issue of learning and simultaneous recognition of multiple patterns even when they are superimposed (or overlapped). Let us consider two patterns $A = 10110001$ and $B = 11001001$ with eight features each. Let them be superimposed to generate a pattern $C = 11111001$. The task of MCP is to decide that the pattern C is not new but rather a combination of A and B. Using standard networks such as the MLP, the Hopfield model, the Kohonen model or the ART model, it is not possible to perform the aforesaid task even if more than one such network is concatenated. However, there exist several networks such as EXIN (Marshall 1990a, b, 1992), SONNET (Nigrin 1990a, b, c, 1992), X-tron [Basak *et al.* 1992, 1993b, 1996, Basak and Pal 1995b] and masking field (Cohen and Grossberg 1986, 1987) which have been exclusively designed for this purpose. The principle of these networks can be exploited for developing application-specific systems for simultaneous recognition of multiple objects. For example, PsyCOP is based on the principle of X-tron.

Since the relevance of MCP to object recognition is significant, we discuss, in brief, its principle and models in appendix A for the sake of completeness of the review.

4.2.2.6. Other models. There exist several other relevant investigations which need to be mentioned in the context of object recognition using neural networks. Poggio and Edelman (1990), Edelman and Weinshall (1991), Edelman (1992) and Poggio *et al.* (1992) developed a three-layered network for producing a standard viewpoint representation of the 3D objects from any given viewpoint. The input layer consists of the coordinate values of the features in the image plane. The hidden layer consists of several nodes with generalized radial basis functions (Gaussian in nature). The centres of the transfer functions code the input viewpoint representation from the coordinate values. The output nodes have linear gain functions. The activations in the output layer represent the coordinates of the features in the standard reference frame. The performance of the model was tested with wire-frame models of 3D objects.

Wechsler and Zimmerman (1988) considered distributed associative memory (matrix associative memory) for recognizing 2D objects with rotational and scale invariance. They used a confomal complex-log mapping to transform the rotation into an additive constraint. From the transformed image the scale and rotations are estimated, and retrieval of the corresponding stored object descriptions is performed with the help of distributed associative memory. The model was further extended (Zimmerman 1992) in conjunction with a re-projection system to tolerate some non-metric transformations due to change in camera positions (the non-metric transformations include foreshortening, self-occlusion, etc.).

The neural network model for position-independent pattern matching (Hirai and Tsuki 1990) consists of three layers, namely the pattern-matching layer, minimum distance layer and a recognition layer. It uses a supervised mode of learning. The effectiveness of the network is tested with very simple synthetic patterns. However, the model does not deal with the problems of rotation invariance and simultaneous recognition of multiple objects.

Chan (1992) developed a model for object recognition with translation invariance. It uses one hypernetwork which generates the target response along with some secondary response values. The next stage consists of a confirmatory network which analyses the target response and the secondary response values to decide which particular object has appeared in the scene. The model achieves translational invariance by replicating the weights of the links over different positions, but it is not able to recognize orientationally variant objects and also the behaviour for mixed objects has not been studied.

There exist several other recently developed connectionist systems for object recognition. A cascade of restricted Coulomb energy (RCE) networks (Li and Nasrabadi 1993) to classify objects. The boundary or shape information of the objects was coded into feature vectors of fixed length, and then cascaded RCE networks were used to resolve progressively the overlapping complex decisiion regions. However, in this technique, the problems of mixed object recognition and localization have not been considered. Higher-order neural networks employing multiplicative (pi) connections had also been employed for position-, scale- and rotation-invariant object recognition (Spikovska and Reid 1993). However, in this technique, the problem of localization within the connectionist framework has also been dealt with.

Feldman (1982, 1985) and Feldman and Ballard (1982) presented the principles of connectionist computing, the principle of stable coalition formation and the WTA network. Note that the neural networks mainly involve the study of emergence of activations of the cells and weights of the links on the basis of mathematical modeling. On the other hand, AI-based techniques mostly deal with inference representation. For solving the recognition problem in the visual domain, the representation of knowledge in the spatial domain needs to be considered. This leads to the concept of structured connectionist models for developing visual recognition systems. Sabbah (1985) also used a structured connectionist framework for object recognition.

4.2.3. *Learning mechanism*. TRAFFIC and MORSEL employ back-propagation learning, that is they operate under a completely supervised mode. The neocognitron operates with a reinforcement weight updating process which may be classified as semisupervised. In other words, the network has the ability to operate under an unsupervised mode, but some of the initial desired outputs are dictated by an external teacher PsyCOP is designed for supervised learning but is different from back-propagation learning. It can take care of mixed and overlapped instances of the objects. MORSEL can also take care of multiple objects simultaneously, but MCP is not embedded in the learning rule of MORSEL, rather it is performed with the help of a subnetwork with fixed interconneciton weights. The learning rules of PsyCOP, on the other hand, directly embed the capability of simultaneous perception of mixed objects. In the DHT model, the feature–object associations are predetermined and are not learned. In the following discussion, we briefly describe the various learning processes involved in neocognitron, PsyCOP, TRAFFIC and MORSEL.

4.2.3.1. Neocognitron. Each S cell in the neocognitron accepts input from a group of activated C cells of the previous layer which represent the collective activity from different parts of an entity or feature. The reinforcement occurs in the links from C cells to the activated S cells. Each S cell (in the layer *l*), in turn, produces an approximately normalized output, which is given as

$$v_S^{(l)}(p,k) = \alpha_l \times \varphi\left(\frac{\beta_l + \sum_{\kappa=1}^{KC_{l-1}} \sum_{\Delta p \in A_l} w_1^{(l)}(\Delta p, \kappa, k)\, v_C^{(1-1)}(p + \Delta p, \kappa)}{\beta_l + [\alpha_l/(1+\alpha_l)]\, w_2^{(l)}(k)\, v_V^{(l)}(p)} - 1\right), \tag{3}$$

where $\varphi(x)$ is the transfer function given as

$$\varphi(x) = \begin{cases} x & \text{for } x > 0, \\ 0 & \text{otherwise.} \end{cases}$$

p denotes the location in the 2D coordinate space with respect to some standard reference frame, Δp is the deviation or positional shift that occurred from the layer of C cells to the layer of S cells, α_l and β_l are constants, κ is the C-cell plane containing similar types of entity (i.e. similar types of feature) and k is the S-cell plane. KC_{l-1} is the total number of C-cell planes in the layer $l-1$. With each S cell, an auxiliary node called a V cell is connected, which normalized the output of the correpsonding S cell. The weights of the links from C cells to S cells are represented by w_1 and those from V cells to S cells are represented by w_2. The weights of the links from C cells to V cells are represented by w_3 and those from S cells to C cells are represented by w_4. A_l is the neighbourhood from which an S cell (in the layer l) and the corresponding V cell connected to the S cell are activated. The outputs of V cells and C cells are given by

$$v_{\mathrm{V}}^{(l)}(p) = \left(\sum_{\kappa=1}^{KC_{l-1}} \sum_{\Delta p \in A_l} w_3^{(l)}(\Delta p)\left[v_{\mathrm{C}}^{(l-1)}(p + \Delta p, \kappa)\right]^2 \right)^{1/2} \tag{4}$$

and

$$v_{\mathrm{C}}^{(l-1)}(p, \kappa) = \psi\left(\sum_{\Delta p \in D_{l-1}} w_4^{(l-1)}(\Delta p) v_{\mathrm{S}}^{(l-1)}(p + \Delta p, \kappa) \right). \tag{5}$$

D_{l-1} is the neighbourhood from which a C cell is activated. $\psi(.)$ is the transfer function of the C cells which is given as

$$\psi(x) = \frac{\varphi(x)}{1 + \varphi(x)}. \tag{6}$$

The mostly activated S cell is selected as the seed cell and the connection weights from the C cells in the previous layer to the mostly activated S cell are reinforced. During reinforcement, w_3 and w_4 are kept fixed, and w_1 and w_2 are updated. The updating rule for reinforcement of the links are given as

$$\Delta w_1^{(l)}(\Delta p, \kappa, k) = \eta w_3^{(l)}(\Delta p) v_{\mathrm{C}}^{(l-1)}(p + \Delta p, \kappa), \tag{7}$$

$$\Delta w_2^{(l)}(k) = \eta v_{\mathrm{V}}^{(l)}(p). \tag{8}$$

Note that the reinforcement process takes place over a neighbourhood in the layer of the C cells depending on the extent of Δp. This helps the process to tolerate noise and positional shifts. η_l determines the rate of weight updating in a particular layer.

The term 'reinforcement' also refers to a semisupervised strategy for learning. Unlike unsupervised learning algorithms (self-organization), the 'seed cells' can be selected by an external teacher instead of always having the winner as the seed cell.

4.2.3.2. PsyCOP. In PsyCOP, during supervised training, transformations from the feature reference frame to the object reference frame and the degrees of importance of the features with respect to different of objects are learned. The weights of the bottom-up (w_1) and top-down (w_2) links store the degrees of importance of the features with respect to the objects, and those from the input to the hidden layer store the transformations. The weights w_1 of the bottom-up links from the hidden layer to the output layer are updated as

$$\Delta w_1(i, j) = \eta\left[\left(\alpha_{ij}\delta_j w_2(j, i) - \alpha_j \frac{w_1(i, j)}{w_2(j, i)}\right) + v_{ij}t_j - (\alpha_{ij}v_{ij} + \alpha_j t_j)w_1(i, j)\right]. \tag{9}$$

The weights w_2 of the top-down links from the output layer to the hidden layer are updated as

$$\Delta w_2(j, i) = \eta\alpha_j t_j[v_{ij} - w_2(j, i)]. \tag{10}$$

In the above, v_{ij} is the output of the hidden node corresponding to the ith input node and jth output node, t_j is the desired output of the jth output node. α_{ij} and α_j are constants associated with the respective hidden and output nodes which decrease with time for which the corresponding nodes remain activated, and δ_j is the error term given by

$$\delta_j = \frac{t_j - v_j}{\gamma g'(u_j)},$$

where γ is a constant, $g(.)$ is the transfer function of the output nodes, u_j is the total input received by the node j, and v_j is the output of the jth output node.

It has been shown by Basak *et al.* (1993b) that, if the degrees of presence of the features and objects are considered only in terms of 0 and 1, that is the features or objects are considered only to be present or absent, than after convergence the weights of the top-down and bottom-up links go to the conditional probability values given as

$$w_2(j, i) = \mathrm{Prob}(f_i | o_j) \tag{11}$$

and

$$w_1(i, j) = \frac{\mathrm{Prob}(o_j | f_i)}{\gamma + \sum[\mathrm{Prob}(f_i | o_j)\,\mathrm{Prob}(o_j | f_i)]}. \tag{12}$$

The weights of the links from input to the hidden layer are updated as

$$\Delta w_3(i, j, T) = \alpha_{ij}[\epsilon_T - w_3(i, j, T)] \tag{13}$$

where T denotes the type of transformation, that is the transformation from the feature reference frame to the object reference frame with respect to position and orientation, and ϵ_T is the offset or the difference between the features' position or orientation with the object's position or orientation. The updating rule provides an iterative averaging of the transformations.

4.2.3.3. TRAFFIC and MORSEL. Both TRAFFIC and MORSEL employ standard back-propagation learning. However, the use of the back-propagation rule has different purposes in these two networks.

BLIRNET, which is a part of MORSEL, uses back propagation for learning the letter clusters in the zone of attention selected by the attention mechanism subnetwork. It has a hierarchical structure similar to the neocognitron and the positional variations are tolerated incrementally in each layer. The positional information is extracted from the attention mechanism subnetwork. The PO network accepts the output of BLIRNET and extracts the strongest activated letter clusters depending on some semantic interrelationship. The attention mechanism subnetwork and PO subnetwork employ fixed excitatory and inhibitory connections and the weights are not updated. The back-propagation rule in BLIRNET does not learn the positional information; rather it is extracted from the AM subnetwork.

In TRAFFIC, on the other hand, back propagation is used to learn the transformations from the feature reference frame to the object reference frame. As can be seen in (1) and (2), the position, orientation and scale of the objects are computed from the feature instances. The variances in position, orientation and scale are computed accordingly. For example,

$$\sigma^2_{x_{o[q]}} = \frac{\sum_{f \in F} M_{fo} c_{f[p]} \left(x_{o[q]} - x_{o[q]f[p]}\right)^2}{\sum_{f \in F} M_{fo} c_{f[p]}}.$$

The confidence $c_{o[q]}$ about the presence of an object o in the output grid location q is computed by

$$c_{o[q]} = \frac{\sum_{f \in F} M_{fo} c_{f[p]}}{\sum_{f \in F} M_{fo}} \left(1 - \sigma^2_{o[q]}\right),$$

where

$$\sigma^2_{o[q]} = \frac{1}{4}\left(\frac{\sigma^2_{x_{o[q]}}}{\max\left(\sigma^2_{x_{o[q]}}\right)} + \frac{\sigma^2_{y_{o[q]}}}{\max\left(\sigma^2_{y_{o[q]}}\right)} + \frac{\sigma^2_{\theta_{o[q]}}}{\max\left(\sigma^2_{\theta_{o[q]}}\right)} + \frac{\sigma^2_{s_{o[q]}}}{\max\left(\sigma^2_{s_{o[q]}}\right)}\right).$$

In the back-propagation rule, the error is computed as

$$E = \tfrac{1}{2} \sum_q \left(c_{t[q]} - c_{o[q]}\right)^2.$$

From the error measure the transformations are learned as

$$\Delta X_{fo} = -\eta \frac{\partial E}{\partial X_{fo}},$$

$$\Delta Y_{fo} = -\eta \frac{\partial E}{\partial Y_{fo}},$$

$$\Delta \Theta_{fo} = -\eta \frac{\partial E}{\partial \Theta_{fo}},$$

$$\Delta S_{fo} = -\eta \frac{\partial E}{\partial S_{fo}}.$$

4.2.3.4. Some remarks. It has been mentioned (Fukushima 1988 a) that, in the case of learning with a teacher, a sufficiently large value is assigned to η_l (equation (7)) so that the reinforcements of the input connections to each seed cell are completed in a few steps of training pattern presentation. Therefore, in the ideal situation, if we have only one iteration with $\eta_l = 1$ for all l, then the output of the winner in a local neighbourhood of layer l becomes

$$v_S^{(l)}(\hat{p}, \hat{k}) = \alpha_l \varphi \frac{[\eta_l/(1+\eta_l)] \sum_{\kappa=1}^{KC_{l-1}} \sum_{\Delta p \in A_l} w_3^{(l)}(\Delta p)\left[v_C^{(l-1)}(\hat{p}+\Delta p, \kappa)\right]^2}{\beta_l + [\alpha_l \eta_l/(1+\eta_l)] \sum_{\kappa=1}^{KC_{l-1}} \sum_{\Delta p \in A_l} w_3^{(l)}(\Delta p)\left[v_C^{(l-1)}(\hat{p}+\Delta p, \kappa)\right]^2}. \quad (14)$$

The output of the cells other than the winner (in the local neighbourhood) becomes zero, since there is no weight updating in those links. Equation (14) shows that Weber's law (as explained in ART (Carpenter and Grossberg 1987 a)) is automatically incorporated through the learning rules. Weber's law has also been incorporated in defining the learning rules for PsyCOP.

In TRAFFIC, the confidence level about the presence of an object in an output cell is solely dependent on the transformation values from the feature reference frame to the object reference frame. Therefore, the transformations are iteratively computed by minimizing the difference of the desired confidence level and the actual confidence level about the presence of the objects. In PsyCOP, on the other hand, the confidence level about the presence of an object depends not only on the transformation values from the feature reference frame to the object reference frame but also on the relative degree of importance of the features constituting the object.

Therefore, in PsyCOP, a two-stage learning paradigm is used in order to learn both the transformation values, as well as the relative degree of importance of the features with respect to the objects.

In MORSEL, the input to the recognition module (i.e. BLIRNET) is selected by the AM subnetwork. This is also the case for PsyCOP. However, in MORSEL, any difficulty arising because of the presence of multiple objects in the image is solely resolved by the attention mechanism subnetwork, that is, it allows only one object to be present in the input of BLIRNET at a time. On the other hand, the recognition module of PsyCOP itself can take care of the difficulty arising from the presence of multiple objects in the input.

The BLIRNET of MORSEL has a structure analogous to the neocognitron. However, in the neocognitron, if supervised reinforcement is performed, then the desired activation in each layer should be known. In BLIRNET, the supervised error popagation is performed only with the knowledge of the desired final output.

4.2.4. *Underlying assumptions, operating conditions and versatility.* The domain of applicability of the aforesaid application-specific systems including their underlying assumptions and operating conditions is described below.

4.2.4.1. Neocognitron

(*a*) The objects need not be structured and rigid.

(*b*) There is no overlap in the objects to be recognized in the input image. If there is any overlap, then the system can recognize only the most prominent one.

(*c*) The objects can be recognized at different positions and scales, but not with any rotational variations.

(*d*) Objects are recognized by hierarchically grouping the pixel information, that is by performing fine to coarse coding of the features. The model can learn the hierarchical groupings of the features under both supervised and unsupervised modes.

4.2.4.2. PsyCOP

(*a*) The objects need to be highly structured and rigid, that is large deformations cannot occur to the objects. The term 'rigid' indicates that the location of the objects (i.e. position and orientation) is almost fixed with respect to the constituent features.

(*b*) Any kind of translation or rotation can occur to the objects.

(*c*) More than one object can be present at the input and they can overlap, but more than one object cannot have the same location, that is, the same position and orientation. One of the very basic assumptions in the design of PsyCOP is that the multiple objects present in the input (possibly overlapping each other) are recognized simultaneously and not one by one.

(*d*) Objects do not suffer a large amount of scale variation.

(*e*) Objects are recognized by transforming the feature instances from the feature reference frame to the object reference frame and computing the cumulative evidence of the features. The model can learn both the transformations from the feature frame to the object reference frame, and the relative importance of the features with respect to the objects as well.

4.2.4.3. MORSEL

(*a*) Objects need not be highly structured and rigid. However, the required rigidity of the input objects is greater in this model than in the neocognitron, and less than in PsyCOP, TRAFFIC or the DHT model.

(*b*) More than one object can be present in the input and it is assumed that these objects are recognized one by one. The objects should not overlap each other. Note that both MORSEL and PsyCOP can accept more than one object simultaneously as input. However, in PsyCOP, input objects can overlap and they are identified simultaneously. In MORSEL, on the other hand, input objects are not mixed, that is they are separated from each other, and these objects are identified one by one by sequentially scanning the image. The sequence in which the objects are identified also provides an interrelationship of the positional arrangement of the objects.

(*c*) Objects do not suffer any rotational or scale variations.

(*d*) Each individual object is recognized by hierarchically grouping the features and the model can learn the hierarchical grouping of the features in order to recognize each individual object. The model identifies a group of objects together (e.g. words constituting more than one letter) and all possible relative positional arrangements of the objects are exhaustively known.

4.2.4.4. TRAFFIC

(*a*) Objects need to be highly structured and rigid, like PsyCOP.

(*b*) Objects can suffer any kind of variation in position, orientation and scale.

(*c*) In the input, more than one object can be present with different locations (i.e. positions and orientations) as in the cases of PsyCOP and MORSEL. However, unlike the later two, it cannot accept multiple instances of the same object.

(*d*) Objects are recognized by transforming the feature instances from the feature reference frame to the object reference frame and computing the cumulative evidence from the features. The model can learn the transformations from the feature reference frame to the object reference frame, that is the relative position of the objects with respect to the constituent features. From the variation in the position of the constituent features, confidence about the presence of an object can be obtained.

4.2.4.5. Dynamic Hough transform

(*a*) Objects need to be highly structured and rigid, like PsyCOP and TRAFFIC.

(*b*) Any kind of rotation and translation can occur to the objects.

(*c*) The objects are recognized by transforming the feature instances from the feature reference frame to the object reference frame and computing the cumulative evidence from the features. The model cannot learn the transformations from the feature reference frame to the object reference frame. Only fixed interconnection weights can be employed from the feature level to the object level. Note that this model does not have any learning mechanism, unlike the other four. Here the objective is not to learn the objects, rather to understand the mechanism whereby feature instances trigger the object instances.

5. Comparison of characteristics

As mentioned in section 3, the object recognition techniques based on the basic networks such as the Hopfield, MLP or Kohonen model (i.e. category 1 in section 4.1) require the descriptions of the objects to be explicitly derived by some other algorithms and then fed to the neural architecture. In other words, the neural architectures here seem to operate as auxiliary processors in the main recognition system. In these techniques, there should always be some expert intervention or some separate process which properly maps the features or relational descriptions on to the network model. The application-specific systems (such as the neocognitron, DHT model, MORSEL, TRAFFIC and PsyCOP), on the other hand, take either the segmented image or spatially distributed features and use them directly as input. It is true that, in some of these systems, features need to be separately extracted, but mapping of the features on to the network is performed by the systems themselves and no separate expert intervention is therefore necessary. In other words, application-specific networks are aimed more towards building up stand-alone systems for object recognition.

Note that these methods were developed by keeping different application domains in mind. It is, therefore, extremely difficult to quantify their relative performances. However, an overall comparison of the characteristics of these methods is provided below.

The Hopfield model has been used for multiple-object recognition by matching the graphs corresponding to the desired objects one at a time with the scene graph, but the problem of MCP (i.e. simultaneous recognition of multiple objects) has not been dealt with. Moreover, the degree of importance of the features cannot be learned. On the other hand, feature importances can be learned with MLP or Kohonen-model-based techniques. Here also, the problem of MCP cannot be handled. Hopfield-model-based techniques are similar in nature to the relaxation-labelling techniques used in classical algorithms whereas the techniques based on the MLP are similar to the statistical or decision theoretic rules in pattern recognition.

As indicated in figure 4, the application-specific systems mainly differ from the points of method (motivation) of design and capability of decision making. The architecture of the neocognitron is motivated from the hierarchical structure of the visual cortex in order to perform hierarchical grouping of features (strategy 1 in section 4.2.1). TRAFFIC employs strategy 2 (section 4.2.1) which is analogous to the classical concept of the GHT to compute explicitly the position, orientation and scaling information of an object. The DHT model also employs this strategy, but the location, orientation and scaling infomation are not explicitly computed; rather the spatial locations of the activated neurons provide these information. MORSEL, on the other hand, employs selective attention mechanism (strategy 3 in section 4.2.1) for locating different letter clusters. PsyCOP integrates both the selective attention mechanism and the HT technique in its architecture. These five application-specific systems incorporate the invariance properties to different extents. For example, the neocognitron and MORSEL do not take care of orientation invariant recognition. Similarly, PsyCOP, in its present form, is not able to perform scale-invariant recognitron.

As far as the capability of decision making is concerned, the task of simultaneous recognition of multiple objects has not been considered in most of the systems (except for MORSEL and PsyCOP). Although TRAFFIC is able to take care of multiple objects, in a limited sense, because it is not able to recognize multiple instances of the same object. MORSEL not only considers the psychological findings (e.g. selective attention)

Table 1. Comparison of the characteristics of application-specific systems

Characteristics	Neocognitron	DHT model	TRAFFIC	MORSEL	PsyCOP
Object type	Numerals, characters	Alphabetical characters	Astral constellation	Words	Industrial objects
Feature type	Pixel	Strokes, junctions	Geometric features	Line segments	Corners
Translation invariance	Yes	Yes	Yes	Yes	Yes
Rotation invariance	No	Yes	Yes	No	Yes
Scale invariance	Yes	No	Yes	No	No
Multiple objects	No	?	Yes	Yes	Yes
Multiple instances (same object)	No	?	No	Yes	Yes
Control	BU/TD	BU/TD	BU	BU/TD	BU/TD
Rigid transformation	No	Yes	Yes	No	Yes
Learning	Supervised	—	Supervised	Supervised	Supervised
Selective attention	Yes	No	No	Yes	Yes
Psychological evidence	Yes	No	No	Yes	Yes
Psychological explanation	?	Yes	No	Yes	?

but also interprets some of the disorders of psychological patients. PsyCOP has been essentially developed on the basis of the psychological finding that identification and localization occur in two separate zones of the visual cortex.

The different characteristics of these systems are summarized in table 1. In this table, the question marks indicate questionable performance. For example, in the task of multiple object recognition, the DHT model faces problems of finding one object in some other object's location if the parameters of the network are not properly tuned. In the table, BU indicates a bottom-up control while TD indicates a top-down control. The neocognitron was originally designed only with a bottom-up control, but later top-down control has been incorporated in order to identify the most prominent one from a mixture of patterns. TRAFFIC uses only a bottom-up process (i.e. the verification from object layer to the feature layer is absent), while the DHT model, MORSEL and PsyCOP use both bottom-up and top-down processes. In the learning process, the neocognitron was trained only with supervised mode, but there is a theoretical formulation for unsupervised learning also. The question marks in the psychological explanation box indicate that it is not clear whether any psychological phenomenon can be explained or not with these models.

6. Conclusions and discussion

An overview of different methods and methodologies in connectionist approaches for object recognition is presented. The different classical algorithms for 2D object recognition are also briefly discussed. Here we have categorized the various methods in two groups depending on whether they are based on basic neural network models or application-specific models. One may consider some other criterion as the basis for categorization. The methodologies (categories) are further partitioned at the next level depending on their underlying assumptions and the types of recognition problem that they handle. The general principles and key features of these categories together with their learning mechanisms are mentioned. A comparison of the characteristics of five application-specific systems under category 2 is provided.

Although neural networks have several advantages, they find some limitations and/or problems in dealing with the tasks of real-life object recognition. Some of them are mentioned below.

(1) *Architecture—selection of the proper architecture for decision making.* Different architectures have been proposed but, for a specific problem, it is difficult to decide which particular architecture would perform best. The neural architectures, in most cases, have been proposed with different objectives or applications in mind, and it is very hard to describe them in a single computational framework.

(2) *Mapping—proper mapping of the features and subparts on to the network with their spatial distribution.* It is always necessary to transform a real-life task properly in terms of the variables acceptable by the computational model of a neural network. Sometimes it may be a rather difficult problem.

(3) *Binding—proper representation of the identity and location together and proper representaiton of the context.* For object recognition, it is necessary to represent the information on 'what' and 'where' of a subpart or a feature (or sometimes the entire object) together, but efficient design of such representation may be difficult. Moreover, in some cases the interpretation of the subparts (features) may change depending on the context. For example, a particular shape or object can have different meanings in different contexts. Representation of this kind of knowledge is still found to be difficult in the connectionist framework.

(4) *Hardware—design of the hardware for real-time performance.* Massive parallelism, a characteristic feature of ANNs, will become useful only when these architectures have suitable hardware realizations or at least can be efficiently simulated on a parallel machine. The main bottleneck of neural network hardware design is the implementation of variable synaptic connections. During the learning phase, neural networks are supposed to change their weights frequently, which is difficult to realize in the available hardware systems.

Appendix A. Mixed category perception: principle and models

The networks dealing with problems such as classification, self-organization and content addressability accept patterns from a single category at a time while performing these operations. However, in many situations, it may be necessary to perceive mixed categories (i.e. more than one category overlapping each other) simultaneously. Such cases may arise in the problem of recognizing multiple objects in a scene at a time or perceiving music coming from more than one source or even in the prediction of disorders from the symptoms of a patient. Such tasks of simultaneous recognition of multiple entities are referred to as mixed category perception (MCP). Here we outline the principle of MCP, various models and their characteristic differences in brief.

A.1. *Principle of mixed category perception*

Ideally (under noiseless conditions), the problem of MCP can be described as follows. Let m different objects O, characterized by the collections of different features f, be represented as

$$O_1 = \{f_{11}, \dots, f_{1n}\},$$
$$\vdots$$
$$O_m = \{f_{m1}, \dots, f_{mn}\}.$$

Let a new feature vector $F_k = \{f_1, \dots, f_N\}$ be formed by the superposition of k different objects, that is

$$F_k = \bigcup_{i \in \{i_1, \dots, i_k\} \subseteq \{1, \dots, m\}} O_i.$$

Then the task of MCP is to identify a set of k objects $\{O_{j1}, O_{j2}, \dots, O_{jk}\}$ such that

$$\bigcup_{j \in \{j_1, \dots, j_k\} \subseteq \{1, \dots, m\}} O_j = F_k.$$

Note that it is desirable to find exactly the same set of k objects which were superposed to form the feature F_k. However, this is possible if F_k results from a unique combination of k objects. Otherwise, another set of k objects may be identified which, when superposed, would result in the same F_k. Thus, MCP appears to be equivalent to the 'set covering' problem.

Several neural network models have been developed in order to perceive mixed categories simultaneously. Sometimes these kinds of networks have been designed to segment properly the input pattern; for example in speech perception a word is often found to be a conjunction of more than one word. Whatever the objective, these networks have a common thread between them which is essentially to understand input patterns, generated by the superposition of more than one pattern, in terms of the stored exemplar patterns. Although, many models in this category have been developed for various purposes other than object recognition, they have their validity in this domain also.

A.2. *Models for mixed category perception*

A model was developed by Peng and Reggia (1989) for the prediction of multiple disorders for a given set of mainfestations. It is a two-layered model where the input layer, representing the manifestations, is connected to the output layer, representing the disorders, though feedforward and feedback connections. The weights of the links were pre-assigned on the basis of the probability of co-occurrence of manifestations and disorders. The output values are updated through feedforward and feedback activations. Although no learning rules have been specified for this model, the structure has been developed on the basis of rigorous mathematical modeling of the problem. Later Cho and Reggia (1993) and Reggia *et al.* (1992) developed a learning scheme for automatically assigning these weights though a competition and cooperation process. The competitive and cooperative process was mathematically formulated and an error back-propagation learning rule was derived. However, the learning rule operates in supervised model only.

In EXIN (an acronym for excitatory and inhibitory connections), developed by Marshall (1990 a, b, c 1992), which is able to perform MCP, a new way of competition between the nodes was introduced. In a WTA network, all nodes inhibit each other equally (i.e. the strengths of inhibitory links between the pairs of neaurons are the same over all the network). In EXIN, the inhibitory strengths are learned, rather than assigning some pre-fixed values. The strengths are updated in such a way that the competition process is confined within nodes of similar nature, that is getting activations from inputs which have sufficient overlap. This is performed by strengthening the connection weights between coactivated neurons and weakening the connections to inactive neurons. This enables the network to achieve a limited form of MCP. The network is able to tell whether the superposed input patterns have little overlap between them. However, the network is able to segment an input pattern with the help of learned exemplars.

Cohen and Grossberg (1986, 1987) developed a massively parallel network, called a masking field, which is essentially a self-similar gain-controlled cooperative-competitive feedback network. It adaptively sharpens the coding property with the repetitive presentations of a patttern at the input. It is able to detect multiple groupings and to code them for some pattern flickering at the input. However, the issues of stability has not been mathematically dealt with in this network.

Another self-organizing neural network (SONNET) was developed by Nigrin (1990 a, b, c, 1992) for MCP. It also consists of two different layers, namely input and output. In the output layer, nodes responding to similar input patterns compete between themselves, which is essentially a similar concept to that used in EXIN. However, SONNET has a novel property of forming stable codes for embedded patterns. If a subpattern occurs in different input patterns quite frequently, than the links from the input to an output node representing the code for the subpattern are strengthened. Additional care has been taken to achieve stability in the code formation. Although the model has originated from speech category perception, its principle has been applied for building up an architecture to recognize one-dimensional patterns with translation invariance.

A three-layered connectionist model was proposed (Basak *et al.* 1992, 1993 b, 1996, Basak and Pal 1995 b) for simultaneous recognition of multiple categories or objects. Later, this model was named X-tron. The input layer accepts numerical values representing the degree of confidence about the presence of the features. The output layer produces the degree of confidence about the presence of categories or objects, and the hidden layer corresponds to the feature–object associations. In X-tron, instead of using a selective form of competition (as in EXIN or SONNET), the competition process is separated from the output layer and is confined in the hidden layer. The learning rules for X-tron are defined in such a way that the weights of the links asymptotically reach some predefined probabilistic measures. During learning, the network automatically adjusts the number of nodes in the hidden layer. X-tron is able to learn under both supervised and unsupervised modes.

A.3. *Comparison of characteristics*

The three models for mixed category perception, namely EXIN, SONNET and X-tron, have been developed based on the principle of ART. However, in ART, the network is able to self-organize only when the input pattern corresponds to single or individual categories. The principle has been suitably modified in these three models to take care of mixed category perception. These are summarized below.

(1) EXIN and SONNET were designed to segment properly the temporal patterns. X-tron, on the other hand, was motivated from simultaneous recognition of multiple visual objects.

(2) EXIN and SONNET achieve MCP, that is coexistence of more than one activated output node under a stable condition by restricting the competition process within similar output nodes. The word 'similar' indicates the fact that the nodes would compete if there exists some overlap between the corresponding feature set. This is taken care of in the learning process itself. On the other hand, X-tron achieves MCP by separating the competition process from the output layer. It employs another layer, namely a hidden layer, to represent the feature–object associations. In this layer, the objects compete between themselves for winning over a feature. Therefore, for different features, different objects may be flagged as winners, and, as a result, multiple winners would coexist in the output layer since they do not compete directly between themselves.

(3) In ART, scale sensitivity is achieved using Weber's law. The word 'scale sensitivity' means that, if two feature sets O_A and O_B of two different objects A and B are such that $O_A \subset O_B$ and if O_B is presented to the network, then it would be able to detect B only and not A. Similarly, if O_A is presented, then the network would be able to detect A only. EXIN was implemented using Weber's law. Then it was modified with the help of scale-sensitive cells, that is the cells corresponding to larger patterns would produce more inhibition to other cells. SONNET employs a scale-sensitive cells only. In X-tron, since the output cells do not inhibit each other directly, it operates only with Weber's law.

(4) Let all these three networks have learned exemplars *ab*, *abc*, *bcd*, *cd*. A new pattern *abcd* is presented to the network. In that case, it is most likely that EXIN and SONNET would activate th enodes for *ab* and *cd*, because *abc* and *bcd* would compete between themselves. On the other hand, in X-tron, the output nodes for *abc* and *bcd* would be activated. In other words, X-tron would possibly be able to detect mixed categories with a higher degree of overlap compared wih EXIN and SONNET, and this is because it degenerates the competition process from the output layer.

(5) One novel property of SONNET is that it can categorize embedded patterns. This means that, if a pattern appears to the network several times, embedded within larger patterns, then SONNET would form a stable category for it. EXIN and X-tron do not possess such capability.

Acknowledgments

This work is supported by the Council of Scientific and Industrial Research, India, under project grant 22(235)/93EMR-II.

References

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J., 1983, A learning algorithm for Bolzmann machines. *Cognitive Science*, **9**, 147–169.

Amari, S., 1972, Learning patterns and pattern sequences by self-organizing nets of threshold elements. *IEEE Transactions on Computers*, **21**, 1197–1206; 1977, Neural theory of association and concept formation. *Biological Cybernetics*, **26**, 175–185.

Amari, S., and Maginu, K., 1988, Statistical neurodynamics of associative memory. *Neural Networks*, **1**, 63–74.

Anderson, J. A., Silverstein, J. W., Ritz, S. R., and Jones, R. S., 1977, Distinctive features, categorical perception, and probability learning. *Psychological Review*, **84**, 413–451.

Ansari, N., and Delp, E. J., 1990, Partial shape recognition: a landmark-based approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, 470–483.

Ansari, N., and Li, K., 1993, Landmark-based shape recognition by a modified Hopfield neural network. *Pattern Recognition*, **26**, 531–542.

Arbuschi, A., Cantoni, V., and Musso, G., 1984, Recognition and localization of parts using the Hough transform technique. *Digital Image Processing*, edited by Levialdi (California: Pitman).

Ayache, N., and Faugeras, O. D., 1986, Hyper: a new approach for the recognition and positioning of two-dimensional objects. *IEEE Transactions on Patter Analysis and Machine Intelligence*, **8**, 44–54.

Ballard, D. H., 1981, Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, **13**, 111–122.

Ballard, D. H., and Brown, C. M., 1992, Principles of animate vision. *CVGIP: Image Understanding*, **56**, 3–21.

Basak, J., Chanda, B., and Dutta Majumder, D., 1994, On edge and line linking in graylevel images with connectionist models. *IEEE Transactions on Systems, Man, and Cybernetics*, **24**, 413–428.

Basak, J., Chaudhury, S., Pal, S. K., and Dutta Majumder, D., 1993 a, Matching of structural shape descriptions with Hopfield net. *International Journal of Pattern Recognition and Artificial Intelligence*, **7**, 377–404.

Basak, J., Murthy, C. A., Chaudhury, S., and Dutta Majumder, D., 1992, A connectionist network for simultaneous perception of multiple categories. *Proceedings of the 11th IAPR International Conference on Pattern Recognition*, The Hague, The Netherlands, pp. 36–40; 1993 b, A connectionist model for category perception: theory and implementation. *IEEE Transactions on Neural Networks*, **4**, 257–269.

Basak, J., Murthy, C. A., and Pal, S. K., 1996, A self-organizing connectionist model for mixed category perception. *Neurocomputing*, **10**, 341–358.

Basak, J., and Pal, S. K., 1995 a, PsyCOP: a psychologically motivated connectionist system for object perception. *IEEE Transactions on Neural Networks*, **6**, 1337–1354; 1995 b, X-tron: An incremental connectionist model for category perception. *IEEE Transactions on Neural Networks*, **6**, 1091–1108.

Bebis, G. N., and Papadourakis, G. M., 1992, Object recognition using invariant object boundary representations and neural network models. *Pattern Recognition*, **25**, 25–44.

Bedini, L., and Tonazzini, A., 1990, Neural network use in maximum entropy image restoration. *Image with Vision Computing*, **8**, 108–114.

Besl, P. J., and Jain, R. C., 1985, Three-dimensional object recognition. *ACM Computing Surveys*, **17**, 75–145.

Bhanu, B., and Faugeras, O. D., 1984, Shape matching of two dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 137–155.

Bhanu, B., and Ming, J. C., 1986, Clustering based recognition of occluded objects. *Proceedings of the Eighth International Conference on Pattern Recognition*, Paris, France, pp. 732–734.

Blanz, W. E., and Gish, S. L., 1991, A connectionist classifier architecture applied to image segmentation. *International Journal of Pattern Rocognition and Artificial Intelligence*, **5**, 603–617.

Bolles, R. C., and Cain, R. A., 1982, Recognizing and locating partially visible objects: the focus feature method. *International Journal of Robotics Research*, **1**, 36–61.

Bullock, B. L., 1978, The necessity for a theory of specialized vision. *Computer Vision Systems*, edited by A. R. Hanson and E. M. Riseman (New York: Academic Press).

Carpenter, G. A., and Grossberg, S., 1987 a, A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics and Image Processing*, **37**, 34–115; 1987 b, Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, **26**, 4919–4930; 1990, ART3: hierarchical search using chemical transmitters in self-organizing pattern recognition architectures. *Neural Networks*, **3**, 129–152.

Carpenter, G. A., Grossberg, S., and Rosen, D. B., 1991, Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system. *Neural Networks*, **4**, 493–504.

Chan, L.-W., 1992, Neural networks for collective translational invariant object recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, **6**, 143–156.

Chaudhury, S., 1989, Development of methodologies for recognition of partially obscured shapes. PhD thesis, Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur, India.

Chaudhury, S., Acharya, A., Subramian, S., and Parthasarathy, G., 1990, Recognition of occluded objects with heuristic search. *Pattern Recognition*, **23**, 617–635.

Chin, R. T., and Dyer, C. R., 1986, Model based recognition in robot vision. *ACM Computing Surveys*, **18**, 69–108.

Cho, S., and Reggia, J. A., 1993, Learning competition and cooperation. *Neural Computation*, **5**, 242–259.

Cohen, M. A., and Grossberg, S., 1986, Neural dynamics of speech and language coding: developmental programs, perceptual grouping, and competition for short-term memory. *Human Neurobiology*,

51, 1–22; 1987, Masking fields: a massively parallel nural architecture for learning, recognizing, and predicting multiple groupings of data. *Applied Optics*, **26**, 1866–1891.

Duda, R. O., and Hart, P. E., 1978, *Pattern Classification and Scene Analysis* (New York: Wiley).

Edelman, S., 1992, A network model of object recognition in human vision. *Neural Networks for Perception*, Vol. 1, edited by H. Wechsler (San Diego, California: Academic Press).

Edelman, S., and Weinshall, D., 1991, A self-organizing multipleview representation of 3-D objects. *Biological Cybernetics*, **64**, 209–219.

Eshera, M. A., and Fu, K. S., 1984, A similarity measure between attributed relational graphs for image analysis. *Proceedings of the Seventh International Conference on Pattern Recognition*, pp. 75–77.

Fahlman, S., and Lebiere, C., 1990, The cascade correlation learning architecture. *Advances in Neural Information Processing Systems*, Vol. 2, edited by D. Touretzky (San mateo, California: Morgan Kaufman).

Feldman, J. A., 1982, Dynamic connections in neural networks. *Biological Cybernetics*, **46**, 27–39; 1985, Four frames suffice: a provisional model of vision and space. *The Behavioral and Brain Sciences*, **8**, 265–289.

Feldmann, J. A., and Ballard, D. H., 1982, Connectionist models and their properties. *Cognitive Science*, **6**, 205–254.

Fu, K. S., 1982, *Syntactic Pattern Recognitiion and Applications* (Englewood Cliffs, New Jersey: Prentice-Hall).

Fukushima, K., 1975, Cognitron: a self-organizing multilayered neural network. *Biological Cybernetics*, **20**, 121–136; 1984, A hierarchical neural net model for associative memory. *Biological Cybernetics*, **50**, 105–113; 1987, Neural network model for selective attention in visual pattern recognition and associative recall. *Applied Optics*, **26**, 4985–4992; 1988a, Neocognitron: a hierarchical neural network capable of visual pattern recognition. *Neural Networks*, **1**, 119–130; 1988b, A neural network for visual pattern recognition. *IEEE Computer*, March 65–75; 1992, Character recognition with neural network. *Neurocomputing*, **4**, 221–233.

Fukushima, K., and Imagawa, T., 1993, Recognition and segmentation of connected characters with selective attention. *Neural Networks*, **6**, 33–41.

Fukushima, K., Imagawa, T., and Ashida, E., 1991, Character recognition with selective attention. *Proceedings of the International Conference on Neural Networks*, pp. I-593–I-598.

Fukushima, K., Miyake, S., and Ito, T., 1983, Neocognitron: a neural network model for a mechanism of visual pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, **13**, 826–834.

Gallant, S. I., 1988, Connectionist expert systems. *Communications of the ACM*, **31**, 152–169.

Ghosh, A., Pal, N. R., and Pal, S. K., 1991, Image segmentation using a neural network. *Biological Cybernetics*, **66**, 151–158; 1993, Self-organization for object-extraction using multilayer neural networks and fuzziness measure. *IEEE Transactions on Fuzzy Systems*, **1**, 54–68.

Gorman, J. W., Mitchell, O. R., and Kuhl, F. P., 1988, Partial shape recognition using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**, 257–266.

Grimson, W. E. L., 1989, On the recognition of curved objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**, 632–643.

Grimson, W. E. L., and Losano-Perez, T., 1985, Recognition and localization of over-lapping parts from sparse data in two and three dimensions. *Proceedings of the IEEE Conference on Robotics and Automation*, St Louis, Missouri (New York: IEEE), pp. 61–66.

Grossberg, S., 1982, *Studies of Mind and Brain* (Boston, Massachusetts: Reidel); 1987, Competitive learning: from interactive activation to adaptive resonance. *Cognitive Science*, **11**, 23–63.

Han, M. H., and Jang, D., 1990, The use of maximum curvature points for the recognition of partially occluded objects. *Pattern Recognition*, **23**, 21–33.

Henderson, T., and Samal, A., 1986, Multiconstraint shape analysis. *Image Vision Computing*, **4**, 84–96.

Hertz, J., Krogh, A., and Palmer, R. G., 1991, *Introduction to the Theory of Neural Computation* (Reading, Massachusetts: Addison-Wesley).

Himes, G. S., and Inigo, R. M., 1992, Automatic target recognition using a neocognitron. *IEEE Transactions on Knowledge and Data Engineering*, **4**, 167–172.

Hinton, G. E., 1981a, A parallel computation that assigns cannonical object-based frames of reference. *Proceedings of the International Joint Committee for Artificial Intelligence*; 1981b, Shape representation in parallel systems. *Proceedings of the International Joint Committee for Artificial Intelligence*; 1990, Mapping part-whole hierarchies into connectionist networks. *Artificial Intelligence*, **46**, 47–75.

Hinton, G. E., and Lang, K. J., 1985, Shape recognition and illusory conjunctions. *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, pp. 252–259.

Hirai, Y., and Tsukui, Y., 1990, Position independent pattern matching by neural network. *IEEE Transactions on Systems, Man, and Cybernetics*, **20**, 816–825.

Hochberg, J., 1987, Machines should not see as people do, but must know how people see. *Computer Vision, Graphics, and Image Processing*, **37**, 221–237.

Hopfield, J. J., 1982, Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences, USA*, 2554–2558; 1984, Neurons with graded response have collective computational properties like those of two state neurons. *Proceedings of the National Academy of Sciences, USA*, 3088–3092.

Hopfield, J. J., and Tank, D. W., 1985, Neural computation of decisions in optimization problems. *Biological Cybernetics*, **52**, 141–152; 1986, Computing with neural circuits: a model. *Science*, **233**, 625–633.

Hush, D. R., and Horne, B. G., 1993, Progress in supervised neural networks. *IEEE Signal Processing Magazine*, 8–39.

Illingworth, J., and Kittler, J., 1988, A survey of the Hough transform. *Computer Vision, Graphics, and Image Processing*, **44**, 87–116.

Jamison, T. A., and Schalkoff, R. J., 1988, Image labeling: a neural network approach. *Image and Vision Computing*, **6**, 203–213.

Kashyap, R. L., and Koch, M. W., 1985, Computer vision algorithms used in the recognition of objects. *Proceedings of the IEEE Conference on Robotics and Automation*, St Louis, Missouri, USA (New York: IEEE), pp. 150–155.

Knoll, T. F., and Jain, R. C., 1986, Recognizing partially visible objects using feature indexed hypotheses. *IEEE Journal of Robotics and Automation*, **2**, 3–13.

Kohonen, T., 1988, *Self-Organization and Associative Memory* (Berlin: Springer-Verlag).

Kosko, B., 1988, Bidirectional associative memories. *IEEE Transactions on Systems, Man, and Cybernetics*, **18**, 49–60; 1991, *Neural Networks and Fuzzy Systems: A Dynamical Approach to Machine Intelligence* (Englewood Cliffs, New Jersey: Prentice-Hall).

Kosslyn, S. M., 1975, Information representation in visual images. *Cognitive Psychology*, **7**, 341–370.

Kosslyn, S. M., Holtzman, J. D., Farab, M. J., and Gazzaniga, M. S., 1985, A computational analysis of mental image generation: evidence from functional dissociations in split-brain patients. *Journal of Experimental Psychology: General*, **114**, 311–341.

Li, W., and Nasrabadi, M., 1989, Object recognition based on graph matching implemented by a Hopfield style network. *Proceedings of the IEEE International Conference on Neural Networks, II*, Washington, DC, USA (New York: IEEE), pp. 287–290; 1993, Invariant object recognition based on a neural network of cascaded RCE nets. *International Journal of Pattern Recognition and Artificial Intelligence*, **7**, 815–829.

Lillo, W. E., Loh, M. H., Hui, S., and Zak, S. H., 1993, On solving constrained optimization problems with neural networks: a penalty method approach. *IEEE Transactions on Neural Networks*, **4**, 931–940.

Lin, W. C., Liao, F.-Y., and Lingutla, T., 1991, A hierarchical multiple-view approach to three-dimensional object recognition. *IEEE Transactions on Neural Networks*, **2**, 84–92.

Lu, S., and Szeto, A., 1993, Hierarchical artificial neural networks for edge enhancement. *Pattern Recognition*, **26**, 1149–1163.

Marr, D., 1982, *Vision* (San Francisco, California: W. H. Freeman).

Marshall, J., 1990 a, Self-organizing neural networks for perception of visual motion. *Neural Networks*, **3**, 45–74; 1990 b, Representation of uncertainty in self-organizing neural networks. *Proceedings of the International Conference on Neural Networks*, Paris, France, pp. 809–812; 1990 c, A self-organizing scale-sensitive network. *Proceedings of the International Joint Conference on Neural Networks*, San Diego, California, USA, pp. 649–654; 1992, Development of perceptual context-sensitivity in unsupervised neural networks: parsing, grouping and segmentation. *Proceedings of the International Neural Networks*, Baltimore, Maryland, USA, pp. 315–320.

Medioni, G., and Nevatia, R., 1984, Matching images using linear features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**, 675–685.

Miller, L. G., and Gorin, A. L., 1993, Structured networks for adaptive language acquisition. *International Journal of Pattern Recognition and Artificial Intelligence*, **7**, 873–898.

Minnix, J., McVey, E., and Inigo, R., 1992, A multilayered self-organizing artificial neural network for invariant pattern recognition. *IEEE Transactions on Knowledge and Data Engineering*, **4**, 162–167.

Minsky, M. L., and Papert, S. A., 1969, *Perceptrons* (Cambridge, Massachusetts: MIT Press).

Mitra, S., and Pal, S. K., 1994, Logical operation based fuzzy MLP for classification and rule generation. *Neural Networks*, **7**, 353–373.

Moody, J., and Darken, C., 1989, Fast learning in networks of locally-tuned processing units. *Neural Computation*, **1**, 281–294.

Mozer, M. C., 1991, *The Perception of Multiple Objects: A Connectionist Approach* (Cambridge, Massachusetts: MIT Press).

Mozer, M. C., and Behrmann, M., 1989, On the interaction of selective attention and lexical knowledge: a connectionist account of neglect dyslexia. Technical Report CU-CS-441-89, Department of Computer Science, University of Colorado, Boulder, Colorado, USA.

Nasrabadi, N. M., and Li, W., 1991, Object recognition by a Hopfield neural network. *IEEE Transactions on Systems, Man, and Cybernetics*, **21**, 1523.

Nigrin, A., 1990 a, Sonnet: a self-organizing neural network that classifies multiple patterns simultaneously. *Proceedings of the International Joint Conference on Neural Networks*, San Diego, California, USA, pp. 313–318; 1990 b, The stable classification of temporal sequences with an adaptive resonance circuit. *Proceedings of the International Joint Conference on Neural Networks*, Washington, DC, USA, pp. 525–528; 1990 c, The stable learning of temporal patterns with an adaptive resonance circuit. PhD Thesis, Duke University, Durham, North Carolina, USA; 1992, A new architecture for achieving translational invariant recognition of objects. *Proceedings of the International Joint Conference on Neural Networks*, Baltimore, Maryland, USA, pp. 683–688.

Pal, S. K., and Dutta Majumder, D., 1986, *Fuzzy Mathematical Approach to Pattern Recognition* (New York: Wiley).

Peng, Y., and Reggia, J. A., 1989, A connectionist model for diagnostic problem solving. *IEEE Transactions on Systems, Man, and Cybernetics*, **19**, 285–298.

Poggio, T., and Edelman, S., 1990, A network that learns to recognize three-dimensional objects. *Nature*, **343**, 263–266.

Poggio, T., Edelman, S., and Fahle, M., 1992, Learning of visual modules from examples: a framework for understanding adaptive visual performance. *CVGIP: Image Understanding*, **56**, 22–30.

Price, K. E., 1984, Matching closed contours. *Proceedings of the Seventh International Conference on Pattern Recognition*, Montreal, Canada, pp. 990-992.

Reed, R., 1993, Pruning algorithms—a survey. *IEEE Transactions Neural Networks*, **4**, 740–747.

Reggia, J. A., D'Autrechy, C. L., Sutton III, G. G., and Weinrich, M., 1992, A competitive distribution theory of neocortical dynamics. *Neural Computation*, **4**, 287–317.

Romaniuk, S. G., and Hall, L. O., 1993, Divide and conquer neural networks. *Neural Networks*, **6**, 1105–1116.

Rumelhart, D. E., and McClelland, J. L., 1986, *Parallel Distributed Processing: Explorations in Microstructures of Cognition*, Vol. I (Cambridge, Massachusetts: Bradford Books–MIT Press).

Rummel, P., and Beutel, W., 1984, Workpiece recognition and inspection by a model-based scene analysis system. *Pattern Recognition*, **17**, 141–148.

Sabbah, D., 1985, Computing with connections in visual recognition of origami objects. *Cognitive Science*, **9**, 25–50.

Segen, J., 1983, Locating randomly oriented objects from partial views. *Proceedings of the Third International Conference on Robot Vision and Sensory Control*, Cambridge, Massachusetts, USA, pp. 676–683.

Shapiro, L. G., and Haralick, R. M., 1982, Organization of relational models for scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **4**, 595–602; 1985, A metric for comparing relational descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **7**, 90–94.

Sietsma, J., and Dow, R. J., 1991, Creating artificial neural networks that generalize. *Neural Networks*, **4**, 67–79.

Simpson, P. K., 1990, Higher ordered and intraconnected bidirectional associative memories. *IEEE Transactions on Systems, Man, and Cybernetics*, **20**, 637–653.

Skrzypek, J., 1989, Neural specification of a general purpose vision system. Technical Report CSD-890072, Computer Science Department, University of California, Los Angeles, California, USA.

Spirkovska, L., and Reid, M. B., 1993, Coarse-coded higher-order neural networks for PSRI object recognition. *IEEE Transactions on Neural Networks*, **4**, 276–283.

Srinivasa, N., and Jouaneh, M., 1993, An invariant pattern recognition machine using a modified art architecture. *IEEE Transactions on Systems, Man, and Cybernetics*, **23**, 1432–1437.

Stockmann, G. C., and Agarawala, A. K., 1972, Equivalence of Hough curve detection to matched filtering. *Graphics, Image Processing*, **20**, 820–822.

Suetens, P., Fua, P., and Hanson, A. J., 1992, Computational strategies for object recognition. *ACM Computing Surveys*, **24**, 5–61.

Triesman, A. M., and Schmidt, H., 1982, Illusory conjunctions in the perception of objects. *Cognitive Psychology*, **14**, 107–141.

Trivedi, M. M., and Rosenfeld, A., 1989, On making computers 'see'. *IEEE Transactions on Systems, Man, and Cybernetics*, **19**, 1333–1335.

Tsang, P. W. M., and Yuen, P. C., 1993, Recognition of partially occluded objects. *IEEE Transactions on Systems, Man, and Cybernetics*, **23**.

Tsang, P. W. M., Yuen, P. C., and Lam, F. K., 1992, Recognition of occluded objects. *Pattern Recognition*, **25**, 1107–1117.

Tsao, T. R., and Chen, V., 1994, A neural scheme for optical flow computation based on Gabor filters and generalized gradient method. *Neurocomputing*, **6**, 305–325.

Turney, J. L., Mudge, T. N., and Volz, R. A., 1985, Recognizing partially occluded parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **7**, 410–421.

Ullmann, J. R., 1993, Edge replacement in the recognition of occluded objects. *Pattern Recognition*, **26**, 1771–1784.

Umeyama, S., 1993, Parametrized point pattern matching and its application to recognition of object families. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**, 136–144.

Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., and Lang, K., 1989, Phoneme recognition using time delay neural networks. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **37**, 328–340.

Wallace, A. M., 1985, Feature determination and inexact matching of images of industrial components. *Proceedings of IEEE*, **132E**, 309–315; 1987, An informed strategy for matching models to images of segmented scenes. *Pattern Recognition*, **20**, 349–363; 1988, A comparison of approaches to high-level image interpretation. *Pattern Recognitiion*, **21**, 241–259.

Wechsler, H., and Zimmerman, G. L., 1988, 2-d invariant object recognition using distributed associative memories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **10**, 811–821.

Wen, W., and Lozzi, A., 1992, Recognition and inspection of two-dimensional industrial parts using subpolygons. *Pattern Recognition*, **25**, 1427–1434.

Zemel, R. S., 1989, TRAFFIC: a connectionist model of object recognition. Technical Report CRG-TR-89-2, Department of Computer Science, University of Toronto, Canada.

Zhao, F., 1991, Machine recognition as representation and search—a survey. *International Journal of Pattern Recognition and Artificial Intelligence*, **5**, 715–747.

Zimmerman, G. L., 1992, Two-dimensional maps and biological vision: representing three-dimensional space. *Neural Networks for Perception*, Vol. 1, edited by H. Wechsler (San Diego, California: Academic Press).