CONCEPTS OF AUTOMATIC PATTERN RECOGNITION

IN COMPUTER VISION

by

Navid Nikravan

B.S. Kansas State University, 1981

A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Computer Science

Kansas State University

Manhattan, Kansas

1987

Approved by:

Major Professor

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Today automation has progressed to the point where further advances definitely require machines that have special talents such as vision. Computer vision is the mechanism by which a machine receives information from the world around it. The concepts and development of machine vision started more than twenty years ago. At the time, Unimation, Inc. developed and installed [24] the world's first industrial robot. This robot was capable of manipulating objects with much higher flexibility than any machine previously developed.

Machine sensing, and in particular computer vision, is the basis for providing sensory inputs to a smart robot. Robotics is the advanced technology of machines which can actually process the humanlike capabilities, reasoning capabilities, and above all sensory inputs. This paper is about the most used sensory input, namely computer vision. Machine vision can be understood by comparing it with human vision. This should not imply that advanced technology exists to duplicate human vision; it need not, and it does not.

Since the early 1950's there has been a steadily increasing interest for embracing machine vision in industry to eliminate boring and unsatisfying work, to improve operator safety, and to boost productivity. With every advancement in robotics there have been corresponding advancements in pattern recognition and image processing. These two different areas of pattern recognition and image processing have developed as separate disciplines, but quite similar techniques are used in each field.

Some researchers consider pattern recognition to be a part of image processing while others see it as a totally different science. Pattern recognition is a process [2] where the output is essentially a data structure, where image processing results in an output image. Figure 1 shows pattern recognition in which the "process" is feature extraction and texture analysis.
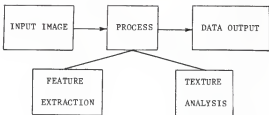
Figure 1. Pattern recognition

Figure 2 shows the diagram of image processing where the
"process" is noise reduction, and edge detection, and
results in an output image.
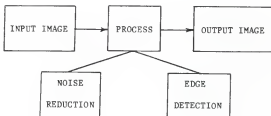
Figure 2. Image processing

A complete and reliable computer vision system could be divided into two categories: image processing and pattern recognition. Most pattern recognition systems perform some kind of basic image processing. Thresholding is an image processing operation which is used in order to get a binary image. Figure 3 illustrates the combined system where a processed image is used as an input image for the pattern recognizer [2].

PATTERN RECOGNITION

IMAGE
PROCESSING

| INPUT IMAGE | IMAGE PREPROCESSOR | PROCESSED IMAGE | IMAGE ANALYSIS PROCESSOR | DATA OUT |

THRESHOLDING
(FILTERING)
(EDGE DETECTION)

FEATURE
EXTRACTION
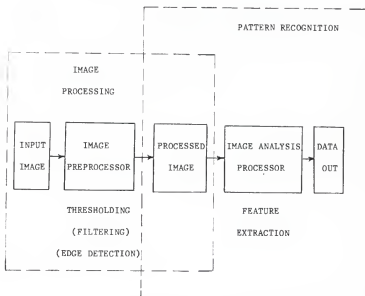
Figure 3. Combined system (Pattern
Recognition/Image Processing)

4

The pattern recognition process in machine vision does not deal directly with the particular object under consideration. In this process a representation of the scene is created in the form of an image. The various features (shape, location, orientation, etc.) of such an image are then described and measured. In general, these extracted features are then matched, pixel by pixel, with various models within the vision system to be recognized.

In the next chapter the preprocessing subfunctions of an image in a pattern recognition system are described. In Chapter 3 the feature extraction and decision models are presented. Chapter 4 presents an analysis of a technique used in object pattern recognition for processing binary images. Concluding remarks and future developments of pattern recognition systems in computer vision are presented in Chapter 5.

Chapter 2

PREPROCESSING

A pattern recognizer can be well thought of as a black box into which goes an image or an "object", and out of which comes some kind of description [5,8,10,18,23,25]. Figure 4 is the break down of the black box "process" of Figure 1.
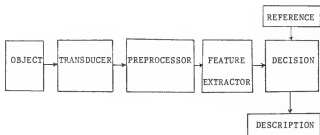
Figure 4. A pattern recognition system

2.1 Transducer

In a pattern recognition system the transducer could be an optical scanner, which we are most concerned with in this

report, where the input could also be in other forms, such as speech. The input is the original object to be classified and the output would be in a machine readable form. In this step the input image is converted into a two dimensional binary-valued matrix where a "0" is used for white cells and a "1" is used for black cells. This is done such that if the output is higher than a preset threshold, the cell is chosen to be a white cell and a "0" bit is recorded on the tape, otherwise, it will be a black cell and a "1" is recorded. There are optical scanners that encode gray-level where they use more bits per cell. For example, an optical resolution for medical projects could easily be six bits which results in 64 gray-levels. Each specific pattern recognition system has different grid size and optical resolution due to the size of the system. This means that a good design for specific objects and pattern recognition algorithms does not necessarily fit the specifications and requirements for another set of objects and pattern recognition system. Therefore we could state that in any experimental situation, for an ideal solution, we need to keep the parameters adjustable and put the scanner under program control. By doing this the scanner will become part of the pattern recognition loop.

2.2 Preprocessing

The function of preprocessing [7,8,9,18,19] is to
extract the proper features from the input image and to
produce another image that is easier for processing. Figure
5 illustrates the preprocessing subfunctions. In the
preprocessing, an image could easily use all subfunctions
whereas another image would not require any preprocessing
(subfunctions call) at all. The preprocessor may not exist
independently, but it is a necessary part of the pattern
recognition system.
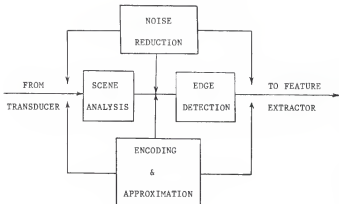
Figure 5. Preprocessing subfunctions

2.2.1 Image Analysis

The input of image analysis is still in the pictorial form where the desired output is s description of the input image. In a pattern recognition system, the feature extraction only functions on individual images that have been given descriptions such as shapes, names, etc. After the transducer operates on the input image, the picture may still consist of broken, incomplete, or multiple objects. Since the feature extractor operates only on single patterns and objects, image analysis is definitely a need in the preprocessing. The image analyzer takes the whole image (broken, incomplete, or multiple objects) and extracts parts of the image that could be determined as single figures. These extracted figures are then sent to the feature extractor one by one where they are processed as single figures. This process is also called segmentation. Image analysis can be a very difficult task, however, successful image analysis systems have been developed that handle problems at different levels of complexity [19,20,21]. The picture could be one of a specified finite set of possibilities, for example, where printed numerals are placed in well separated boxes on a formatted coding sheet. In this case the image (picture) is matched with a finite set of templates in order to specify which character is actually present.

Another case would be the images that consist of specified parts [19,20] in specified spatial relationships, for example, a human face containing eyes, nose, etc. In such a case it looks for the specified parts, and as these are found, it proceeds to find the other parts. It should be mentioned that even the individual parts may not be trivial to recognize, where in that case, models for the parts are also needed for recognition. A more complex case would be the segmentation of hand-printed text where character shaping and spacing are very highly variant. In such a case the segmentation is usually performed manually and limits the automatic recognition process to a character-by-character analysis. The most successful hand-printing recognizers are run on-line, where the operator inputs characters one by one. The operator inputs characters at a quick succession of strokes and pauses between characters. The obvious disadvantage of this system is the pausing between characters that greatly slows down the process. In attempting to analyze a picture it is necessary to make use of whatever prior knowledge is available about the class of pictures to which the input picture belongs. Such knowledge can be regarded as a model.

2.2.2 Edge Detection

The purpose of an edge detector [6,14,15,16] is to search for the points separating the figure from the background of the image. Edge detectors are also called boundary or contour detectors. These detectors operate on images which may or may not have been previously segmented. Most of the edge detectors only operate on a region that consists of only a single figure (segmented input).

Rosenfeld [15,18] distinquishes between the "edge" and the "border" of a pattern. Figure 6 illustrates the border according to Rosenfeld. The border consists of only the blacks cells that are adjacent, either vertically or horizontally, to white cells. In Figure 6, the border is illustrated by the black cells that are superimposed with an "X".
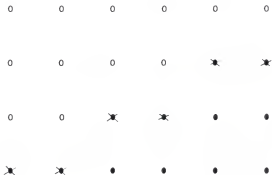
Figure 6. Illustration of the border

11

Figure 7 illustrates the edge according to Rosenfeld.
The edge consists of points that are determined as being
located midway between the black cells and the vertical or
horizontal adjacent white cells. In Figure 7 the edge is
illustrated by the "X".

Rosenfeld defines the border "as being made up of
outermost elements of the pattern", and the edge, "as lying
midway between horizontally or vertically adjacent pairs of
pattern/background points" [15].

```
    O       O       O       O       O       O

                                    X       X

    O       O       O       O   X   ●       ●

                    X       X

    O       O   X   ●       ●       ●       ●

    X       X

    ●       ●       ●       ●       ●       ●
```
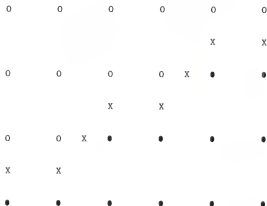
Figure 7. Illustration of the edge

Rosenfeld combines edge and border outline definitions
and presents an adapted scheme that is shown in Figure 8.

The combined method consists of points that are diagonally posisitoned midway between black cells and white cells.

```
    o       o       o       o       o       o


                                    X       X

    o       o       o       o       ●       ●

                X       X       X

    o       o       ●       ●       ●       ●

        X       X

    ●       ●       ●       ●       ●       ●
```
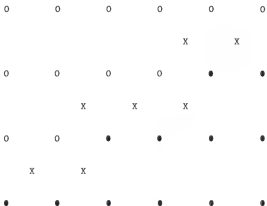
        Figure 8.   Combination method adopted of the border
                    and edge

     Kitchin and Pugh [15] point out that the disadvantage of the Rosenfeld definition of the edge is that such a definition requires the resulting array of points that has twice the point density of the original array. The new array has the density and spacing of the original array of edge points, with the difference that each point in the new array is shifted in both X and Y directions by one-half point spacing. Therefore, in the new array each point lies at the center of four points within the original array [15].

                                13

In most pattern recognition system applications it is very useful to determine the edge of an object so that general measurements can be extracted from the image.

In the gray-level the approach is to threshold the input image, creating a binary image that contains values zero or one. This is done such that the binary image replaces each and all values in the original image by a zero or one, depending on whether the value is above or below the desired threshold value. In the binary image a "1" is placed for a black point which corresponds to the foreground, and a "0" is placed for a white point which corresponds to the background. Therefore, the edge point of the figure can easily be determined by white points touching black points.

Edge detection is another method which is based on derivative operators or local gradients of gray-level. Derivative operators give a high value at points in the original image where the gray-level of the image is changing rapidly [5,19,20]. For example, in the original image each point could be assigned the maximum value of the set formed by taking the absolute difference of the original point value and its eight neighboring values. Its value (the gradient) at a point represents the "edge score" for each point and these scores are used to locate the figure.

Another method is the combination of the computational efficiency of the global threshold method with the data

sensitivity of the gradient method. The threshold in this particular method is set dynamically rather than being preset. Most of the useful information [5,19,20,25] in an image is contained in the regions where a sudden change of the gray-level occurs from one pixel to another. Such changes usually indicate a boundary, for example, an edge between two distinctly different objects in the image. This information consists of the direction in which the change of gray-level or the intensity changes most rapidly (edge direction) and also the size of the transition (edge magnitude). The direction of the local gradient is the angle between the coordinate axis and the composite vector, and the magnitude of the local image gradient is the length of the composite vector. These quantities can be computed from the partial derivatives of the image function. This information allows segmentation of an image which is an important criteria for identification and classification of objects. Edge or boundary information can also be used in order to determine the threshold value to isolate specific image regions.

2.2.3 Encoding and Approximation

In the preprocessing of an image, as the picture is passed from one procedure to another it is necessary to change the picture's representation in order to minimize the amount of memory and processing time required. One method

15

of such representation is run-length encoding. For example, in the case of a binary picture [5] the conversion to run-length is performed by scanning the original image row-wise from left to right. The search is done until the first black cell is found, its coordinates are then recorded and the scan again continues until the scanner encounters a white cell. A "black strip" is found and its length is determined and recorded along with the initial coordinate pair. In such an operation, the scan continues until all the strips are determined and recorded. This would definitely require less storage space for pictures in the horizontal direction.

2.2.4 Noise Reduction

Noise reduction is an information reducing technique. It is used to simplify the system logic where the approximation is used to ease the data load on the system.

Smoothing is a technique [5,19,20] that is frequently used for noise reduction in image processing. This method replaces each cell in the image by some reasonable value. For example, it replaces the cell with the integer part of the average of the cell values and its two vertical and horizontal neighboring pixels. In any binary image this method would change any isolated one-valued (black) cell to zero (white) which results in cleaning up the image.

Chapter 3

FEATURE EXTRACTION AND DECISION MODELS

3.1 Feature Extraction

The purpose of the feature or property extraction
process is to "generate an N-dimensional vector from the
input system, which capture the `essence` of the pattern to
be recognized. It is this elusive `essence` that makes
feature extraction the least understood and hence the most
difficult part of designing pattern recognition systems"
[11].

Rosenfeld believes that it is sometimes advantageous to
make feature extraction "a two-step process in which
pictures are first mapped into functions of a single
variable, and the functions are, in turn, mapped into real
numbers" [18].

In the pattern recognition system a picture still
remains after the preprocessing. Feature extraction
describes [7] the mapping of such two-dimensional spatial
object into a feature space. There are several techniques
that are used to extract features. The following section
describes some of these techniques that are most commonly
used.

3.2 Template Matching

This method as the name implies, is a comparison of the image of an object with "templates" or other images already stored in memory. In this method the original image of an object that is to be recognized is precisely compared (pixel by pixel) to each of the learned images. The quality of that match is then determined by the total number of pixels which disagree between the two images.

There are four different low-level templates [5] that are used. These templates are total templates, partial templates, piece templates, and flexible templates. Each of these categories is more flexible and the matching process is more complex than in the preceding category.

3.2.1 Total Templates

Total templates require an exact match between an image and a template. The matching process is very restrictive, where any displacement or orientation error of the correct pattern will not result in a perfect match and will be rejected. Also, the number of pixels in the input image must be exactly the same as the number of pixels in the template in order to have a perfect match.

3.2.2 Partial Templates

Partial templates are more flexible than total templates. In partial templates the image is totally

independent from the background, therefore, allowing multiple matches against a particular image. Since there is no restriction on displacement in partial templates the obvious disadvantage of this method would be the incorrect matches that could result when the actual image is embedded in a larger pattern. As an example, the "F" template would be perfectly matched against the "E" template.

3.2.3 Piece Templates

Piece templates break up a single pattern into different components and use these components to obtain a match. The pattern " A " could be broken up into " / ", " \ ", and " - " components (piece templates) to be recognized. The disadvantage of this method is that even by having all the components an incorrect or unintended match could be obtained, for example, where the symbol " $\bigvee$ " and an " A " have the same piece templates. It should be noted that in this method the largest piece template must be compared first in the matching process since it could contain smaller piece templates and it also contains the most information.

3.2.4 Flexible Templates

Flexible templates are capable of handling misorientation problems. These templates are also called rubber masks. Cohen and Feigenbaum explain that in the

process of obtaining a best match "the flexible template starts with a good prototype of a known object. After each comparison with the unknown object, the rubber mask is parametrically modified to obtain a better fit. This relaxation procedure is continued until no more improvement is obtained. The object can now be encoded as the template plus a series of modifications, which can be compared against the results with other starting templates to determine the best match" [5].

3.3 High Level Template Matching

In this method, rather than pixel to pixel matching of an image with a template, images and templates are described symbolically where descriptions such as first and second moment of inertia or area are matched to template descriptions. The first and second moments of inertia of the input image are computed as an image is being received. The first moment of inertia determines the location of the center of gravity of the input object and the second moment of inertia determines the object's orientation. In this method it is required to rotate an object image to correspond to the orientation of the templates. A match is obtained if the number of unmatched pixels is below a preset threshold. The disadvantage of this method is the computational complexity and intensiveness.

3.3.1 Global Feature Method

This method was developed at SRI International. It computes the global features such as width, total area, perimeter, total hole area, and number of internal holes. By having this information available for an image, the vision system performs a search among all of the templates that are actually stored within the vision system`a database. The global feature method is widely used and is now available commercially.

3.4 Decision Models

There are two classes of decision models in the pattern recognition system, logical (syntactic) and statistical. In the statistical approach the original image after the feature extraction process is reduced to an N-dimensional feature vector, usually a real-valued vector. The decision making process is based on the description of the object that is represented by the vector. For example, if the object to be recognized is an alphabet element, then the description is simply a class name, "A", "B", "C", etc. Whereas in the syntactical approach, a pattern is represented as a graph (tree, string) of pattern primitives and their relationships. In general, the process of decision making is a parsing procedure.

3.4.1 Logical Models

There are two classes of logical models, namely decision-tree models and syntactic (grammatical) models. These models are more easily understood since they are close to natural language.

3.4.1.1 Decision-Tree Method

Decision-tree methods are typically more appropriate to use on specific sets of objects that are to be recognized without being segmented into more than one or two simple sub-objects. The decision-tree approach could be thought of as a "flow chart" approach to identify an object.

3.4.1.2 Syntactic Method

The syntactic (structural, linguistic, grammatical) method in pattern recognition takes, as the input, some sub-objects and their specified relationships with each other, and outputs the name of the associated object. The syntactic method could be easily designed in a pattern recognition system to deal with object classes of a very complex internal structure. The basic idea behind the syntactic pattern recognition approach [11] is to describe highly complex patterns in terms of a hierarchical composition of simpler patterns. This approach is more easily understood since it holds the similarity between the hierarchical (treelike) structure of patterns and the syntax

of languages. Figure 9 shows the picture of geometric figures that can be described in terms of the hierarchical structural description as shown in Figure 10 [9,10].
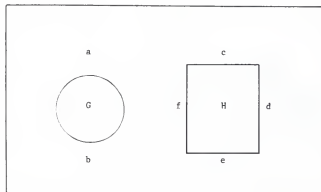
Figure 9. Picture of geometric figures
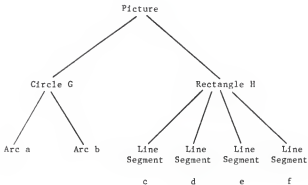
Figure 10. Hierarchical structural description

As shown in Figures 9 and 10, in this approach a set of features are extracted from the pictorial patterns and these extracted features are interpreted as the coordinates of points in a vector (feature) space. Then partitioning the feature space into sub-objects that correspond to patterns within the same class occurs.

3.4.1.2.1 Relational Graph

A relational graph [9,10] is the representation of the structural information of a pattern. For example, the picture of geometric figures in Figure 9 could be described

in great detail where relations between various sub-objects and primitives are explicitly specified. It should be noted that syntactic pattern recognition could be very useful if and only if the pattern primitives (fundamental subpatterns selected) are much simpler to classify than the original patterns themselves. Figure 11 [10] is the relational graph of Figure 9.
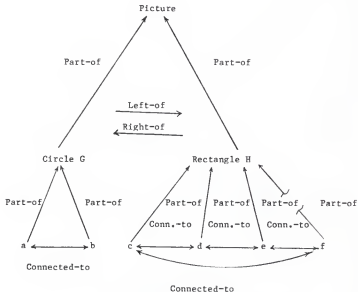
Figure 11. Relational graph of Figure 9

Fu states [8,9] that the most attractive aspect of the syntactic approach is its capability to use the recursive nature of a grammar, which is applying the grammatical rules any number of times, to express in a very compact way some basic structural characteristics of infinite sentences. As stated before, syntactic approach is very practical only if a large set of complex objects could be described by using small sets of simpler sub-objects and grammatical rules.

3.4.2 Statistical Approach

The statistical approach [4] in pattern recognition, in spite of its incredibly fast development, is still in its infancy. As mentioned earlier, in the statistical method the pattern is represented as an N-dimensional feature vector and the process of decision making is totally based on similarity measurements. The geometrical interpretations are usually more explicit ways of explaining some basic concepts of the statistical method in pattern recognition. For example, consider the set of two-dimensional measurements of the two pattern classes A and B [4] in Figure 12.
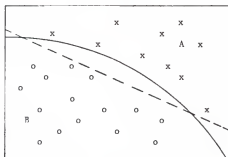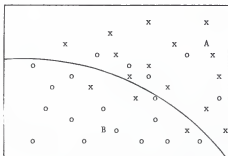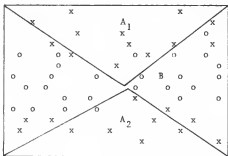
Figure 12. Two-dimensional measurements of two
pattern classes A and B

As shown in Figure 12(a), a nonlinear decision boundary is constructed in order to partition the pattern without an error into class A and class B. It should be noted that the two classes that are disjoint in the pattern are called separable classes. The dashed line represents that the decision boundary is linearly separable. Figure 12(b) shows that the two classes cannot always be a single decision boundary, although if the statistical decision theory is applied, it could only be partitioned with a minimum error. In Figure 12(c), class A is partitioned into two sub-classes, sub-class A_1, and sub-class A_2, therefore requiring the decision boundaries to be constructed between class B and sub-class A_1.

Statistical pattern recognition is very highly diversified and still with a number of scattered results. In this introductory exposition, it is almost impossible to describe, even in broad terms, the many varied disciplines within statistical pattern recognition.

CHAPTER 4

AN APPROACH FOR PROCESSING OF BINARY IMAGES

A considerable amount of research has been done toward
development of precise and complete methods of processing of
two dimensional images stored as binary matrices. A very
large portion of this development has been directed towards
understanding and solving problems of character recognition
as well as object recognition. Solving such problems
[12,15,18,26] involves not only the assigning of an image to
one of a set of prespecified classes but also requires a
description of the image. The number of possible
descriptions is very large so it is not practical to regard
one description as defining a class. A description could
refer to different subsets of an object where it specifies
properties of these subsets. In order to determine such a
description, an automatic pattern recognition system must be
well capable of segmenting the object subsets as described
in Section 2.2.1 of this paper. It should be stated that
there is actually no standard method of segmentation as yet
developed. Different types of subsets can actually be
objects themselves, of course, this is totally dependent
upon the type of description that is desirable. This
chapter presents an analysis of a technique for the

processing of binary images that was developed over a period of five years by P. W. Kitchin, Patscentre Benelux, Belgium, and Alan Pugh of the Department of Electronic Engineering, University of Hull, England [15].

4.1 Shape and Size

The objects could be in any arbitrary geometric shape [3,22] and the number of objects is essentially unlimited. This technique also assumes that all objects presented to the assembly machine must be exact templates of the reference objects. Also in this model, there is absolutely no allowance for any deviation in size or shape of the object to be recognized. Any deviation directly results in an absolute rejection of the object. Therefore, the description of the object must contain only the information that originally exists within the stored image of the data base.

4.2 Position and Orientation

An object can be presented to the system in any position and any orientation in the field of view. Though it should be noted that for such a system a position and orientation invariant description is required in order for it to recognize the object components. It is important to measure these parameters in order to make it possible for further manipulations, whereas, the individual alphabets do

30

not require these measurements because they are presented in a relatively constrained orientation.

4.3 Basic Parameters

There are several parameters that must be derived from an object in order to provide essential and valuable information about the object. These parameters are namely the area, perimeter, minimum enclosing rectangle, center of area, minimum radius vector and maximum radius vector, and holes.

4.3.1 Area and Perimeter

The area and perimeter of the object provides the dimensionless shape factor which is (area)/(perimeter). This factor is an important parameter in the pattern recognition system. The area and perimeter of the object also provide some classification criteria independent from the position and orientation of the image.

4.3.2 Minimum Enclosing Rectangle

Minimum enclosing rectangle provides information on the size of the object depending on the coordinates of the minimum enclosing rectangle. Of course, these coordinates would be totally dependent upon the size and orientation of the object.

4.3.3 Center of Area

The center of area is the origin for the minimum and maximum radius vectors. The center of area is totally independent of the object orientation and could be easily determined. It is an important parameter for determining the location and, therefore, the recognition of the object.

4.3.4 Minimum and Maximum Radius Vectors

The minimum and maximum radius vectors are the distances from the center of area to two points on the edge of an object. This determines the minimum and maximum length and direction of the vectors from the center of area to those points.

4.3.5 Holes

The holes within an object could be approached as different objects relative to the object that they are in, having size, shape, and position. The number of holes, if any, within an object, is an important parameter for recognition.

4.4 Edge Extraction

In this system it is assumed that the edge detector produces an unbroken sequence of boundary points. An "edge" is described in Section 2.2.2 (Edge Detection) and a formal definition of border and edge is given according to

32

Rosenfeld [15]. Figures 6 and 7 illustrate the differences between the edge and the border, and Figure 8 is the combination method of Figures 6 and 7.

4.4.1 Edge Trace

Edge trace could be thought of as a procedure that provides the system with many different parameters of the edge to be traced. Figure 13 illustrates these parameters.
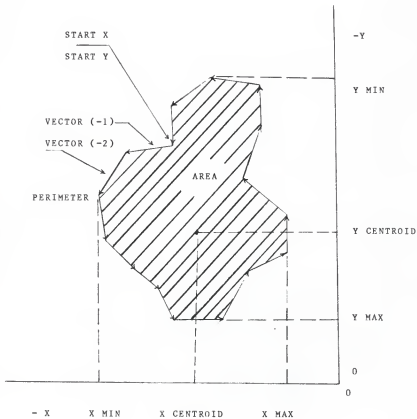
Figure 13. Parameters that are returned by Edge Trace

As Figure 13 indicates, the coordinates of the initial start-up points, START X and START Y are loaded initially into the system. These parameters are summarized in Table 1 [15]. These parameters are stored as global variables that contain a complete description of the edge in the form of a list of vectors.

VARIABLE	VALUE
PERIMETER	The length of the traced edge (positive integer)
AREA	The enclosed area (positive integer)
X MAX, X MIN Y MAX, Y MIN	The maximum and minimum X and Y coordinates reached by the edge trace (negative integers)
X CENTROID Y CENTROID	The X and Y coordinates of the center of enclosed area (negative integers)
VECTOR COUNT	The number of elemental vectors within the traced edge. This number is equal to the number of edge points (a negative integer).
CHAIN VECTOR	An ordered array of the directions on the vectors making up the outline. The array has VECTOR COUNT elements.

Table 1. Summary of parameters returned by Edge Trace

The vector notation used in this system is defined as the path between any pair of connected points in the image matrix where "a point in the image matrix is defined as being connected to another point if it occupies one of the eight immediately adjacent locations in the matrix" [15]. Figure 14 illustrates the eight possible paths between two points in the image. These paths are called elemental vectors and their directions are labeled (-8) to (-1).
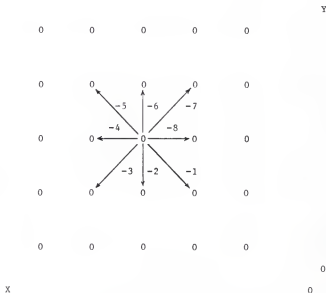
Figure 14. The elemental vectors

In order to be able to describe a pattern precisely, a complete description of the vectors' directions that links one particular point to the next point in the image is determined and these vectors and their corresponding directions are stored in an array. For example, consider the simple outline in Figure 15. The chain vector numbers and vector values (vector direction) of Figure 15 are also shown in Table 2.
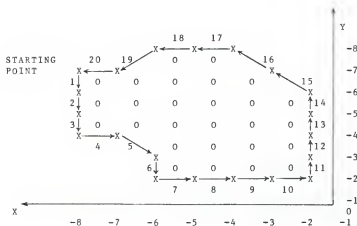
Figure 15. The outline in the vector form

38

CHAIN VECTOR NUMBER	VECTOR VALUE
-1	-2
-2	-2
-3	-2
-4	-8
-5	-1
-6	-2
-7	-8
-8	-8
-9	-8
-10	-8
-11	-6
-12	-6
-13	-6
-14	-6
-15	-5
-16	-5
-17	-4
-18	-4
-19	-3
-20	-4

Table 2. The chain vector numbers and vector values of Figure 15

4.6 Perimeter Computation

The perimeter of an outline (image matrix) such as Figure 15 could easily be determined as the sum of the magnitudes of the constituent elemental vectors such that the even numbered directions of the chain vector numbers would have magnitudes of 1 unit and the odd numbered directions would have magnitudes of $\sqrt{2}$ units. The process is such that at each point in the image matrix is located (chain vector number and its corresponding direction), one of the two integers, even-perim or odd-perim is incremented. At the end of the search the perimeter is determined by

$$\text{PERIMETER} = \text{EVEN-PERIM} + \text{ODD-PERIM}(\sqrt{2})$$

where the perimeter is rounded to the nearest integer number. It should be noted that the image matrix is assumed to be of unit spacing as shown in Figure 14.

4.7 Area Computation

The area of the image matrix is determined as the summation of the areas between each and all elemental vectors from an arbitrary line. In this case, as Figure 15 indicates, the arbitrary line is chosen to be the line Y=0, and based on the line Y=0, any other Y values for each point in the image matrix could be obtained.

In the calculations of the area of the image matrix, since Y is assumed to be a negative number, a positive area corresponds to the elemental vectors having a decreasing X

40

component where a negative area corresponds to the elemental vectors having an increasing X component. Figure 16 illustrates the elemental vectors and their corresponding areas.
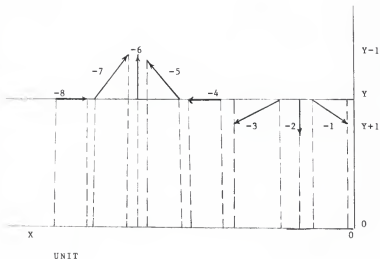
Figure 16. Elemental vectors and their corresponding areas

As Figure 16 shows the elemental vectors having directions -5, -4, and -3 contain positive areas whereas elemental vectors having directions -8, -7, and -1 have negative

areas. We can therefore conclude that the net area of an image matrix traced in an anti-clockwise direction would result in a net positive area.

The formulas for the area corresponding to each elemental vector is shown in Table 3 along with the double area for ease of computation.

ELEMENTAL VECTOR	AREA	2(AREA)
-8	(Y)(1)	2Y
-7	(Y-1/2)(1)	2Y-1
-6	0	0
-5	(Y-1/2)(-1)	-2Y+1
-4	(Y)(-1)	-2Y
-3	(Y+1/2)(-1)	-2Y-1
-2	0	0
-1	(Y+1/2)(1)	2Y+1

Table 3. Formulas of area for all elemental vectors

42

In order to determine the area of an image matrix the chain vector value and the corresponding Y value (in this case from the line Y=0) for each and all chain elemental vectors is obtained from Figure 15. The vector element numbers and values, the Y values, and the summation of double areas are all summarized in Table 4. As stated above, in order to avoid unnecessary calculations of the 1/2 factor for the area, the sum of double areas are calculated and in the end this area is divided by two for the result of the actual area.

VECTOR ELEMENT NUMBER	VECTOR ELEMENT VALUE	Y	2(AREA)	$\sum$2(AREA)
-1	-2	-7	0	0
-2	-2	-6	0	0
-3	-2	-5	0	0
-4	-8	-4	-8	-8
-5	-1	-4	-7	-15
-6	-2	-3	0	-15
-7	-8	-2	-4	-19
-8	-8	-2	-4	-23
-9	-8	-2	-4	-27
-10	-8	-2	-4	-31
-11	-6	-2	0	-31
-12	-6	-3	0	-31
-13	-6	-4	0	-31
-14	-6	-5	0	-31
-15	-5	-6	+13	-18
-16	-5	-7	+15	-3
-17	-4	-8	+16	+13
-18	-4	-8	+16	+29
-19	-3	-8	+15	+44
-20	-4	-7	+14	+58

NET AREA = 29

Table 4. Results of the area computations

44

4.8 First Moments of Area and Centroids Computations

From the calculated area of the outline, the X and Y coordinates of the center of the enclosed area could now be determined. In order to do so the first moments of area about the X and Y axis are desired. Tables 5 and 6 show all elemental vectors and their corresponding formulas for the first moment of area about the X and Y axis, respectively. In both cases the sum of the moments of area will result in a desired negative number. When this total negative moment of area is divided by the actual positive area, a negative value for both X and Y will be obtained as desired. It should be noted that the odd elemental vectors contain a constant term, +1/6 or -1/6 in their formulas. In an image matrix (closed outline) the number of elemental vectors could easily become very large where the constant terms will result in cancelling themselves. Also, the constant term is very small in comparison with the total enclosed area or the value of Y^2 or X^2, therefore, this constant is eliminated in Tables 5 and 6 where the expression for double moments of area about X and Y are given.

The total of $2(M_x)$ and $2(M_y)$ are then determined as sum-moment-X and sum-moment-Y where the division of these negative numbers by the double area will simply result in the values of X and Y centroids. See Appendix A for the calculations of moments of area about X and Y axis and the resulting X-centroid and Y-centroid.

VECTOR DIRECTION	M_x	$^*2(M_x)$
-8	$Y^2/2$	Y^2
-7	$Y^2/2 - Y/2 + 1/6$	$Y(Y-1)$
-6	0	0
-5	$-Y^2/2 + Y/2 - 1/6$	$Y(-Y+1)$
-4	$-Y^2/2$	$-Y^2$
-3	$-Y^2/2 - Y/2 - 1/6$	$Y(-Y-1)$
-2	0	0
-1	$Y^2/2 + Y/2 + 1/6$	$Y(Y+1)$

*The constant terms +1/6 and -1/6 are eliminated

Table 5. Formulas for moment of area about the X axis

VECTOR DIRECTION	M_y	$*2(M_y)$
-8	0	0
-7	$X^2/2 + X/2 + 1/6$	$X(X+1)$
-6	$X^2/2$	X^2
-5	$X^2/2 - X/2 + 1/6$	$X(X-1)$
-4	0	0
-3	$-X^2/2 + X/2 - 1/6$	$X(-X+1)$
-2	$-X^2/2$	$-X^2$
-1	$-X^2/2 - X/2 - 1/6$	$X(-X-1)$

$*$The constant terms +1/6 and -1/6 are eliminated.

Table 6. Formulas for moment of area about the Y axis

Now that the X-centroid and Y-centroid of the object are known, this coordinate (X,Y) will be the center of area of the object, which is the origin for the minimum and maximum radius vectors. From this origin, minimum and maximum radius vectors could easily be found by tracing the edge of the object in a counterclockwise direction. The lowest and highest value from the center of area to the object edge will be the minimum and maximum radius vectors.

The holes, as mentioned earlier, could be treated as single objects relative to the object that they are in. The procedures to find perimeter, area, moments, etc. as described could be applied directly for individual holes. It should be stated that if an object contains one or more holes, an additional parameter comes into consideration. This parameter is the distance(s) of the center(s) of area of the hole(s) from the center of area of the object. This is an important parameter and is widely used for similarity measurements in pattern recognition systems.

In addition to the hole pattern in terms of the distance between the centers of area of the holes from the center of area of the object that they are in, an additional parameter is obtained, namely, the relative angular position of the holes to the center of area of the object. Figure 17 illustrates the holes within an object and their parameters as a model where R_1, R_2, and R_3 are the distances of the centers of area of the holes from the center of area of the

48

object, and a_1, a_2, and a_3 are their relative angular positions, respectively.
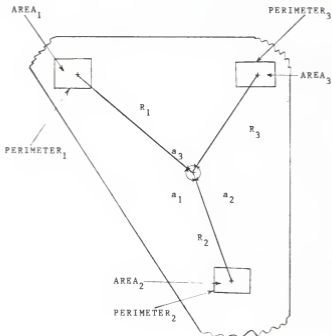
Figure 17.   The parameters of the holes located within an object

4.9 Circles Model

The circles model is a technique that takes advantage of the center of area principles in order to recognize an object, including all alphabetic characters, regardless of shape, size, and orientation criteria. The process of recognition starts for any given image (object) by determining the center of area of such an image. See Section 4.3.3 for center of area. From this center of area a number of circles of a given radius is then superimposed on the center of area of the given image. The number of circles, centered on the center of area of the object is totally dependent on the size, shape, and degree of precision desired of the match with the stored known object in the data base. This means the more circles superimposed on the object, the more feature points are then determined, therefore, a more precise match could be obtained from those feature points. It should be noted that by increasing the number of circles superimposed on the object, and therefore increasing the number of feature points, results in a more precise and accurate, but at the same time, slower recognition process.

Recognition by the circle model is such that a circle of a given radius is superimposed on the center of an object where the intersections of the circle with the image outline are defined as the feature points. These feature points are namely the distances (radius) from the intersection points

of the circle(s) and their relative angular positions from the center of area of the image.

The intersection points are found as an ordered list. The list could start from any predefined position within the image and then continues in a counterclockwise direction. The rotation of the input ordered list or the reference list of intersection points relative to one another may then be required in order to find a match. Figure 18 illustrates the feature points that are obtained by superimposing two circles on the center of area of an image. In the next section, the placement of the circle is presented.

θ_A - R_2

θ_B - R_2

θ_C - R_2

θ_D - R_1

θ_E - R_1
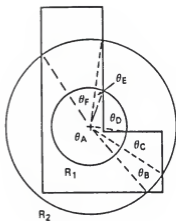
θ_F - R_2

a. Original image

θ_B - R_2

θ_C - R_2

θ_D - R_1

θ_E - R_1

θ_F - R_2
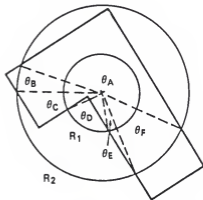
θ_A - R_2

b. Image rotated

Figure 18. Circles superimposed on the image

It is necessary to superimpose more than one circle on
the center of area in order to specifically define the
orientation of an object, or to differentiate it from
another similar object. In Figure 18 the ordered lists are
in the order that they are found in a counterclockwise
direction. The difference obtained when comparing the two
radius-angle lists illustrates two different orientations of
the object. In order to get a match, the two images and
their feature points are compared in the following steps:

1. Both ordered lists must be checked to ensure
   that they have the same number of entries
   (radius and its relative angular position).

2. The entries within the two lists are then
   compared one by one for a match between the
   two ordered lists for both radius and relative
   angular position entries.

3. If no match is obtained between the two lists
   then one ordered list is rotated by one
   angular position value in the counterclockwise
   direction and the comparison is repeated.

In Step 1, in order to simplify the comparison process,
the measured angular position values need to be compared if
and only if an exact match of the two ordered lists of
radius numbers was previously found.

The orientation of one object relative to another could be determined after the correlation between the two lists containing radius numbers and the angular positions is obtained. As shown in Figure 18(a) and 18(b), this degree of rotation can be viewed as the accumulation of angular position(s) in the ordered list. This accumulation is the result of removing one or more angular value position(s) from the top and placing it at the bottom of the stacked ordered list. The procedure is repeated until a match is found. In the case of a match, the accumulated values will determine the degree of rotation relative to the original image stored in the data base. Table 7 shows angular positions, their values, and corresponding radius that correspond to Figure 18(a) and 18(b).

Original Image				Image Rotated		
Angle	Value	Radius		Angle	Value	Radius
θ_A	188	R_2		θ_B	16	R_2
θ_B	16	R_2		θ_C	25	R_2
θ_C	25	R_2		θ_D	78	R_1
θ_D	78	R_1		θ_E	12	R_1
θ_E	12	R_1		θ_F	41	R_2
θ_F	41	R_2		θ_A	188	R_2

Table 7. Orientation measurements of Figure 18(a) and 18(b)

4.9.1 Criteria for Selecting the Radius of the Circles

In selecting the radius of the circles superimposed on
the center of the area of the object the following are
considered to be suitable criteria.

1. In order to select the radius of the largest circle, the
    largest distance from the center of area to a point on
    the outline image (image after edge detection) must be
    determined. The largest circle must then have a radius
    less that the distance obtained.

2. The number of intersection points should not be too
    small or too large, approximately greater than two and
    less than or equal to ten, totally dependent on the size
    of the object.

3. For precision and accuracy of the match and orientation
    purposes, the radius should be as large as possible,
    thus obtaining more intersection points by superimposing
    such a circle.

4. The radius should be selected such that for small
    changes in the radius new points should not be obtained
    nor should intersection points disappear.

5. The intersection points obtained by the superimposed
    circles should uniquely define the orientation of the
    image.

6. For orientation purposes of a component or for differentiating one component from another, it is definitely better to superimpose more that one circle on the center of area of the component. This would indeed result in a more accurate match and orientation.

4.9.2   Advantages of the Circle Model

There are many advantages of the circle model for pattern recognition, specifically object recognition. The circle model solves the problems of wrong way up position, orientation, and detection of overlapping objects.

4.9.2.1   Wrong Way Up Position

The advantage of the circle model according to Alan Pugh [15] is its ability to recognize and distinguish the "wrong way up" situation. Figure 19 illustrates the "wrong way up" position. In such a situation the object is being viewed from the "flip side" or in other words, the back side of the image. In order to distinguish the "wrong way up" situation the angular position values with their corresponding radius numbers are compared with the original image features within the data base. Also the angular position values and their difference values appear in reverse order for the "wrong way up" component.
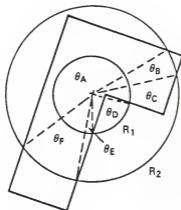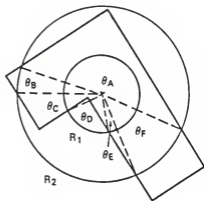
θ_B – R_2	θ_F – R_2
θ_C – R_2	θ_E – R_2
θ_D – R_1	θ_D – R_1
θ_E – R_1	θ_C – R_1
θ_F – R_2	θ_B – R_2
θ_A – R_2	θ_A – R_2

Figure 19. Wrong way up position

58

The circle model is indeed a very powerful technique for extracting feature points and specifying features of an arbitrarily shaped object for recognition process and orientation purposes of an object.

4.9.2.2 Orientation

Another advantage of the circle model is its ability to present an object to the machine in any orientation in the field of view. Of course, in the case of character recognition it should be noted that individual alphabets do not require orientation measurements since they are normally presented to the system in a relatively constrained orientation.

4.9.2.3 Detection of Overlapping Objects

Another advantage of this model is its detection of overlapping objects [13,15] in the early stages of the recognition process. Detecting overlapping objects in pattern recognition systems is an essential task, but at the present time is still in its early developmental stages. The approach that I recommend for a pattern recognition system (specifically for object recognition) is based on the principle that if point X_1 is outside a closed object "O" and is connected by a straight line to another point X_2 in any arbitrary plane, then X_2 must also be outside the object "O" if and only if the straight line X_1X_2 intersects object "O" an even number of times. Therefore, it is determined

that if line X_1X_2 only intersects the object "0" an odd number of times then X_2 is indeed inside the object "0" and thus it belongs to object "0". Figure 20 illustrates this principle.

(a) X_2 does not belong to object "O"
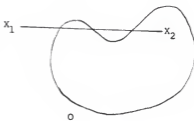
(b) X_2 belongs to object "O"

Figure 20. Position of point X_2 with the object "O"

Based on this principle, the problem of overlapping objects could be approached in such a way that after edge detection, if the image outline is intersected an even number of times, then the outline image could be segmented and processed as two different outline images. If the image outline is intersected an odd number of times, the outline cannot be segmented and is therefore traced as a single outline image.

In the vector elemental form of an outline image an intersection point is defined as when two points lie in two separate planes, in other words, they are located on both sides of the outline image (boundary or contour line). In the case where two points are overlapping it is not considered an intersection point. Figure 21 illustrates such intersection points.

(a) An intersection point   (b) Not an intersection point

Figure 21. Intersection points of boundary lines

In Figure 21(a) the intersection counter is only incremented once even though the outline image P_1 has three vector elements overlapping with the boundary line B_1, whereas, in Figure 21(b) the intersection counter is not incremented at all.

In conclusion it could be stated that if an outline image is within (inside) another outline image having no intersection points, or an odd number of intersection points, they both are considered one single object. In the case of an even number of intersection points, the object is segmented into two single objects.

4.9.3 Disadvantages of the Circle Model

A disadvantage of this model is the size and shape dependability. As stated before, the objects could be in any arbitrary geometric shape but the description (size and shape) of the unknown object must exactly match the information that exists about the original image stored within the data base. In other words, the object to be presented to the system must be of exact templates to the reference object or it results in a definite mismatch. There are a few techniques that have recently been developed to resolve the size dependability problems in the object recognition process. However, in such techniques, they eliminate the two most valuable parameters, namely, the area and perimeters of the object to be matched. It should be

63

noted that not all pattern recognition systems need to be size and shape independent. For example, in a part inspection pattern recognition system, size and shape dependability is a must, whereas in a more general pattern recognition system, when the only purpose is the recognition of the object (belonging to a specific object category), the size independability is required. The size independability can easily be in favor of a specific pattern recognition system, however, it could be unfavorable for another system, totally depending on the purpose of the specific pattern recognition system.

4.10 Machine Vision Vs. Human Vision

The ultimate goal of computer vision/pattern recognition researchers is to approximate as closely as possible the human vision system [1,17] in order for further advancements in machine vision. As stated in Chapter 1, this should not imply that advanced technology exists to exactly duplicate human vision, it need not, and it does not.

The human vision system is very highly complex and quite different in comparison to machine vision. Tomaso Poggio states that

"Indeed, it is widely expected that a coming generation of computers and robots will have sensory, motor and even `intellectual` skills closely resembling our own. How might such machines be designed? Can our rapidly growing

knowledge of the human brain be a guide? And at the same time can our advances in `artificial intelligence` help us to understand the brain?

At the level of their hardware (the brain`s or a computer`s) the differences are great. The neurons or nerve cells in a brain are small, delicate structures bound by a complex membrane and closely packed in a medium of supporting cells that control a complex and probably quite variable chemical environment. They are very unlike the wires and etched crystals of semiconducting materials on which computers are based. In the organization of the hardware the differences also are great. The connections between neurons are very numerous (any one neuron may receive many thousands of inputs) and are distributed in three dimensions. In a computer the wires linking circuit components are limited by present-day solid-state technology to a relatively small number arranged more or less two-dimensionally" [17].

Human vision with its complex structure does indeed have an advantage over machine vision for its ability and capability to analyze and recognize qualitative aspects of an image or a scene, although machine vision has the ability to measure quantitative data. Glorioso and Colon Osorio [11] present a theory known as the Muller-Lyer illusion, as Figure 22 illustrates. The vertical lines seem to be of different lengths, however, both vertical lines are the exact same length.
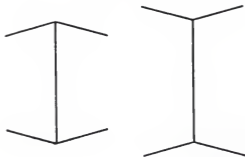
Figure 22. Muller-Lyer illusion

Glorioso and Colon Osorio state that "In terms of the stored model theory, an explanation of this illusion is that the leftmost image represents an outside corner of a square object, the rightmost image represents an inside corner of, for example, a room, and the perceptual system shrinks one and enlarges the other to compensate for the distortion caused by perspective" [11]. From this we can conclude that indeed human vision could sometimes be less accurate and precise than machine vision in measuring quantitative data.

Machine vision systems have limited capabilities and performances compared to human vision. Tables 8 and 9 summarize [24] the capabilities and evaluations of performance of both machine and human vision, respectively.

Capabilities	Machine Vision	Human Vision
Distance	Limited capabilities	Good qualitative capabilities
Orientation	Good for 2-D	Good qualitative capabilities
Motion	Limited, sensitive to image blurring	Good qualitative capabilities
Edges/Regions	High contrast image required	Highly developed
Image Shapes	Good quantitative measurements	Qualitative only
Image Organization	Special software needed; limited capability	Highly developed
Surface Shading	Limited capability with gray scale	Highly developed
2-D Interpretation	Excellent for well defined features	Highly developed
3-D Interpretation	Very limited capabilities	Highly developed
In General	Best for quantitative measurement of structured scene	Best for qualitative interpretation of complex and unstructured scene

Table 8. Evaluation of capabilities

Performance Criteria	Machine Vision	Human Vision
Resolution	Limited by pixel array size	High resolution capability
Processing Speed	Fraction of a second per image	Real time processing
Discrimination	Limited to high contrast images	Very sensitive discrimination
Accuracy	Accurate for part discrimination based upon quantitative differences. Accuracy remains consistent at high production volume.	Accurate at distinguishing qualitative differences. May decrease at high volume.
Operating Cost	High for low volume, lower than human vision at high volume	Lower than machine at low volume
In General	Best at high production volume	Best at low or moderate production volume

Table 9. Evaluation of performance

Present day machine vision systems are not even comparable to human vision, which is highly complex and delicately structured. But at the same time, human vision can definitely be a guide to seek solutions and make advancements in artificial intelligence, specifically computer vision. See Appendix B for Commercial Machine Vision System Specifications [24].

Chapter 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

The framework of the automatic pattern recognition systems and its terminology as presented and analyzed in the preceding chapters provides a useful technique for solving pattern recognition problems, specifically object recognition. This report has analyzed techniques for feature extraction from the binary image of an object in order to derive parameters used in the recognition process. An approach for processing binary images was then presented to determine the existence of a match between an object and the stored image of this particular object in the data base. This approach allows the object to be presented to the machine in any orientation. The design of a particular pattern recognition system is totally based on ad hoc considerations since the system is dependent on the input data and cost constraints.

In designing an automatic pattern recognition system, careful consideration is required in the preprocessing stages (see Chapter 2), specifically for selecting the type of data that the system is designed to handle. In other words, specifying the type of data that is sent to the

transducer for processing. Indeed with all advancements in artificial intelligence, specifically automatic pattern recognition systems, this advanced technology definitely requires human intelligence and interaction as a guide through various rough spots in the recognition process.

The pattern recognition approach that was presented and analyzed in Chapter 4 is a powerful technique for object recognition. This model is well suited for recognition of objects, inspection, orientation, and position purposes. The perimeter, area, moments about X and Y axes, and the center of area, are indeed useful parameters in the recognition process. However, computations of such parameters from the binary images and their vector elemental representations highly depend on and involve mathematical theory and concepts that would directly decrease the processing speed of pattern recognition systems.

The circle model is indeed a useful and powerful technique for feature extraction and, in addition, for recognizing objects and their orientations. It also detects overlapping situations of any arbitrarily shaped object from its outline image. The ability of the circle model for recognizing the "wrong way up" situation makes this model more attractive.

It should be stated that even though an automatic pattern recognition system can solve the recognition problems, it must be kept in mind that human vision not only

solves such problems, it also seeks solutions. With all
advancements in artificial intelligence, specifically in the
area of computer vision/pattern recognition, human vision is
an ultimate source of information for understanding and
approaching artificial intelligence problems.

5.2 Future Work

Artificial intelligence, specifically computer vision/pattern recognition, is in its early incremental development stages where it requires the machine to have special talents, such as stereopsis, decision making, and above all, the capability of seeking solutions. Today, it is possible to design and build model-based vision systems that could operate well with very limited numbers of specified objects. It is well expected improvements and advancements in performance of such model-based vision systems will be furthered developed in the near future.

Further research and development in the area of three-dimensional imaging and stereopsis (integration of depth information) definitely requires further advancements. Three-dimensional interpretation of an image will definitely provide some quantitative constraints which can be used to greatly reduce the amount of search needed for pattern recognition processes. Another topic for research development that is directly related to three-dimensional object recognition is the area of development and reduction of complexity in the three-dimensional scene. By developing models and algorithms that include a complete and precise description of the object to be recognized, the segmentation time will be reduced considerably.

New pattern recognition algorithms and models need to be developed to recognize the relationships between the

object and its neighboring objects that exists in a natural environment. With all advancements in pattern recognition, the object to be recognized still requires a relatively orderly environment. Advancements need to be accomplished so that the objects can be recognized in a less structured environment.

Further research needs to be performed in the area of evidential reasoning to develop an automated technique (learning systems) for evidence and interpretations of objects in a coherent scene. The performance of these learning systems will eventually increase by accumulating visual input information.

Overlapping and touching of objects also require further research. Current pattern recognition systems may detect overlapping situations, however, they do not recognize overlapping objects.

APPENDIX A

Chain Vector Number	Values	Y	$2(M_x)$	$\sum 2(M_x)$
-1	-2	-7	0	0
-2	-2	-6	0	0
-3	-2	-5	0	0
-4	-8	-4	16	16
-5	-1	-4	12	28
-6	-2	-3	0	28
-7	-8	-2	4	32
-8	-8	-2	4	36
-9	-8	-2	4	40
-10	-8	-2	4	44
-11	-6	-2	0	44
-12	-6	-3	0	44
-13	-6	-4	0	44
-14	-6	-5	0	44
-15	-5	-6	-42	2
-16	-5	-7	-56	-54
-17	-4	-8	-64	-118
-18	-4	-8	-64	-182
-19	-3	-8	-56	-238
-20	-4	-7	-49	-287

- -

$2(M_x) = -287$
$(2)Area = 58$
$Y\text{-centroid} = 2(M_x/(2))Area$

$Y\text{-centroid} = -4.95$

Chain Vector Number	Values	X	$2(M_y)$	$\sum 2(M_y)$
-1	-2	-8	-64	-64
-2	-2	-8	-64	-128
-3	-2	-8	-64	-192
-4	-8	-8	0	-192
-5	-1	-7	-42	-234
-6	-2	-6	-36	-270
-7	-8	-6	0	-270
-8	-8	-5	0	-270
-9	-8	-4	0	-270
-10	-8	-3	0	-270
-11	-6	-2	+4	-266
-12	-6	-2	+4	-262
-13	-6	-2	+4	-258
-14	-6	-2	+4	-254
-15	-5	-2	+6	-248
-16	-5	-3	+12	-236
-17	-4	-4	0	-236
-18	-4	-5	0	-236
-19	-3	-6	-48	-284
-20	-4	-7	0	-284

- -

$2(M_y) = -284$
(2)Area = 58
X-centroid = $2(M_y/(2))$Area

X-centroid = -4.89

APPENDIX B

			APPLIED INTELLIGENT SYSTEMS	AUTOMATIX			COGNEX
	COMPANY		PIXIE	AUTO-VISION II	ROBO-VISION IIA	CYBER-VISION III	OATAMAN
	MODEL						
TYPE OF SYSTEM	BINARY						
	GRAY SCALE		64 LEVELS	●	●	●	●
RESOLUTION (PIXEL ARRAY)			128 x 128	244 x 248	244 x 248	244 x 248	.020" CHARACTER HEIGHT
PROCESSING SPEED (RATE/MINUTE)			REAL TIME	360 PARTS; 2300 HOLES	150 INCHES	360 PARTS	900 CHARACTERS
STANDARD COMPONENTS	CAMERA		SOLID STATE	SOLID STATE OR VIDICON	SOLID STATE OR VIDICON	SOLID STATE OR VIDICON	VIDICON
	LIGHT SOURCE		●				
	COMPUTER		AIS COMPUTER	AI32	AI32	AI32	LSI-11/23
	SOFTWARE		THREADED CODE	RAIL LANGUAGE	RAIL LANGUAGE	RAIL LANGUAGE	MACRO ASSEMBLER
	OUTPUT INTERFACE		RS-232	RS-232	RS-232	RS-232	RS-232
	MONITOR		●	●	●	●	RCA
	DATA ENTRY		KEYBOARD	KEYBOARD	KEYBOARD	KEYBOARD	KEYBOARD
	OTHER				AIO 800 ROBOT	AIO 600 ROBOT	
TYPICAL SYSTEM COST ($000)			25	30	100 (with ROBOT)	130 (with ROBOT)	30
MOST SUITABLE APPLICATIONS	INSPECTION	MEASUREMENT	●	●			
		VERIFICATION	●	●			●
		SURFACE INSPECTION	●	●			
	PART I.D.	MATERIAL SORTING/HANDLING	●	●	●	●	●
		CHARACTER RECOGNITION	●				●
		BIN PICKING	●				
	CONTROL	WELDING	●		●		
		PROCESSING/ MACHINING	●		●		
		FASTENING/ ASSEMBLY	●			●	

		CONTROL AUTOMATION	COPPERWELD		EVERETT/ CHARLES	GENERAL ELECTRIC	
COMPANY							
MODEL		CAV-1000	OPTO-SENSE	OPTO-SENSE MENTOR	ERMAC 2500	OPTO-MATION II	I PARS
TYPE OF SYSTEM	BINARY	●	●	●	●	●	
	GRAY SCALE						●
RESOLUTION (PIXEL ARRAY)		128 x 128	244 x 248	244 x 248	255 x 255	244 x 248	
PROCESSING SPEED (RATE/MINUTE)		120 PARTS				900 PARTS	
STANDARD COMPONENTS	CAMERA	SOLID STATE	SOLID STATE	SOLID STATE	VIDICON	SOLID STATE (CID)	VIDICON
	LIGHT SOURCE		●	●		STROBE LIGHT	STROBE LIGHT
	COMPUTER	HP 85	LSI-11/2	Z-80	INTEL 8010B	GE PN2304	GE PN2304
	SOFTWARE	BASIC LANGUAGE		TEACH CONTROL MODULE	DIRECT LOGIC LANGUAGE	VPL LANGUAGE	
	OUTPUT INTERFACE	RS-232C	RS-232G	RS-232G	RS-232	RS-232C	RS-232
	MONITOR						
	DATA ENTRY	KEYBOARD			KEYBOARD	KEYBOARD	
	OTHER						
TYPICAL SYSTEM COST ($000)		20	50	50	23	40-50	23
MOST SUITABLE APPLICATIONS — INSPECTION	MEASUREMENT	●	●	●	●		
	VERIFICATION	●	●	●			
	SURFACE INSPECTION						
PART I.D.	MATERIAL SORTING/HANDLING	●	●	●	●		
	CHARACTER RECOGNITION						●
	BIN PICKING						
CONTROL	WELDING						
	PROCESSING/ MACHINING						
	FASTENING/ ASSEMBLY		●	●			

80

		HAM INDUSTRIES		INSPECTION TECHNOLOGY		MACHINE INTELLIGENCE CORPORATION	
COMPANY							
MODEL		HS-1000	HS-2000	ITI-2020	ITI-2030	VS-100	VS-110
TYPE OF SYSTEM	BINARY	●	●			●	●
	GRAY SCALE			●	●		
RESOLUTION (PIXEL ARRAY)				320 x 240	320 x 240	256 x 240	256 x 240
PROCESSING SPEED (RATE/MINUTE)						900 PARTS	900 PARTS
STANDARD COMPONENTS	CAMERA	VIDICON	VIDICON	VIDICON	VIDICON	SOLID STATE OR VIDICON	SOLID STATE OR VIDICON
	LIGHT SOURCE			STROBE LIGHT	STROBE LIGHT	STROBE LAMP	STROBE LAMP
	COMPUTER			●	●	LSI-11	LSI-11
	SOFTWARE			●	●	SRI ALGORITHMS	SRI ALGORITHMS
	OUTPUT INTERFACE	●	●	●	●	RS-232C	RS-232C
	MONITOR	●	●	●	●		
	DATA ENTRY			KEYBOARD	KEYBOARD	LIGHT-PEN	LIGHT-PEN
	OTHER						
TYPICAL SYSTEM COST ($000)		5-8	5-8	20	32	35-40	35-40
MOST SUITABLE APPLICATIONS — INSPECTION	MEASUREMENT	●	●	●		●	●
	VERIFICATION	●	●	●	●	●	●
	SURFACE INSPECTION			●	●		
PART I.D.	MATERIAL SORTING/HANDLING					●	●
	CHARACTER RECOGNITION						
	BIN PICKING						
CONTROL	WELDING						
	PROCESSING/MACHINING						
	FASTENING/ASSEMBLY						

COMPANY		OBJECT RECOGNITION SYSTEMS				
		100	200	Q	1000	I-BOT 1
TYPE OF SYSTEM	BINARY					
	GRAY SCALE	●	●	●	●	●
RESOLUTION (PIXEL ARRAY)						
PROCESSING SPEED (RATE/MINUTE)		300 PARTS	300 PARTS	240 CHARACTERS	90-120 PARTS	36 PARTS
STANDARD COMPONENTS	CAMERA	VIDICON	VIDICON	VIDICON	VIDICON	CCD CAMERA
	LIGHT SOURCE			FIBER OPTICS	FIBER OPTICS	
	COMPUTER	INTEL 80/10	INTEL 80/10	INTEL 80/10	INTEL 80/24	16-BIT
	SOFTWARE	ORS	ORS	ORS	ORS	PASCAL, VAL
	OUTPUT INTERFACE		RS-232C	RS-232C	RS-232C	RS-232C
	MONITOR		●	●	●	●
	DATA ENTRY	PHOTOCELL TRIGGER	KEYBOARD		KEYBOARD	KEYBOARD
	OTHER				X-Y TABLE	
TYPICAL SYSTEM COST ($000)		20	25	19		25
MOST SUITABLE APPLICATIONS — INSPECTION	MEASUREMENT					
	VERIFICATION	●	●	●	●	●
	SURFACE INSPECTION			●	●	●
MOST SUITABLE APPLICATIONS — PART I.D.	MATERIAL SORTING-HANDLING	●			●	●
	CHARACTER RECOGNITION			●		
	BIN PICKING					●
MOST SUITABLE APPLICATIONS — CONTROL	WELDING					●
	PROCESSING/ MACHINING					●
	FASTENING/ ASSEMBLY		●		●	●

			OCTEK		PROTHON	ROBOTIC VISION SYSTEMS	
	COMPANY						
	MODEL		4200 ROBOT VISION MODULE	INSPECTOR GENERAL	ROBOTIC VISION SYSTEM	ACOMS 1100	ROBO SENSOR 200
TYPE OF SYSTEM	BINARY		•		•		
	GRAY SCALE			•		•	•
RESOLUTION (PIXEL ARRAY)			320 x 240 OR 320 x 480	320 x 480	.1% OF FIELD OF VIEW	.05" FEATURE SIZE	.5% OF VIEWING RANGE
PROCESSING SPEED (RATE/MINUTE)			300 PARTS	300 PARTS	1800	60-120 MEASURE-MENTS	60-120 MEASURE-MENTS
STANDARD COMPONENTS	CAMERA			SOLID STATE OR VIDICON	VIDICON	SOLID STATE	SOLID STATE
	LIGHT SOURCE					LASER	LASER
	COMPUTER			DATA GENERAL OR DEC		HP 1000	HP 1000
	SOFTWARE		FORTRAN	FORTRAN		HP	HP
	OUTPUT INTERFACE		RS-330	RS-330		RS-232	RS-232
	MONITOR			•	•	•	•
	DATA ENTRY			KEYBOARD		KEYBOARD	KEYBOARD
	OTHER		IMAGE ANALYZER				
TYPICAL SYSTEM COST ($000)			10	40-50	10	75	75
MOST SUITABLE APPLICATIONS	INSPECTION	MEASUREMENT	•	•	•	•	•
		VERIFICATION	•	•	•	•	•
		SURFACE INSPECTION				•	•
	PART I.D.	MATERIAL SORTING-HANDLING	•	•	•		
		CHARACTER RECOGNITION		•			
		BIN PICKING					
	CONTROL	WELDING					•
		PROCESSING/ MACHINING					•
		FASTENING/ ASSEMBLY	•	•			•

		UNIMATION		VIEW ENGINEERING
COMPANY				
MODEL		UNI-VISION I	UNI-VISION II	719
TYPE OF SYSTEM	BINARY	●	●	●
	GRAY SCALE			
RESOLUTION (PIXEL ARRAY)		256 x 240	256 x 240	251 x 244
PROCESSING SPEED (RATE/MINUTE)		900 PARTS	900 PARTS	
STANDARD COMPONENTS	CAMERA	SOLIO STATE OR VIDICON	SOLIO STATE OR VIDICON	VIDICON
	LIGHT SOURCE	STROBE LIGHT	STROBE LIGHT	
	COMPUTER	LSI-II	LSI-II	VIEW ELECTRONICS
	SOFTWARE	VAL LANGUAGE	VAL LANGUAGE	VIEW LANGUAGE
	OUTPUT INTERFACE	RS-232C	RS-232C	RS-232C
	MONITOR	●	●	●
	DATA ENTRY	LIGHT-PEN	LIGHT-PEN	KEYBOARD
	OTHER			
TYPICAL SYSTEM COST ($000)		35-40	35-40	20
MOST SUITABLE APPLICATIONS	INSPECTION MEASUREMENT	●		●
	VERIFICATION	●		●
	SURFACE INSPECTION			
	PART I.D. MATERIAL SORTING/HANDLING	●		
	CHARACTER RECOGNITION			
	BIN PICKING			
	CONTROL WELDING		●	
	PROCESSING/ MACHINING			
	FASTENING/ ASSEMBLY			

REFERENCES

1. Besl, Paul and Ramash Jain, "Range Image
Understanding", *IEEE Computer Society Conference on
Computer Vision and Pattern Recognition*, June 1985, pp.
430-449.

2. Boven, Geert Van, Patrick Wambacq, Andre Oosterlinck,
H. Van den Berghe, "Industrial Image Computer Systems:
A Comparative Study", *Architecture and Algorithm for
Digital Image Processing*, 1983, pp. 229-242.

3. Callahan, James and Richard Weiss, "A Model for
Describing Surface Shape", *IEEE Computer Society
Conference on Computer Vision and Pattern Recognition*,
June 1985, pp. 240-245.

4. Chen, Chi-hau, Statistical Pattern Recognition, Spartan
Books, 1973.

5. Cohen, Paul R., and Edward A. Feigenbaum, The Handbook
of Artificial Intelligence, Vol. 3, HeurisTech Press,
1982.

6. Cunningham, Robert, "Segmenting Binary Images",
Robotics Age: In the Beginning, 1983, pp. 82-101.

7. Foroutan, I. and J. Sklansky, "Feature Selection for
Piecewise Linear Classifiers", *IEEE Computer Society
Conference on Computer Vision and Pattern Recognition*,
June 1985, pp. 306-311.

8. Fu, King Sun, "Pattern Recognition for Automatic Visual
Inspection", SPIE Vol. 336, *Robot Vision*, 1982, pp.
12-19.

9. Fu, King Sun, Syntactic Methods in Pattern Recognition,
Vol. 112, Academic Press, 1974.

10. Fu, King Sun, Syntactic Pattern Recognition and
Applications, Prentice-Hall, Inc., 1982.

11. Gloriosa, Robert M. and Fernando C. Colon Osorio,
Engineering Intelligent Systems Concepts, Theory, and
Applications, Digital Press, 1980.

12. Gonzalez, R. C. and R. Safabakhsh, "Computer Vision
Techniques for Industrial Inspection and Robot Control:
A Tutorial Overview", *Tutorial on Robotics*, IEEE
Computer Society Press, 1983, pp. 299-324.

13. Hattich, W., "Recognition of Overlapping Workpieces by Model-Directed Construction of Object Contours", _Artificial Vision for Robots_, Kogan Page, 1983, pp. 77-92.

14. Hildreth, Ellen C., "Edge Detection in Man and Machine", _Robotics Age: In the Beginning_, 1983, pp. 136-142.

15. Kitchin, P. W. and Alan Pugh, "Processing of Binary Images", _Robot Vision_, Springer-Verlag, 1983, pp. 21-42.

16. Kjell, Bradley P. and Charles R. Dyer, "Edge Separation and Orientation Texture Measures", _IEEE Computer Society Conference on Computer Vision and Pattern Recognition_, June 1985, pp. 306-311.

17. Poggio, Tomaso, "Vision by Man and Machine", _Scientific American_, Vol. 250, No. 4, April 1984, pp. 106-116.

18. Rosenfield, Azriel, Picture Processing by Computer, Academic Press, 1969.

19. Rosenfield, Azriel and Avinash C. Kak, Digital Picture Processing, Vol. 1, Academic Press, 1976.

20. Rosenfield, Azriel and Avinash C. Kak, Digital Picture Processing, Vol. 2, Academic Press, 1982.

21. Schowengerdt, Robert A., Techniques for Image Processing and Classification in Remote Sensing, Academic Press, 1983.

22. Segen, Jakub, "Learning Structural Descriptions of Shape", _IEEE Computer Society Conference on Computer Vision and Pattern Recognition_, June 1985, pp. 96-99.

23. Simon, Herbert A. and Laurent Siklossy, Representation and Meaning: Experiments with Information Processing Systems, Prentice-Hall, Inc., 1972.

24. Tech Tran Corporation, "Machine Vision Systems: A Summary and Forecast", _Tech Tran Corporation_, 1983.

25. Ullmann, J. R., Pattern Recognition Techniques, Crane, Russak & Company, Inc., 1973.

26. West, Perry C., "Machine Vision in Practice", _IEEE Transactions on Industry Applications_, Vol. IA-19, No. 5, September/October 1983, pp. 794-801.

BIBLIOGRAPHY

Augesteijn, Marijke F. and Charles R. Dyer, "Model-Based Shape From Contour and Point Patterns", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 1985, pp. 306-311.

Berger, Michael, "Robots in Japan: Leading in Implementing But Lacking Smarts", Electronics, November 17, 1983, p. 122.

Besl, Paul and Ramash Jain, "Range Image Understanding", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 1985, pp. 430-449.

Boven, Geert Van, Patrick Wambacq, Andre Oosterlinck, H. Van den Berghe, "Industrial Image Computer Systems: A Comparative Study", Architecture and Algorithm for Digital Image Processing, 1983, pp. 229-242.

Braggins, Don, "Robots Sharpen Up Their Vision", New Scientist, December 15, 1983, p. 811.

Callahan, James and Richard Weiss, "A Model for Describing Surface Shape", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 1985, pp. 240-245.

Chang, Ning-San, Image Analysis and Image Database Management, UMI Research Press, 1981.

Chen, Chi-hau, Statistical Pattern Recognition, Spartan Books, 1973.

Cipovic, H., D. Milivojevic, Z. Kajmakovic, "Planar Object Recognition by the Computer Vision Method", Intelligent Robots: Third International Conference on Robot Vision and Sensory Controls, November 1983, pp. 9-16.

Cohen, Paul R., and Edward A. Feigenbaum, The Handbook of Artificial Intelligence, Vol. 3, HeurisTech Press, 1982.

Cunningham, Robert, "Segmenting Binary Images", Robotics Age: In the Beginning, 1983, pp. 82-101.

Denker, Dr. Stephen P., "Choosing a Vision System That Improves Testing Effectiveness", Test & Measurement World, October 1984, pp. 192-200.

Faedo, W., L. Foulloy, and R. B. Kelley, "Three Stage
    Algorithm for Recognizing Stable States and Reorienting
    a Workpiece", Intelligent Robots: Third International
    Conference on Robot Vision and Sensory Controls,
    November 1983, pp. 448-456.

Foroutan, I. and J. Sklansky, "Feature Selection for
    Piecewise Linear Classifiers", IEEE Computer Society
    Conference on Computer Vision and Pattern Recognition,
    June 1985, pp. 306-311.

Fu, King Sun, "Pattern Recognition for Automatic Visual
    Inspection", SPIE Vol. 336, Robot Vision, 1982, pp.
    12-19.

Fu, King Sun, Syntactic Methods in Pattern Recognition, Vol.
    112, Academic Press, 1974.

Fu, King Sun, Syntactic Pattern Recognition and
    Applications, Prentice-Hall, Inc., 1982.

Geisler, Wolfgang, "A Versatile Data Base for A Robot Vision
    System", Intelligent Robots: Third International
    Conference on Robot Vision and Sensory Controls,
    November 1983, pp. 632-636.

Gloriosa, Robert M. and Fernando C. Colon Osorio,
    Engineering Intelligent Systems Concepts, Theory, and
    Applications, Digital Press, 1980.

Gonzalez, R. C. and R. Safabakhsh, "Computer Vision
    Techniques for Industrial Inspection and Robot Control:
    A Tutorial Overview", Tutorial on Robotics, IEEE
    Computer Society Press, 1983, pp. 299-324.

Hattich, W., "Recognition of Overlapping Workpieces by
    Model-Directed Construction of Object Contours",
    Artificial Vision for Robots, Kogan Page, 1983, pp.
    77-92.

Herbert, M. and T. Kanade, "The 3-D Profile Method for
    Object Recognition", IEEE Computer Society Conference
    on Computer Vision and Pattern Recognition, June 1985,
    pp. 458-463.

Hildreth, Ellen C., "Edge Detection in Man and Machine",
    Robotics Age: In the Beginning, 1983, pp. 136-142.

Hobrough, Gilbert and Theodore Hobrough, "Stereopsis for Robots by Iterative Stereo Image Matching", Intelligent Robots: Third International Conference on Robot Vision and Sensory Controls, November 1983, pp. 94-102.

Kitchin, P. W. and Alan Pugh, "Processing of Binary Images", Robot Vision, Springer-Verlag, 1983, pp. 21-42.

Kjell, Bradley P. and Charles R. Dyer, "Edge Separation and Orientation Texture Measures", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 1985, pp. 306-311.

Kovalevsky, V. A., Image Pattern Recognition, Springer-Verlag, 1980.

Kuan, Darwin T., "Three-Dimensional Vision System for Object Recognition", Intelligent Robots: Third International Conference on Robot Vision and Sensory Controls, November 1983, pp. 366-372.

Lowe, David G., Perceptual Organizations and Visual Recognition, Kluwer Academic Publishers, 1985.

Masuda, Isao, Norihiro Hayama, Teruo Akiyama, Tokiichiro Takahashi, and Seiichiro Naito, "Approach to Smart Document Reader System", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 1985, pp. 550-557.

McFarland, William D. and Robert W. McLaren, "Problems in Three-Dimensional Imaging", Intelligent Robots: Third International Conference on Robot Vision and Sensory Controls, November 1983, pp. 148-157.

McPherson, Charles A., "Three-Dimensional Robot Visions", Intelligent Robots: Third International Conference on Robot Vision and Sensory Controls, November 1983, pp. 116-126.

Neveu, Charles F., Charles R. Dyer, and Roland T. Chin, "Object Recognition Using Hough Pyramids", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 1985, pp. 328-333.

Niemann, H., Pattern Analysis, Springer-Verlag, 1981.

Poggio, Tomaso, "Vision by Man and Machine", Scientific American, Vol. 250, No. 4, April 1984, pp. 106-116.

Porter, G. B. III and J. L. Mundy, "Visual Inspection System Design", _Tutorial on Robotics_, IEEE Computer Society Press, 1983, pp. 338-346.

Rearick, T. C., "A Texture Analysis Algorithm Inspired by a Theory of Preattentive Vision", _IEEE Computer Society Conference on Computer Vision and Pattern Recognition_, June 1985, pp. 312-317.

Reeves, Anthony P., Richard J. Prokop, and Russell W. Taylor, "Shape Analysis of Three Dimensional Objects Using Range Information", _IEEE Computer Society Conference on Computer Vision and Pattern Recognition_, June 1985, pp. 452-457.

Rosenfeld, Azriel, Picture Processing by Computer, Academic Press, 1969.

Rosenfeld, Azriel and Avinash C. Kak, Digital Picture Processing, Vol. 1, Academic Press, 1976.

Rosenfeld, Azriel and Avinash C. Kak, Digital Picture Processing, Vol. 2, Academic Press, 1982.

Schowengerdt, Robert A., Techniques for Image Processing and Classification in Remote Sensing, Academic Press, 1983.

Segen, Jakub, "Learning Structural Descriptions of Shape", _IEEE Computer Society Conference on Computer Vision and Pattern Recognition_, June 1985, pp. 96-99.

Shapiro, Sydney F., "Vision Expands Robotics Skills for Industrial Applications", _Computer Design_, September 1979, pp. 78-90.

Simon, Herbert A. and Laurent Siklossy, Representation and Meaning: Experiments with Information Processing Systems, Prentice-Hall, Inc., 1972.

Tech Tran Corporation, "Machine Vision Systems: A Summary and Forecast", _Tech Tran Corporation_, 1983.

Thompson, Alan M., "Introduction to Robot Vision", _Robotics Age: In the Beginning_, 1983, pp. 82-101.

Ullmann, J. R., Pattern Recognition Techniques, Crane, Russak & Company, Inc., 1973.

West, Perry C., "Machine Vision in Practice", _IEEE Transactions on Industry Applications_, Vol. IA-19, No. 5, September/October 1983, pp. 794-801.

CONCEPTS OF AUTOMATIC PATTERN RECOGNITION

IN COMPUTER VISION

by

Navid Nikravan

B.S. Kansas State University, 1981

AN ABSTRACT OF A MASTER`S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Computer Science

Kansas State University

Manhattan, Kansas

1987

ABSTRACT

Computer vision/pattern recognition is in its early developmental stages. Presently, a number of limitations do exist in designing a complete and reliable pattern recognition system. These limitations can be overcome by introducing new concepts of automatic pattern recognition in computer vision.

This report begins with the description of the overall structure of an automatic pattern recognition system. It deals with considerations and requirements for designing an automatic pattern recognition system and for selecting the type of data the system is designed to handle. Human intelligence and interaction as a guide are indeed required for further advancements in artificial intelligence, specifically pattern recognition. The latter portion of this report presents and analyzes an approach for the processing of binary images and their recognition process. The recognition process is facilitated through parameters that are derived from an object in order for the recognition process to proceed. These parameters used in the recognition process are obtained through feature extraction.

The future trends and developments of pattern recognition systems are discussed. Some ideas regarding the future work of object pattern recognition systems are also presented.