DESCRIPTION AND SYNTHESIS

IN

PATTERN RECOGNITION

BY

REKHA RANI

A thesis submitted in partial fulfilment

for the Award of the Degree of

MASTER OF PHILOSOPHY

SCHOOL OF COMPUTER AND SYSTEMS SCIENCES

JAWAHARLAL NEHRU UNIVERSITY

NEW DELHI 110 057

1978

SCHOOL OF COMPUTER AND SYSTEMS SCIENCES
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI-110057

The research work embodied in this dissertation has been carried out by Miss Rekha Rani at the school of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi - 110057.

This work is original and has not been submitted in part or full for any other degree or diploma of any other University.

R. Sadananda
DR. R. SADANANDA
SUPERVISOR

N. P. Mukherjee
DR. N.P.MUKERJEE
INCHARGE

SCHOOL OF COMPUTER AND SYSTEMS SCIENCES
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI - 110057.

ACKNOWLEDGEMENT

Working with Dr. R. Sadananda, my dissertation supervisor, has been a rewarding experience. The willingness and cooperation that he has extended at all times has been a constant source of encouragement to me. I also thank him for his patience and the guidance he has given during the course of this dissertation.

My gratitude is also due to Dr. N.P. Mukerjee, In-charge, School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, for not only providing all possible help and facilities, but also for his interest and enthusiasm in the present work.

I thank my colleagues and the technical staff of the Computer Centre, for having helped me at various stages of this dissertation.

CONTENTS

			Page
		ACKNOWLEDGEMENT	i
		CONTENTS	ii
		SYNOPSIS	iv
CHAPTER	1		
	1.1	Artificial Intelligence	1
	1.2	Computers and Psychology	3
	1.3	Scene Analysis and Computers	4
	1.4	Scope of Present Work	5
CHAPTER	2		
	2.1	Pattern	7
	2.2	Approaches to Pattern Recognition	8
		2.2.1 Statistical Decision Theory	8
		2.2.2 Syntactic Approach	11
	2.3	Linguistic Methods and the Present work	12
	2.4	Grammar Controlled Pattern Analyser	13

CONTENTS (CONTD.)

Page

CHAPTER 3

3.1 The Image Former 15

3.2 The System ANS 17

3.2.1 I_DATA 19

3.2.2 TEE 19

3.2.3 HID_L 21

3.2.4 ST_LN 21

3.2.5 TR_RN 23

3.2.6 QU_RN 23

CHAPTER 4

4.1 The Domain of ANS 26

4.2 The Weight System 27

4.3 Learning 27

4.4 Future Trends 29

APPENDIX A

A.1 IDATA 30

A.2 TEE 33

A.3 HID_L 35

A.4 ST_LN 35

A.5 TR_RN 35

A.6 QU_RN 36

APPENDIX B 37

BIBLIOGRAPHY 38

SYNOPSIS

Since recognition of patterns is intrinsic to intelligence, the study of pattern recognition as an independent discipline has gained a lot of importance in the past few years. Research and development in pattern recognition has been mainly along two general paths - decision theoretic approach and the syntactic approach. In the decision theoretic approach, a set of characteristic features are extracted from the patterns and a pattern is transformed to a vector in the feature space. Recognition of each pattern is usually done by partitioning the feature space into the pattern class and assigning the input pattern to a particular pattern class on the basis of the value of its feature vector. In the syntactic approach, the pattern primitives are selected and their relations in the pattern are described by a set of syntactic rules. The result is a structured description of the input. Both approaches have led to a number of useful results of theoretical and practical importance. The choice of the approach to be followed depends on the kind of patterns to be recognised. It has now been realised that a good pattern recognition system uses decision theoretic, syntactical and heuristic methods at various stages.

Since many problems of pattern recognition are concerned with picture analysis and description, description schemes based on a syntax are very useful and powerful methods of solving such problems. One of the areas of picture analysis and description where descriptive methods should prove to be useful is in recognition of partially occluded objects. With a powerful descriptive apparatus, it is possible to synthesise the objects into recognisable wholes in a Gestalt sense, once the scene is subjected to a descriptive analysis.

In this dissertation an attempt has been made to use descriptive schemas to find out the hidden lines in a pattern and on the principles of Gestalt psychology to determine the shapes of figures, some of which are hidden by others.

The input pattern after pre-processing is reduced to a labelled sketch consisting of straight line edges. The relevant information pertaining to the points and lines under consideration is also extracted. On the basis of this information and the grammer supplied to this system, it determines which may be the possible hidden lines, i.e., lines which are overlapped by another figure. The system proceeds by first determining the kind of joints at various vertices, i.e., how many lines meet at a point and their configuration with respect to each other. It then finds out which of these line segments may be parts of a larger line, a part of which is hidden by another figure. On the basis of this information and principles of Gestalt psychology, the two segments are treated as one straight line. Using different types of syntactic rules, depending on the shape of the geometric figure to be recognised, the system attempts to recognise the figures according to their shapes.

Different kinds of geometric figures have different sets of syntactic rules for their recognition. By finding out which sets

of production rules or descriptions are satisfied by points in the pattern, the system attempts to recognise the various figures as belonging to different categories, typified according to their shape. Some possible extensions of the system are also suggested.

CHAPTER 1

The human brain is a very complex entity and little is known about its internal mechanisms and organisation. Any attempt at simulating its activities in the hope of understanding it can be of some success only if we have devices which can not only store huge amounts of information but also organise and process this information at very high speeds. But it can not be concluded that these achievements in hardware alone can pave the way for understanding the neurological mechanisms and organisation. Perhaps, more significant may be the role of software development. We may need a software technology to handle vast amounts of data and methods to handle with ease combinatorially explosive problems - a performance of ordinary importance to biological beings.

The rapid development of computers from the fifties onwards in terms of memory size, speed and versatility in terms of input-output devices, led to the use of computers for a wide variety of applications. It was no longer just a computing machine which could do complex computations at incredible speeds. Programs were written which attempted to make the computer "behave intelligently" and do such functions as theorem proofs, solve analogy tests, understand natural languages or play games (8, 13, 14, 23, 24, 28, 29, 30, 38,39).

1.1 Artificial Intelligence

A. Turing in his classic paper (43) argues that if a machine and an intelligent being are posed with the same set of questions, the answering of which requires intelligence and they cannot be distinguished from their answers, then the machine is entitled to be called intelligent. It does not matter what strategy the machine applies to solve this problem. The advocates of this approach argue that though computers can only do what they are programmed to do, humans can also learn, think and create because bilogically they have been programmed to do so (2, 9, 43). Furthermore, this biological program is modified by interaction with the environment, similarly if a computer is programmed to behave intelligently, it can do so. Clearly this program should be able to analyse its own performance, and enhance its own effectiveness. It is quite possible that this program may exhibit a behaviour more intelligent than it was programmed for, especially, since with the complexity of information processing possible in the problem-solving-learning machine, it is not always possible to predict the behaviour of the program. The task becomes even more complex when there are a large number of information processing operations and the behaviour of these operations after their interaction with the task environment has to be predicted (9).

Since high speed computers are available, it may appear that problems involving intelligence can be attempted by invoking exhaustive search processees. But it can be shown that the number of steps involved even in simple problems are astronomical. Not only it is impossible to meet this challenge of combinatorial explosion now, it does not appear that it would be possible even in the foreseeable future if one takes into account the technological trends in hardware. The clue to intelligence is a highly selective search of the possible alternatives to a given problem in such a way that this leads to paths with solutions. Therefore, a machine if it is to be labelled intelligent should do a very drastic pruning of the possible solution trees and choose paths which lead to solutions.

Some other problems that have to be tackled in the process of programming computers to be intelligent is how is one to make computer learn new heuristics? Should these heuristics be special-purpose with low failure rate or should they be general-purpose with a wide variety of applications? But since our knowledge of learning mechanisms is rudimentary, the answers to these questions are only partly known.

1.2 Computers and Psychology

One of the approaches to making machines intelligent is to base programs on principles of human learning and behaviour. In this way the descriptions of thought-processes can be turned into models of machines or the design of programs. Pioneering work has been done in this field among others by Newell, Simon, Shaw (28, 29, 30, 38, 39).

Perceptual grouping appears to be the basis of human visual organisation. It is responsible for the formation of wholes from parts. Objects formed from parts of scenes may themselves be grouped. Thus perceptual grouping yields a heirarchy of parts and sub-parts (18). To automate this kind of perceputal grouping, it was necessary to write programs in such a way that results from various local feature detectors or elements were combined into larger units such as parts or even complete objects. In a general case the number of such objects is very large and in the assembly stage of the larger object, all the elements may not even be present, while there may be a large number of extraneous ones. To try out all the combinations becomes impossible in practice (21). Biological information processing systems seem to overcome this problem by having evolved a set of processing strategies according to which the individual features are to be combined. Such a procedure naturally reduces the number of possible combinations that need to be compared against some model.

As Kanal and Chandrashekhran (19) point out "long before the emergence of Pattern Recognition which is an important aspect of all activities to which the appellation "intelligent" can be applied psychologist had concerned themselves with the phenomena of perception and concept formation. Patterns, after all, cannot be recognised without the concept of patterns. Therefore, the belief that if complex pattern recognition problems can be solved, a giant stride towards modelling human intelligence would be on the way."

The Gestalt school of psychologist after considerable experimental and theoretical research, have listed some basic principles of perception. While one may not be able to estimate the clinical relevance of these concepts, the Gestalt principles do successfully explain many of the questions relating to perception. Many successful programs in the area of visual pattern recognition have been designed and developed taking into account Gestalt principles.

1.3 Scene Analysis and Computers

One area where machines have been programmed to exhibit intelligent behaviour is the area of scene analysis. A central problem in the area of scene analysis is that of segmenting a scene into scenes which correspond to natural objects. Let us define this visual process as segmenting or grouping. In recent years several efforts have been directed towards the study of physical cues for grouping and computer methods which are effective for a particular type of scene which involve a particular cue (49).

Pioneering work has been done by Roberts (32) in this field. His program is capable of building up an internal representation of a relatively simple configuration of solid objects given a perspective view as input. This description can then be manipulated to some other configuration as hypothesized by the program, from a changed perspective.

Guzman (15) in his doctoral thesis worked successfully on a program which finds bodies in scenes, formed by three-dimensional objects. Some of these bodies may be partially hidden from view. Guzman's program SEE analysed the program in terms of vertices, lines and surfaces. The primitive cues in his program are various types of joints eg. a T joint indicates an edge of one block disappearing into another, a Y joint indicates a three way corner and so on. Other such edge and vertices cues give similar implications. This program then combines all the implications to decide which faces form the blocks.

1.4 Scope of Present Work

The aim of this system is to build up and classify geometric figures from the data given to the system in terms of vertices and lines. The system deals with figures in two dimensions in the Euclidean plane. Some of these figures may be partially occluded by other figures. It recognises the visible parts of the occluded figures and by involving Gestalt principles generates the occluded parts. It then integrates the two to make the decomposition of the scene possible. The system deals with straight line figures like triangles and quadrilaterals.

CHAPTER 2

2.1 Pattern

A pattern may be regarded (18) as

1) any arbitrary ordered set of numbers representing particular values of a finite number of variables. Each arbitrary set is given a label which thereby defines its class, with the result that two such sets might be associated with a particular class although there may be no obvious similarity between the two sets.

Or we could also define it as

2) an ordered set of numbers representing a particular value of a finite number of variables in which there are certain definable relationships between the numbers in the set. Two such sets would be given the same classifying label if they are observed to conform to the same definable relationships.

According to these view points we can have the following types of patterns

1) Random Patterns being patterns of the type described in definition 1.

2) Point Patterns in which the essential quality of the pattern could be represented by a set of points distributed with a relative orientation in the input field (15).

3) Texture Patterns in which there is a repetition of well defined groups across the input field (although the group itself may be a random pattern). It is the presence of repetitions which is important here.

4) Line Patterns in which the essential quality of the pattern could be represented by a system of zero-width lines.

5) Area Patterns in which the essential quality of the pattern can be represented by a system of areas.

2.2 Approaches to Pattern-Recognition

The study of Pattern-Recognition as an independent discipline is rather recent. Research and development in Pattern Recognition has been mainly along two broad paths:

1. The Statistical Decision Theory
2. The Syntactic Approach.

2.2.1 Statistical Decision Theory

The statistical decision theory provides one of the most powerful (5, 19, 20, 26, 41) mathematical tools to pattern recognition. A mathematical theory of pattern recognition is concerned with classification of unknowns into one of several categories.

A suitable model on which such a theory can be built is n dimensional Euclidean space in which members of categories and the unknowns are represented as points. For any problem, each members is described by a finite member of measurable properties, each of which (property) is represented as a coordinate in n dimensional space. The 'learning' process is the development of a specific classification procedure from samples from the categories; the 'recognition' process is the use of that procedure for classification of an unknown.

Pattern Recognition problem is concerned with determining into which of a number of categories the unknown best fits. The selection of categories reflects the subjective notion of the experimenter. The classification procedure at best can be a probabilistic one, i.e., one can only say that the input pattern has a certain probability of belonging to a particular category.

A mathematical theory on a n space model (6, 10, 11, 20, 26) cannot ensure that these techniques are adequate for any representation. Data from the real world must be properly represented in n space. The techniques will be effective in classification only to the extent that properties measured adequately typify the various categories of interest. A n space representation can correspond to data described by a finite number of distinct properties. In this theory one implicitly assumes that each category has associated with it a multivariate probability distribution in n space describing the distribution of members of that category. Category distributions if incompletely known initially are estimated from samples and it is expected that this determination can be improved for larger number of samples and for greater prior knowledge of forms of these distributions.

The classification procedure then consists in effect of a division of the space into K regions each associated with a category. An unknown is classified into the category corresponding to the region into which it falls. The decision procedure partitions the space into disjoint regions. Given an unknown point x, and its probability distribution over the categories, then this point x can be associated to any category with a corresponding probability for each such assignment. But we are interested in specifically assigning the unknown into one category only and for this generally threshold procedures are used (6, 20, 26).

Searching for the 'best' decision procedure is a very subjective process and depends solely on what values the experimenter wishes to attach to various effects caused by errors of classification. Threshold classification result in errors and we attempt to find decision rules which minimise some aspects of these errors. But once again the particular aspects chosen reflect the subjective nature of the experimenter (10, 11, 20, 26).

But there are certain inherent inadequacies in this general methodology (26) because in dealing with pictures or class of pictures the really relevant and significant problems are concerned with their generation and description. Since pattern recognition is the problem of pattern analysis and description (26), the frame work coping with general recognition problem should be capable of analysing the input picture and generating a structured description of it. Recognition techniques based on structural description of the input picture is a more comprehensive scheme.

A typical example of this kind of approach is scene analysis where the picture to be analysed is quite complex and the number of features required is often very large. The statistical decision theory does not appear very appealing here. When the patterns are complex, and when the numbers of possible descriptions is very large, it is impractical to regard each description as defining a class. Then the requirement of recognition can only be satisfied only by a description of each pattern rather than task classification (10, 11).

2.2.2 Syntactic Approach

The syntactical approach draws analogy between the heirarchical structure of patterns and the syntax of languages (10, 11, 26). A class of patterns is defined as satisfying a certain set of relationships between suitably defined primitives. The relationships may be boolean expressions or might be specifiable by a generative grammar. The primitives themselves are left undefined within the model. The structural relations specify how the primitives are juxtaposed to qualify as a certain object. Obviously for the success of this approach, the primitives themselves should be more easily recognisable than the pattern itself. After each primitive within the pattern is specified the recognition process is accomplished by performing a syntax analysis or parsing of the sentence describing the given pattern to determine whether it is syntactically correct with respect to the specified grammar. The syntax analysis also produces a structural description of the sentence representing the given pattern. One very attractive aspect of this approach is that sometimes it is possible to design a recursive grammar which may lead to a very compact representation of even complex and lengthy description. The major problem that arises in this approach is the design of the grammer. What are the primitives to be used and how the production rules are to be structured are questions of prime importance. Moreover, the grammar should be such that it generates all descriptions of a particular kind as required, and should not generate any extraneous object. In practice this may become a difficult objective to achieve.

2.2.2.1 Semantics

When we are dealing with pictures, the interpretation involves a further syntactic structure of the event pictorially depicted - this constitutes the semantics of the picture. Given a pictorial expression there is a multiplicity of possible picture syntactic descriptions. Such multiplicity would present an impossible task to the syntax directed parser. If, however, the parser is directed not only by the picture grammar but also by the necessity to recover well formed descriptions with respect to a certain frame of reference it is plausible that a variety of potentially assignable descriptions can be drastically reduced (26).

The key to pattern-recognition problem does not lie wholly in statistical approaches or heuristics or syntactical approaches. Rather a good pattern recognition scheme uses all these approaches at various levels of preprocessing with each tool being applied where it is most optimum (12,20). Among others pioneering work has been done in this field by Narsimhan (26,27), Evans (6,7), Guzman (15,16), Clowes (3), Rosenfeld (35), Winston (46, 47, 48).

2.3 Linguist Methods and the Present Work

Since scene analysis is an area where the pictures to be analysed are quite complex and the number of features required for classification are very large, syntactic methods for recognition are more appropriate than statistical decision procedures.

Moreover, since, for a recognition procedure to be linguistic it is not necessary that it should explicitly formulate expressions in formal language and process it to reach the classification decision. All that is necessary is that recognition of pattern be based on the presence and proper juxtaposition of specific elements of pattern (6, 20).

Keeping this definition of a syntactical procedure and principles of Gestalt psychology in mind (21), a grammar controlled pattern analyser is formulated for a class of pictures consisting of triangles and quadrilaterals.

2.4 Grammar Controlled Pattern Analyser

A Grammar is defined as a finite set of production rules. Each rule (6) consists of four parts

(a) the name of the syntactic quantity being defined

(b) either one list (A,B,C) or a list of lists say (A,B),(CC,D) of names of constituents to be used in (c) and (d) below.

(c) A predicate in terms of constituents that must be satisfied if they are to constitute an object of the type being defined

(d) an expression that is used only if predicate in (c) returns true in which case a new object has been found.

A grammar may be recursive and there may be as many alternate definitions of a syntactic type as desired. The inputs required by the analysis (6) program consists of

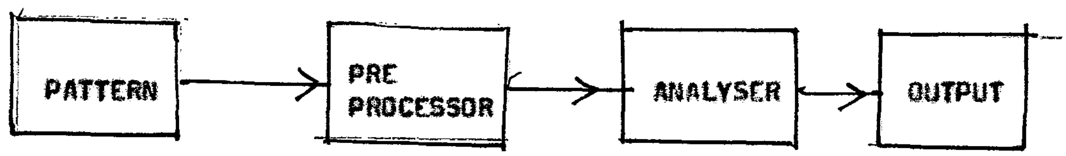
(a) a grammar

(b) an input pattern in the form of lowest level constituents with any desired information attached to them for later use in analysis process.

An extensive preprocessing may be required to transform the input pattern into the form required by the analyser. The demarcation between the functions of the preprocessor and the functions of the analyser is arbitrary depending on the kind of grammar, form of input etc.

The grammar in our case consists of various modules in the system ANS which defines the rules for occurrences of various syntactic types. As Narsimhan (26) points out "this descriptive schema should be such that it must assign structure to a scene in terms of occurrences of specific objects and their spatial distribution. It must also assign structures to an object in terms of occurrences of sub-parts and their composition. Assign structures to sub-parts in terms of more primitive entities and their composition. Thus given a picture one should be able to generate descriptions of the following sort - in the given picture such and such objects occur with such and such configuration. An object itself would be described as being composed in a specified manner of such and such sub-parts with specified parts".

The grammar controlled pattern analyser discussed above and in the following chapter is designed keeping the above in consideration.



CHAPTER 3

Many visual scenes contain objects which are partially hidden from view. From the portion of the object that is visible one can generally guess what the object as a whole is (21), eg. in Fig. 3.1, the object B will be recognised as a "house", even though a part of it is hidden from view.

In this dissertation an attempt has been made to design a system which overcomes the difficulties arising from the existence of hidden and overlap lines in scene analysis. Only the simplest cases of hidden and overlap lines pertaining to straight line figures are considered below as a first step towards the analysis of scenes consisting of straight line figures and objects some of which may be partially occluded by others.

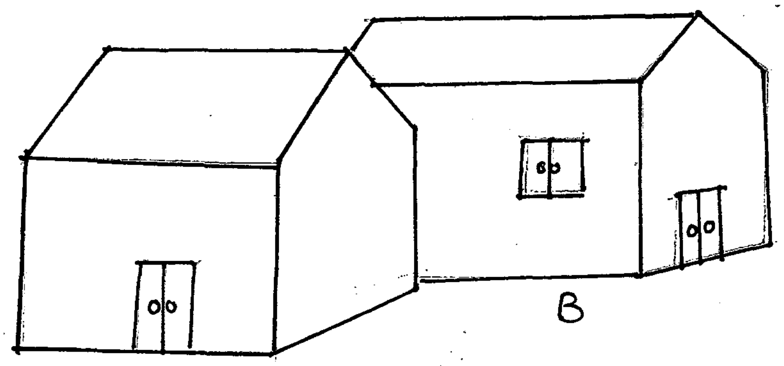
It has been pointed out by Evans (7) "among the problems of picture analysis with which syntactic analysis techniques can help are difficulties of overlap and hidden lines."

The system consists of two sub-systems. The main system that handles problems of partial occlusion is named ANS (ANALYSIS AND SYNTHESIS). The input to this system is from another system IFORM (IMAGE FORMER) which preprocesses the pictorial objects into a format acceptable to ANS.

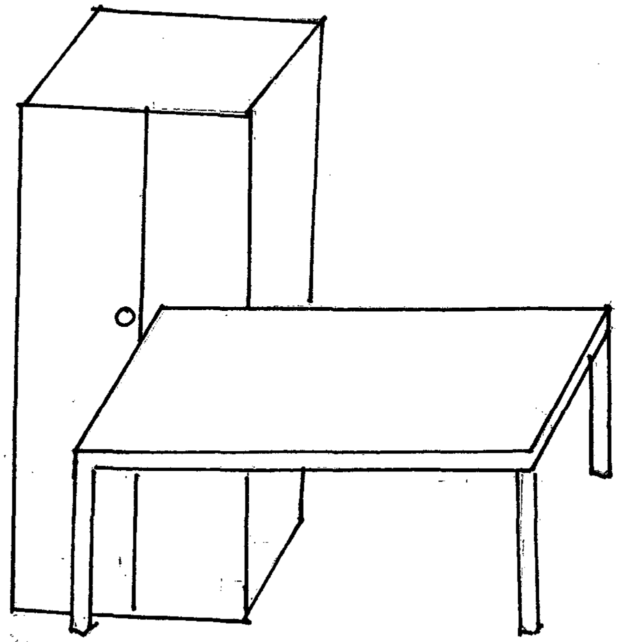
3.1 The Image Former (IFORM)

In this phase of the implementation the picture is converted to two dimensional multilevel image. There are a variety of techniques for doing this (44), specifically a raster may be used. The input to

FIGURE 3.1



AN EXAMPLE OF ONE FIGURE OVERLAPPING ANOTHER



ANOTHER REAL-LIFE SITUATION OF ONE FIGURE OVERLAPPING ANOTHER

IFORM is a picture. First this picture is converted to a two dimensional array of numbers, each number representing an intensity level. With the help of Boolean threshold functions, noise, cleaning smoothing, gap filling, thinning of the edges are done. The image is in the form of a line diagram, together with some auxiliary information about the abstract local features, eg. this information could be in the form of labels of points and lines and also the coordinates of the vertices. The picture which will be input to ANS will be of the kind shown in Fig. 3.2.

3.2 The System ANS

The output of IFORM is fed to the computer using any of the available edge detection techniques (34). A contour trace of the figure can locate various vertices and also the number of line segments meeting at a point. The system ANS acts on the labelled sketch and extracts information pertaining to the global aspects of the sketch like connectivity, boundary shapes of delimited regions. The output of ANS can be fed to another program which determines relationships between these regions (i.e. scene analysis), information about Gestalt clusters (i.e. configuration aspect, sub-patterns and so on).

The system ANS consists of the following sub-modules which are executed sequentially. Any module may cross-reference any other module for executing a specified function.

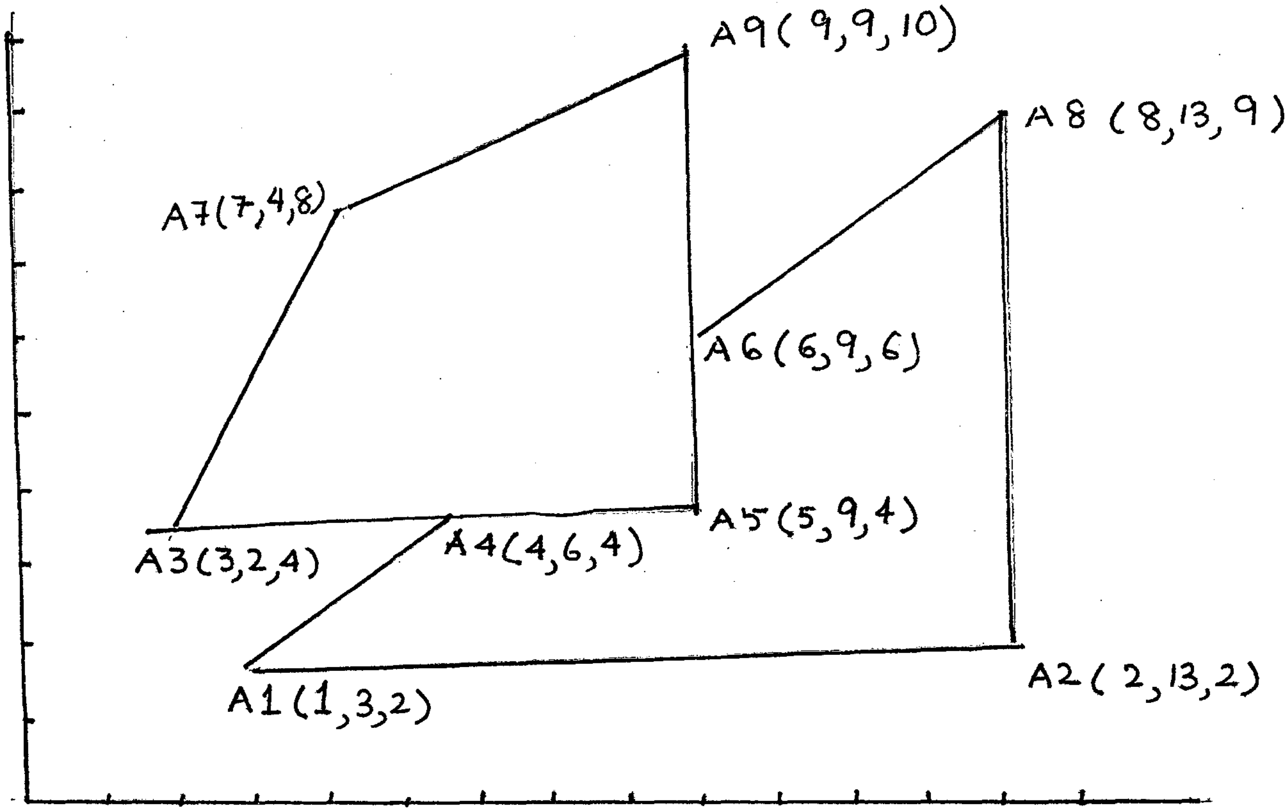


FIGURE 3.2

3.2.1 I DATA

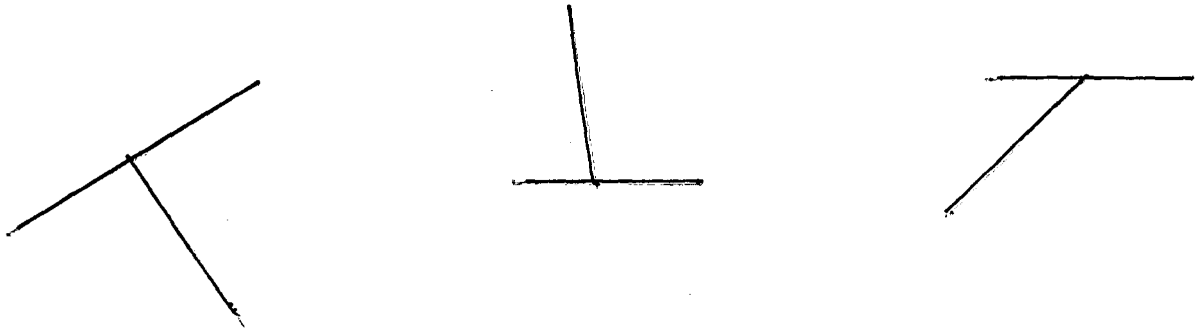
IDATA is the first module. In this phase the figure is taken as consisting of straight line edges and each point and line is described as a heirarchy of various types of information relating to the entity in question, eg. any vertex is characterised by its abscissa and ordinate and the number of line segments meeting at that point. This can be done with the help of specialised data-structures or an array, the elements of which specify the values of particular attributes. The dimensions of the array would depend on the number of attributes one would like to specify (APPENDIX A).

3.2.2 TEE

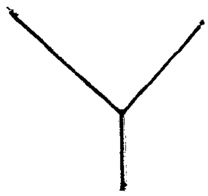
This is the second phase of the system. This module detects at which of the vertices there are three segments forming a 'T'. The scope of this module could be enlarged to define and include various other configurations of segments meeting at a vertex like a " fork joint" or an "arrow joint" (fig. 3.3). The interpretation of these joints would depend on the context of their occurrence and this would form the semantics of the system. In all these cases, the corresponding interpretation, i.e. how these segments are to be related with respect to the figure where they occur, forms an integral part of the program.

The system ANS considers simple cases of overlapping, i.e. where one figure overlaps the other. This module acts on the data-structure envisaged in Phase I - IDATA. The output is in the form of an array of all vertices. Those vertices that have a T joint have the corresponding element in the array as '1', otherwise it is '0'.

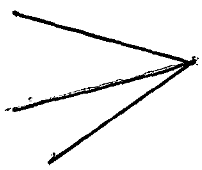
FIGURE 3.3



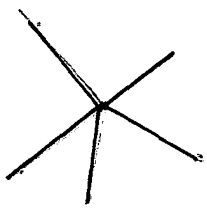
SOME EXAMPLES OF ‘T’ JOINTS



A ‘FORK’ JOINT



AN ‘ARROW’ JOINT



A MULTI JOINT

DIFFERENT KINDS OF JOINTS

3.2.3 HID_L

This module examines the vertices which have a 'T' joint and finds out if there is a straight line between any of these vertices, which is hidden by an overlapping figure. This procedure is an attempt to take into account the principles of Gestalt Psychology(21). This is done by developing equations for all segments that meet at vertices which have a T joint. These equations are then reduced to a canonical form, and compared with the equations of other line segments. The sets of equations that match clearly represent the same line indicating continuity. The vertices which have 'T' joint and the segments meeting at the vertex have the same set of equations, the system assumes, that these two segments are actually just parts of a larger segment, a part of which is hidden by an overlaping figure, eg. line segment $D_1\ F_1$ (fig. 3.4.1) is considered as "hidden" and segments $A_1\ D_1$ and $F_1\ H_1$ are considered to be parts of large segment $A_1\ H_1$. In fig. 3.4.2, $D_2\ F_2$ is considered as "hidden" and segments $A_2\ D_2$ and $F_2\ G_2$ are taken as parts of a larger segment $A_2\ G_2$.

The output of this phase is an ordered list of segments which are considered to be a part of a larger segment. The ordered list will contain entries of the type ($A_1\ D_1$, $F_1\ H_1$), ($A_2\ D_2$, $F_2\ G_2$).

3.2.4 ST_LN

The input to this phase is the output of HID_L, i.e. names of segments which are parts of a larger segment, a part of which is occluded by another figure. This module combines them to form one straight line. This facilitates in recognition of figures which is done in the later stages.

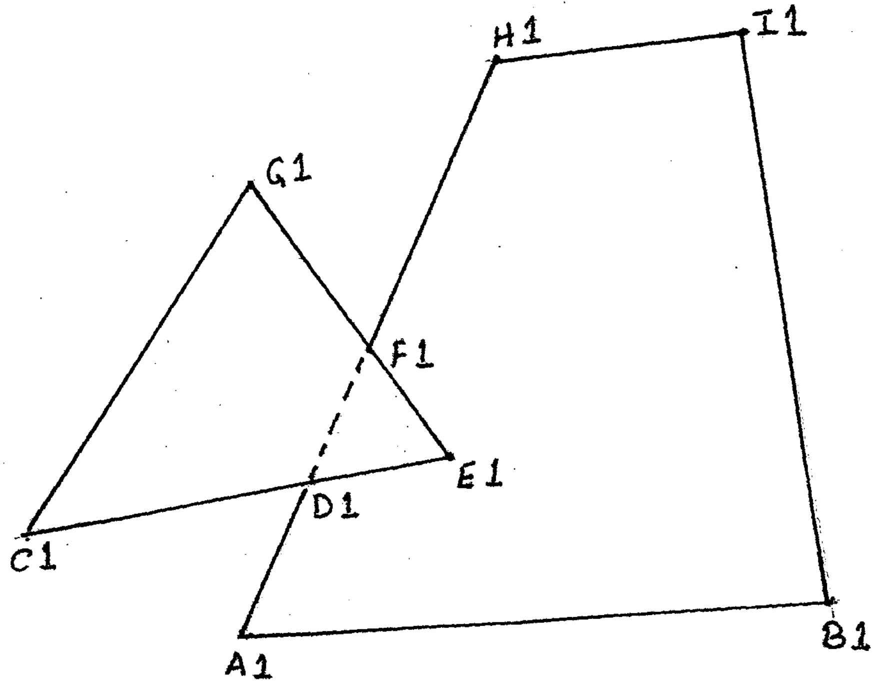


FIGURE 3.4.1

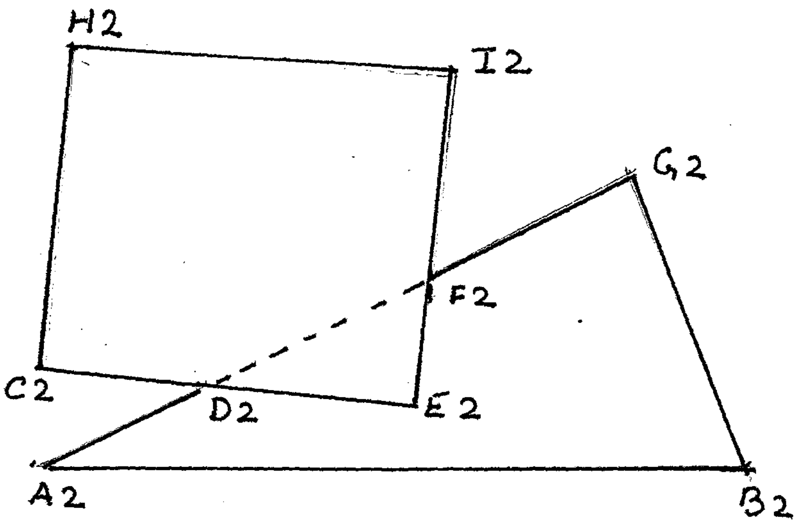


FIGURE 3.4.2

This module functions on the basis of results provided by its sub-module or sub-routine M_LIN. M_LIN procedure finds out the coordinates of the end points which form the merged straight line.

The output of ST_LN is an array of ordered pairs of vertices. Each member of the ordered pair denotes that there is a line between members forming the ordered pair, eg. for Fig. 3.5, the output would be of the form (1,8), (1,2), (3,7), (8,9), (3,5), (5,7), (2,9) for Fig. 3.5.1 and (1,7), (1,2), (3,5), (2,7), (5,8), (7,8), (3,7) for Fig. 3.5.2.

3.2.5 TR–RN

This module finds out which lines form triangles in the input pattern. For this it finds out if

1) there exist straight lines between pairs of points in triplets of vertices

2) No two lines in (1) should be collinear. For this equations of the lines are compared. The output is a triplet of points which form triangles.

3.2.6 QU_RN

This phase of the algorithm finds out which vertices form quadrilaterals. The grammar for this case is

1) there exist straight lines between adjacent points in sets of all points considered as quadruples.

2) No two such straight lines should be collinear. The collinearity condition is found out by finding out the equations of the lines. The output is a quadruple of points representing a

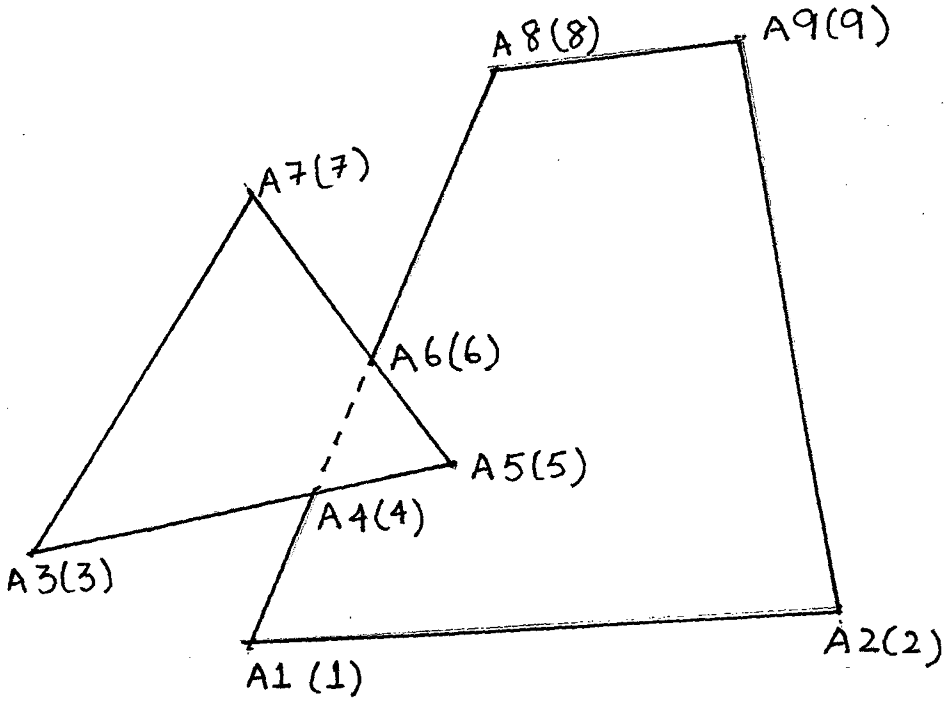


FIGURE 3.5.1

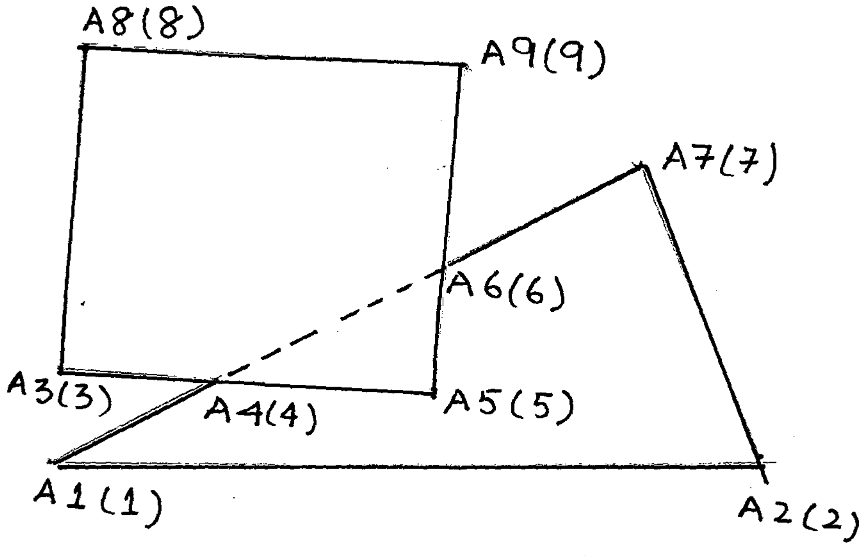


FIGURE 3.5.2

quadrilateral in the figure.

Modules which recognise other kinds of geometric figures could also be incorporated into the system, to broaden its scope. Also one could incorporate in an extended version of this program, which quadrilaterals are squares, rectangles, which triangles are scalene, isosceles or equilateral. For this a more comprehensive grammar would have to be built in the system.

CHAPTER 4

The importance of developing algorithms for handling partially occluded pictorial information cannot be over stressed. This kind of problems are frequently encountered in handling military photographs, mining data, design of robots etc. (15, 46, 47, 48). The design of A N S is a step towards this end. Below we consider some of the limitations and some possible future extensions of ANS.

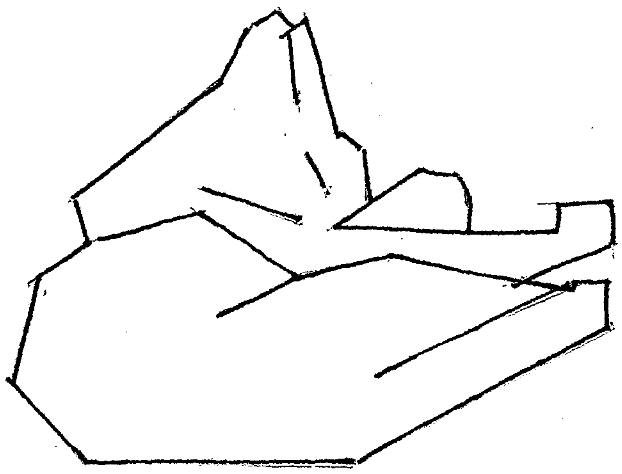
4.1 The Domain of ANS

The system ANS works in a restricted domain of straight line figures. The reason for doing this was that the relevant information pertaining to the attributes of the figure (e.g. the slope of a line) can be extracted from well-developed mathematical formulae. Moreover, many real life scenes preserve their pictorial meaning even after a transformation into scenes containing straight line objects only provided the transformation is appropriate. The classic example is Atteneneave's sleeping Cat (1).

Only very simple geometric figures like triangles and quadrilaterals have been considered. More complex geometric figures can also be studied by incorporating into the system an appropriate grammar to describe them.

The Image Former (IFORM) by preprocessing the input images enables further computation by providing the necessary descriptive inputs. These can be considered to form a vocabulary in a language defined by a suitable grammar. The computations done in ANS are related to computations on the phrase, sentence and intersentence entities of picture language.

FIGURE 4.1



ATTENEAVE'S SLEEPING CAT

4.2 The Weight System

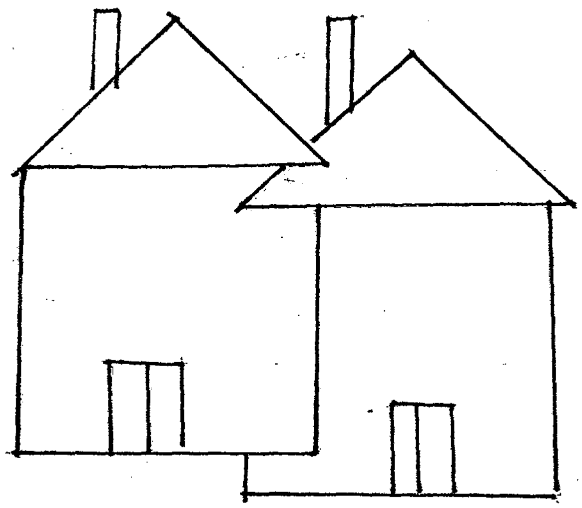
The scope of ANS can be broadened by allowing it to choose from a set of different "alternative" meanings for the partially occluded part. Each of the possible meanings could be associated with a certain weight which (weight) would depend on the configuration of the environment, eg. if the system encountered a hidden line in a triangle then depending on whether the surrounding scene is that of a house or a machine part, the occluded portion could either be a straight line or a saw-tooth like figure (Fig. 4.1). Therefore, even though the parts of the figure that are visible are the two sides of a triangle, the "meaning" attached with the hidden parts are entirely different.

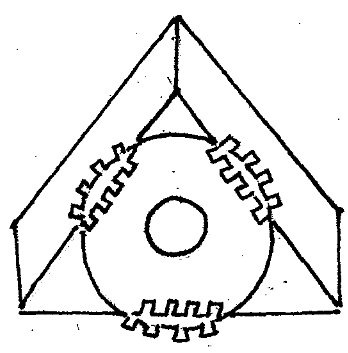
4.3 Learning

Most learning devices incorporate some kind of a weighing process. The crucial process in this task is the choice of an appropriate weight system, eg. the Perceptron (5, 33). With a suitable weight system designed for this system, the system ANS could be exposed to learning situations where the typical kind of objects it would have to identify are already built into the system. Then ANS would 'know' what set of weights to assign to each occurrence of a hidden line depending on the context in which it appears. Such a set of weights would depend on the set of weights that were incorporated in it during the training phase. The choice of the weight system could depend on statistical considerations, i.e. what weight system gives maximum number of correct outputs.

All learning schemes do not necessarily involve the weight system or statistical inference methods. Some of them depend on heuristics

FIGURE 4.2





and formulation of concepts in the training phase (31, 36, 37, 42). Perhaps, it would be a worthwhile attempt to integrate an appropriate set of weighting procedures with heuristic and paradigm of concept formation.

One of the possible ways to realise a "learning machine" is to equip it little at the design stage and support it with heavy learning machinery. Another way to realise is to have a system with no training phase and with a heavy dependence on a programmer built design for its functioning.

4.4 Future Trends

Due to the enormity of problems concerned with simulation of any thought process or the brain functions in terms of the solution spaces to be searched, more stress should be laid on the design of algorithms which offer a reduction in the search space of solutions and avoid combinatorial explosions.

APPENDIX A

The implementation of the system ANS was attempted on EC 1020 B installed in Jawaharlal Nehru University, New Delhi.

PL/I was chosen for developing the programs because of its versatility and some of its non-numeric computational aids. LISP would have been a more powerful non-numeric list processing language on which to implement ANS. PL/I was chosen due to the non-availability of LISP. Since PL/I D Level Computer was available and since many of the data-structures and facilities which are available in PL/I F level computers were envisaged in the system ANS, a modified version was run on the computer. Some of the important aspects of the program are described below.

A.1 IDATA

This module consists of data declarations. The vertices and lines are associated with auxiliary information which is structured as a heirarchy. The vertices are input as a data structure of the following kind:

```
1 POINT (P)
  2 X_CRD
  2 Y_CRD
  2 NAME
  2 S_NME (Q)
  2 NO_LIN_SGMT
  2 ANGLE (S)
```

where P = total number of points in the picture

Q = number of line segments meeting at a point

S = number of angles between the line segments

X_CRD = represents the value of the x coordinate of the point

Y_CRD = denotes the y coordinate of the point

NAME = label of the point

S_NME(Q) = ordered list of names of segments meeting at a point

NO_LIN_SGMT = represents number of line segments that meet at a point

ANGLE(S) = denotes the ordered list of magnitude of angles between the line segments.

eg. the point L in Fig. A.1 would have the following information.

X_CRD = 5

Y_CRD = 10

NAME = L

S_NME(3) = (A1, A2, A3)

NO_LIN_SGMT = 3

ANGLE (2) = (40, 140)

The second data structure that is used pertains to the information about the line-segments. It is also structured as a heirarchy of auxiliary information. The following is an example:

1 LINE (I)

2 SMT_NME

2 X1_CRD

2 Y1_CRD

2 X2_CRD

2 Y2_CRD

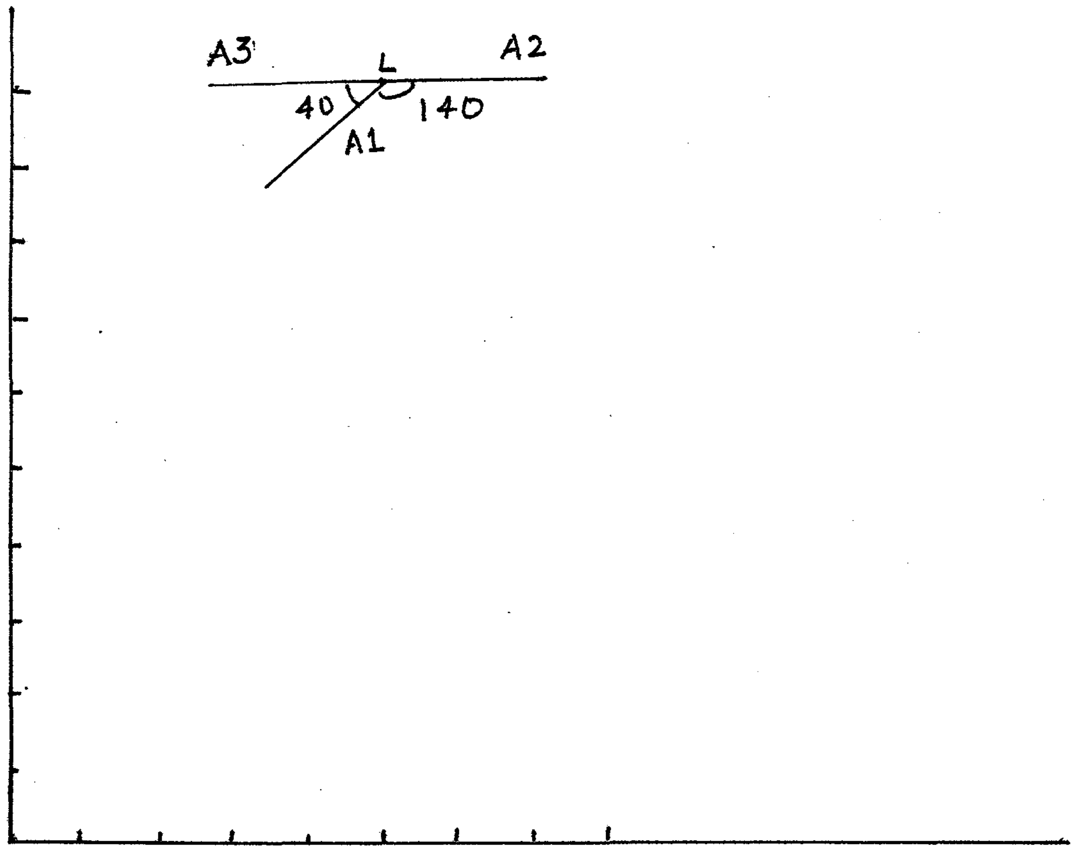


FIGURE A·1

I = total number of line-segments in the picture

SMT_NME = denotes the label of the segment

X1_CRD, X2_CRD = denotes the x-coordinates of the end points of the segment

Y1_CRD, Y2_CRD = denotes the y-coordinates of the end points of the line segment

The line-segment L1 in Fig. A.2 would have the following information associated with it.

SMT_NME = L1

X1_CRD = 2

Y1_CRD = 7

X2_CRD = 6

Y2_CRD = 2

A.2 TEE

This is the procedure which finds out which vertices have 'T' joints. For this it finds out

1) the number of line segments meeting at a point

2) sum of the interior angles between segments meeting at a point.

The module gives an output '1' for the point under consideration, if number of line segments in (1) is three and value of (2) = 180, otherwise the value is '0' for the point under consideration. Thus, the output is an array of 0s and 1s depending on which of the vertices have a 'T' joint. The vertices are so numbered so that there is a 1-1 correspondence between the vertices which have a T joint and the

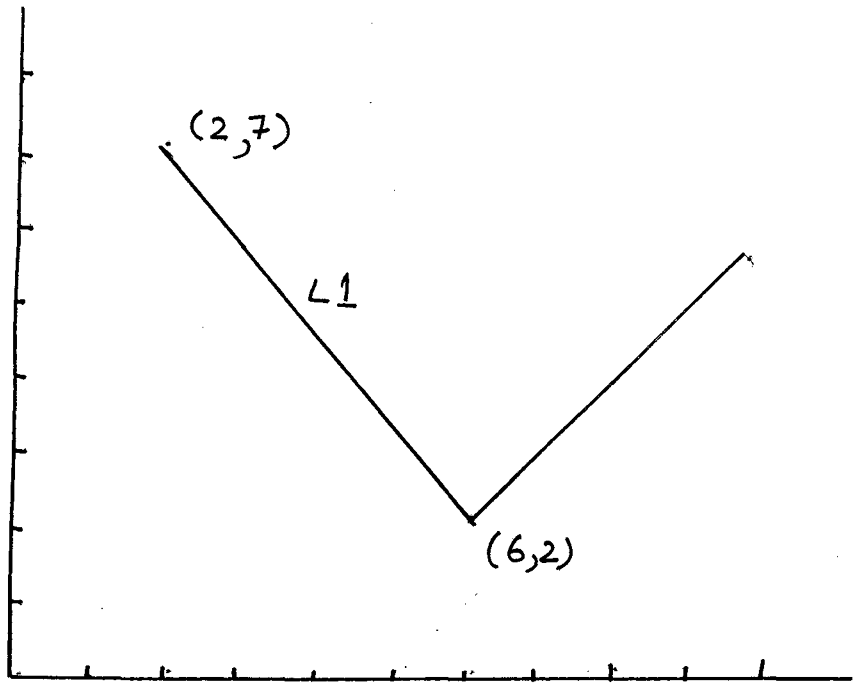


FIGURE A.2

vertices which have a '1' output.

A.3 HID_L

This is the procedure which determines the segments in pairs which are probably parts of a larger segment which is hidden by another figure. In this procedure, the slopes and intercepts of line segments which form a T joint are calculated using the results of analytic geometry (Appendix B). The output is an ordered pair of list of segments between which may be the hidden line.

A.4 ST_LN

Calls subroutine M_LIN to determine the end points of the larger line segment. This is done by treating the arguments returned by HID_L as forming one larger line segement. It returns '1' for those coordinates which have a straight line between them. ST_LN uses this information to give the label of the points between which are straight lines.

A.5 TR_AN

This is the procedure which determines the coordinates of the triangles in the input figure. The information from ST_LN is utilised and the grammar utilised to determine the existence of triangles between any three points is

1) there should be a straight line between pairs of points considered in triplets
2) no two lines in (1) should be collinear.

The output is a list of triplets which form triangles.

A.6 QU_RN

This is the procedure which recognises quadrilaterals in the input figure. The procedure finds out

1) if there exist straight lines between adjacent points in sets of all points considered as quadruples.

2) No two straight lines in (1) should be collinear.

If the above grammar is satisfied, then the system gives as output a list of quadruples which are the labels of the vertices of a quadrilateral.

A P P E N D I X   B

Given below is a list of mathematical formulae used in ANS.

1. Given the coordinates of two points (x_1, y_1) and (x_2, y_2) the distance d between these two points is

$$d = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2}$$

2. Given two line segments they can be considered to be along one straight line if

$M_1 = M_2, \quad C_1 = C_2$

where

M_1, M_2 = slopes of the two line segments

C_1, C_2 = are intercepts which these segments form with the y-axis

The general equation of the line can be given as

$$y = MX + C$$

where

M = slope

C = intercept on the y-axis.

B I B L I O G R A P H Y

1. Atteneave, F., 'Some Informational Aspects of Visual Perception', Psych. Review, 61 , 183, 1954.

2. Blakemore, C., Mechanics of Mind, Cambridge Press, 1977.

3. Clowes, M., 'Transformational Grammars and the Organisation of Pictures', Automatic Interpretation and Classification of Images, ed A. Grasselli, Academic Press, New York 1969.

4. Cooper, D.B., and Freeman, J.H., 'On the Asymptotic Improvement in the outcome of Supervised Learning Provided by Additional Non-supervised Learning', IEEE Trans. on Computers, C 19 , 1055, Nov. 1970.

5. Cooper, P.W., 'Fundamentals of Statistical Decision and Learning' Automatic Interpretation and Classification of Images ed. A. Grasselli, Academic Press, 1969.

6. Evans, T.G., 'Descriptive Pattern Analysis Techniques', Automatic Interpretation and classification of Images, ed. A. Grasselli, Academic Press, 1969.

7. Evans, T.G., 'Descriptive Pattern Analysis Techniques - Potentialities and Problems', Methodologies of Pattern Recognition ed. S. Watanabe, Academic Press, 1969.

8. Evans, T.G., 'A Heuristic Program to Solve Geometric Analogy Problems', Ph.D. Dissertation, M.I.T, 1963 also in 'Semantic Information Processing ed. M. Minsky, The M.I.T. Press, 1968.

9. Feignbaum, E.A., and Feldman, J., Computers and Thought, Mc Graw-Hill, 1963.

10. Fu, K.S. Introduction in Digital Pattern Recognition ed. K.S. Fu, W.D. Keidel and H. Wolter, Springer-Verlag.

11. Fu, K.S. 'Introduction' in Syntactic Pattern Recognition Application, ed. K.S. Fu, Springer-Verlag.

12. Fu, K.S. 'On Syntactic Pattern Recognition and Stochastic Languages', Frontiers of Pattern Recognition ed. S. Watanke, Academic Press, 1972.

13. Gelernter, H. 'Realisation of Geometry Theorem Proving Machine', Proc. ICIP, 1959.

14. Gelernter, H., Hansen, J.R., and Loveland, D.W., 'Empirical Exploration of Geometry Theorem Proving Machines', Proc. Western Joint Computer Conf., 143, 1960.

15. Guzman, A. 'Computer Recognition of Three Dimensional Objects in a Visual Scene', Ph.D. Thesis, Electrical Engg. Dept., M.I.T., December 1968.

16. Guzman, A. 'Decomposition of a Visual Scene into Three Dimensional Bodies', Automatic Interpretation and Classification of Images, ed. A. Grasselli, Academic Press, 1969.

17. Jarvis, R.A. 'Interactive Image Segmentation', Data Structures, Computer Graphics and Pattern Recognition, ed. A. Klinger, K.S. Fu and T.L. Kunii, Academic Press, 1977.

18. Julesz, Bela. 'Cluster Formation at various Perceptual Levels' Methodologies of Pattern Recognition, ed. S. Watanbe, Academic Press, 1969.

19. Kanal, L. and Chandrasekhran, B., 'Recognition, Machine Recognition and Statistical Approaches', Methodologies of Pattern Recognition, ed. S. Watanbe, Academic Press, 1969.

20. Kanal, L. and Chandrasekhran, B., 'On Linguistic, Statistical and Mixed Models for Pattern Recognition', Frontiers of Pattern Recognition, ed. S.Watanbe, Academic Press, 1972.

21. Kasvand, T. 'Linquistics for Scene Analysis' Data Structures, Computer Graphics and Pattern Recognition, ed. A. Klinger, K.S. Fu, T. Kunii, Academic Press, 1977.

22. Kuliowsky, C.A. 'Pattern Recognition Approach to Medical Diagnosis' IEEE Trans. on Systems Sciences and Cybernetics, SSC 6 , No.3, 1970.

23. Lindsay, R.K. 'Inferential Memory as the Basis of Machines which Understand Natural Languages', Computers and Thought, ed. E.A. Feignbaum and J. Feldman, Mc Graw-Hill, 1963.

24. Minsky, M. 'Artificial Intelligence', Information, a Scientific American Publication, 1963.

25. Munson, J.H. 'Some Views on Pattern Recognition Methodology' in Methodologies of Pattern Recognition, ed. S. Watanbe, Academic Press, 1969.

26. Narsimhan, R. 'On the Description, Generation and Recognition of Classes of Pictures' in Automatic Interpretation and Classification of Images ed. A. Grasselli, Academic Press, 1968.

27. Narsimhan, R. 'Syntax Directed Interpretation of Pictures', Communication of the Association for Computing Machinery, 9, 166, 1966.

28. Newell, A. 'Learning Generality and Problem Solving', Proc. of the IF IPS, 107, 1962.

29. Newell, A. and Simon, H.A., 'The Logic Theory Machine', IRE Trans. on Information Theory, IT 2, No.3, Sept. 1956.

30. Newell, A., Shaw, J.C. and Simon, H.A., 'Report on a General Problem Solving Program', Proc. of the ICIP, Paris, France, June 1959.

31. Nilsson, N.J. 'Learning Machines: Foundations of Trainable Pattern Classifying Systems', McGraw-Hill, New York, 1965.

32. Roberts, L. 'Machine Perception of Three Dimensional Objects', Ph.D. Thesis, Dept of Electrical Engineering, M.I.T., May 1963.

33. Rosenblatt, F. 'Principles of Neurodynamics', Spartan Books, New York, 1962.

34. Rosenfeld, A., and Thurston, M., 'Edge and Curve Detection for Visual Scene Analysis', IEEE Trans. on Computers, C 20 , 562, 1971.

35. Rosenfeld, A. Picture Processing by Computers, Academic Press, New York, 1969.

36. Sadananda, R. 'Descriptive Inferences in Pattern Recognition', Ph.D. Thesis, IIT Kanpur, August 1975.

37. Sadananda, R. and Mahabala, H.N., 'CONCEPTFERENCE: A System That Develops Concepts and Infers on Pictorial Data', IEEE Trans. on Systems, Man and Cybernetics, March, 1978.

38. Samuel, A.C. 'Some Studies in Machine Learning Using the Game of Checkers', IBM Journal of Research and Development, 3 , 210, 1959.

39. Samuel, A.C. 'Some Studies in Machine Learning Using the Game of Checkers II - Recent Progress', IBM Journal of Research and Development, 11 , 601, 1967.

40. Sayre, K.M. 'Recognition: A Study in the Philosphy of Artificial Intelligence' Univ. of Notre Dane Press, 1965.

41. Sebestyen, G.S. 'Decision Making Processes in Pattern Recognition, Macmillan Co., New York, 1962.

42. Slagle, J. Artificial Intelligence : The Heuristic Programming Approach , McGraw-Hill.

43. Turing, A.M. 'Computing Machinery and Intelligence' in Mind, 59 , Oct. 1950.

44. Ullman, J.R. Pattern Recognition Techniques, London Butterworths, England, 1973.

45. Watanbe, S. 'Methodologies in Pattern Recognition', Academic Press, 1969.

46. Winston, P.H. 'The MIT Robot' in Machine Intelligence, ed. D. Michie and B. Melzer, Vol. 7, American Elsevier, New York, 1972.

47. Winston, P.H. 'Scene Understanding Systems in Frontiers of Pattern Recognition', ed. S. Watanbe, Academic Press, 1972.

48. Winston, P.H. 'Learning Structural Descriptions from Examples' Ph.D. Dissertation, M.I.T., Cambridge, 1970.

49. Zobrist, A.L. and Thompson, W.B., 'Building a Distance Function for Gestalt Grouping, IEEE Trans. on Computers, C 24 , 718.