Scholars' Mine

Doctoral Dissertations   Student Theses and Dissertations

1971

Modeling the visual pathway for interactive diagnosis of visual fields

Chiam Geoffrey Goldbogen

Follow this and additional works at: https://scholarsmine.mst.edu/doctoral_dissertations

Part of the Mathematics Commons

Department: Mathematics and Statistics

MODELING THE VISUAL PATHWAY FOR INTERACTIVE

DIAGNOSIS OF VISUAL FIELDS

by

CHIAM GEOFFREY GOLDBOGEN, 1944-

A DISSERTATION

Presented to the Faculty of the Graduate School of the

UNIVERSITY OF MISSOURI-ROLLA

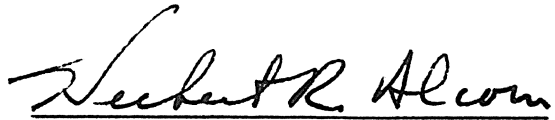In Partial Fulfillment of the Requirements for the Degree
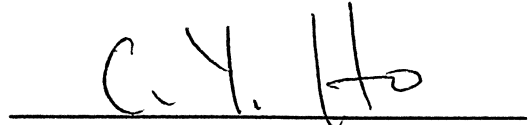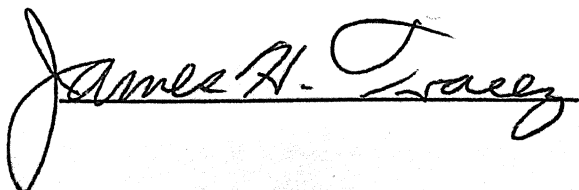
DOCTOR OF PHILOSOPHY

in

MATHEMATICS

1971

Advisor

ABSTRACT

Visual fields are an important tool for the ophthalmologist in the detection, diagnosis, and monitoring of certain diseases and maladies of the visual pathway. The aim of the present research is to build a computer system which utilizes a learning machine to develop a mathematical model of the visual pathway. It is hoped that this system may be used in the field of ophthalmology as a teaching aid, or may assist in various aspects of diagnosis. Faults corresponding to blind or impaired areas of visual fields are extracted from medical records of a patient's condition. The structure of the model allows both forward and backward simulation of the faults in a manner related to the multidimensional path sensitizing technique utilized in the diagnosis of digital systems. An important feature of the method is close manmachine interaction, accomplished with the aid of a graphic display, which enables the path sensitizing and the learning to be observed as it progresses.

PREFACE

This work grew out of a study and proposal of an automated visual

field system by Mr. B. Shelman at Argonne National Laboratory.[1] The

system he proposed considered "...automation in the preparation, use

and retrieval of (visual field) patterns...with close man-machine in-

teraction." After the initial study, design of a complete system was

postponed until a study could be made to evaluate the classification

of the visual fields for retrieval purposes. I was assigned this task

during a summer research appointment at Argonne National Laboratory in

1969. During my initial research[2] it occurred to me that I could use

the path sensitizing technique, which I studied under Dr. Szygenda,[3]

in dealing with maladies of the visual pathway. In this regard I modi-

fied the goal of the research from the classification task to a model-

building and diagnosis task with the following objectives:

1) Build a learning machine with a practical application.

2) Develop a good mathematical model of the visual pathway.

3) Use path sensitizing in a new diagnostic application —
   topological diagnosis of a biological neural net.

4) Develop a diagnostic aid for the trained professional.

5) Build a tutorial and instructional tool in the study of
   ophthalmology.

6) Develop a new form of learning machine and implement it
   with original algorithms.

7) Find further applications of this work.

The result of this research, system VISUAL, is useful as a stand-

alone system; however, its design will allow it to be immersed in an

"automated visual field system." A brief description of the system

VISUAL was first presented at the IEEE Conference on Feature Extraction and Selection in Pattern Recognition in October 1970.[4]

Chapter I of this dissertation is entirely introductory. Chapter II is designed to give the mathematicians, engineers, and computer scientists enough background on the anatomy of the visual pathway and the definition of the visual fields to understand the system. Medical people will probably want to omit this chapter. Chapter III describes the model. The model is a unifying concept in all facets of the system. A paper[5] delivered in March 1971 at the Second Annual Pittsburgh Conference on Modeling and Simulation describes many of the modeling considerations that are presented in this chapter. Next, Chapter IV presents an overview of the system.

Chapter V introduces the learning machine structure, which I feel is one of the most original contributions of this work. I hope that work on similar applications will be done using the concepts established here. Finally, the results and conclusions are presented in Chapter VI.

I felt that some details which were not essential to the basic understanding of the system but yet important to an in-depth study of this research should be placed in the appendices. This information is necessary to those researchers who may wish to further this study. Appendix A establishes mathematically the formal structure of the learning machine employed in VISUAL. Reference in this appendix is made to the mathematical structure presented by Nilsson[6] so that his structure can be compared and contrasted with the structure used in VISUAL. Appendix B details in flowchart form two of the processes — radius selection and diagnosis — described in the main body of the

paper. Appendix C lists some approaches of learning that during a time in this research were considered, but were rejected for this application. I feel that in another application perhaps one of these algorithms will be effective and hence should be included in this work. Appendix D describes and flowcharts the details, concepts, and techniques used in the nerve fiber bundle movement of the learning machine.

At present, it is felt that further field testing of the system needs to be performed before all aspects of the system performance can be judged. To this end, I am preparing forthcoming reports, including a Users Manual and Field Studies as well as a System Documentation. These reports will provide excellent companions to this work in completing this study.

In the final essence, VISUAL is a research tool and a vehicle for the study of interactive graphics, path sensitizing applications, and learning machines. Although it does have value as a diagnostic and teaching aid, to become financially and operationally practical, the system must be adapted for a computer other than the present one employed here at Argonne National Laboratory.

typing and editing. Most of all, to my wife, Deirdre, for her patience and moral support throughout my graduate years — my greatest thanks.

TABLE OF CONTENTS

Table of Contents (continued)   Page

LIST OF ILLUSTRATIONS

List of Illustrations (continued)

List of Illustrations (continued)

LIST OF TABLES

Chapter I. INTRODUCTION

A patient's visual field contains important information that an ophthalmologist uses in the detection, diagnosis, and monitoring of certain diseases and maladies of the visual pathway. This paper describes a computer system, VISUAL, that the ophthalmologist may use interactively to aid him to understand a particular patient's condition. Secondly, the system may be used as an ophthalmological teaching aid. At present, only tumors and lesions that affect the fibers in the chiasm are specifically dealt with; however, the straightforward extension of the method to other diseases (in other locations of the visual pathway) is indicated.

Before the system can be used as a diagnostic aid, a model must be developed. This is accomplished with a learning machine, which will be described.

It is necessary to outline two fields of background so that the design of the model can be understood. First, a discussion is presented of the important aspects of the anatomy of the visual pathway, the functioning of the fiber bundles in the transmission of visual forms, and the effect of diseases on the fibers in the visual pathway and on the visual fields. Secondly, an explanation of path sensitizing and its role in a simulation and diagnostic environment is presented. After these preliminaries, heuristic arguments for the model design can be examined in detail.

At this point the over-all system is described. The desirability of a graphical display in conjunction with a learning machine is an important consideration in determining the computer environment necessary to support the envisioned system. In fact, this necessity has required

that the proposed research be conducted under the direction of the Argonne Universities Association at Argonne National Laboratory, where special graphical and computer facilities are available. The system is described independently of the equipment, but the special equipment used is indicated in Chapter IV.

The software operates in six modes, most importantly in learning mode, display mode, and diagnostic mode. Operating in learning mode, the learning machine builds and improves the model; in diagnostic mode, the system acts as a diagnostic aid; and in display mode, the system performs as an ophthalmological teaching tool. All three modes refer to the model during decision processes, learning processes, and display processes, and are therefore dependent on the status of the model for the quality of their performance. During the development of the model, it is convenient to use the modes of diagnosis and display to monitor the progress of learning because in these modes a good graphical picture of the model can be seen.

The learning machine incorporates much of the state-of-the-art theory in pattern recognition and learning machines. It is at the same time a type of learning machine that differs from the classical form. This new learning machine is called the "adapted machine."

One of the main features of the research is the combination of three separate areas of specialization: path sensitizing, learning machines, and medical diagnosis.

Chapter II.   VISUAL FIELDS AND THE VISUAL PATHWAY

A.   Introduction to Visual Fields

Harrington[7] defines the visual field as "that portion of space in which objects are simultaneously visible to the steadily fixating eye. It is somewhat more than one-half of a hollow sphere, situated before and around each eye of the observer, within which objects are perceived while the eye is fixating at a stationary point on its inner surface."

The stationary point is called the point of fixation, while the straight line extending from the eye to the point of fixation is the visual axis.

The interested reader is encouraged to explore the apt description of visual acuity of the visual field as "an island hill of vision surrounded by the sea of blindness" for a thorough understanding of visual perception.[7] The hill image comes from the fact that visual acuity is not uniform within the hollow sphere, but sharper and more sensitive near the point of fixation, and less sensitive away from the point. The hill of vision is shaped like a pointed mound, with the highest elevation of the hill in the center where visual acuity is greatest.

A patient's visual perception is recorded on charts by examination techniques of perimetry.[7] The resultant record is called the visual field. The chart has a reference grid as shown in Figure 1. The chart has two sets of three concentric circles, one set for each eye. The center of a set of circles corresponds to the point of fixation of the eye. The circles correspond to regions of points of perception $10°$, $20°$, and $30°$ from the visual axis. (These angles are defined with vertex at the eye, one edge the visual axis and the other edge the line from the eye to the point of perception.)
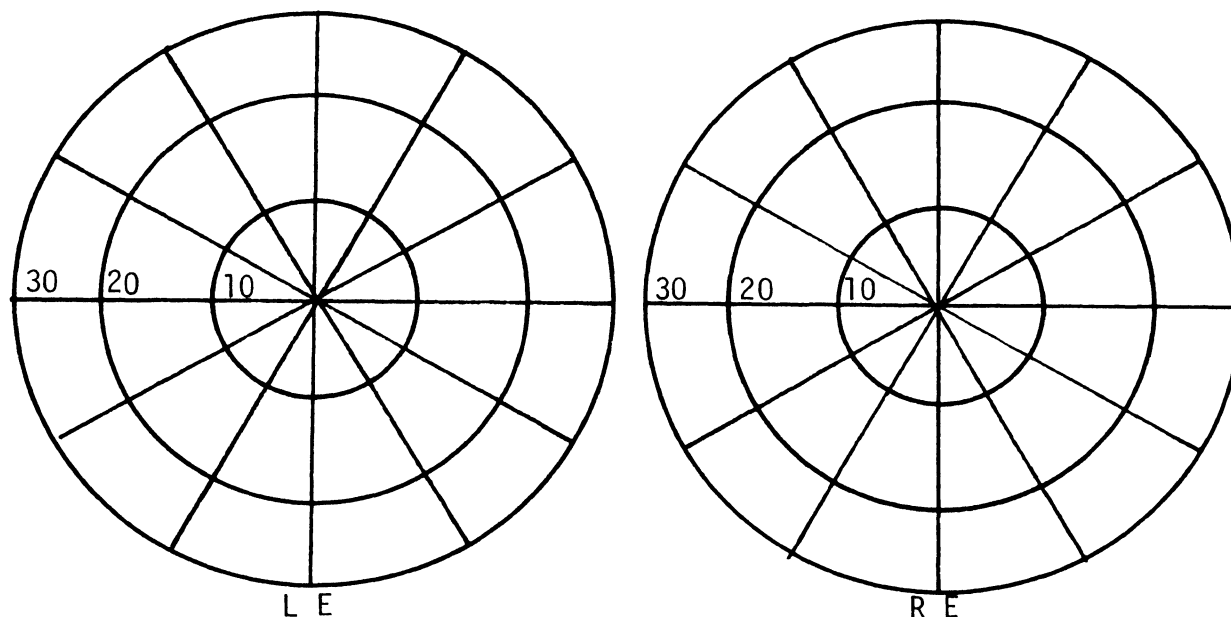
Figure 1. Chart for Recording Visual Field
Using Tangent Screen Examination.

The chart shown in Figure 1 is a chart used in the tangent screen

examination. Although there are other means and instruments for perim-

etry (notably the Goldman Perimeter[7]), discussion of the tangent screen

examination is sufficient for the purposes of this paper. The patient

sits in front of a black screen, with one eye covered. The seeing eye

is fixated at a point on the tangent screen. The examiner takes a test

object (a white ball on the end of a long black wand) and moves the

test object to various points on the tangent screen (see Figure 2) not-

ing where the object is perceived and where it is not. This process is

continued until the examiner has mapped out a contour on the chart sur-

rounding the seeing area. This is a contour of one level of perception

of the island hill of vision. These contours, called *isopters*, show

the region of perception of a test object that subtends a constant

visual angle. Clinically, the isopter is measured in terms of d, the

distance between the eye and the test object, and S, the size of the

object, both measured in millimeters and expressed as a fraction, S/d.
A test object of size S and distance d subtends a fixed visual angle
throughout the visual field. Table I lists some typical isopter sizes
and the corresponding visual angle produced by the test object.

Notice that the range of the normal visual field varies with
the visual angle. The last column in Table I will be explained in
Chapter V.

Table I.

ISOPTERS AND ASSOCIATED NORMAL VISUAL FIELDS

Visual Angle	Isopter	Extent in Degrees of Normal Visual Field				Radius-Indices Masked
		Temporal	Nasal	Inferior	Superior	
10.32'	1/330	80	55	60	50	None
3.42'	1/1000	25	25	25	25	9
6.84'	2/1000	26	26	26	26	9
10.20'	3/1000	38	30	30	26	None
1.70'	1/2000	24	24	24	24	8,9
5.10'	3/2000	30	30	30	24	None
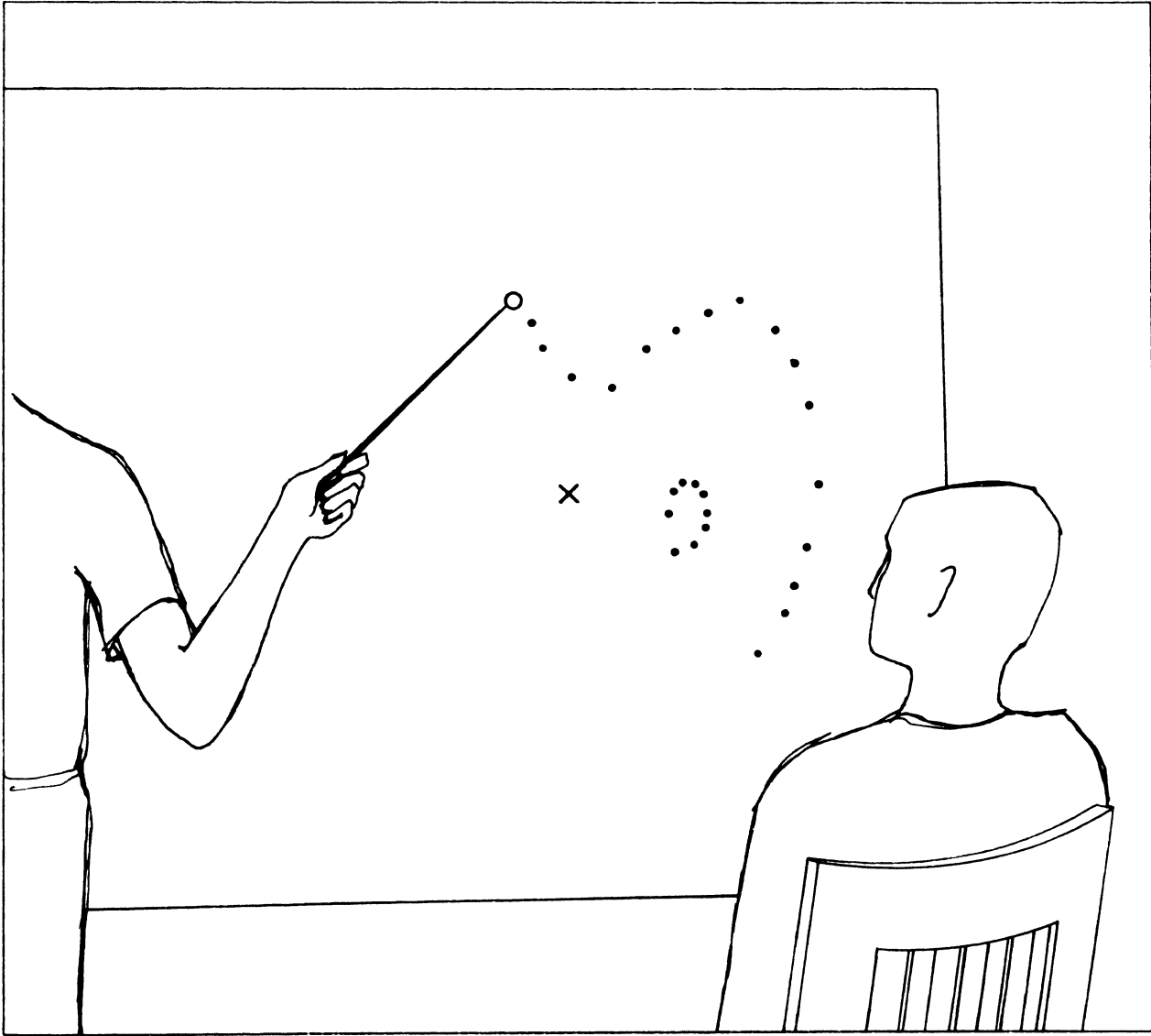
Figure 2. Visual Field Examination Using Tangent Screen.

Figure 3 shows a sample record of a visual field taken with a tangent screen. Two isopters were measured and charted with the anoptic portions of the field shaded. In the visual field of the right eye, 3mm test objects on a tangent screen at a 2000mm distance were visible to the patient except in the upper right quadrant of the field. The smaller 1mm objects were again not visible in the upper right quadrant, and furthermore, not visible in portions of the lower right quadrant of
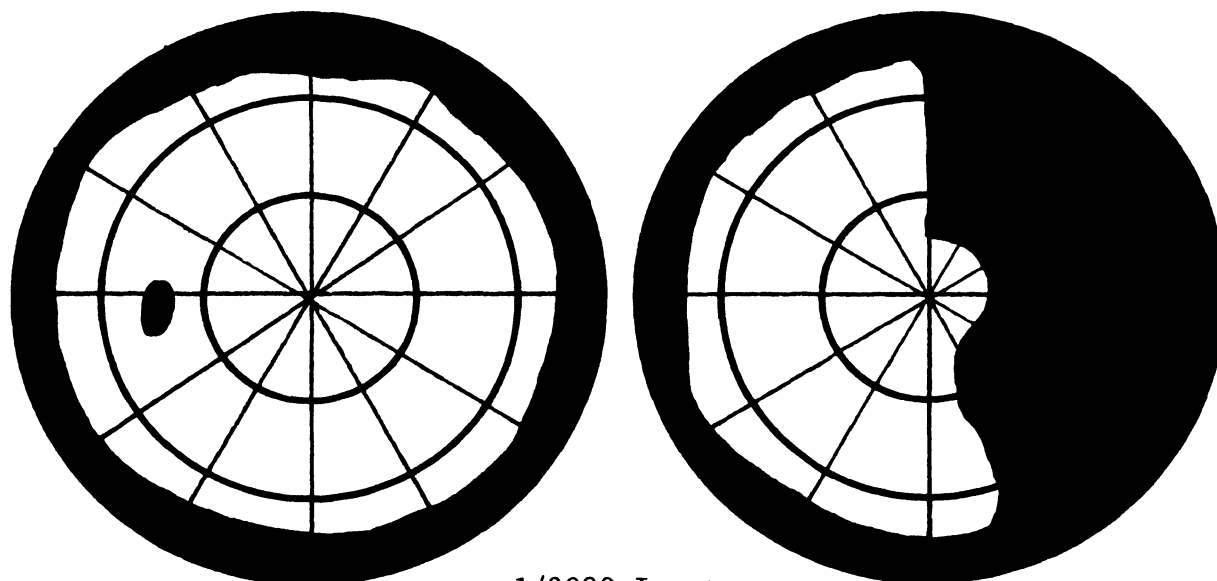
1/2000 Isopter
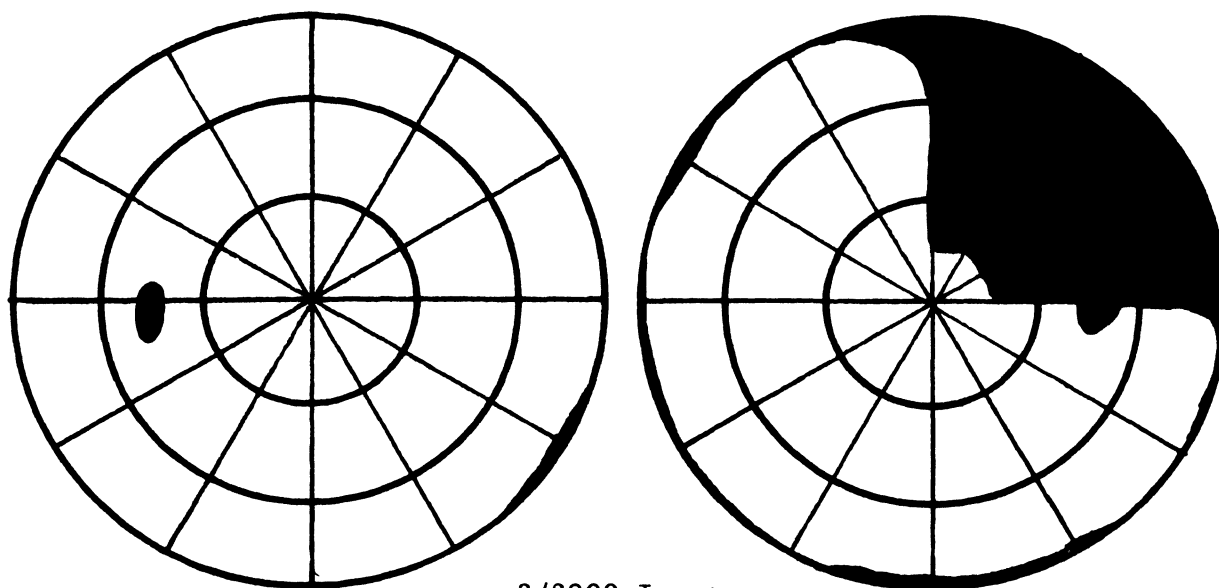
3/2000 Isopter

Figure 3. Sample Visual Field.

the right visual field. The left eye has normal vision for both isopters. Notice the normal blind spot in each visual field. Since, as illustrated in this example, there is a substantial difference in shape of the visual fields from one isopter to another, a thorough visual field test consists of more than one isopter.

Many factors affect the accuracy of the information obtained in the visual field examinations. The two factors most vulnerable to variation are the examiner and the patient. Because of his physical condition, the patient may not respond properly to the examiner's inquiries and may signify incorrectly where the regions of perception are. The technique of the examiner, his thoroughness, and precision, affect the data. Other variables which can be controlled more readily are illumination of the room, test objects, and screen. Finally, the eye fixation of the patient is an important consideration. During the examination, the examiner must continually insure that the patient's eye is fixated at the fixation point.

Recently, some attempts to automate (and hopefully improve) the data gathering of visual field records have been made. The author is familiar with two approaches. B. Shelman's work at Argonne National Laboratory preceded this research. The approach was essentially along the lines of an automated tangent screen.[1] Dr. J. Lynn has attempted to perform the data gathering by using a cathode ray tube display driven by an IBM 1130.[8] The tube serves the function of a very close tangent screen, while the computer controls the intensity and location of test dots on the tube. Whether either system will improve the accuracy of the present technology remains to be seen.

B. <u>Anatomy and Physiology of the Visual Pathway</u>

The anatomy of the visual pathway is very complex.[7,9-13] The physiology, or the functioning, of the pathway is correspondingly complex and, unfortunately, there are some important processes of the miracle of visual that are not yet understood. This section introduces some basics in the anatomy and physiology of the visual system so that the mathematical model presented in Chapter III can be compared to the physical counterpart. It should be remembered that the model developed in Chapter III does not attempt to simulate all the physiological and psychological aspects of vision, but rather to simulate a special phase in the process of vision — the transmission of data through the visual pathway.

The anatomy of the eye is well known. Important to understanding the visual fields is the structure of the eye. The eye inverts an image in the same manner as a pinhole camera. This inverted image activates and stimulates photoreceptors in the retina. At the retina, the image is transformed by several layers of nerve cells to electric impulses that are carried through the optic nerve and chiasm to the lateral geniculate body. There another transformation of electrical impulses is performed, and the new impulses are carried through the postchiasmal pathway to the striate cortex of the occipital lobes of the brain. (Refer to Figure 4.)

In Figure 5, cells from each of the layers of the retina are represented. The first layer of cell contains the photoreceptors (rods and cones); the bipolar and horizontal cells are in the next layer, along with amacrine cells; the final layer is composed of the ganglion cells whose axons form the optic nerve. It is not important to this

work to discuss in detail the functioning of the retina; however, one should note that in the retina the mapping of (transformation of) signals performed by the bipolar cells of the photoreceptors to the ganglion cells is many-to-many. Combining this property with the functioning of the amacrine and horizontal cells, the fibers in the optic nerve transmit no less than three types of signals: "on" signals, "off" signals, and "on-off" signals.

Because of the many-to-many mapping transformation, it is convenient to refer to the receptive field of a corresponding on (or off) fiber in the optic nerve. The receptive field for an "on" fiber, for example, is shown in Figure 6. The "on" signal is initiated by photoreceptors in the center circle, the region of greatest sensitivity. The "off" signal is initiated by the receptors in the outer annulus, while the intermediate region evokes the "on-off" signal.

The anatomy of the visual pathway varies microscopically and to some extent macroscopically from individual to individual. For the purposes of this work, the variation is not significant. Also, the symmetry of the fiber arrangement in the pathway is considered not sensitive to the degree of variation of the fibers. Figure 7 shows a schematic representation of the visual pathway showing how the temporal retinal fibers remain on the same side of the visual pathway, while the nasal fibers cross over in the chiasm to the opposite side of the visual pathway. A special name is given to the fiber bundles whose receptive field senses objects near the center of the visual fields. The location of these "macular fibers" is an important feature of the anatomy. In Chapter III we shall examine the structure of the visual pathway and especially the chiasm, in more detail.
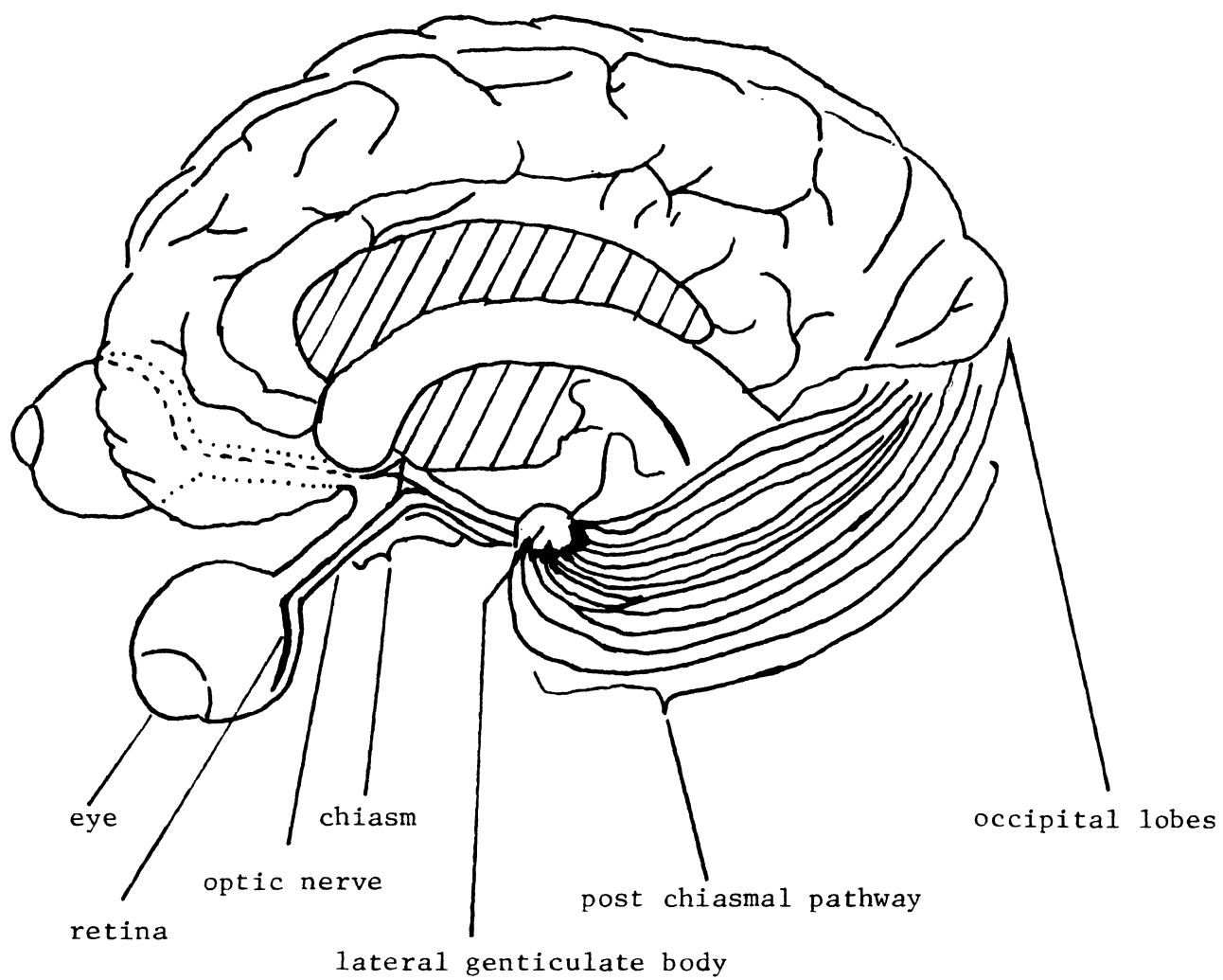
Figure 4. Architecture of the Visual Pathway.

nerve fiber layer

ganglion
cell

amacrine
cell

bipolar
cell

horizontal
cell

rod cell

cone cell
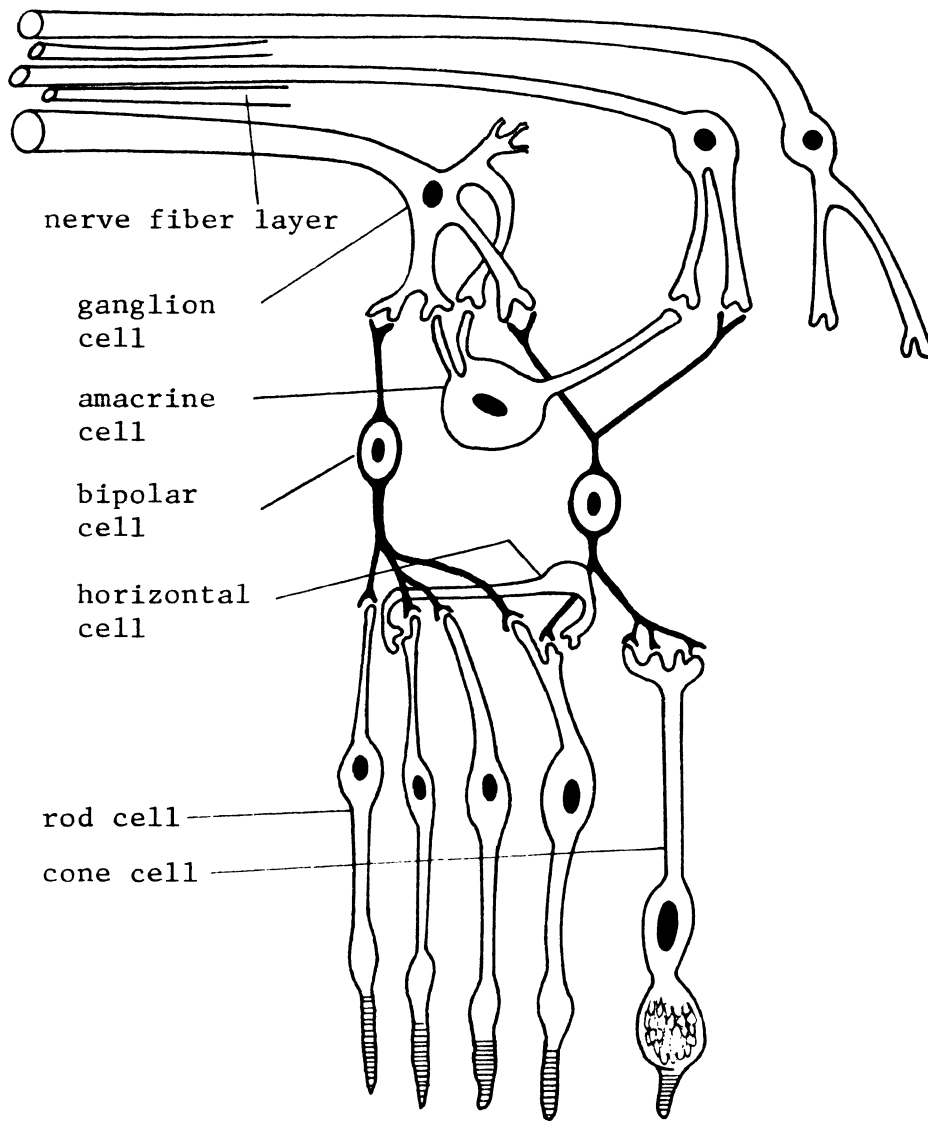
Figure 5. Anatomy of the Retina.
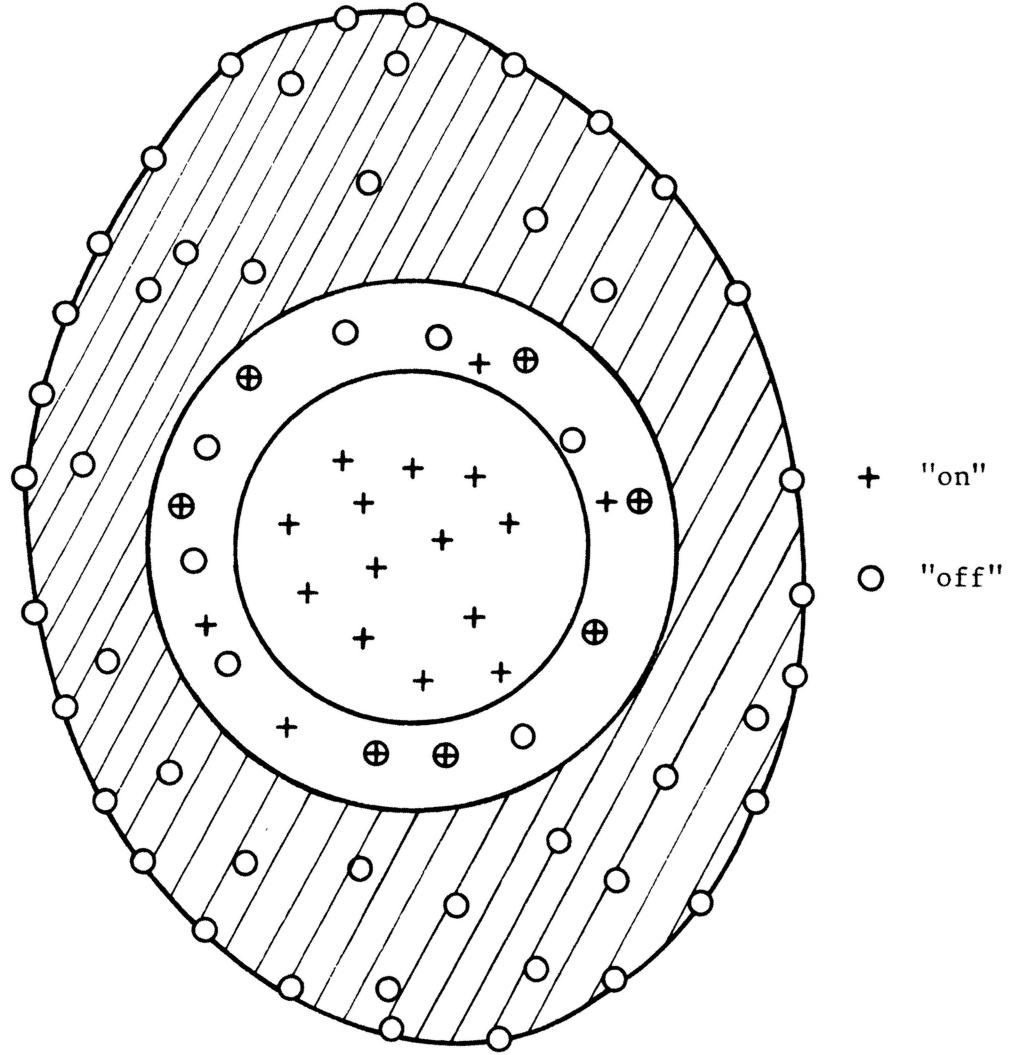
Figure 6. Organization of a Receptive Field.

C. Diseases of the Visual Pathway

Diseases of the visual pathway characteristically affect the visual field.[7] Diseases such as brain tumors in the vicinity of the visual pathway cause pressure on the fiber bundles in the pathway and disrupt the transmission of signals along the fibers. Figure 7 schematically indicates various sites of total disruption of fibers (shown with solid lines or shaded circle) and correspondingly the resultant abnormal visual field.

Example 1 in Figure 7 shows that the visual field on the side of the lesion is totally anoptic, while the contralateral field is normal. In example 2 of Figure 7, notice that the nasal retinal fibers of both eyes are damaged, and correspondingly the temporal half of the visual field for both eyes is anoptic. Further examination of this diagram shows how each particular area of damage affects the visual field uniquely and hence characteristically. For a more complete discussion, see Reference 7.
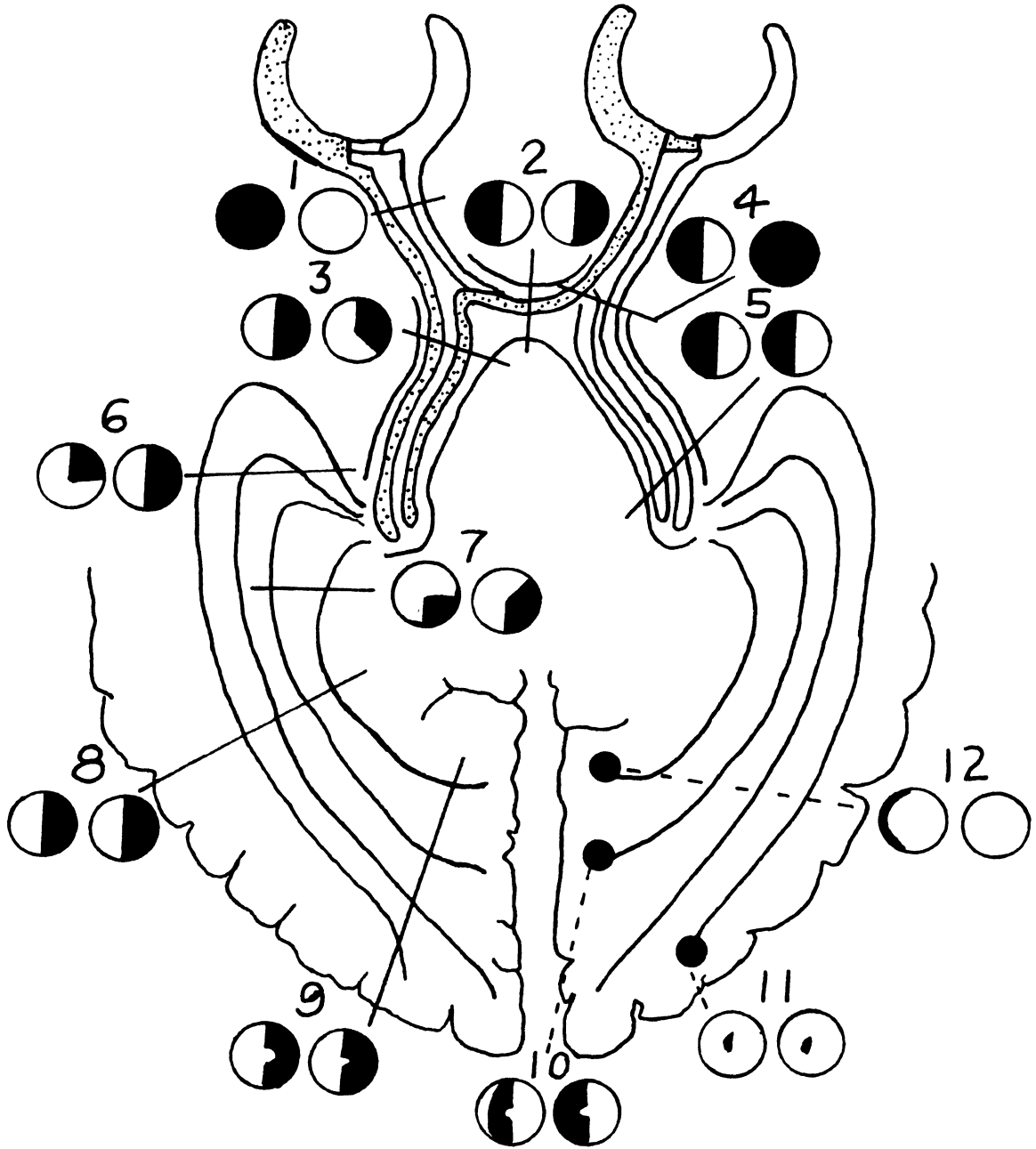
Figure 7. Visual Fields Resulting from Total
Fiber Disruption at Various Locations.

Chapter III.   DESIGN OF THE MODEL

In this chapter we mention first the elements of the model.   Then
we will explain path sensitizing and how the system uses that technique
in conjunction with the model.   At that point we can present heuristic
arguments for the selection of the position of various elements in the
model.

A.   Elements of the Model

There are three element types in the model: sense points, nerve
fibers, and c-sects.   Although there are thousands of nerve fiber
bundles, a set of 252 representative nerve fiber bundles was selected.
Each fiber bundle begins at a theoretical point at the retina, called
the sense point.   At various parts of the visual pathway, a plane cuts
the pathway.   The intersection of the plane and the pathway makes a
two-dimensional surface called a c-sect.   Information of the location
and orientation of the c-sect is contained in the model.   The fiber
bundles extend from their sense points back through the visual pathway
to the occipital lobes.   The location of the intersection of each nerve
fiber bundle with the c-sects it passes through is recorded in the list
structure of the model.

B.   Path Sensitizing in Digital Systems

Before describing the path sensitizing simulation process, basic
concepts of path sensitizing are outlined.   Because the path sensitiz-
ing technique has been well defined and studied in depth, and because
the philosophy of the technique rather than the method itself has been
used in the VISUAL system, path sensitizing is reviewed conceptually
as a test generation technique.

The importance of diagnostics for digital systems increased from the first appearance of electronic digital computers in the 1950's to the present with the complex computers of today's technology. One method of diagnostics is test generation. In this method the set of tests may be applied to a digital system, and from the results of the tests, diagnostics about the status of the digital system is made, the presence or absence of one or more particular faults may be determined, and possibly the location of the fault may be given. Path sensitizing is a method of generating the tests used primarily in combinational systems, although it has been used as a heuristic for sequential circuits.[3] The discussion here is restricted to combinational circuits. Let us define combinational digital system, D, as a circuit consisting of "and", "or", "nand", "nor", and "not" gates connected by lines with no feedback or memory. The only lines of the system whose values may be externally controlled are the primary inputs $(i_1, i_2, \ldots, i_N)$. Likewise, the only lines of the system whose values are externally evident are the primary outputs $(0_1, 0_2, \ldots, 0_p)$. That is why in describing a diagnostic test, the primary inputs need to be specified, and the results interpreted through the primary outputs.

Although Eldred[14] published perhaps the first paper on test generation for combinational circuits, the term path sensitizing was coined by Armstrong.[15] Armstrong's method is more precisely one-dimensional path sensitizing. The "sensitized path" is the path of propagation of the fault from the site of failure to a primary output. For example, consider the circuit (Figure 8) with line 3 stuck-at-one.
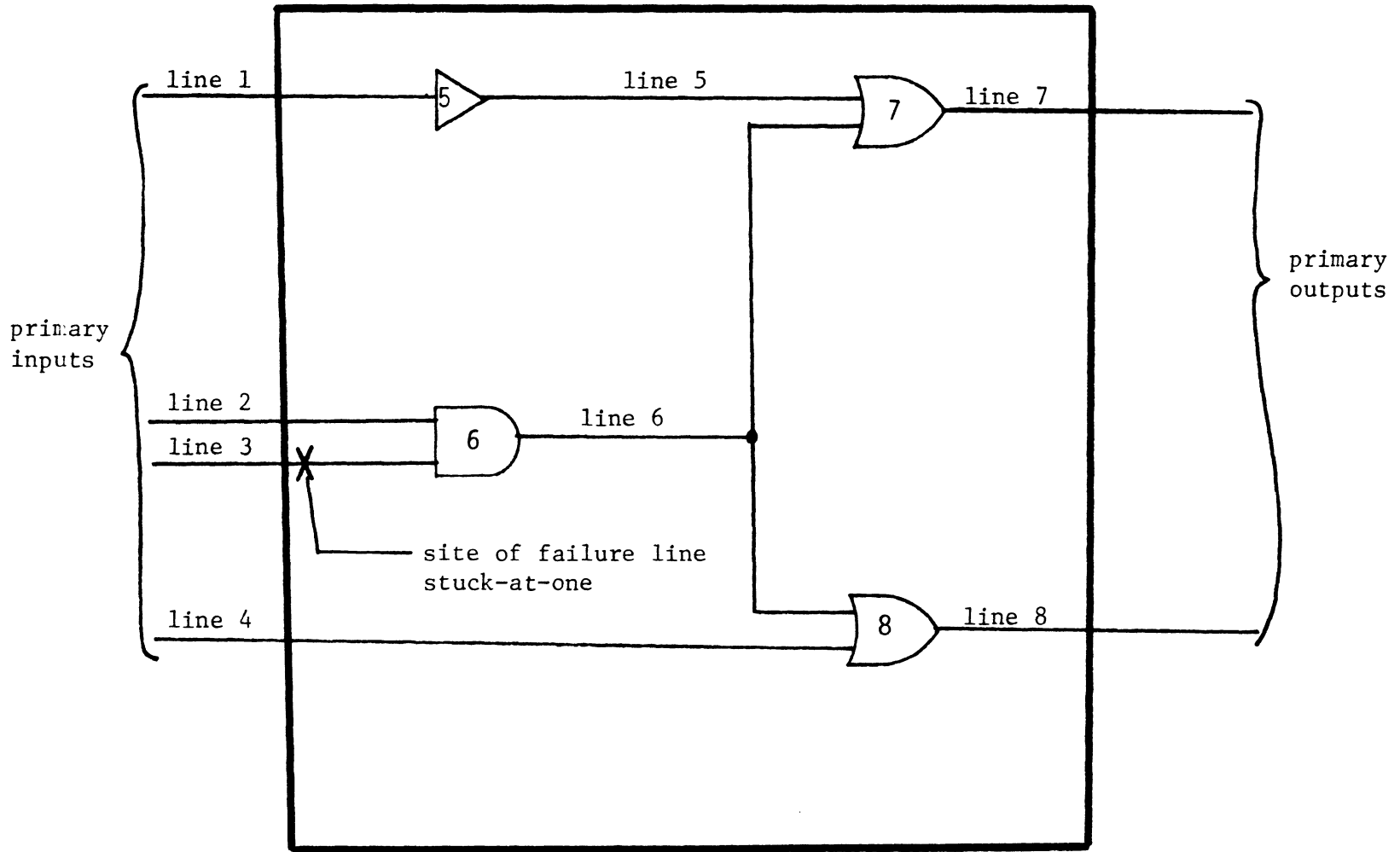
Figure 8. A Digital System with a Fault.

With input vector (1,1,0,1) the effect of the stuck-at-one fault
is propagated through line 3-6-7. Diagrammatically this can be illus-
trated (Figure 9) by using the symbol d to represent line value of 0
for a fault-free circuit, and line value of 1 for a circuit with the
fault.

In list form, the line values are

1	2	3	4	5	6	7	8
1	1	d	1	0	d	d	1

In this example, the sensitized path is 3-6-7. It is a one-dimensional
path since there is one path of propagation of the fault from the site
of the fault to a primary output. If we set the primary inputs at
(1100) however, the circuit looks like that shown in Figure 10.

The fault is propagated through two paths: lines 3-6-7 and lines
3-6-8. This is an example of two-dimensional path sensitizing, or,
more generally, multidimensional path sensitizing. In a general com-
binational circuit, it is necessary to check more than just the one-
dimensional path to determine whether a diagnostic test can be gen-
erated for a fault. This result was proven in Schneider's famous
example.[16] Roth formalized the mathematics[17] underlying the technique.
He called the process of propagating the fault toward the primary out-
puts, the forward trace. The backward trace was the term defined to be
the tracing of 0 and 1 line values to the primary inputs.

The work of Eldred, Armstrong, and Roth as outlined above was well
established, when in 1969 Szygenda and Goldbogen presented research[3]
from which the present work has borrowed philosophic and technical
concepts.
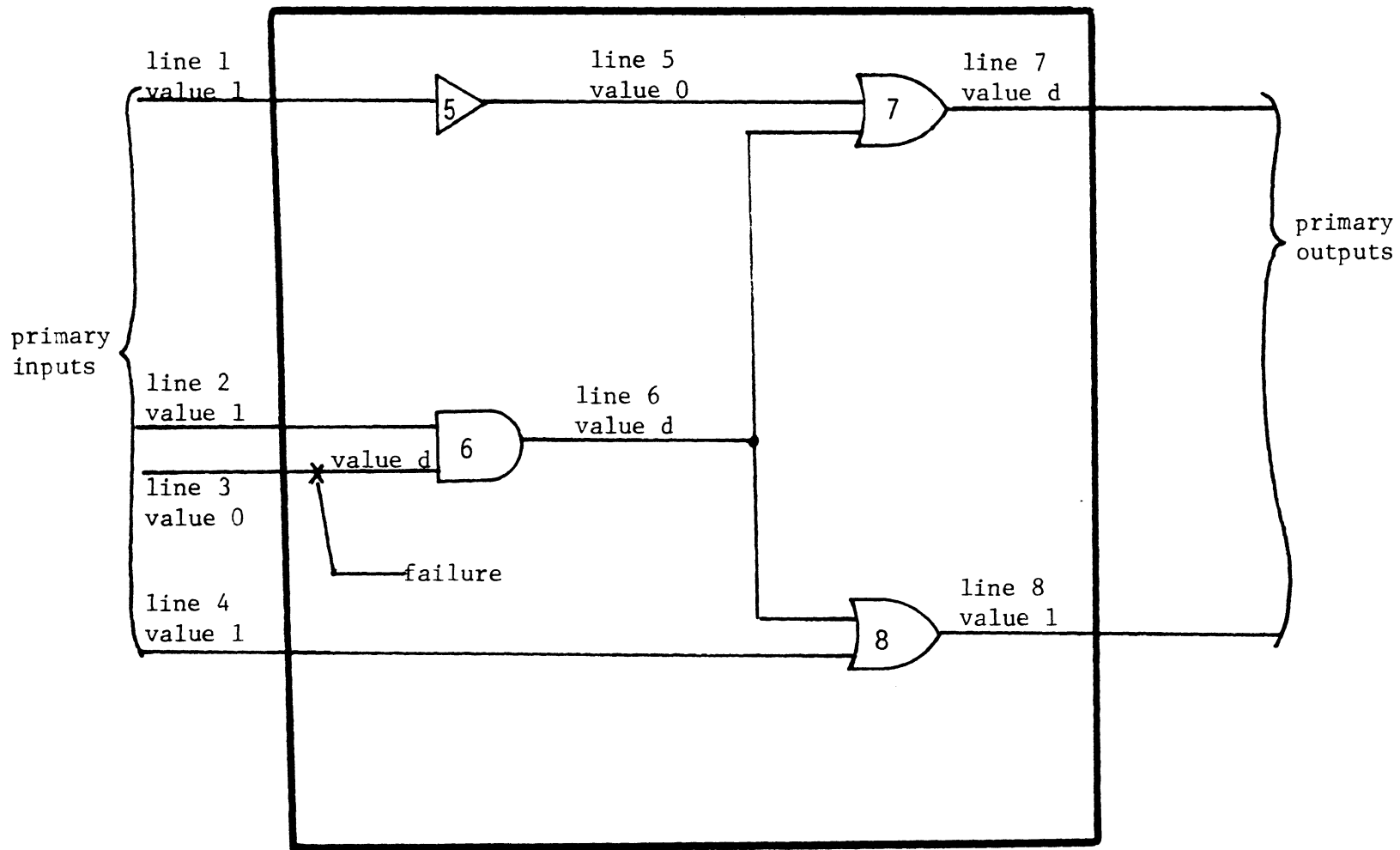
Figure 9. Line Values which Propagate the Fault Along One Path.

line 1
value 1

line 5
value 0

line 7
value d

5

7

primary
inputs

primary
outputs

line 2
value 1

line 6
value d

value d

6

line 3
value 0

failure

line 4
value 0

line 8
value d

8
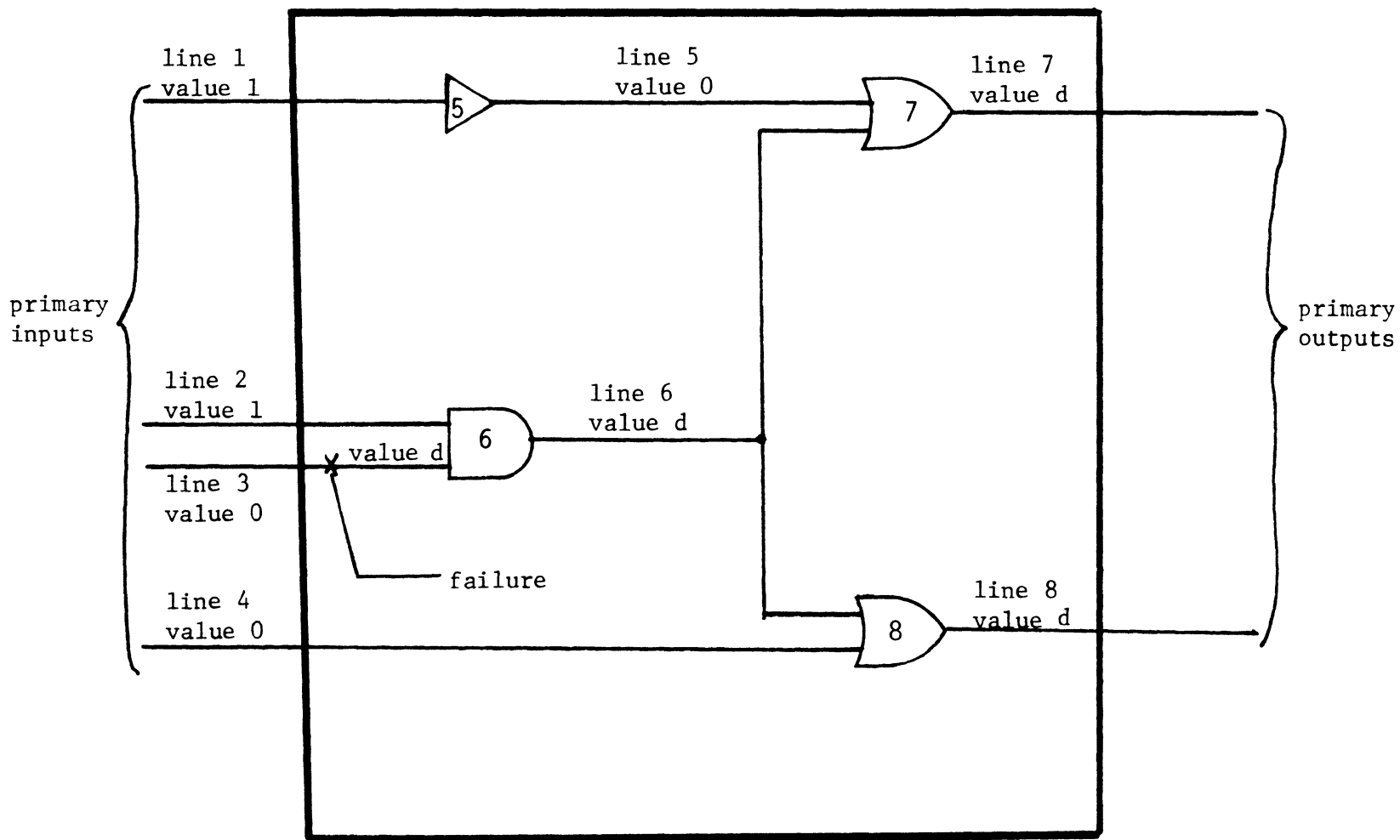
Figure 10.   Line Values which Propagate the Fault Along Two Paths.

In the implementation of the path sensitizing algorithm,[3] an

n-tuple (where n is the number of lines in the system) is developed.

At first the n-tuple contains a primitive cube of failure and remaining

lines set to 'don't care' conditions as denoted by x. Then as the for-

ward trace propagates faults, and backward trace propagates line values,

many of the x-valued lines assume values of 1, 0, d, or $\bar{d}$ (complement

of d value). The final n-tuple described the test generated. In our

example the n-tuple grows in the following manner:

```
1 2 3 4 5 6 7 8

x x d x x x x x        primitive d-cube

x 1 d x x d x x

x 1 d x 0 d d x

1 1 d x 0 d d x
```

From the final n-tuple we see the primary inputs as (1,1,0,x), and
$$1\ 2\ 3\ 4$$

primary outputs as (d,x).   The final n-tuple also has the information
$$7\ 8$$

that the test generated will not distinguish between the faults:

line 3   stuck-at-one

line 6   stuck-at-one

line 7   stuck-at-one   .

The final d-cube reveals the candidates of the site of failure by the

presence of a d (or $\bar{d}$).

The second important concept presented by Szygenda and Goldbogen

was embedding a path sensitizing routine in a simulation and diagnosis

system. This is the main concept that is used in this research.

In the work of path sensitizing in digital circuits, the path sensitizing algorithm generates a test which has information concerning presence or absence of faults at locations designating line numbers. In the implementation of the algorithm, an n-tuple is used to store information about each line, that information being:

Content	Meaning
x	line value unspecified or unknown ('don't care' value)
1	line value 1
0	line value 0
d	line value $\begin{cases} 0 \text{ for good circuit} \\ 1 \text{ for faulty circuit} \end{cases}$
$\bar{d}$	line value $\begin{cases} 1 \text{ for good circuit} \\ 0 \text{ for faulty circuit} \end{cases}$

In the present work, the path sensitizing generates information about the location of a tumor or lesion in the visual pathway as to a global location (which c-sect) and local location (within the c-sect area affected). In the implementation, there are N storage arrays, one for each c-sect.

C.   Path Sensitizing in VISUAL

The model consists of uncoupled nerve fibers. Each fiber will be treated as an electrical line that receives a 0 signal from its sense point (primary input) and transmits the signal through each c-sect intersected by the fiber to the occipital lobes (primary output). A nerve fiber bundle is assumed to be impaired from a visual field test if the area of the visual field opposite to the sense point of the nerve fiber bundle is in an anoptic region. In the present system, a 1 signal is transmitted through each faulty nerve fiber from the sense

point through each intersecting c-sect. As the faulty signal (1 signal) is propagated through the system, N arrays (one for each c-sect) record where the faulty nerve intersects.

The c-sects are represented by lattice arrays with dimension 96 by 96. Each element in the array is either 0 or 1, and hence represented by a bit. (This concept is slightly modified in Section C, Chapter V.) The array represents a two-dimensional surface, and the x-y coordinate on the c-sect corresponds to the ith, jth element of the array where i=x and j=y. Thus the x-y coordinates on the c-sect are scaled to range from 1 to 96 in each dimension. Because of this scaling and because c-sects have different physical size, the array distances may represent different physical sizes.

In the VISUAL system as in a digital system, path sensitizing involves two directions of propagation. In the VISUAL system, however, fault signals only are propagated. The direction of propagation from sense point back through the visual pathway is called the forward trace and is used in diagnostic and learning mode. The back trace, used in display mode, traces fault signals directly from the affected c-sect to the sense points.

In the forward trace, N lattice arrays consisting of 96 by 96 bits are set to all 0. Then the sense points are set according to the visual field used in the learning or diagnosis cycle. Sense points which are perceiving are set to 0, while the anoptic points are set to 1. The fiber bundles whose sense points are set to 1 are path sensitized. The "path sensitizing" process causes bits to be set to 1 at lattice points corresponding to locations through which pass faulty nerve bundles.

D. Selection of Location of Model Elements

Mathematically, the model is a sequence of 252 lists, one for each representative fiber bundle in the model:

$$S_1, S_2, S_3, \ldots, S_i, \ldots, S_{252} \qquad .$$

Each list has 12 triples.

$$
\begin{aligned}
S_i = \ & c_{i,1} \ x_{i,1} \ y_{i,1} \\
& c_{i,2} \ x_{i,2} \ y_{i,2} \\
& \quad \vdots \\
& c_{i,j} \ x_{i,j} \ y_{i,j} \\
& \quad \vdots \\
& c_{i,12} \ x_{i,12} \ y_{i,12}
\end{aligned}
$$

where $c_{i,j}$ is the index of the jth c-sect through which fiber bundle i passes coming from sense point i and traveling toward the occipital lobes

$(x_{i,j}, y_{i,j})$ is the (x,y) coordinate of the point of intersection.

The model also has specifications in three dimensions as to where the c-sects lie. The present configuration of c-sects is illustrated in Figures 11 and 12. The final mathematical information in the model is the location of the sense points in the retina as illustrated in Figure 13. This location is inverted and then referenced to the scale on a visual field. For example, sense point 17 located at ray 2, circle 8 (of the left eye) perceives objects in the left visual field at 210° from the right horizontal and 25° from the point of fixation.
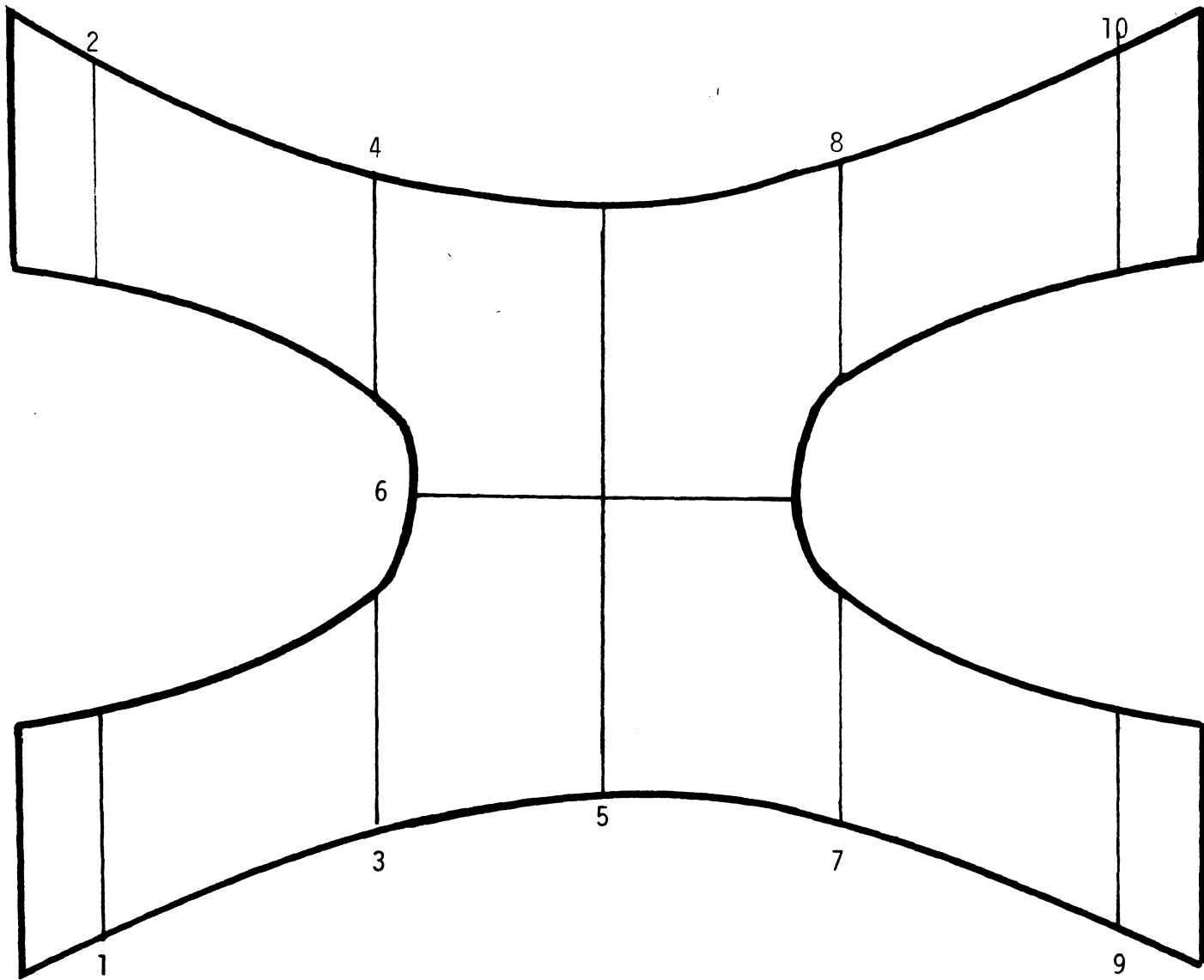
Figure 11. Top View of Chiasm and c-sects.
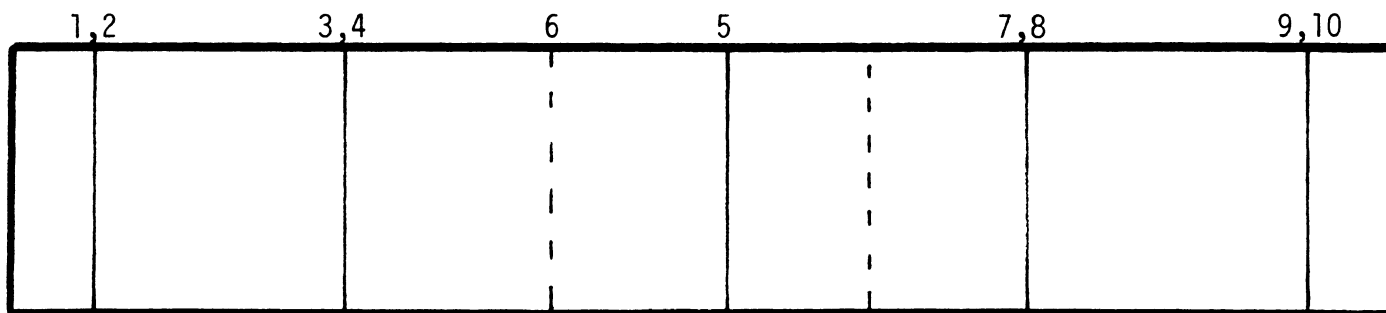
Figure 12. Side View of Chiasm and c-sects.
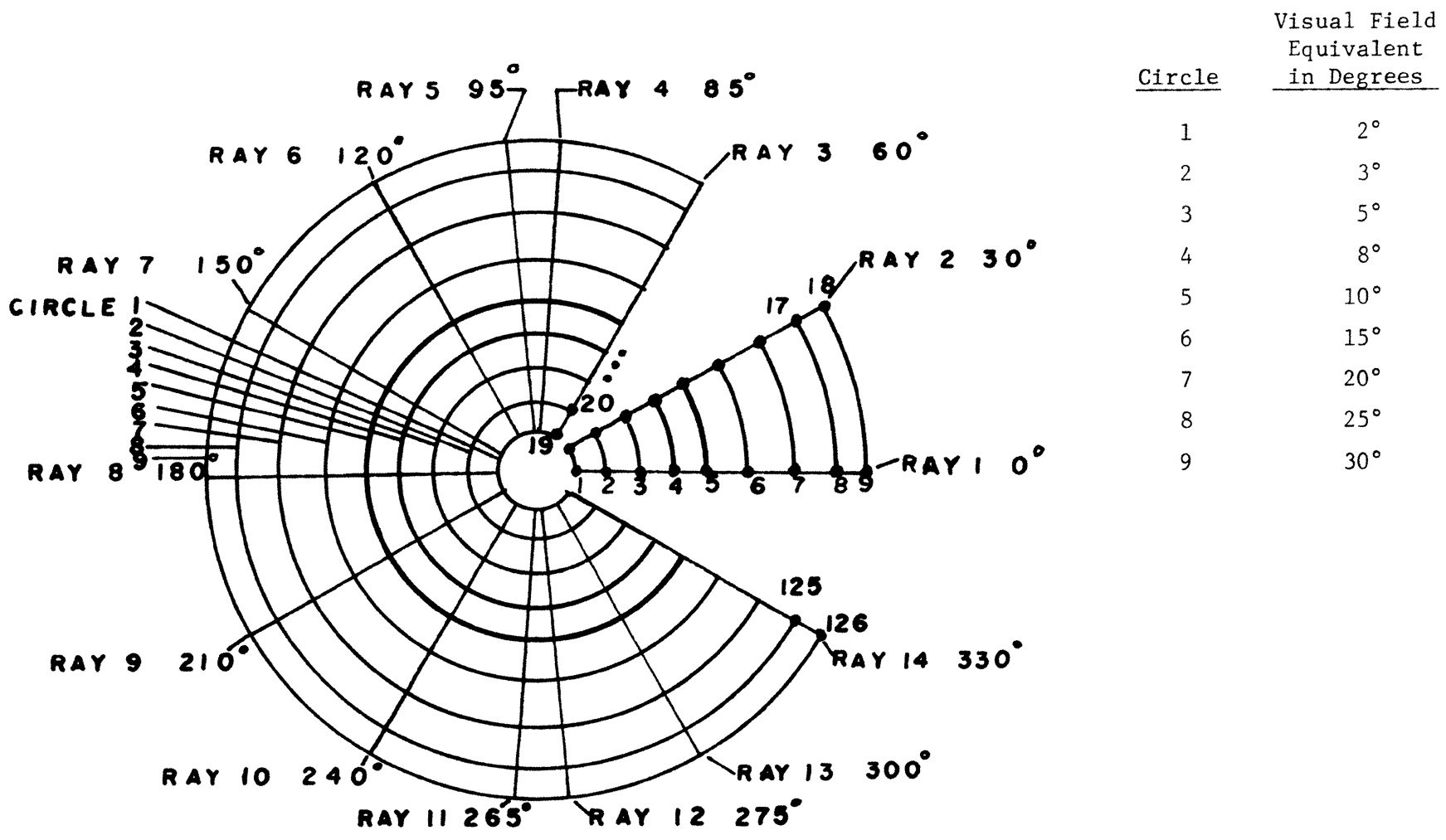
Figure 13. Sense Points Locations.

1. C-sect Locations

The choice of c-sect locations is heuristic, and many factors influenced the choice. At almost every cross section in the chiasm there is a different arrangement of fiber bundles; however, at a specific set of cross sections there are different combinations of fiber bundles (see Figure 14). C-sect 1 has bundles from the left eye only. C-sect 3 has, in addition, fiber bundles from the lower nasal quadrant of the retina. These fiber bundles loop through c-sect 3 at two locations. Further along, c-sect 5 shows these fiber bundles intersecting at only one location. C-sect 7 has fiber bundles from the upper nasal quadrant of the left eye looping out, while the upper nasal fiber bundles from the right eye come in to the c-sect. Finally, in c-sect 9 only nasal fiber bundles from both eyes are present. Along the right side of the chiasm we can make symmetric arguments for c-sects 2, 4, 5, 8, 10. C-sect 6 along the center of the chiasm shows nasal fibers which cross over. Although the arrangement of the fiber bundles may be different in some of the cross sections not chosen, the information loss is probably small and perhaps may be retrieved with further work. More precisely, it is thought that an interpolation process may supply some of the lost information. For example, suppose one considers a cross section between c-sect 7 and c-sect 9. The fiber bundles which pass through the cross section also pass through c-sects 7 and 9. Also, although the arrangement of bundles is changing between the two c-sects, the bundles approximate straight lines to some degree, and this suggests linear interpolation. In fact, since the location of the c-sects relative to the chiasm is known, and since the location of the fiber bundles relative to the c-sect is known, linear interpolation is straightforward.

A second factor in the selection of c-sect locations is the location of common tumor or lesion sites. This factor is motivated by two reasons. One is that the machine will diagnose better when an unknown tumor is in the region of a c-sect, rather than between c-sects. The other reason is that the learning file is built with clinical patterns, most common of which are the more frequent diseases. For example, pituitary adenoma will exert pressure somewhere in c-sect 6. Because the relation of the pituitary gland to the chiasm differs from person to person, the exact area of this pressure will vary, but it will usually be in c-sect 6 at some spot, hence this is a motivation for choosing c-sect 6 as a c-sect for the model.

The remaining factors affecting the choice are mathematical, programming, and display considerations. From a programming point of view, it is desirable to have as few intersections between a particular fiber bundle and a c-sect as possible. The c-sects chosen have at most double intersections. Some other candidates had triple intersections, but these c-sects have not been used in the model. The c-sect should completely cut the chiasm from one border to another. This minimizes confusion as to whether or not a fiber intersects a c-sect. Also, it may allow chiasmal shift to be (more readily) dealt with. Finally, for display purposes the c-sect should be horizontal or vertical. This facilitates the mathematics in calculating the displays as well as makes the displays on the 2250 Graphic Display more intelligible.

2. Sense Point Locations

Selection of the sense point locations is again heuristic. Figure 13 shows the location of sense points in the left retina. These number 1 through 126. Sense points 127 to 252 lie in the right retina in similar locations. It is desired to have sufficient coverage by the sense points to be rather contour sensitive. In addition, the sense points are chosen to accentuate some of the major descriptors that the ophthalmologist normally uses. The presence of the vertical line in a contour is important, motivating sense points along rays 4, 5, 11, and 12. The vertical is also important, and rays 1 and 8 sense this area. If our machine were to diagnose glaucoma, however, it is probable that more sense points along the vertical would be necessary. The major paths of fiber bundles as shown in Figure 14 dictate that quadrantal representation is necessary. Thus, in the left eye, for example, rays 2 and 3 sense the lower temporal quadrant of the visual field. Rays 6 and 7 sense the lower nasal quadrant; rays 9 and 10 the upper nasal quadrant; and rays 13 and 14 the upper temporal quadrant of the visual field. Since the macular fiber bundles travel in their own unique path, there are special sense points on each ray which sense the central field. The sense points on the circle 1, 2, and 3 perceive images at points in the visual field at 2°, 3°, and 5° from the point of fixation. Therefore, 3 sense points (out of 9) on each ray correspond to macular fibers. This aids to extract information from the visual field concerning the status of the macular fibers; i.e., whether spared, split, or whatever extent involved.[7] The remaining sense points on a ray except the 4th are spaced 5° apart, perceiving at 8°, 10°, 15°, 20°, 25°, and 30° for the 4th, 5th, 6th, 7th, 8th, and 9th sense points on a ray.

ANTERIOR

INFERIOR

SUPERIOR
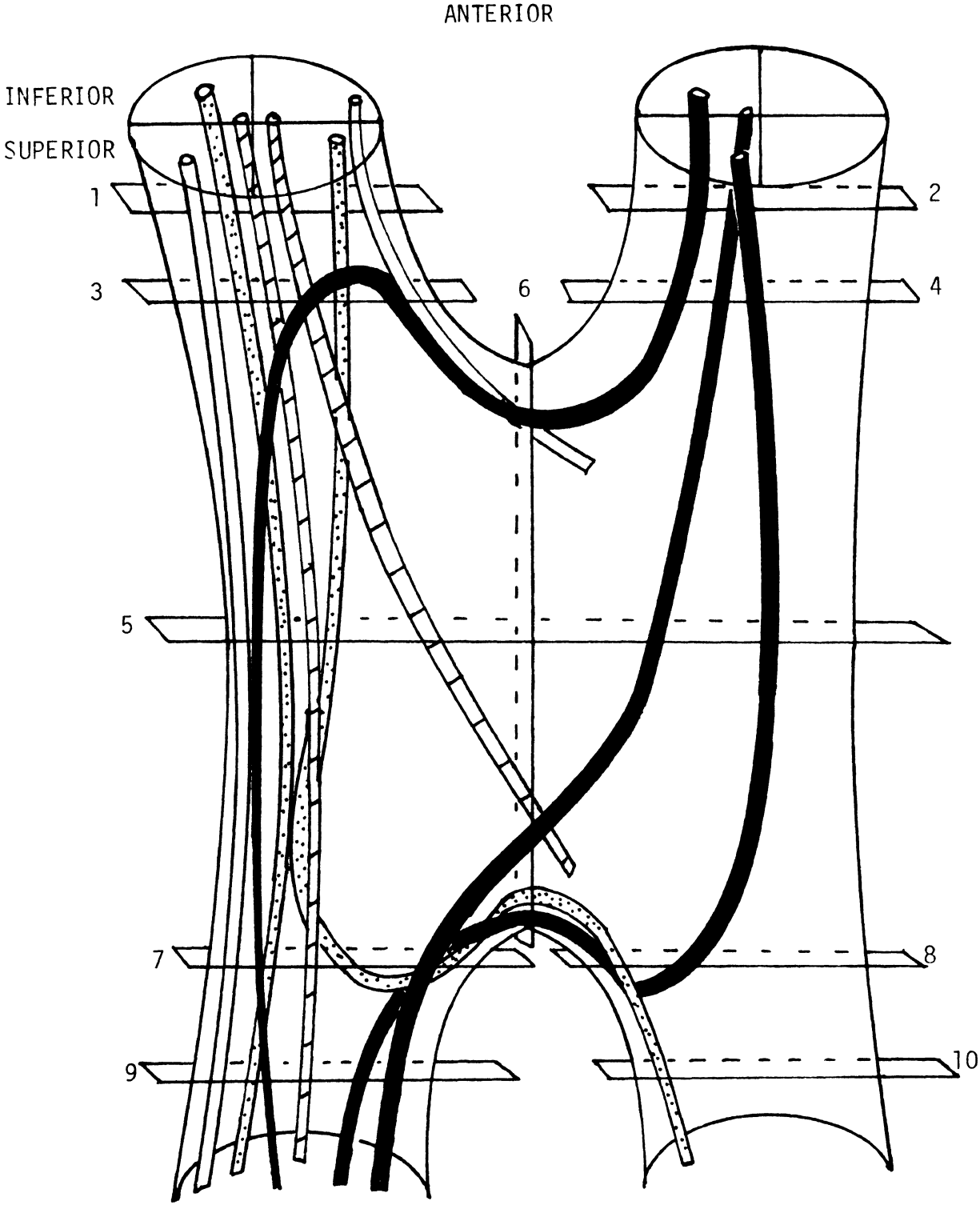
Figure 14. Location of Some Nerve Fiber Bundles in Chiasm.

Notice that the sense points are perceiving visual fields only to 30°. This means that our model is neglecting the outer 30° to 90° data, which could be obtained through Goldman perimeter tests. There is strong motivation for this, since evidence in ophthalmology shows that most diagnosis can be made using the central visual fields.[7] This also means that the fiber bundles corresponding to sense points which might lie in this outer region are not modeled.

Notice that the blind spot is also ignored in this manner. Sense point ray 1 circle 6 for the left eye would perceive that region of the visual field that lies in the normal blind spot. The system treats the blind spot in this manner: when a normal blind spot lies in a seeing field, it is designated as seeing; when it lies in an anoptic field, it is designated anoptic.

With regard to the sense points and the model, although sufficient coverage is necessary and requires a certain amount of density, it is hoped that the spacing of the sense points is sparse enough to allow for minor variations between individuals. In particular, the spacing of the sense points should be enough so that the receptive fields of the corresponding nerve fiber bundles do not overlap.

3.   Fiber Bundle Considerations

Finally, we need to list the assumptions on the fiber bundles as electric lines. It is assumed that when a sense point perceives a test object, a zero is transmitted through the corresponding line to the brain when the patient signifies that the object is visible. This occurrence we called event 1. A second possibility, event 2, is that the object is perceived by the sense point and the zero signal transmission is begun; however, because of a tumor or lesion, the signal is

blocked and never reaches the brain. The patient signifies that the object is not visible. These two events are shown in the table below:

Object is perceived by sense point	Patient signifies as visible	Tumor obstructs line	Event
YES	YES	NO	1
YES	NO	NO	5
NO	YES	NO	6
NO	NO	NO	3
YES	YES	YES	7
YES	NO	YES	2
NO	YES	YES	8
NO	NO	YES	4

Events 3 and 4 can occur when testing a small isopter whose normal visual field extends less than 30°. In Chapter V, Section B, we discuss how these outcomes are screened out.

Because vision is a psychological as well as a biological process and because we are dealing with people (the patient and the examiner), events 5, 6, 7, and 8 in the table may occur. For example, consider event 5; even though a patient perceives the test object, he may — because he is tired or because he is mentally or physically incapable — not signify that he sees the object. Although this outcome as well as other outcomes are possible, we assume our information is accurate and that the visual field data described events 1 or 2 for each sense point area. We also have intentionally not modeled the functioning of the visual pathway in extreme detail. The form of the signal transmitted, the preprocessing performed by the retinal cells, and the transformation of signals to the occipital lobes are considerations for models which perform as their main function imitating in detail the "psychological process of vision." [9,10,18,19]

Chapter IV. DESCRIPTION OF THE SYSTEM VISUAL

A. Graphical Display

An interactive graphical display device augments the utility of a general-purpose digital computer in three ways. First, user facility and convenience are achieved by presenting alphanumeric output in a soft form, where data can be searched efficiently without wasting time and material on volumes of hard copy which might be discarded after the gleaning of a single number. Secondly, the user gains speed and saves user time through interactive communication. Finally, the user gains interactive *graphical* communication with the computer. Graphical communication allows the user to go beyond the bounds of languages and numbers to the visual world of graphical and diagrammatical representation. This is, after all, where man does his best problem solving. In some learning and pattern recognition tasks, there is already some evidence indicating that man aided with an interactive graphical device performs better than man alone or computer alone. Michie and Chambers[20] explored the graph transverse problem of the most efficient connection of a single line to 50 points. Their results indicated man-machine interaction performed the task best. Recently, John W. Sammons, Jr.[21] presented an interactive graphic system intended for general pattern recognition use, which he feels "...will provide new and powerful methods for the exploration and ultimate understanding of complex patterns." Lichlider and Clark in 1962 saw heuristic programming as a long-term problem whose solution will enhance man-computer partnerships.[22] In fact, as far back as 1945, man expected and awaited interactive graphics.[23] Today many systems generally use interactive graphics in mathematical problem solving, data manipulation, and information

display.[24-33] In these systems the ability to display and converse

graphically is an important adjunct in problem solving. Speed and

convenience are also important ingredients in these systems. When

problem solving involves formulation of a plan or heuristic, testing

the process and evaluating the results, the ability to work in the

immediate time domain aids the user indispensably. Thoughts can be

formulated and tested without the dulling of painful and tedious pro-

gramming.

With these guidelines, the system VISUAL has been constructed to

show graphically as much as the user desires of the status and progress

of the model before, during, and after the learning. The system allows

the user to communicate graphically both in commanding the learning

process and in handling the data. Of course, the user also gets the

speed and facility of the graphical display.

B.   Equipment

The system VISUAL is a large Fortran program designed for an

IBM 2250 Model 3 Graphic Display. The IBM 2250 is attached to an

IBM 360 Model 75 computer. VISUAL uses Fortran subroutine calls from

Graphical Subroutine Package (GSP)[33] to control the 2250 interactively,

while the 360 is under control of the MVT version of OS/360.

Figure 15 shows an IBM 2250. The 2250 has the ability to display

dots, straight lines, and characters on a 12" x 12" screen. The user

can communicate with the system VISUAL by three devices whose function

and utility are created by the program: The program *function keyboard*

contains 32 buttons. The system can be programmed to sense when and

which button is pressed. The *alphanumeric keyboard* is a standard   type-

writer keyboard. The software allows alphanumeric information to be

transferred from the keyboard to a buffer in the 360. The *light pen* can be pointed at the display screen, and the program can sense which image has been "touched" by the light pen.

DISPLAY SCREEN

LIGHT PEN

ALPHANUMERIC KEYBOARD

PROGRAM FUNCTION KEYBOARD
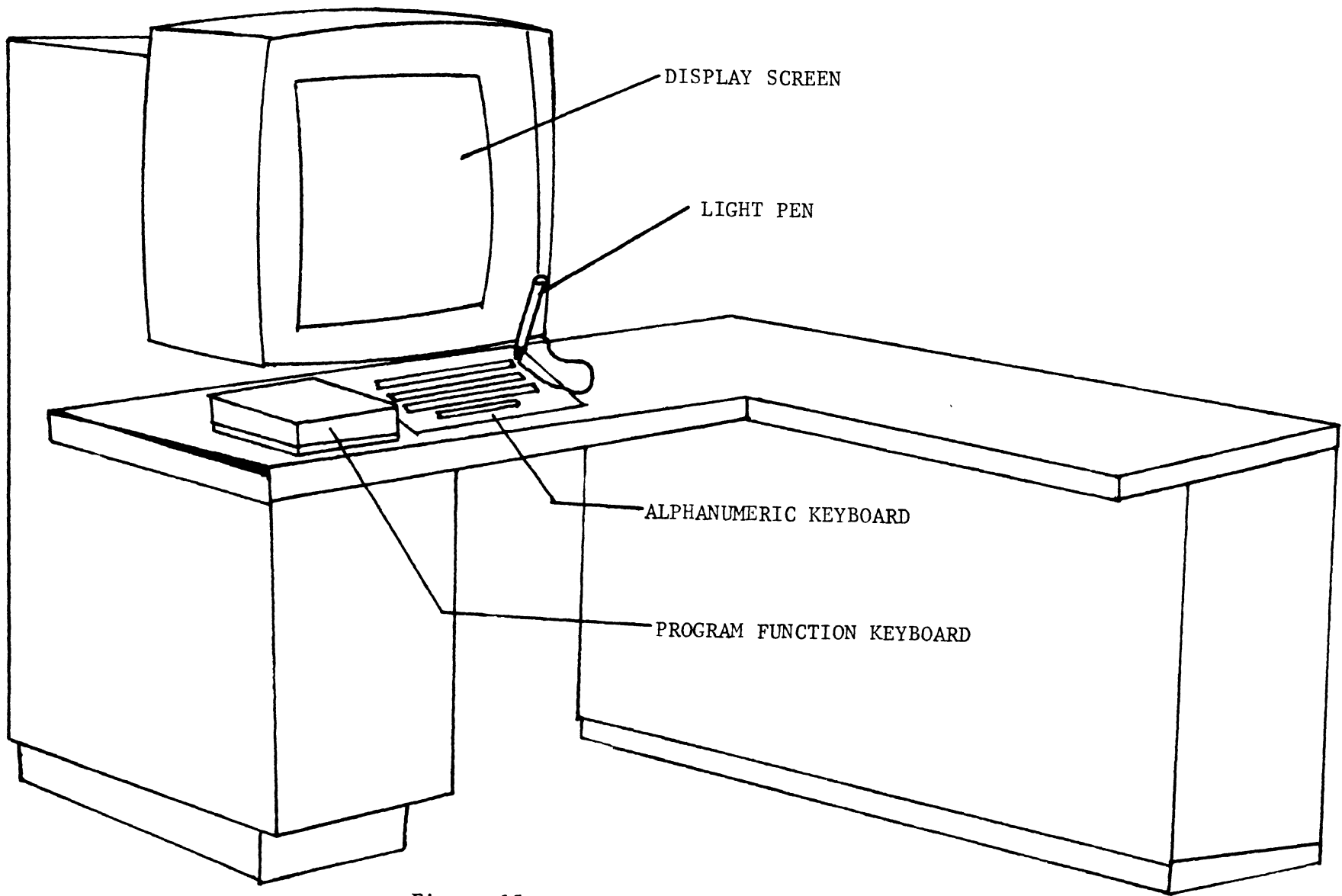
Figure 15.   IBM 2250 Graphic Display Unit.

C. File Structure

There are two main files in VISUAL: the *model* and the *pattern*
file. The model has been described in Chapter III. It is updated dur-
ing the learning process and consulted during the simulation process.
The model is embedded in a 3 by 12 by 252 array, and is formatted as a
loose list of 3-tuples as shown below in Figure 16.
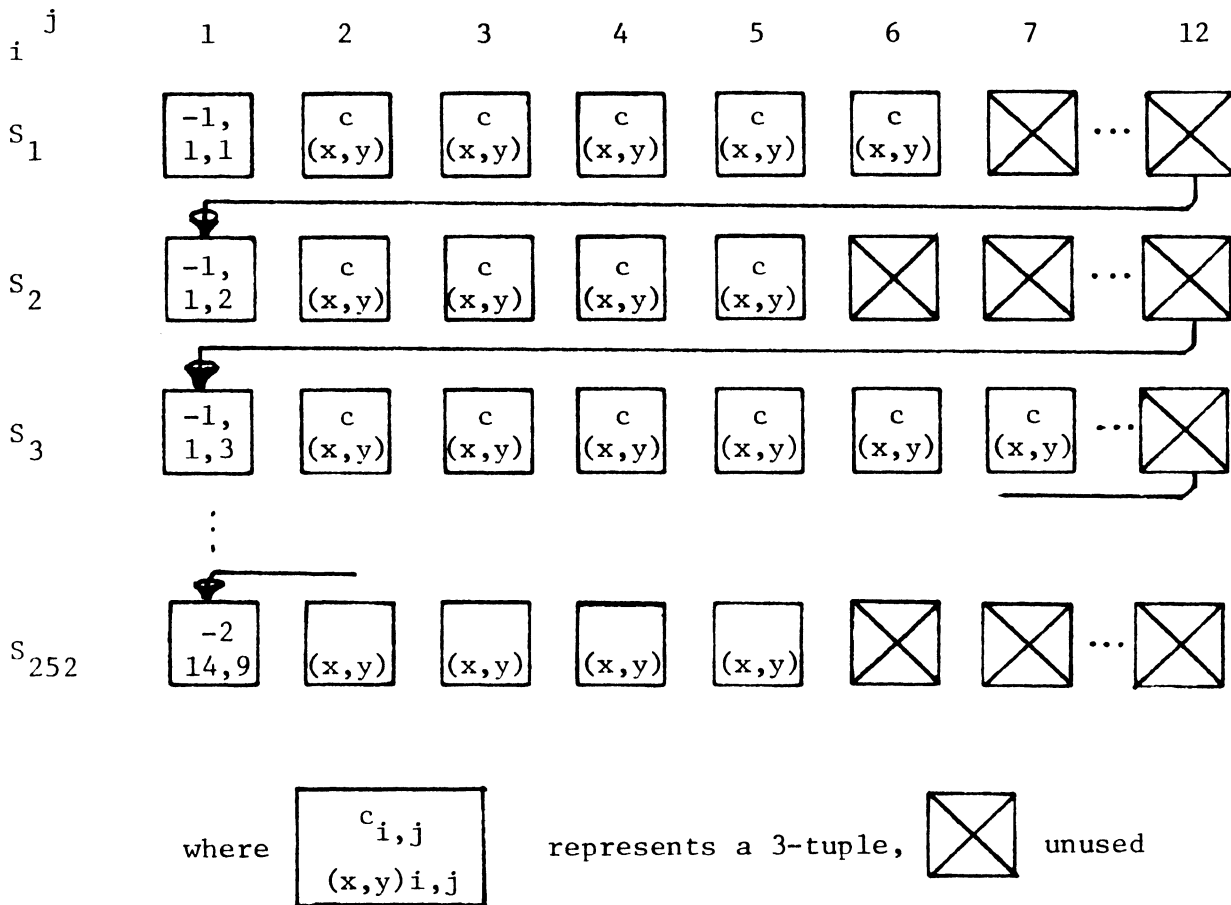
Figure 16. Loose-List Structure of Model.

This loose-list structure enables a fiber bundle description (a
sublist S_i) to be located immediately and to be processed quickly by
incrementing a single index. The loose structure allows addition of
elements to a sublist during learning. This process has not been

implemented, but at one time was considered, and hence is discussed in Appendix C.

There are other reasons for adding elements to the model. A more accurate model of the chiasm will require a larger model. Also, when the model is extended to the post-chiasmal pathway, the model will be larger. These increases will result in more c-sects and hence some of the unfilled locations in the loose list will be filled. The loose-list structure allows flexibility and ease in these changes in the model.

The pattern file has information for each pattern consisting of:

a) (x,y) coordinates contouring the visual field.

b) Test object distance and size.

c) Packed sense points, which is a digitized form of the information in a).

d) Extent of the damage given in terms of the parameter r.

e) Patient information includes name, age, condition, etc.

f) Examination information contains comments on examination conditions as well as examiner's name, data, number of isopters.

g) Diagnosis data lists x,y center and verbal description of diagnosis.

Since a pattern represents a single isopter, and because an examination usually consists of more than one isopter, the pattern file is divided into a two-level file. The first level contains the "pattern" information of a, b, c, and d. Items e, f, and g denoting patient information are in the second level of the file. In order to link between these two levels, each record has pointers to associates in other levels. Figure 17 diagrams this file structure.
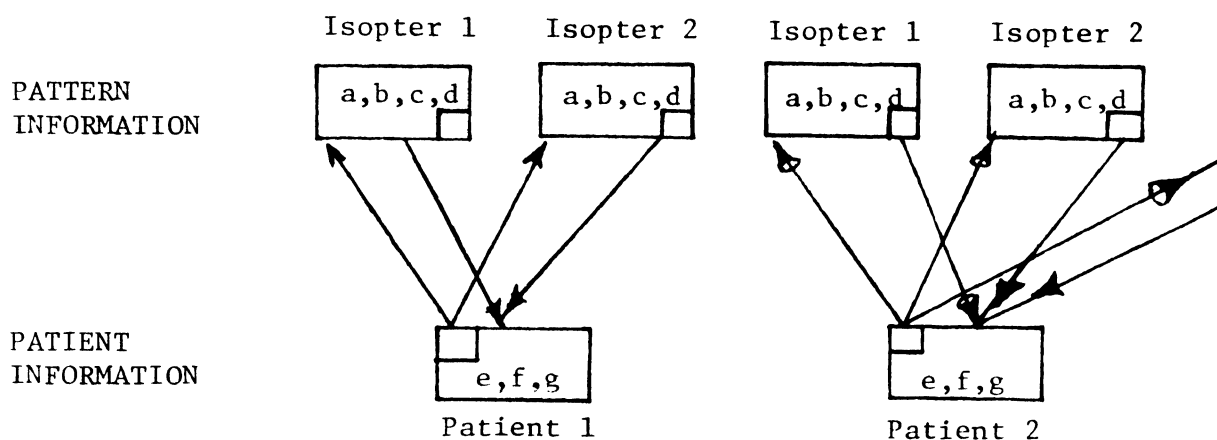
Figure 17. Pattern File Structure.

D. Graphic Data Sets

The images and text which appear on the graphical display are grouped by the software into units called graphic data sets.

Some of the graphic data sets are static. They are constructed once and remain unchanged throughout the execution of the program. These may be omitted (subroutine OMIT) from the screen or returned to the screen (subroutine INCL). These graphic data sets are

ICHAST	Top view of chiasm and c-sects
ICHASS	Side view of chiasm and c-sects
ICUT	Numbers of c-sects, light pen sensitive
IGRID	Normal grid for visual fields
IGRID2	Large grid for input

The graphic data set that shows the information in the pattern file is the pattern board (IPAT). The pattern board presents a list and a brief description to the user. Preceding each entry in the list is a light pen-sensitive file number, which when touched causes the corresponding pattern to be retrieved from the file. The pattern board changes as patterns are added or deleted from the file.

The dynamic graphic data sets are created repeatedly in different forms during the various modes of operation of the system.

IFORM	Contour picture of the visual field pattern
IDAT	Patient and pattern data associated with the form
IPATHT	Top view of sensitized paths
IPATHS	Side view of sensitized paths
IRET	Retina sense point display
ISECT	Display of sensitized fibers intersection with c-sect
IQ	C-sect front view outline and number
IVF	Redrawn visual field resulting from back trace

IRINGT
IRINGS } IRING - Diagnosis of location and extent of damage
IRINGF

JRINGT
JRINGS } JRING - Correction of location and extent of damage, also used to set lesion for back trace simulation
JRINGF

43

E.   The System VISUAL

VISUAL operates in six modes: initialization, input mode, diag-
nostic mode, learning mode, display mode, and output mode.

Initialization is always the beginning step when operating the
system.   The initialization phase sets up in an initial model, and an
initial pattern file.   It also creates the static graphic data sets and
the initial pattern board.

When the initialization phase is completed, the user may select
any of the remaining modes or may terminate the processes.   (See
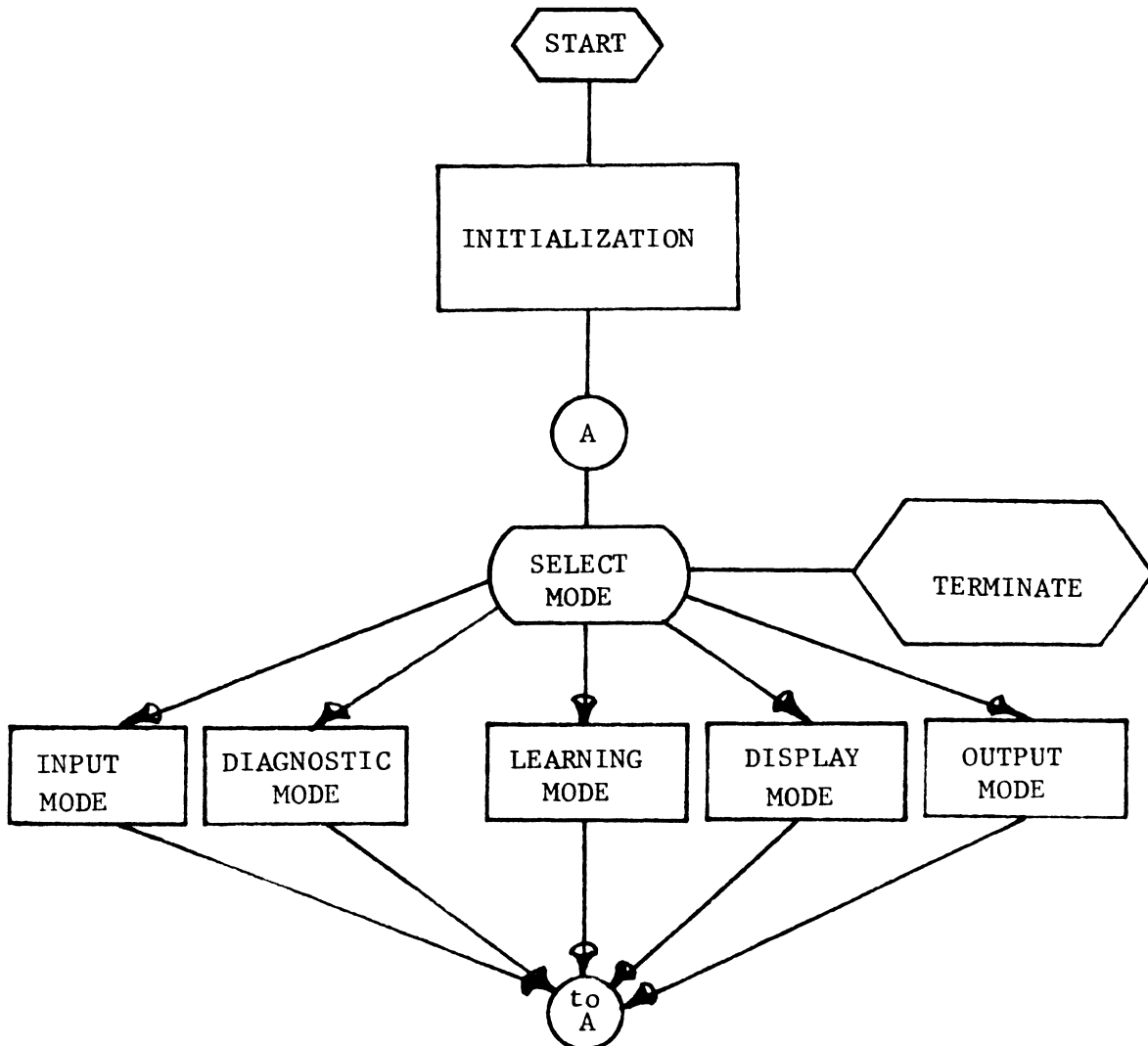Figure 18.)

Figure 18.   Overview of VISUAL FLOW

Input mode permits the user to "sketch" a visual field, type in patient and pattern data, process the visual field picture to produce the sense point array, and store the information on the pattern file.

Output mode allows the output of the present model and/or selected patterns from the pattern file for a file that may be used in future runs of the system.

The diagnostic, learning, and display modes are the heart of the system.

Learning mode utilizes a single subroutine LEARN which implements the local learning discussed in Chapter V, and Appendices A and B. The user need only select the c-sect and the pattern sequence, S, either set values, or accept default values on the set of learning parameters, and call the subroutine LEARN.

Display mode performs the back trace propagation to simulate a hypothetical lesion. The user first selects a c-sect with the light pen. Then using subroutine PRESSR, a ring of dots (JRING) is manipulated until the user positions and sizes the graphic data set as he wishes. At this point, the system performs the back trace, sets the sense point arrays, and draws the resulting visual field. The flow of the display mode is given in Figure 19, while Figure 20 shows the graphic display status at the end of the display mode process.

In diagnostic mode the pattern board is first displayed. The user light pen selects a pattern to be displayed. The pattern board disappears and the visual field appears (IFORM) along with the accompanying retina sense points (IRET). The graphic display is shown in Figure 21. The user has the option of selecting sense points with the light pen to view their corresponding fiber bundle paths. In Figure 22,

the user has light-pen-selected fiber bundle 171 to view. In Figure 23, he selects all sensitized fibers. The path sensitizing operation presents the same options, sensitizes only sense point touched by the light pen or sensitizes all fibers. Here the user may view the fault arrays sequentially or by command of the light pen. Figure 24 shows the fault array 2 being displayed. Next, the diagnosis algorithm is called and the diagnosis (IRING) displayed in Figure 25. The dotted circles signify the location and extent of damage. A subroutine is called to allow the user to present the system with a correction as shown in Figure 26, in solid circles. The flow of diagnostic mode is given in Figure 27.

F.   Parameters

The system has three types of parameters: model parameters, learning parameters, and decision parameters. The learning parameters as discussed can be changed interactively. These parameters control the local learning process. The decision parameters can be changed (but not interactively) to cause the decision algorithm to perform differently. The model parameters are inputted during the initialization mode of the system. The user can control c-sect number and location, and sense point number and location. The path-sensitizing mapping properties of the system as well as other functions can proceed as usual, once the model parameters are set. User control of these parameters is an important aspect of the flexibility of the system VISUAL.
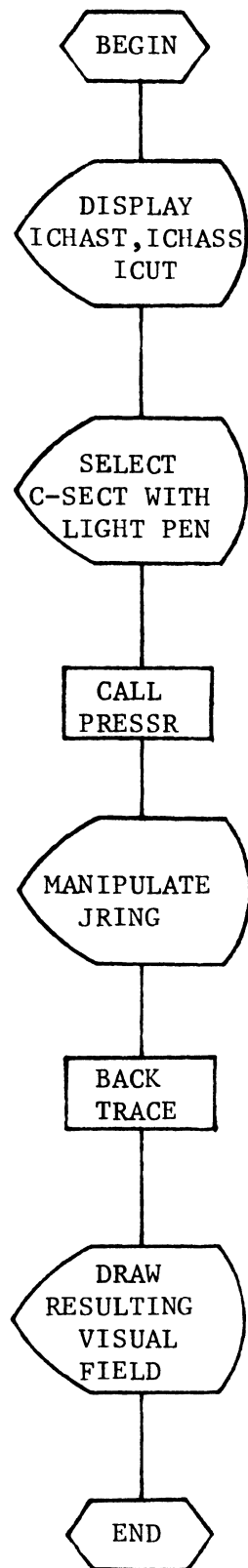
Figure 19. Display Mode Flow.
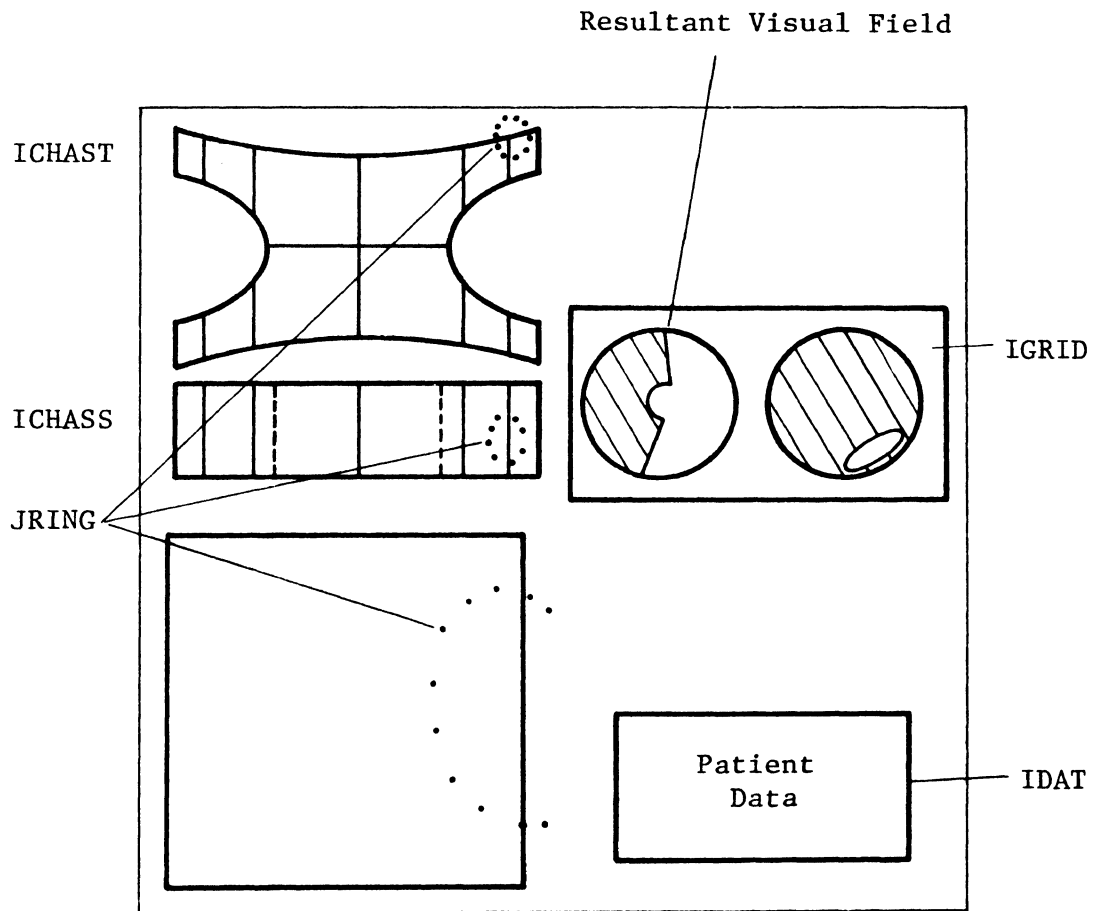
Figure 20. Graphic Display at End of Display Mode Cycle.
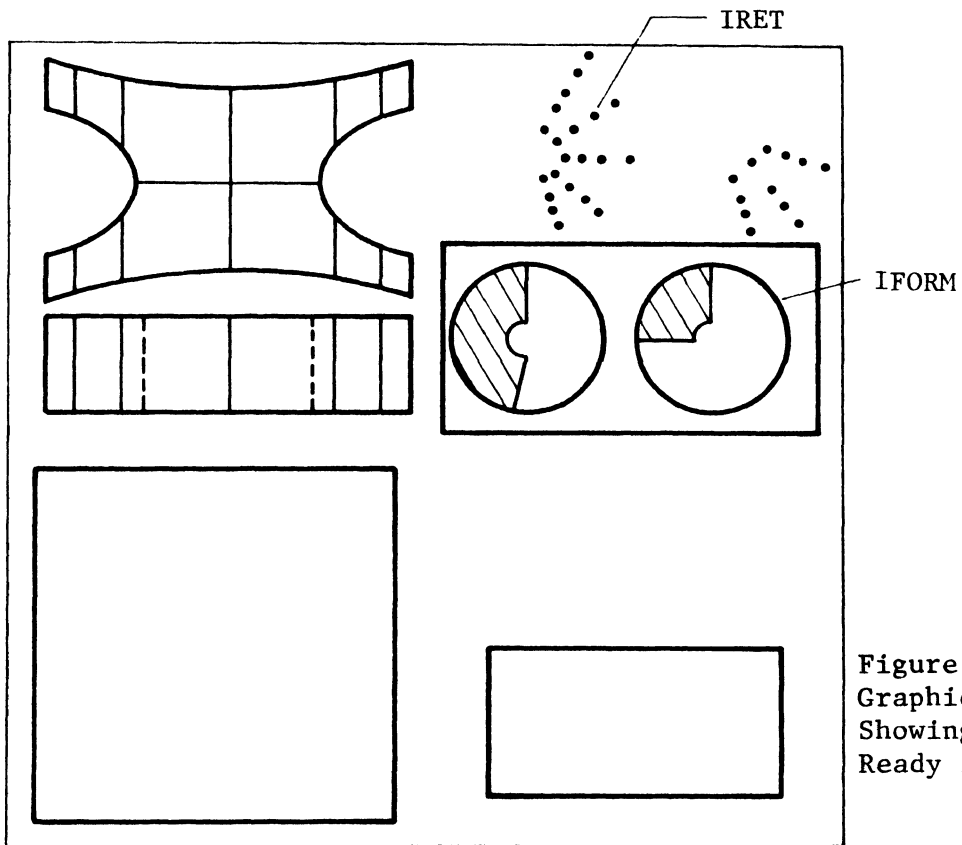
Figure 21.
Graphic Display
Showing Pattern
Ready for Diagnosis.
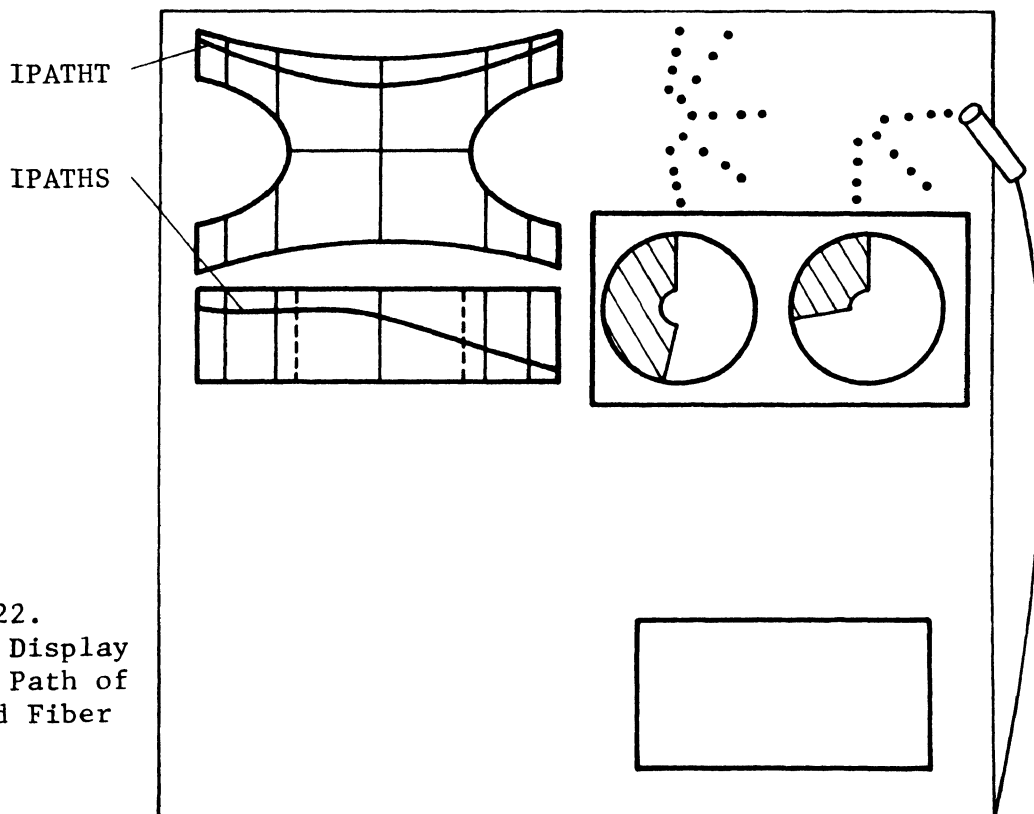
Figure 22.
Graphic Display
Showing Path of
Selected Fiber
Bundle.
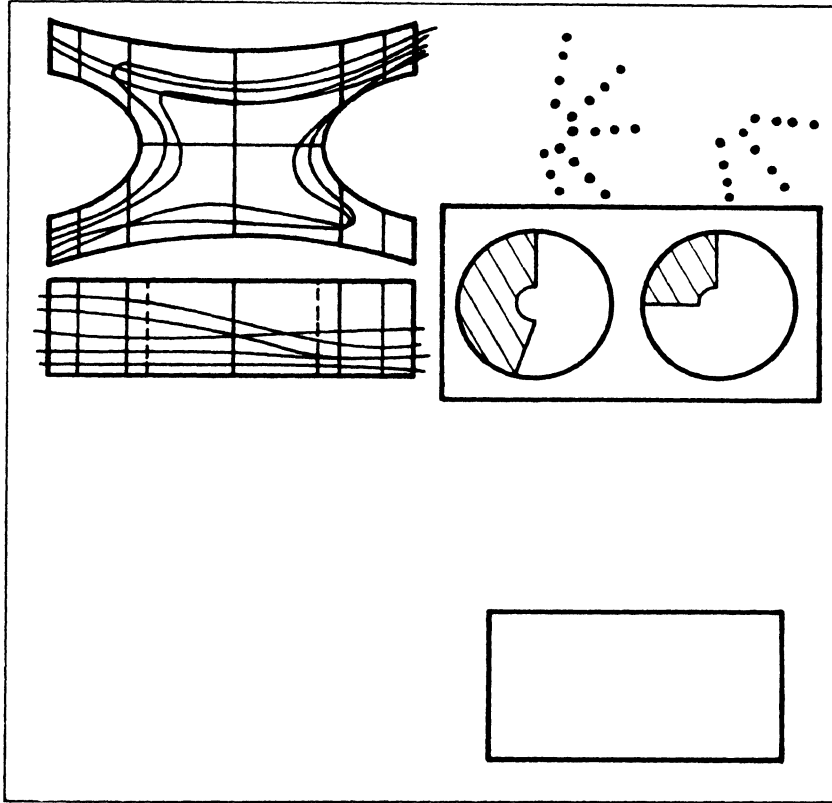
Figure 23.
Graphic Display
Showing All Sen-
sitized Paths.
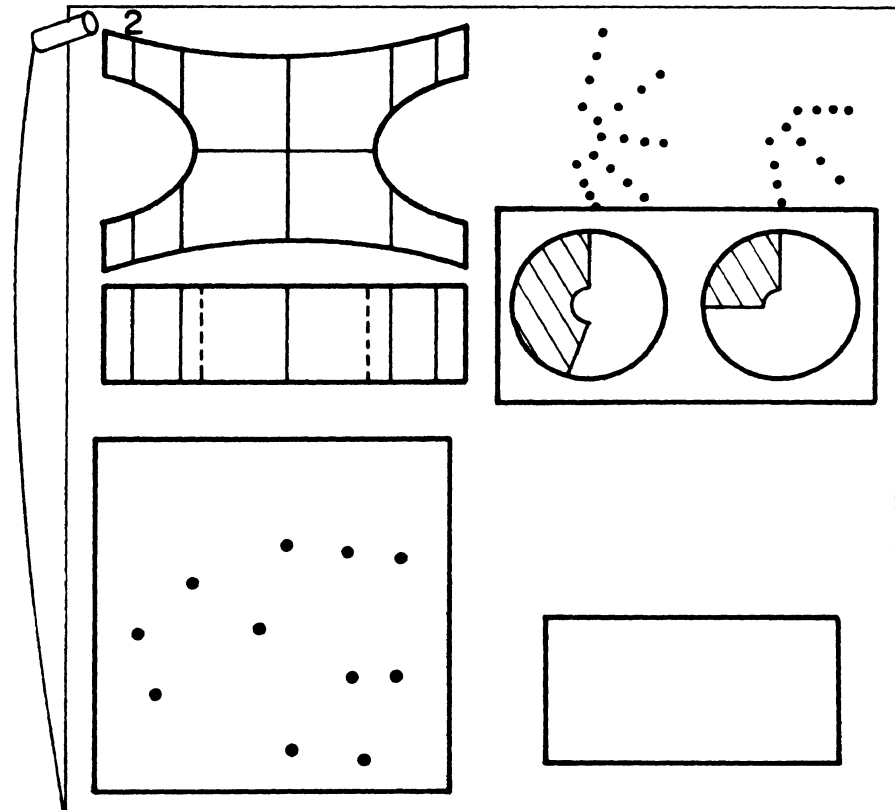
Figure 24.
Graphic Display
Showing c-sect 2
Fault Array.
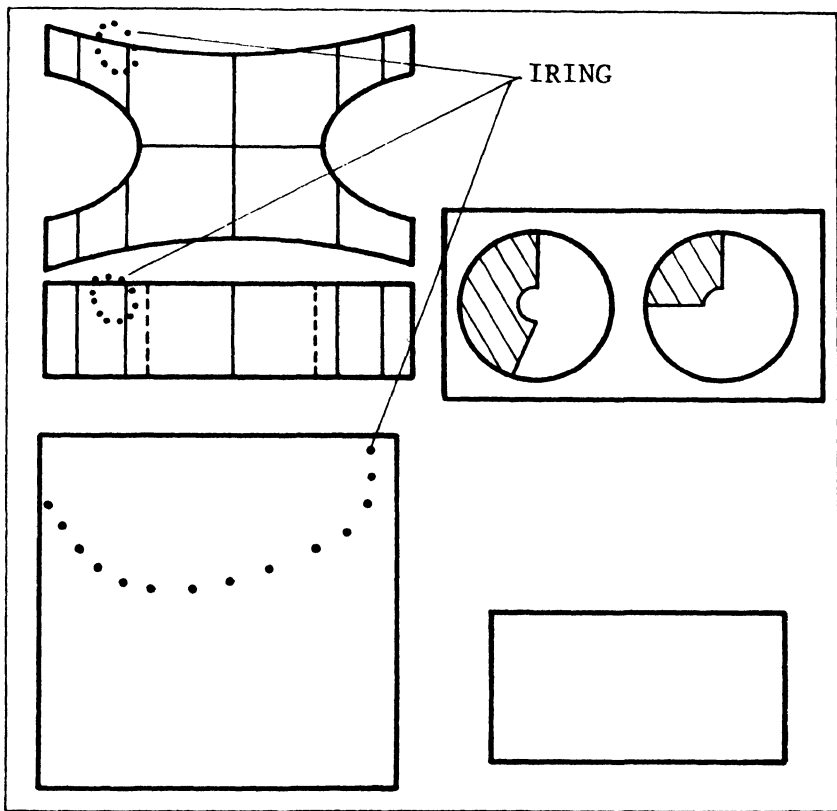
Figure 25.
Graphic Display
Showing Computer
Diagnosis.
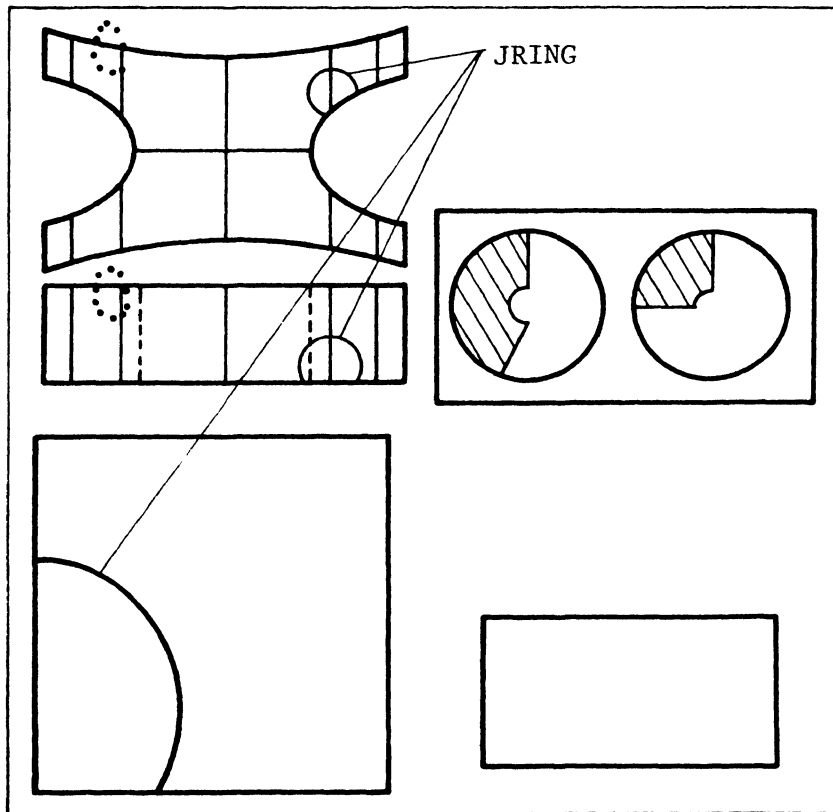
Figure 26.
Graphic Display
Showing Corrected
Diagnosis.
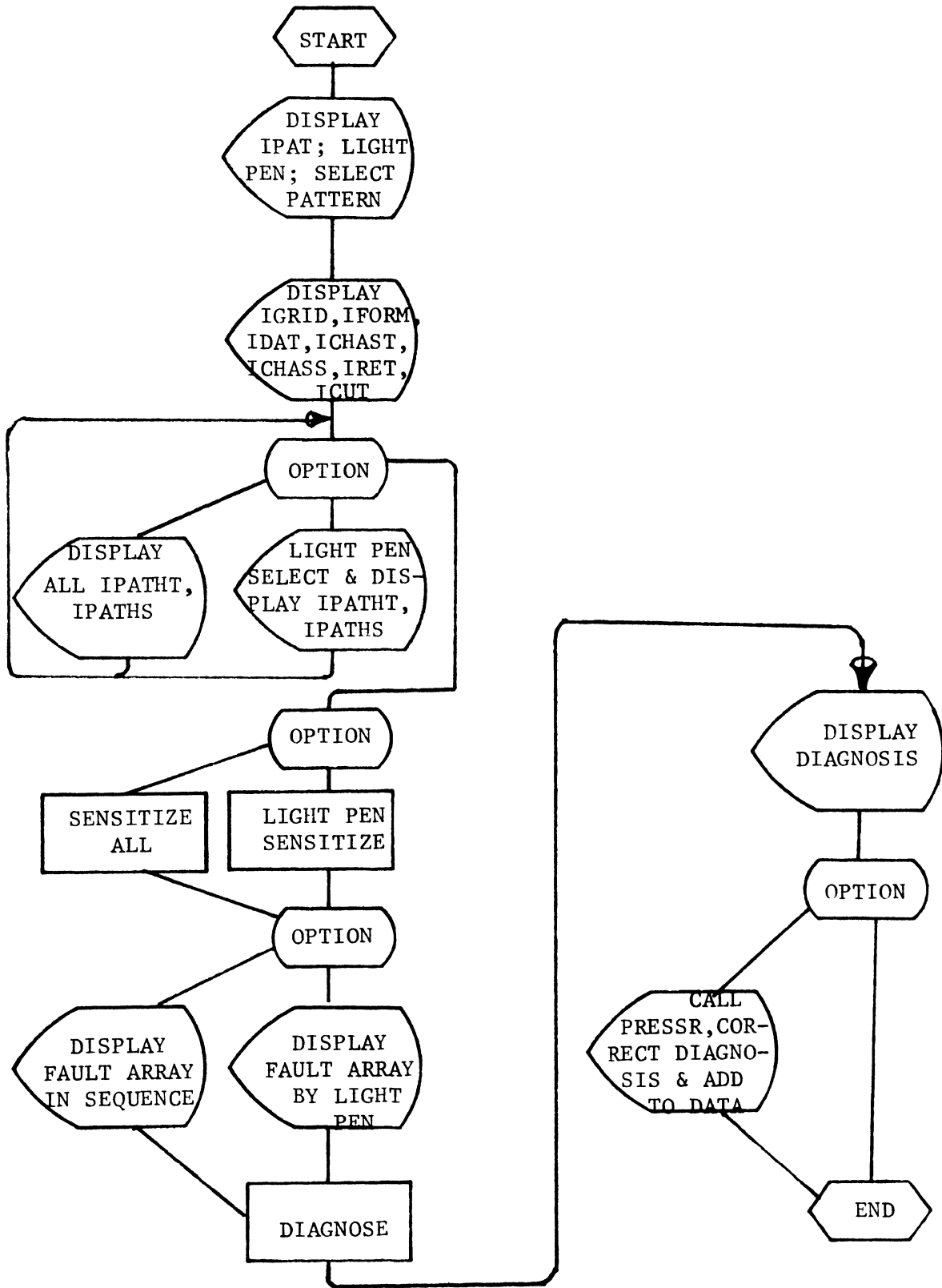
Figure 27. Diagnostic Mode Flow

Chapter V. THE LEARNING MACHINE

In this chapter we wish to show the major concepts of the learning machine in VISUAL. The mathematics of the system are detailed in Appendix A.

A. Classical Learning Machine

A learning machine is a device which, through experience, adapts a method of performing a given task or a set of tasks. Learning machines have been primarily pattern-recognition devices whose task is to examine an unknown input, I, and respond, diagnose, classify, or interpret the input signified by emitting a response, R. This type of learning machine has been applied to primarily character recognition. Other types of learning machines have been made to play simple games (tic-tac-toe), highly complex games (chess, go), perform mathematical hill climbing, simulate human cognitive processes, and solve general logic problems.[34,35,36] It is beyond our scope of concern to investigate all the configurations of learning systems; however, a basic learning system of the first type is presented below. It is the one representative of a learning machine which is very commonly used in many applied problems and has received thorough mathematical treatment in prominent works in the field of pattern recognition.[6,37-41] For this reason, it is called a classical learning machine. These machines learn on a training set of known inputs. When this training phase is completed, the machine performs its specific task on a second file of inputs, called simply the input file.

The pattern recognition device may further be divided into three subsystems: the preprocessor, the feature extractor, and the pattern classifier. The preprocessor transforms the input from the input space

R' Known Response
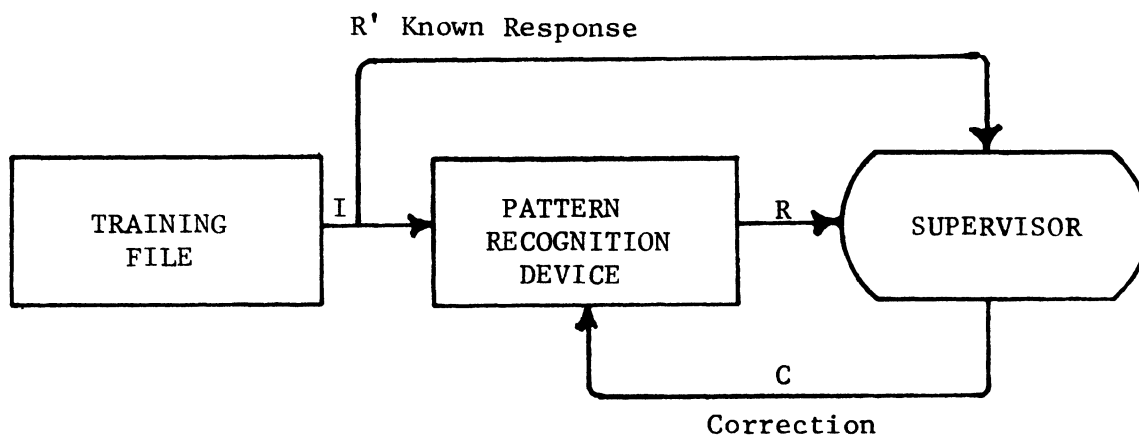
Figure 28. Classical Learning Machine in Learning Phase.
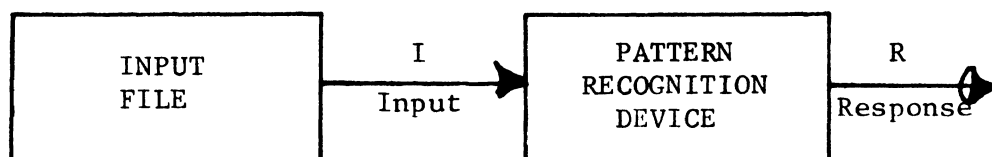
Figure 29. Classical Learning Machine Performing Tasks.

to a pattern space, filtering out noise, and if possible enhancing the input. In this discussion we include in the preprocessor all the functions required to change the input in the real world to a digital form in the computer system. The feature extractor is important when the representation of the input is in a form difficult to handle mathematically. The feature extractor will transform the input from a pattern space to a feature space using a mapping that emphasizes distinguishing features of the input data while filtering noise and irrelevant characteristics from the input data. The pattern classifier analyzes the feature space representation of the input pattern, and responds. It maps, therefore, a point in the feature space to a point in the response space. Usually the pattern classifier accepts in input (the point in the feature space) as an N-tuple, X, and evaluates M discriminant functions, one for each response class. One of the values (say the maximum) is selected and that response class chosen. The performance of the pattern classifier is dependent on the performance of the feature extraction. The functioning of the whole pattern recognition device is dependent on the functioning of each of the subsystems so that the interrelationships of the subsystems and the relationship of each subsystem to the whole system is a prime consideration, and the separation of the functions of the subsystems is difficult. In practice, however, we break up the system into the three subsystems to facilitate the discussion of our system. Before proceeding, it is important to note that until recently, most of the work in the field of pattern recognition concentrated on the pattern classification problem.[42] The field of feature extraction has had few works yielding significant problem independent results. Most work, in fact, has problem-dependent heuristic

feature extractors which utilize the ingenuity of the designer to overcome a particular problem.

The classical pattern classifiers are usually termed parametric or nonparametric. Parametric classifiers have received much mathematical analysis[37-40]; the parametric classifiers assume a functional form for conditional probability distributions of the input patterns. These distributions are assumed known except for a set of parameters. The function of the learning machine is to learn optimum values of the parameters. Once these parameters are evaluated, they are used in discriminant functions which form the decision or response.

Because the parametric learning machines depend upon the assumed probability densities, and because in applied problems these functional forms are often difficult to estimate, in practice the nonparametric learning machines are used in most useful applications. Adaptive distribution-free procedures are used in nonparametric learning machines.

A general (nonparametric) learning machine is described in detail in Appendix A. There the mathematical definitions of the preceding terms as well as theorems and procedures are discussed, and the mathematical support for the system is given.

In summary of this introduction to classical learning machines, Figure 30 shows the relation of the mentioned terms. During learning, the supervisor compares the system response with the known response, and emits a correction. Notice that the correction from the supervisor feeds back to the pattern classifier. In parametric machines, the parameters are updated and improved. In nonparametric machines, weights may be changed as in the case of linear machines.
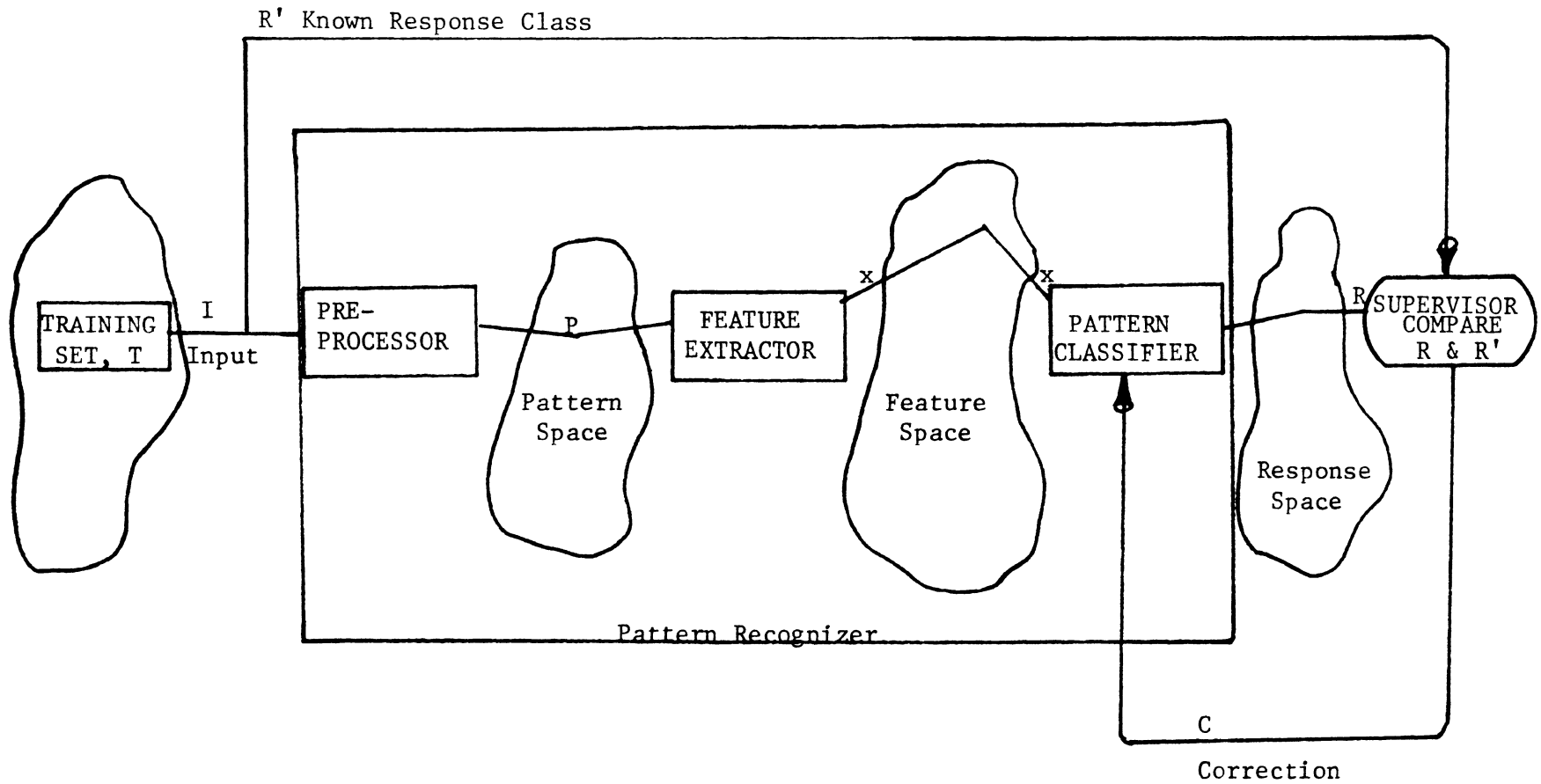
Figure 30. Subsystems and Image Spaces of the Classical Learning Machines.

B.   Adapted Learning Machine

In this section we will show structurally the relation between the classical learning machine and the adapted learning machine used in the VISUAL system.   The VISUAL system has its counterparts of the preprocessor, feature extractor, and pattern classifier subsystems. In the VISUAL system, the preprocessor performs the task of setting the sense points from a given visual field and isopter size.   A sense point perceives a spot in the visual field at a location opposite, in the way shown in Chapter III. If this spot lies in an anoptic field, the sense point is set to one (1); otherwise it is set to zero (0). If the spot is not visible in normal vision under the isopter size, the sense point is set to -1, signifying a 'don't care', and that point and associated nerve fiber is ignored.   This masking of the 'don't care' nerves is done according to limits of the normal visual field as shown in Table I.   For example, in a 1/1000 isopter test, objects are not normally seen at 26° from the point of fixation. Objects at 30° are perceived by sense points in circle 9.   Hence, for a 1/1000 isopter, sense points on circle 9 are set to -1.   This information is given in the last column in Table I.

Thus the preprocessor transforms the visual field in a form of a clinical record to a set of sense points set to zero, one, or 'don't care'.   The pattern space consists of more than $2^{2^{52}}$ points, since each of 252 sense points can be set to zero or one.   The "more than $2^{2^{52}}$" comes from the addition combinations with the 'don't care' masks.

The feature extractor, using path sensitizing, transforms the sense point array to c-sect arrays as described in Chapter III, Section C.   The system is designed so that when the system is operating

well, the faults cluster best in the c-sect which has the lesion in its vicinity. The pattern classifier determines the response by picking out the best clustered c-sect array. Clustering will be defined in the next section.

An important adaptation in the VISUAL system is the correction of the model instead of the weights or parameters in the pattern classifier. This correction process is really performed during Local Learning, which is discussed later in this chapter. A diagrammatical comparison of the two systems follows in Figure 31.
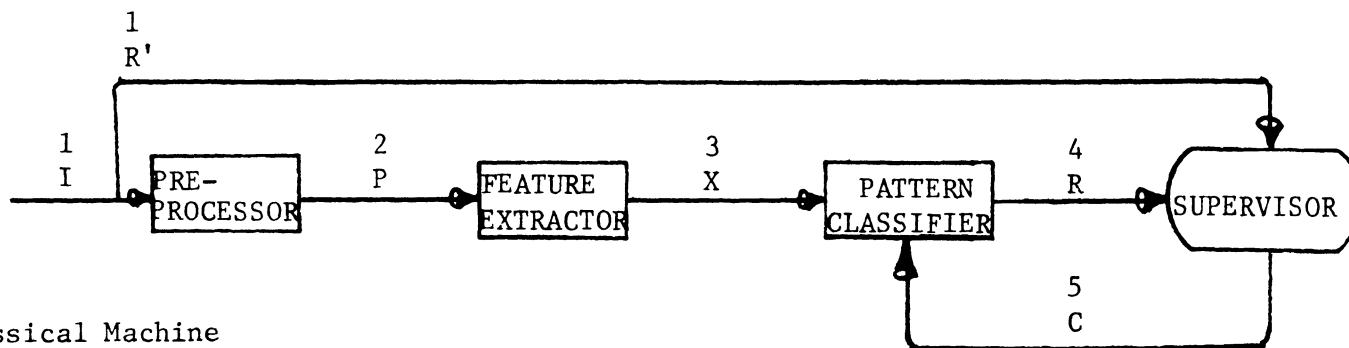
(A) Classical Machine
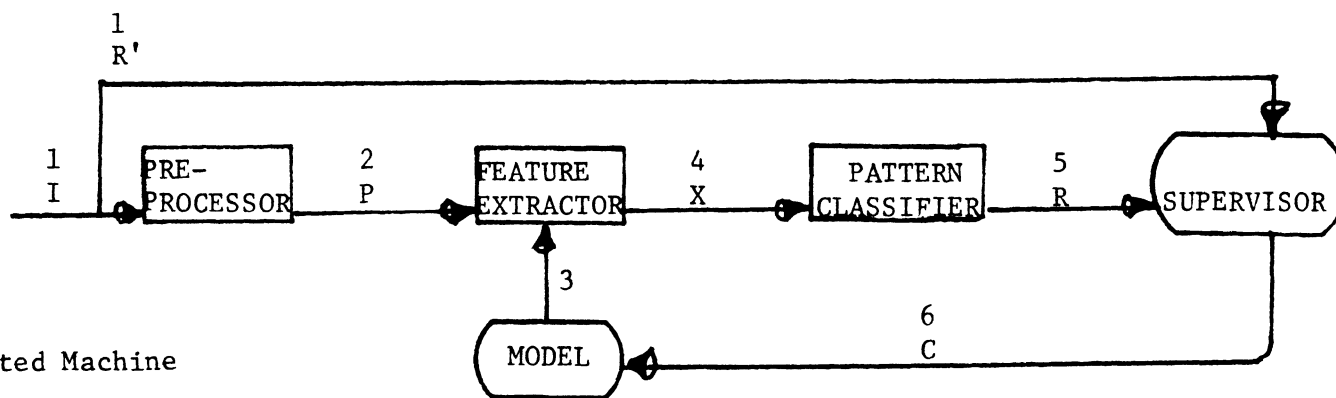
(B) Adapted Machine

Figure 31. Structural Comparison of (A) A Classical Learning Machine, and (B) The Adapted Learning Machine.

C. Clustering

In the current pattern recognition literature, clustering is not universally defined. As a property of data, it is the tendency of the same class to cling together in hyperspace. It is used to extract information about the structure of a data set. For example, in a population of sample points in the pattern space, clustering can be used to determine the number of major groups among the sample points. Clustering may reveal in addition some distance relations by finding a minimal spanning tree.

In this paper, clustering is used to compare c-sect arrays in the decision process of the pattern classifier. The c-sect arrays have been set by the feature extractor using path sensitizing; they appear to the pattern classifier as lattice arrays with 96 by 96 points. At each point there may be a 1 if an affected fiber passes through the point, or a 0 if an unaffected fiber passes through, or an X if no nerve intersects the c-sect at that point. The cluster of faulty fibers in each c-sect array is measured with three heuristic measures: "count" (KCC), "density" (SMOM), and "pureness" (SMX/NCC). The best cluster of faulty fibers should correspond to the location of the lesion in the patient.

KCC is a count of the number of affected nerves which pass through the c-sect. The faulty nerve lattice points have a centroid (MX,MY) in the c-sect, and an average distance from the centroid (SMOM). Because the fiber bundles are sensitized one at a time, the information needed to calculate these measures is not available at one time; hence, MX,MY and SMOM are calculated on the run and are continually updated.[43] When an unsensitized nerve fiber is encountered, it is examined to see

whether it lies within SMOM of the point (MX,MY). If it does, the counter SMX is incremented by one. Thus SMX is an approximate count on the number of unaffected nerves within SMOM of (MX,MY). The counter NCC is the count of the number of nerve intersections (affected and un-affected) which pass through the c-sect. The pureness, or p factor, is defined to be SMX/NCC. Two assumptions are made:

1. The area of damage in the patient is such that the affected fibers are adjacent and the area of damage is contiguous. (Exceptions here include multiple lesions and chiasmal shift.[7] These do occur in practice, but in the present system are not considered.)

2. Unaffected nerve fibers will not be surrounded by affected nerves in a cross section at the area of damage.

The first assumption requires that the best clustered fault array is more densely clustered than other arrays. The measurement SMOM yields a heuristic measure of this. By searching for the fault array with smallest SMOM, the decision rule shows preference to contiguous affected fibers. Consider the following geometric examples:

SMOM

(MX,MY)

SMOM = 10

SMOM = 13
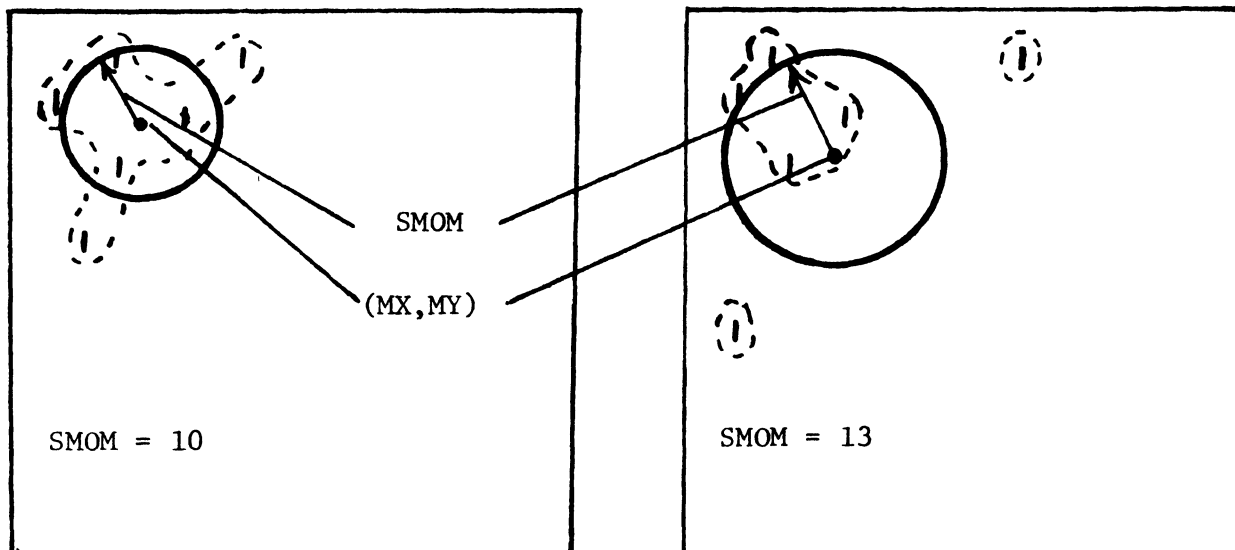
Array A   Array B

Figure 32. Two Fault Arrays.

Intuitively, array A is better clustered than array B. Array B seems to violate the condition that the affected fibers be adjacent. (This does not restrict the affected fibers to lie at lattice points one unit apart, because in the c-sect the fibers do not fill every lattice point. The fibers are assumed to lie in a distribution that is a uniform distribution throughout the c-sect.) Array B would perhaps represent three disjoint fault clusters, while array A seems one contiguous fault cluster as indicated by the dotted lines.

Assumption 2 accentuates the clustering idea of assumption 1, but more importantly, is an important cluster feature in its own right. Consider the following figure:
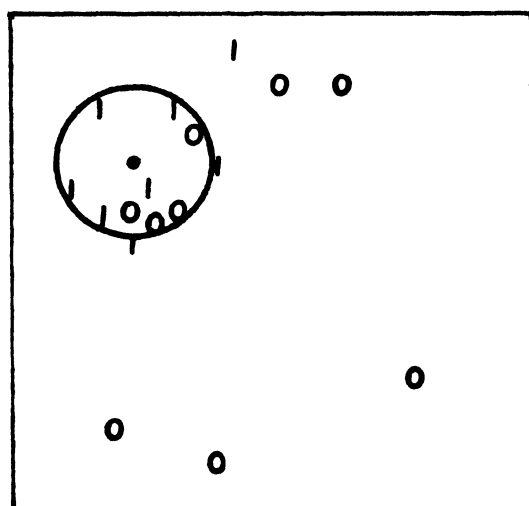
(A)

$NCC = 17$

$SMOM = 10$

$SMX = 4$

$P = \dfrac{4}{17}$

$NCC = 17$

$SMOM = 10$

$SMX = 3$

$P = \dfrac{3}{17}$
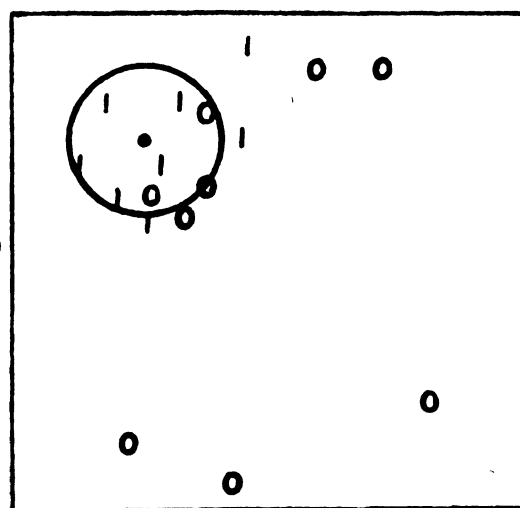
(B)

Figure 33. Fault Arrays with Unaffected Nerves.

NCC = 17

SMOM = 10

SMX = 1

$P = \dfrac{1}{17}$

(C)

NCC = 17

SMOM = 10

SMX = 0

$P = \dfrac{0}{17}$

(D)
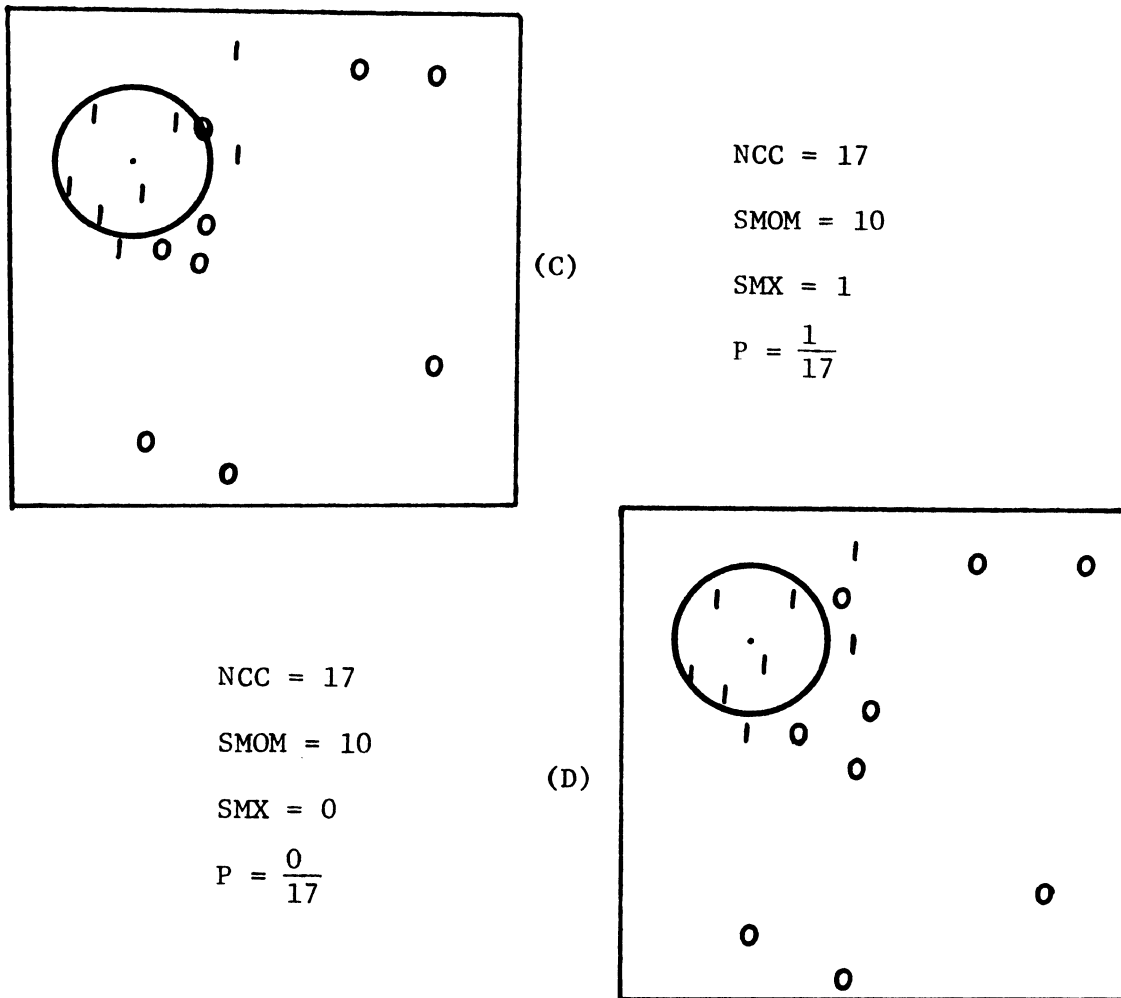
Figure 33 (contd.). Fault Arrays with Unaffected Nerves.

The fault array in (A) has many unaffected nerves among the cluster of faulty nerves. The array in (B) has less. The array in (C) has even less, while the array in (D) is the best cluster according to assumption 2. The pureness is accordingly descending in arrays A, B, C, D, illustrating its worth has a heuristic to accentuate assumption 2. To illustrate how the P factor accentuates the presence of assumption 1, in Figure 34 arrays A and B of Figure 32 are redrawn with some unaffected nerves added. When the SMOM is larger, the chances of engulfing unaffected nerves within this radius are greater. Hence, array B has a higher P factor than array A.
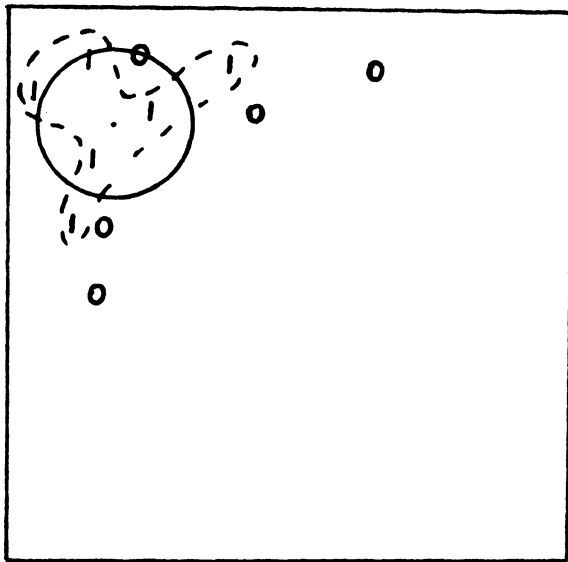
Array A   Array B
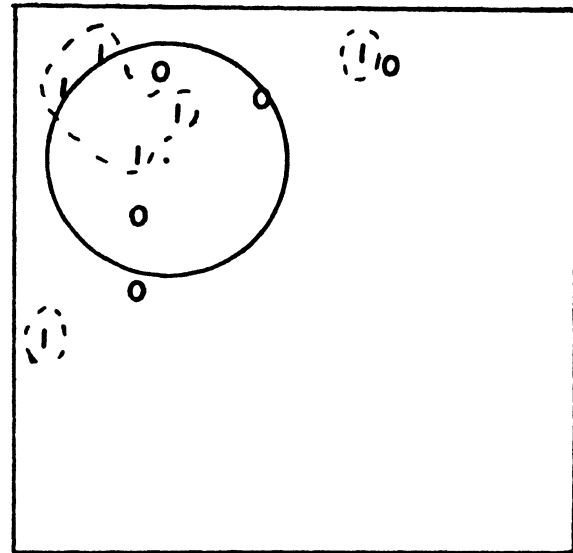
NCC = 11	SMX = 1	NCC = 11	SMX = 3
SMOM = 10	$P = \dfrac{1}{11}$	SMOM = 13	$P = \dfrac{3}{11}$

Figure 34. Fault Arrays of Figure 32
with Unaffected Fibers Added.

These measurements are meant to be heuristics. They are fast, efficient means of measuring the loosely defined quality of cluster. Possibly one might devise unusual fault arrays where these measurements do not work well, but in the present application they perform adequately.

D. Types of Learning

Learning in the VISUAL system is performed by perturbating the model; the status of the model contains the substance or knowledge that has been learned. This substance is somewhat intangible and defined and measured in terms of two heuristics, Local Learning and Global Learning. The system has two functions: building a good model of the visual pathway and performing diagnoses and simulation correctly. Local learning is concerned with the former, global learning with the latter.

1. Local Learning

Local learning concerns the behavior of the system with regard to each c-sect separately. Consider one fixed c-sect, c-sect i, and for the moment, consider the reduced model, M_i, to be one lattice array of dimensions 96 by 96 lattice points, and N elements (nerve fiber bundles) mapped into the 9126 lattice points. From the training set T pick a subset T_i of training patterns whose lesions are associated with c-sect i . Each member t of T_i will have the following information about the lesion. In addition, there is a parameter FRAC which is used with all patterns.

x = x coordinate of center of lesion

y = y coordinate of center of lesion

r = radius of extent of damage from center

where

$1 \leq x,y \leq 96$, x and y integers

$5 \leq r \leq 96 * \sqrt{2}$

$0.0 \leq FRAC \leq 1.0$.

The lesion represented by a test pattern is such that affected nerves

lie within r of the center of the lesion and that unaffected nerves lie outside a distance of f = r * FRAC from the center. Figure 35 illustrates a possible lesion associated with a test pattern t, where x = 1, y = 96, r = 72.0, FRAC = 0.6667.
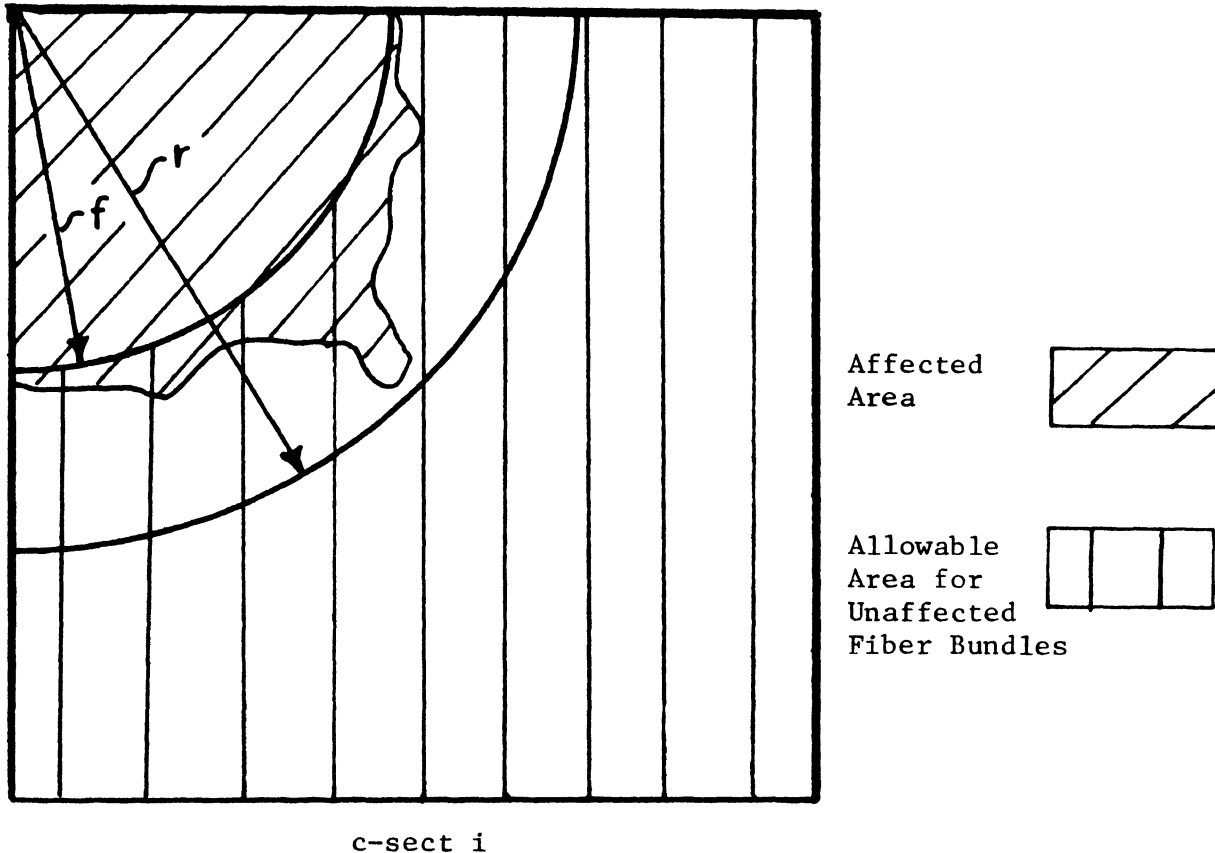
c-sect i

Figure 35. Area of Damage in c-sect i.

Note that although the regions outlined by the radii are sections of circles, the lesion is not restricted to that shape. The lesion shape does, however, contain the circle of radius f. Within the band between the circles of radius r and the circle of radius f, the shape of the lesion may vary considerably. The lesion will not extend past the circle of radius r. Unaffected nerves may lie anywhere outside the circle of radius f. Since we are dealing with maladies such as tumors and lesions, the shape of the affected regions may vary

considerably. The geometry of circles seems to align itself with the idea that the pressure will probably be proportionate to the distance from the source of the malady.

When a training pattern t is inputted to the system, the sense points are set and the path sensitizing is performed. In c-sect i, there will be two types of elements: 1's representing sensitized fibers and 0's representing unsensitized fibers; 'don't care' fibers being neglected. If the 1's lie within r and the 0's lie outside f, the model satisfies the training pattern t and no correction is necessary. If the model does not satisfy the training pattern, the model is perturbated. The 1's are moved toward the center and the 0's are moved away. The manner in which these elements are moved is discussed in Appendices A and D. Geometrically, it suffices here to note that the 1's and 0's are moved a fixed increment, INC. Figure 36 illustrates c-sect i with pattern t applied. In (A) the test pattern has been applied and three 1's and two 0's are in incorrect regions. (B) shows the movement initiated through a single application of the learning process. (C) shows the c-sect after the movement of the 1's and 0's. After one pass of learning has been applied, there is still a 1 in an incorrect region. More applications of pattern t are necessary to move it further toward its acceptable region.
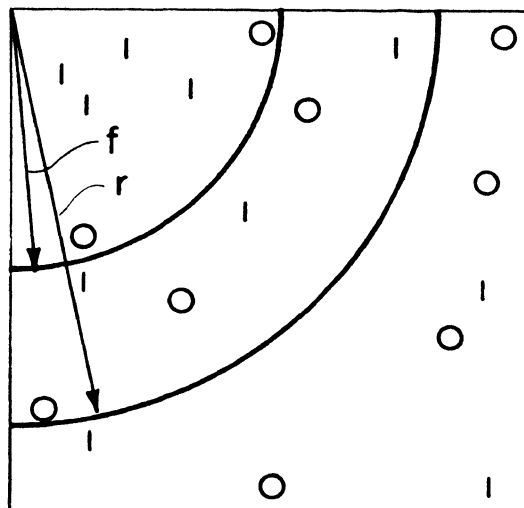
(A) Before Learning
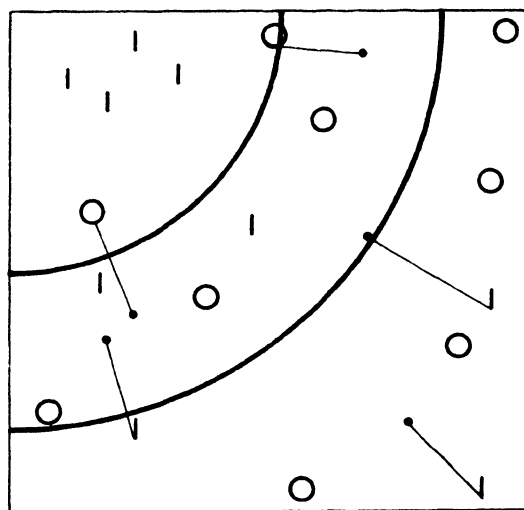
(B) During Learning
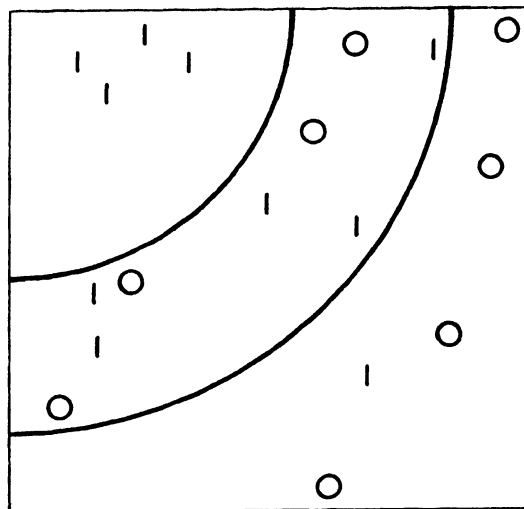
(C) After Learning

Figure 36. Arrangement of Fibers in c-sect
Before, During, and After Learning.

It is an important feature of the learning system that there are two types of movements of the fiber bundles: toward the center, and away from the center. It has been suggested to call this action "push-pull" learning. The system can learn that fiber bundles pass near an area when damage to that area causes loss of vision in those bundles. Conversely, the fiber bundles corresponding to the portion of vision not impaired should not lie near the affected area.

With regard to a single c-sect and a single test pattern, the local learning process using incremental movement can be applied repeatedly until the test pattern is satisfied. When more than one test pattern is applied to a c-sect, certain new situations must be dealt with. In particular, there are three major types of inconsistencies that may be seen when two test patterns are applied to a c-sect. Minor problems also arise from the implementation considerations.

The first type of inconsistency occurs when a nerve bundle is supposed to lie in each of two disjoint regions of the c-sect. Consider the following example:

Test Pattern (i)	Center x_i	y_i	r_i	Sense Point 7 Set to
1	1	96	40	1
2	10	1	30	1
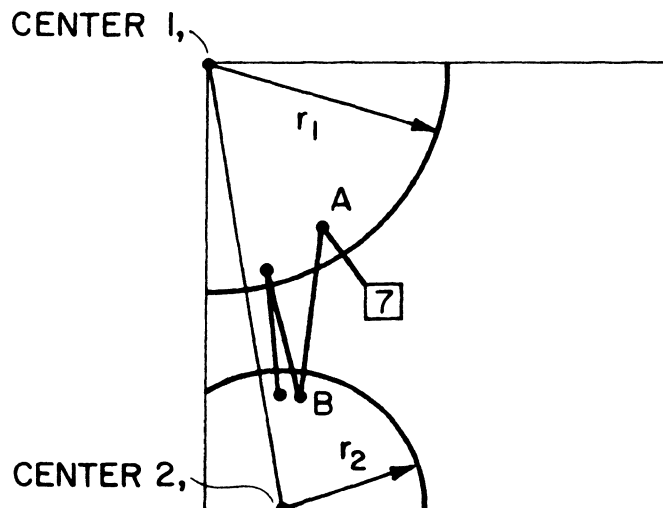
Figure 37.
Type I Inconsistency.

Nerve bundle 7 is in a location outside of r_1. Hence application of pattern 1 causes the fiber bundle to be moved to point A. Then application of pattern 2 moves the fiber bundle to point B inside the circle of radius r_2; however, during this movement the fiber bundle has moved out of the first region. Because the regions are disjoint, repeated applications of these two patterns will cause fiber bundle 7 to oscillate between the two regions along the line joining the centers.

The second type of inconsistency occurs when the fiber bundle is being sent out of overlapping regions. For example:

Test Pattern (i)	Center x	y	r_i	f_i	Sense Point 8 Set to
3	1	45	90	70	0
4	96	50	92	75	0
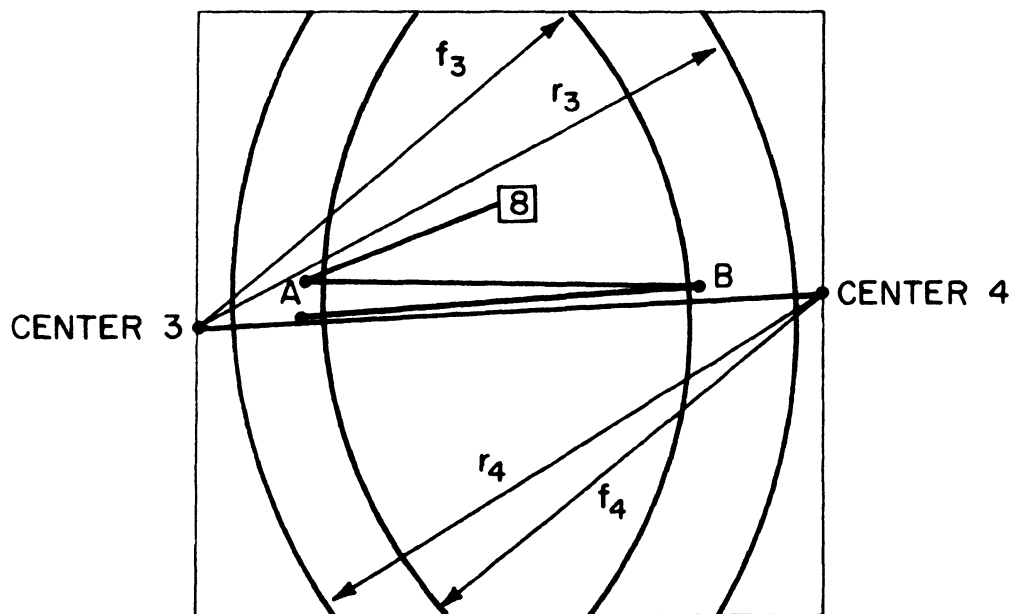
Figure 38. Type II Inconsistency.

Application of pattern 4 brings fiber bundle 8 to point A outside

of the radius f_4. Then application of pattern 3 moves fiber bundle 8

to point B outside of f_3. Since there is no area in the c-sect not

covered by circle f_3 or circle f_4 or both, repeated applications of

pattern 3 and pattern 4 result in oscillation of nerve 8 along the

line joining center 3 and center 4.

The third type of inconsistency occurs where a fiber bundle is

moved into a region and at the same time out of another region that

completely engulfs the first region. For example:

Test Pattern (i)	Center		r_i	f_i	Sense Point 9 Set to
	x	y			
5	40	1	30	25	1
6	42	1	46	40	0

CENTER 5   CENTER 6
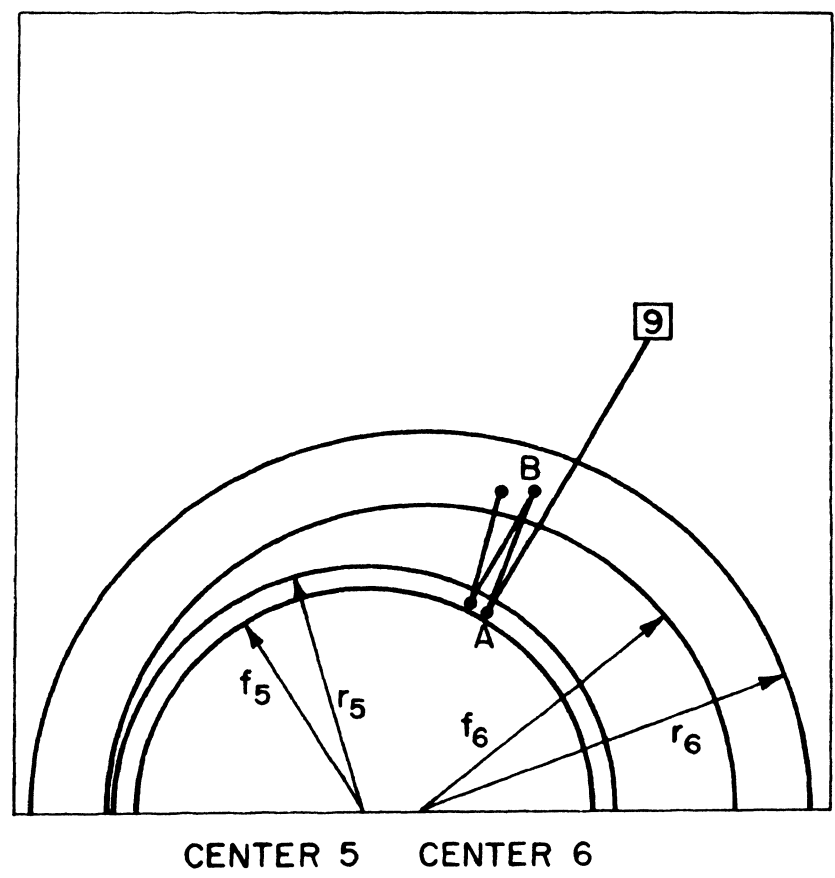
Figure 39. Type III Inconsistency.

Fiber bundle 9 is moved to point A after application of pattern 5.
Then applying pattern 6 moves bundle 9 to point B outside of circle f_6,
but also outside circle r_5. Repeated applications of pattern 5 and
pattern 6 will cause fiber bundle 9 to oscillate, since there is no re-
gion inside r_5 and outside f_6. The area where this oscillation will oc-
cur depends on many factors, some of which are discussed in Appendix D.
It is possible for the oscillation to progress along the circumference
of both circles to a region near point c, but other movement factors
may prevent this.

The inconsistencies arrive mainly because the extent of the damage
of a tumor or lesion is difficult to estimate, and the estimated value
of r may be in error. Revisions in the radii help undo these inconsis-
tencies. Also, if the shape of the tumor does not fit well into the
approximated regions specified for it, revision in r may aid the fit.

Consider now a finite sequence of test patterns $S = \{t_1, t_2, \ldots t_k\}$
being applied to the model, and at each application incremental move-
ment being used. Learning on each member of the sequence is called a
learning cycle.

With these concepts, the entire learning process adds another
level of learning (the revision of the radii) as shown in Figure 40.

The algorithm that selects the radius to be revised in step 5 of
the local learning process tries to pick the radius whose change will
help the most oscillating nerves. This algorithm is detailed in
Appendix B.
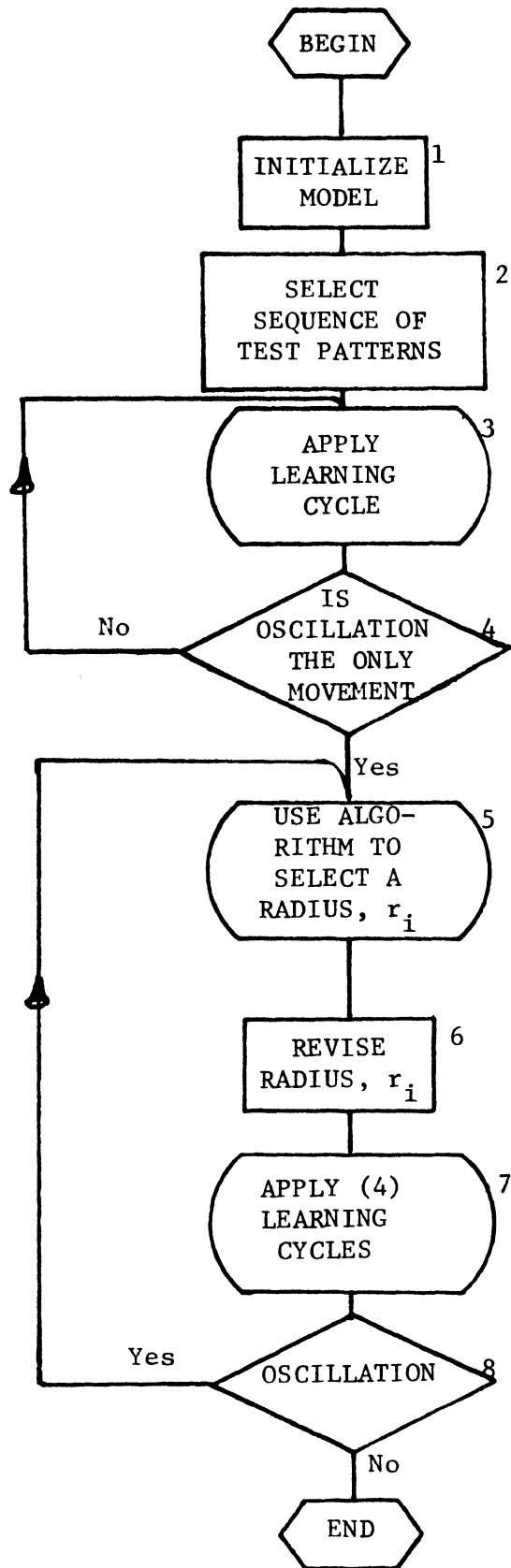
Figure 40. Local Learning Process on Sequence of Test Patterns.

In review then, a particular fiber bundle, j, has one of three relations with a given test pattern, i. The fiber may be sensitized, may be unsensitized, or may be unrelated to the test pattern with a 'don't care'. Each of these relations imposes a region, R_j^i, in which the fiber bundle should lie. When the fiber is sensitized, the region R_j^i is the interior of the circle r_j; for the unsensitized fiber, the region R_j^i is the exterior of f_j; and for the 'don't care' condition, the entire c-sect qualifies as the region.

The local learning process is a movement of each fiber bundle to the region $R^i = \bigcap_{j=1}^{k} R_j^i$ which is the *logical intersection* of the regions R_j^i. If there is an inconsistency between any two of the regions, the intersection is null. Revisions of the radii may alleviate most of the inconsistencies. When there are no inconsistencies, R^i is not null, and the movement will bring the fiber to a point within R^i. When inconsistencies exist, the movement results in oscillation as shown in Figures 37, 38, and 39. Both the movement and the radius revisions are part of the local learning process.

The progress and effectiveness of the local learning process on a given c-sect, i, and pattern sequence $t_1, t_2, \ldots, t_k$ is measured in two manners. The first measures the speed of convergence of the local learning process. Convergence is defined as the minimization of the number of fiber bundles moved per learning cycle. Alternately, when some fiber bundles are moved more than once during a cycle, counting the total number of movements per cycle gives a more sensitive measure of convergence of local learning. In Chapter VI, some sample runs of local learning on pattern sequence are given. Convergence is shown by plotting fibers moved and total movement versus cycles.

The second measure of local learning is an attempt to measure the validity of the final values of the radii and the final position of the fiber bundles. To implement this measure, a number of different starting configurations of the fiber positions are generated randomly. To each starting configuration, the local learning process is applied, using the same pattern sequence (and starting values of radii). The final position of each fiber bundle is compared for each starting configuration. The closer together these final positions are the more precise the learning process. Tightness of the clusters of the positions also indicates the amount of information in the pattern sequence. Comparing the final values of the radii for each starting configuration yields a measure on the validity of the learning process. This measure of the local learning is further discussed and examples presented in Chapter VI.

2. Global Learning

The behavior to the system with regard to diagnosis is the function which is considered global in nature. The term global implies the dependence of this function on more than one c-sect. Recall that the diagnosis algorithm compares the fault clusters of the different c-sects. In this regard, global learning is the ability of the system to change the model so that the clustering can yield proper diagnosis through the diagnosis decision algorithm.

Global learning occurs indirectly when the local learning process is invoked. For this reason the learning feedback loop implied in step 6 of Figure 31 is not an immediate correction response, but an accumulated correction response that comes from repeated applications of the local learning process. When an application of a pattern, t, is

applied in diagnostic mode, the user, upon viewing an incorrect diagnosis, corrects this diagnosis as to c-sect, (x,y) coordinates, and extent (r). The pattern t with the corrections may then be inserted into a test pattern sequence. Application of the sequence using the local learning process corrects the model. This type of local learning on many c-sects is important to the global learning progress.

Global learning is measured in terms of percent of correct diagnosis. The measurement is applied to two separate sets — the training set composed of patterns that have been used in a local learning process and a test set of patterns not used in the learning process. The results of the global learning are discussed in Chapter VI.

E.   Difference of Area Measurement

A final measure of the simulation ability of the system is to take patterns from the test set and simulate the damage at the c-sect, (x,y) coordinate, and extent given by the pattern. Using display mode, this results in a visual field. The difference in the area of the two visual fields is the measurement sought. The VISUAL system attempts to minimize this difference in area. This process is illustrated in Figure 41.

In summary, three types of measurement are used to monitor the learning process of the model and the system. Local and global learning measure the forward trace aspects, while the difference of area measurement is used to monitor the back trace simulation. Before leaving the subject of learning and measurement, note that Appendix C includes some of the approaches to learning that were tested and not used because of various reasons. However, these approaches are described to give suggestions for future research or refinements that may be made to the system.

Given Visual Field

c-sect
(x,y)
r

Simulate Lesion

Resultant
Visual Field

Compare
Visual Fields

Darkened Area Shows Where
the Visual Fields Differ
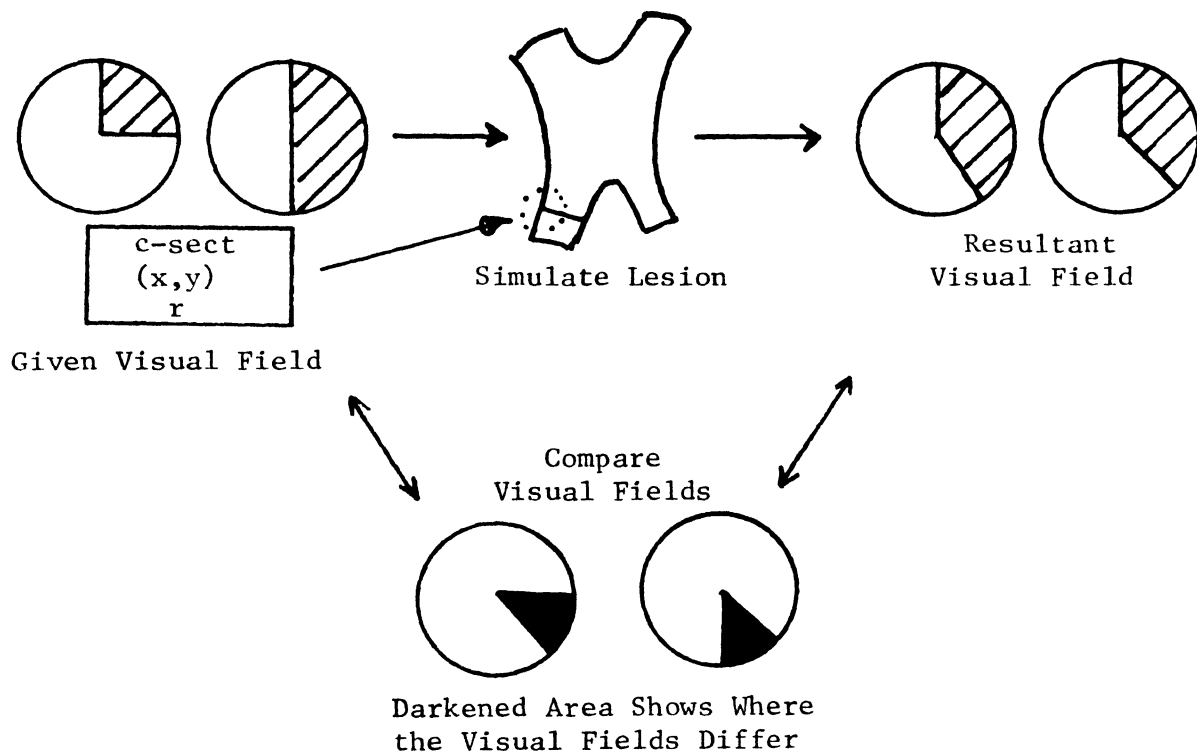
Figure 41. Difference of Area Measurements.

Chapter VI. RESULTS

In this chapter, some local learning results and global learning results are discussed. Also, considerations of the types of model development are presented.

A. Local Learning Results

Two aspects of local learning results are discussed: convergence of local learning on a pattern sequence, and local learning from different starting positions. In each case, examples are given first and the qualitative generalization.

In regard to local learning on a pattern sequence, some parameters may vary, affecting the convergence. INC is the parameter which controls the length of movement of a fiber bundle during application of a test pattern. (Refer to Appendix D for complete discussion of movement.) RINC is the parameter that is the change in the length of a radius which is determined by the radius selection algorithm discussed in Appendix B. FRAC, as shown in Chapter V, determines f by the relation f = r*FRAC. The unit of measure of the abscissa (time) in the following examples is LCYCL, the number of cycles per unit. Recall that a cycle is one pass through the training sequence.

Example 1

This is a typical example of local learning on a sequence. This example shows actual computer results of a sequence of eight patterns on c-sect 6 with FRAC = .8. Other data is:

	1	2-49	50-74	75-100
		Units		
LCYCL	25	6	3	1
INC	4	4	2	1
RINC	4	4	2	1

Pattern	X	Y	Starting Radius	Final Radius
1	1	48	17.7	25.7
2	48	1	91.6	87.6
3	48	1	58.0	54.0
4	48	1	16.9	44.9
5	96	96	135.4	135.4
6	96	96	120.0	108.0
7	96	96	31.5	91.5
8	96	48	60.0	28.0

The arrangement of the patterns and their (starting) radii is shown in Figure 42.
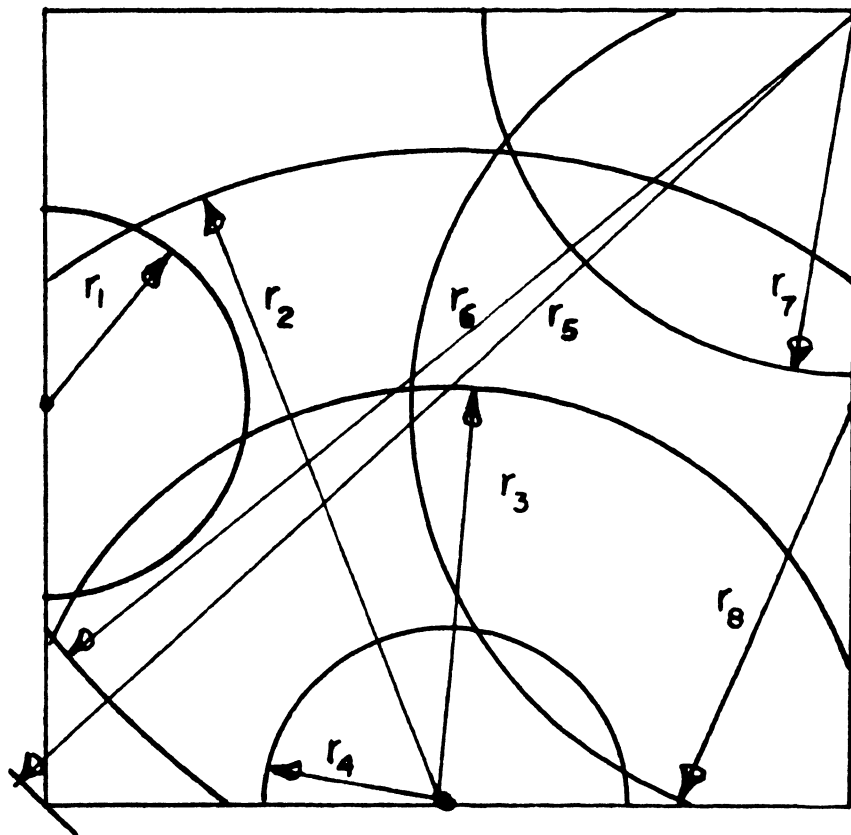
c-sect 6

Figure 42. Arrangement of Pattern Regions for Example 1.

Figure 43 plots the convergence of the local learning. Both the total fiber movement (t graph) and the number of fibers moved (g graph) are charted. The total fiber movements must always be greater or equal to the number of fibers moved. The strict inequality holds when one or more fibers is moved more than once in a cycle. After each unit, the radius changed is marked by vertical lettering ri ± j, where radius i is increased (+) or decreased (-) by j. At the end of the first cycle the value of the t graph and the g graph was 180 and 70, respectively. After 25 cycles (one unit) the values came sharply down to 55 and 30. The general trend of both graphs then was to decrease until convergence to zero at 52 time units. There are some areas of increase in the graph at 7, 8, 9, at 22, and at 24. At these points the measurement of learning indicates more fiber bundles are being moved because of the change of a radius. This does not indicate real "negative" learning because the radii value at these points is still converging to the final value that satisfies the pattern sequence. In the interval marked (a) on the graphs, the convergence shows no marked improvement, and continual monotone change of a radius (in this case radius 4) seemingly does not improve the model. On the contrary, the model is continually being improved in this region. Probably at this point inconsistencies are encountered which need large alterations in the radii to become consistent. Since the radii are changed in a "fixed-increment mode," the measure of learning seems constant until the accumulated change in the radii is sufficient.
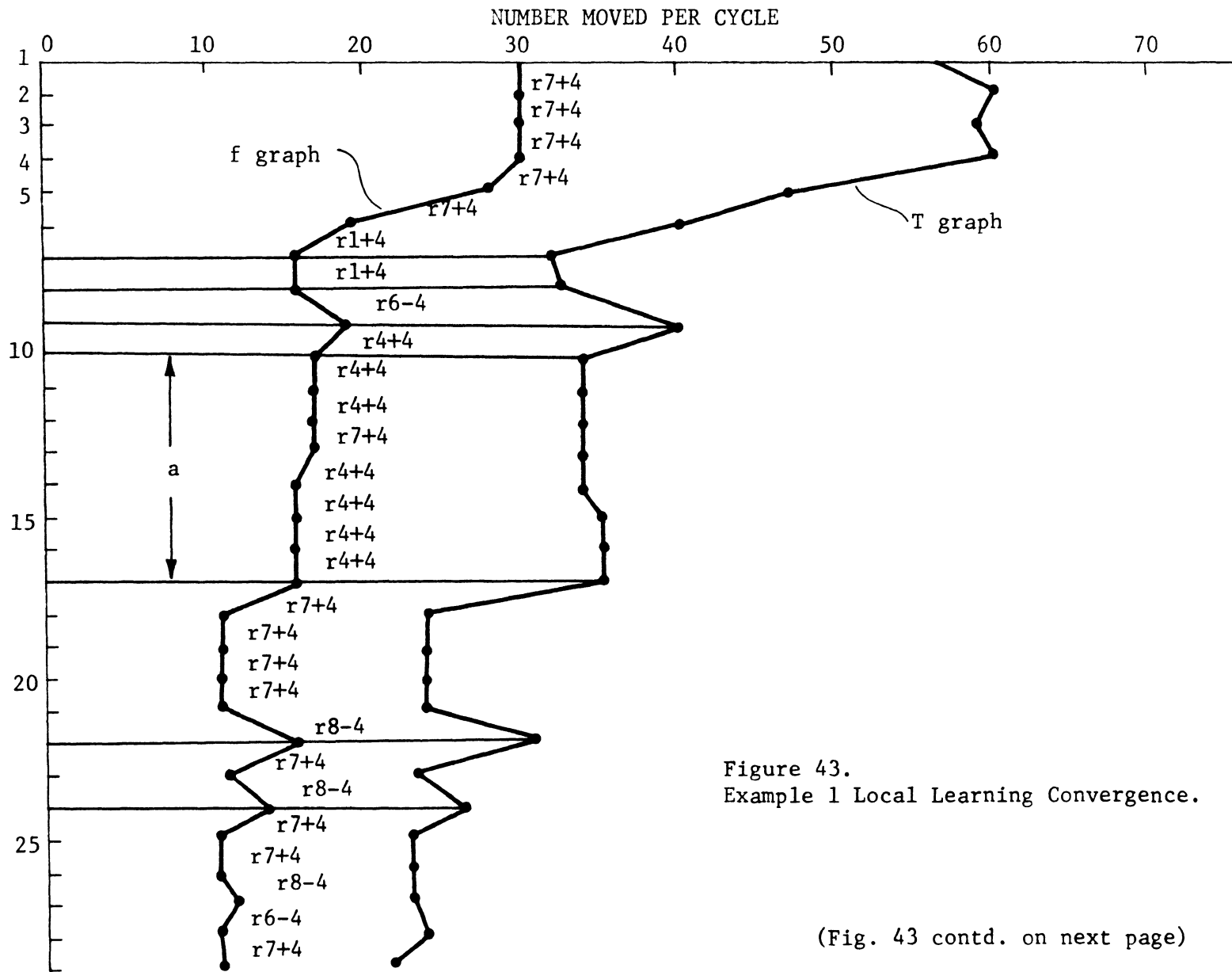
Figure 43.
Example 1 Local Learning Convergence.
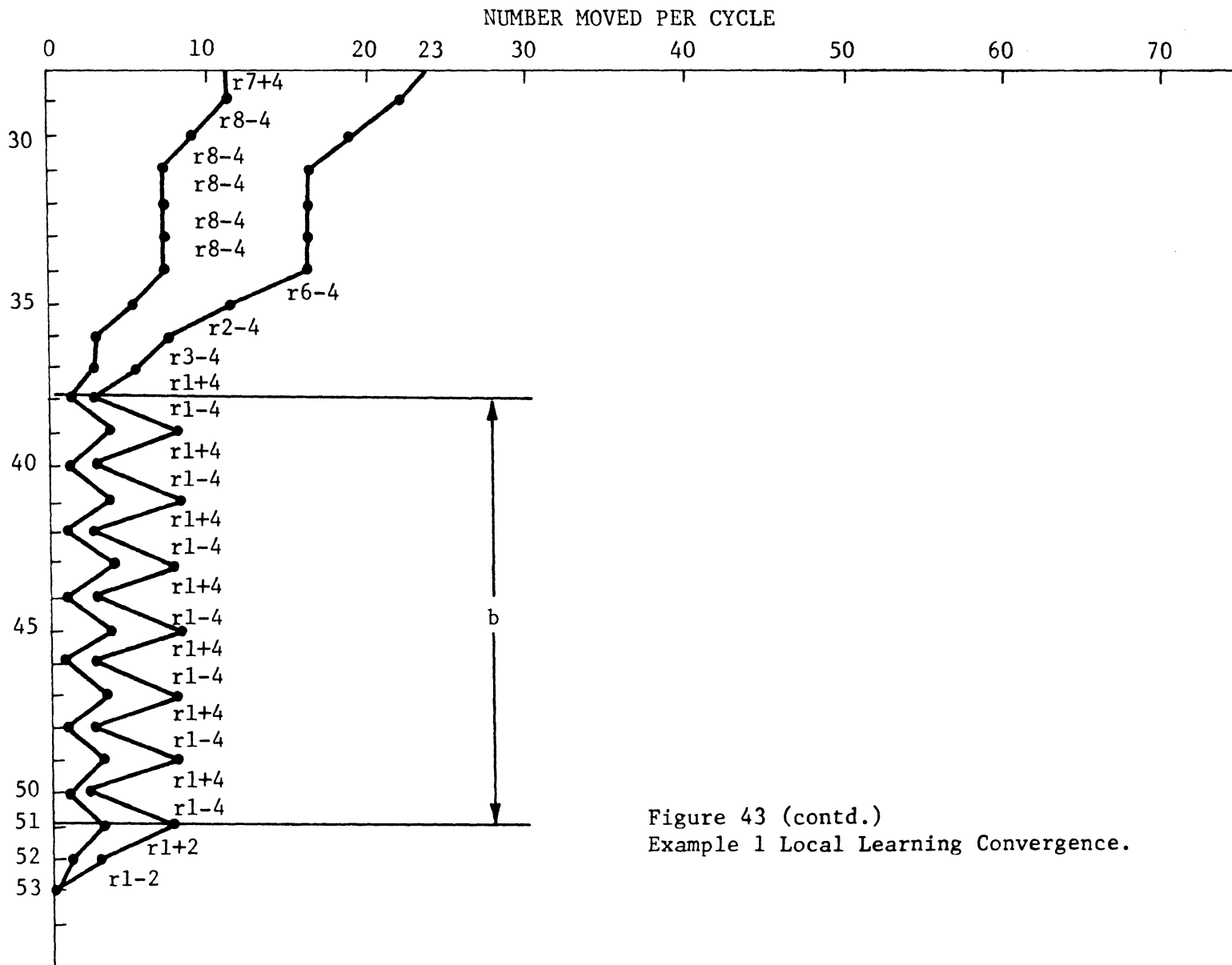
(Fig. 43 contd. on next page)

Figure 43 (contd.)
Example 1 Local Learning Convergence.

Another phenomena which is indicated in Figure 43 is oscillation of the radii values. In the interval b, the value of radius 1 oscillates between 29.7 and 25.7. Movement of a few fiber bundles oscillate also. In the 50th time interval, the parameters RINC and INC are halved, ending the oscillation and resuming convergence. The movement is very sensitive to INC, since in time slots 51 and 52, rl has no total change, but the change in INC allows the fiber bundles to slip into different lattice points. (See Appendix D for details on movement.)

In general, the convergence can be improved somewhat by using the absolute-error-correction type movement, which is discussed in Appendix D. The fixed-increment movement, although slower, is easier to control, and allows more flexibility. Especially in cases where convergence does not go completely to zero, the fixed-increment movement performs smoother. Consider Figure 44, where fiber bundle 6 needs to be in regions r_1, r_2, and r_3 to satisfy the pattern sequence. There are cases (when data is in error, for example) when these situations occur and the radii cannot be changed without introducing an equal or greater inconsistency involving perhaps other fibers. Absolute-error-correction movement will result in the final movement oscillation shown by the dotted lines, while fixed increment movement causes the bundle to oscillate less, indicated by the solid lines.

A second reason to prefer fixed increment mode is that it allows the system to handle patterns with multiple regions within a c-sect. For example, consider pattern 9, which is caused by pressure from below on c-sect 6. The pressure below is sufficient to push the chiasm up and press the top of the chiasm against tissue above the chiasm.

This results in some damage to fibers in two regions of the chiasm, as shown in Figure 45.
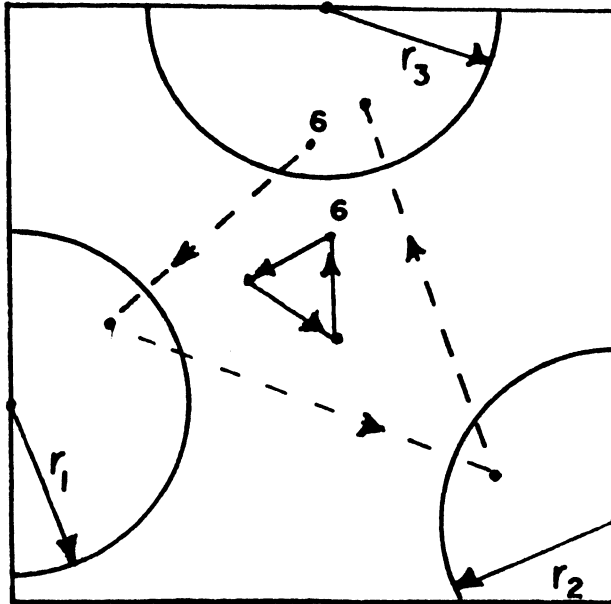
Figure 44. Oscillation for Two Types of Movement.

Figure 45. Region of Damaged Fibers in Pattern 9.
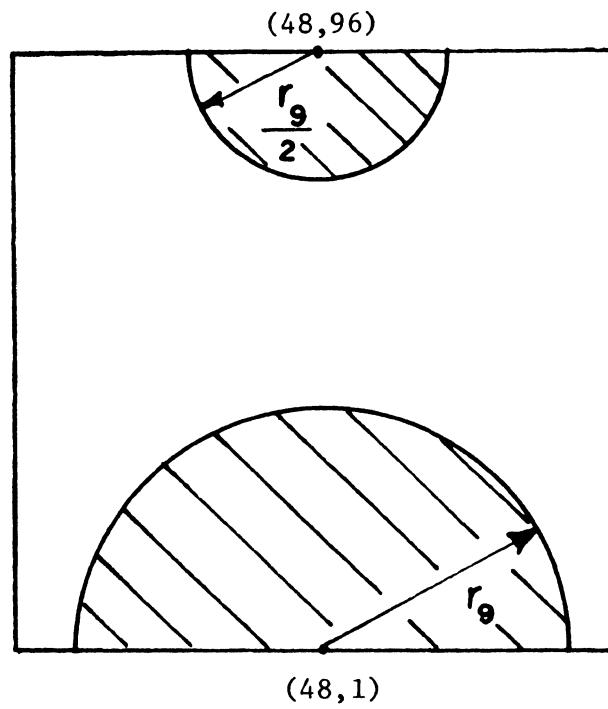
When pattern 9 is added to the pattern sequence in Example 1, the local learning cycle proceeds as before with slight modifications. First, pattern 9 is not used in the first five time intervals. This is to allow the fiber bundles to perform some major migration before being influenced by pattern 9. When pattern 9 is added to the learning sequence, each fiber is influenced only by the region of pattern 9 to which it is closest. Absolute-error-correction movement would cause implementation problems at that point.

Figure 46 shows the local learning profile of Example 1, with pattern 9 added, using the adaptations listed above.

NUMBER MOVED PER CYCLE

TIME

r7+4
r7+4
r7+4
r7+4
r8+4

f graph

r7+4

Add Pattern 9

r6-4
r6-4
r6-4
r6-4
r7+4
r7+4
r1+4
r6-4
r1+4
r1+4
r1+4
r7+4
r1+4
r4+4
r4+4
r1+4
r6-4
r1+4
r7+4

T graph

r6-4
r4+4
r3+4

Figure 46. Local Learning Convergence with Pattern 9 Added.
(Fig. 46 contd. on next page)

Figure 46 (contd.) Local Learning Convergence with Pattern 9 Added.

Example 2

This example uses the same pattern sequence and data as Example 1. Six different starting configurations are used, and local learning performed for each configuration. The starting configurations are generated randomly uniform over the discrete 96 by 96 lattice points.

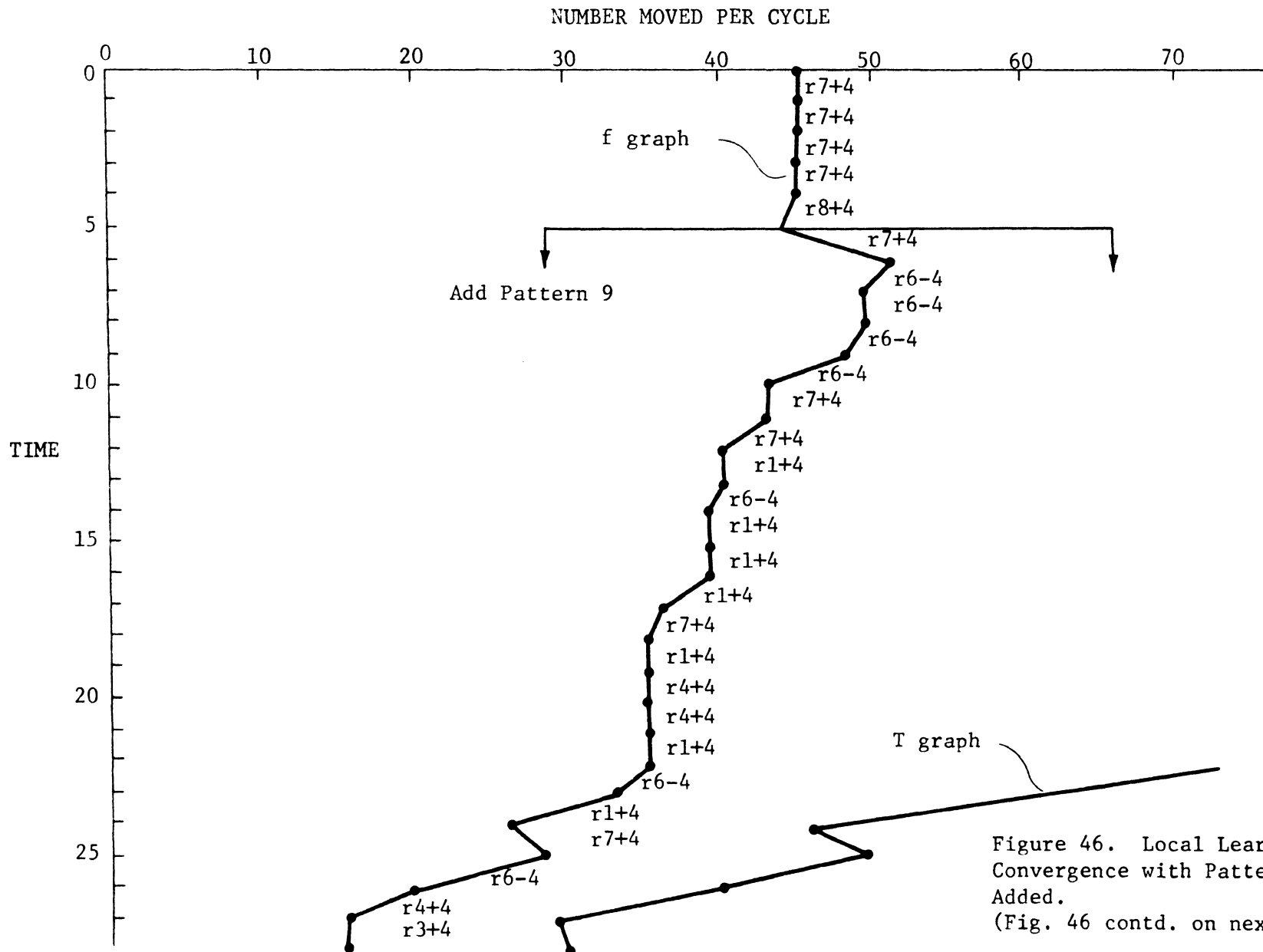For each fiber bundle, the starting position and ending position is noted for the six learning runs. The starting centroid (SC) and average distance from the centroid (ASD) for the six starting locations are calculated. Likewise, the ending centroid (EC) and average distance of the ending locations from the ending centroid (AED) are calculated. This AED is the most important measurement. A small AED indicates that (for that fiber) no matter where the starting position is set, the fiber moves to nearly the same ending position. This also means that both the learning heuristic is good, and that the pattern sequence contains a significant amount of information concerning that fiber. Figures 47, 48, and 49 show starting and ending configurations for fibers 4, 24, and 222, in Example 2. S_i and E_i are positioned at the ith starting and ending locations, respectively. The data shown graphically in the figure is listed in Table II.

The average distance is reduced by 93.2 percent; 87.5 and 83.6 percent for fibers 4, 24, and 222. In general, the reduction rate is about 85 percent in Example 2.
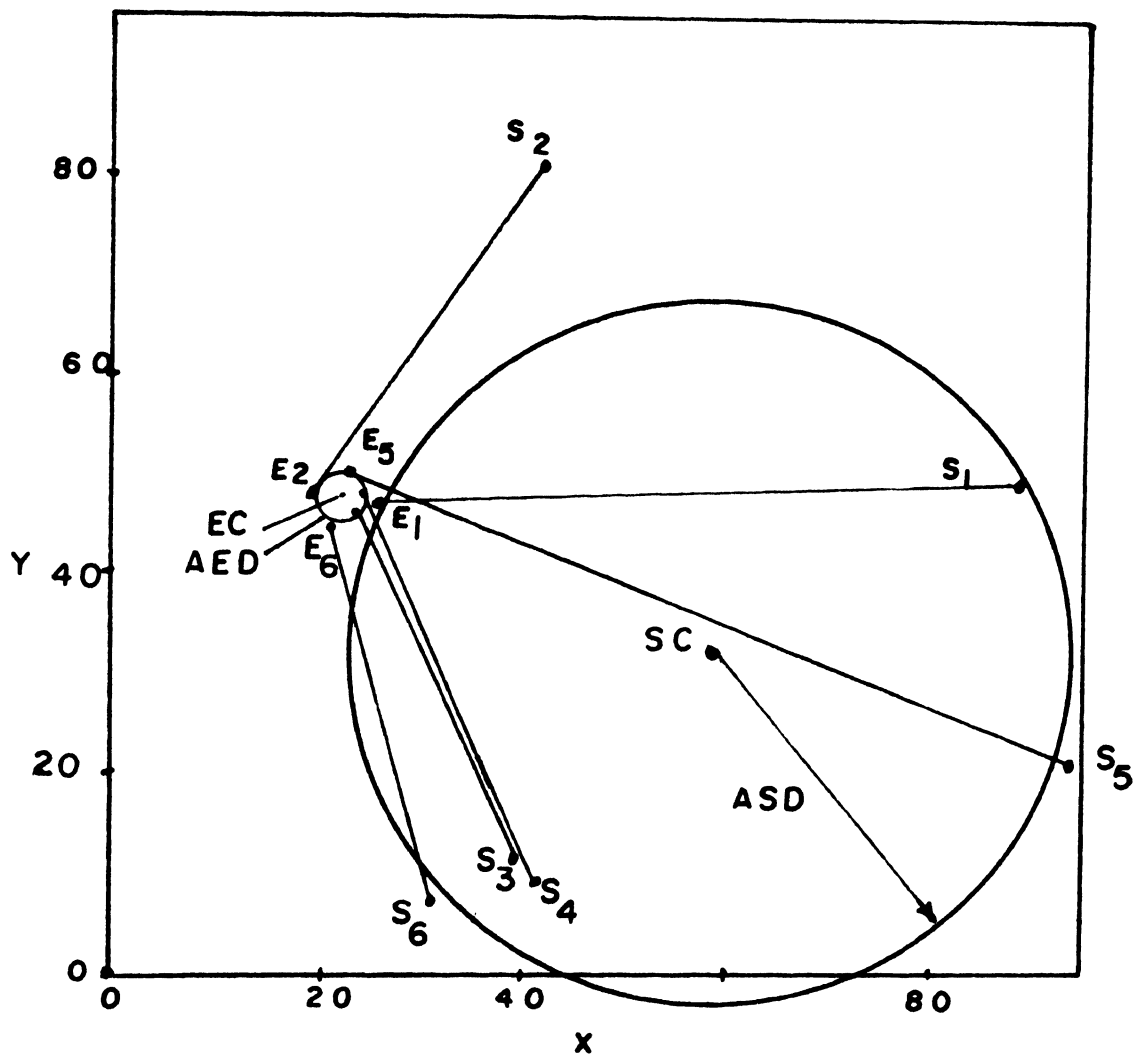
Figure 47. Distribution of Starting and Ending
Locations for Fiber Bundle 4.
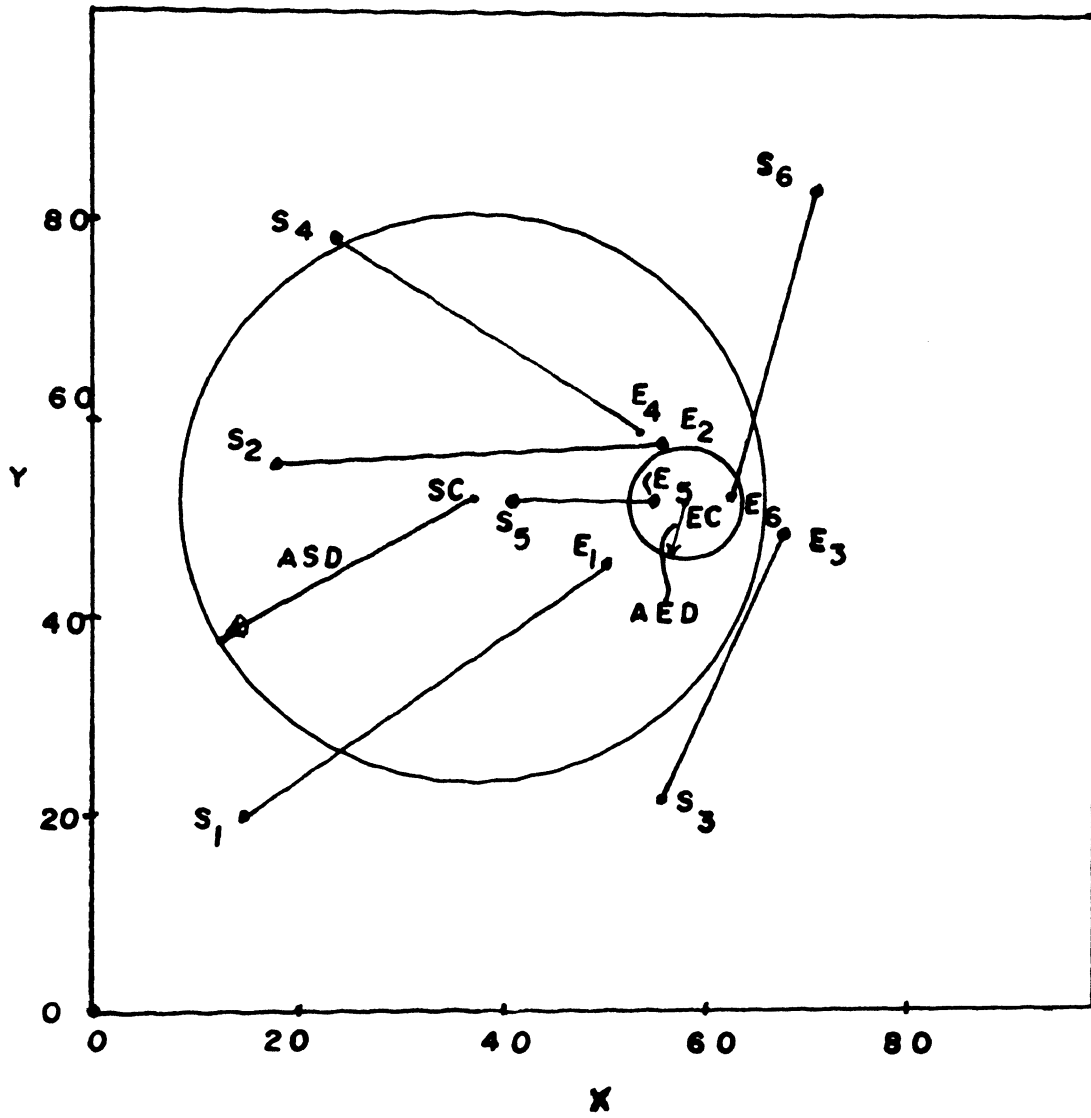
Figure 48. Distribution of Starting and Ending
Locations for Fiber Bundle 24.
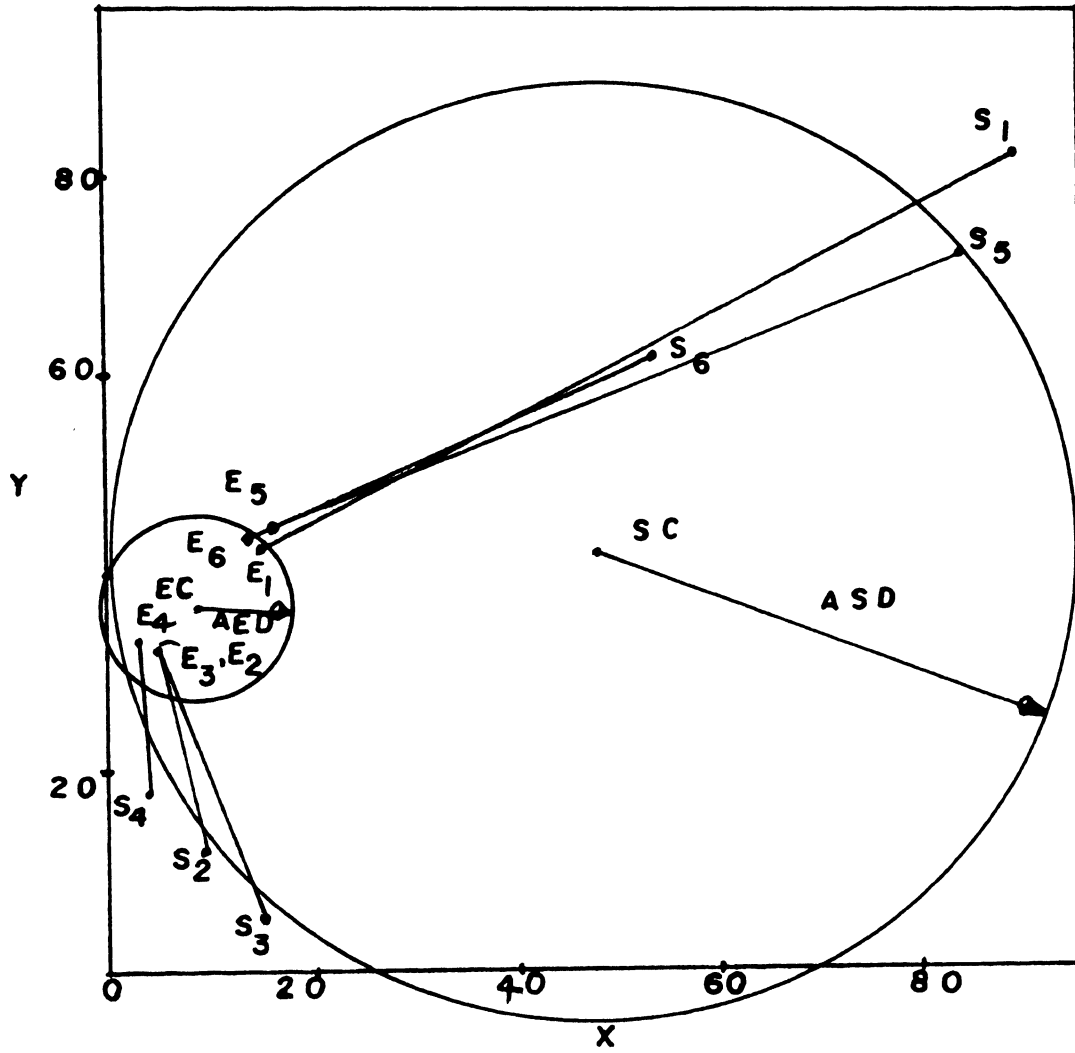
Figure 49. Distribution of Starting and Ending
Locations for Fiber Bundle 222.

Table II.

DATA FOR EXAMPLE 2

Position	Fiber	Run Number												Centroid		Average Distance from Centroid
		1		2		3		4		5		6				
		X	Y	X	Y	X	Y	X	Y	X	Y	X	Y	X	Y	
Starting	4	87	49	42	81	40	12	43	10	95	21	32	7	56.8	30.3	35.11
Ending	4	26	47	19	48	24	46	24	48	45	49	21	47	23.2	47.5	2.39
Starting	24	15	19	19	55	56	21	24	77	42	51	71	81	37.9	50.7	28.67
Ending	24	51	45	57	58	68	47	55	57	56	52	63	51	58.3	51.7	6.75
Starting	222	89	92	10	12	15	5	4	17	84	72	55	60	48.8	43.0	46.26
Ending	222	15	43	4	32	4	32	3	33	16	44	15	43	9.5	37.8	8.03

Another measurement to note here is the consistency of the final values of the radii. Table III shows that the value of radius 4 very consistently stops near 56.9, even though its value change is one of the largest of the radii. The value of radius 2 varies the greatest. Addition of one or two more patterns in the pattern sequence helps lessen this variance.

Table III.

RADIUS VALUES FOR EXAMPLE 2

Radius	Starting	1	2	3	4	5	6
1	17.7	19.7	17.7	20.7	17.7	14.7	17.7
2	91.6	79.6	79.6	59.6	71.6	63.6	79.6
3	58.0	62.0	58.0	58.0	58.0	58.0	58.0
4	16.9	56.9	56.9	56.9	56.9	56.9	58.9
5	134.4	134.4	134.4	134.4	134.4	134.4	134.4
6	120.0	118.0	120.0	118.0	114.0	116.0	116.0
7	31.5	87.5	99.5	88.5	88.5	87.5	91.5
8	60.0	28.0	22.0	27.0	25.0	28.0	22.0

B.   Global Learning Results

Global learning is the ability of the system to improve its diagnostic performance through local learning in more than one c-sect. For example, suppose pattern A is diagnosed as belonging to c-sect i rather than c-sect j (its true location). This may be caused by one of many decision steps in the diagnosis algorithm (see Appendix B). Further, suppose the SMOM(i) (the SMOM measurement of pattern A in c-sect i) is less than SMOM(j) (the SMOM measurement in c-sect j). This would result in a misclassification. After some local learning in c-sect j and c-sect i, it is probable that the SMOM(i) increased, while SMOM(j) decreased, rectifying that decision step in the diagnosis algorithm for pattern A. A similar consideration can be shown concerning the pureness factor and other parameters. The results of local learning to improve the diagnosis can be seen in the following example.

Example 3

Two sets of patterns are presented to the system. Each set has some patterns belonging to c-sects 3, 6, and 7. The set T will be used to train the system. That is, the patterns of T belonging to c-sect 3 will be applied to c-sect 3 for local learning. The same will be done for the patterns of T belonging to c-sects 6 and 7. The patterns in set Q will not be used for training, since Q will be a "test" set.

Before learning the patterns, T had 9 incorrect and 1 correct diagnosis for a 10 percent performance. Set Q had 9 incorrect and 1 correct for 10 percent performance.

After the learning, the system performance was 4 incorrect and 6 correct from set T for 60 percent and 50 percent improvement, while the test set Q showed 5 incorrect and 5 correct for a performance of 50 percent and 40 percent improvement.

As anticipated, the system did better with the T set than the Q set; however, it is important to notice that significant improvement is accomplished for the Q set, even though none of the patterns in Q were used in learning. Because of this phenomenon, it is felt that after a substantial amount of local learning, the system will perform well even on patterns that were never used in learning.

C. Difference of Area Results

Example 4

The global learning can also be measured by the "difference in area method" mentioned in Chapter V. That is, using the backtrace of display mode, simulate a lesion in a particular location and find the resultant visual field; compare this visual field to the known visual field and measure the difference of areas of the two fields. As this example shows, this area is lessened by local learning (see Figures 50 and 51). Pattern A was a member in set T of the preceding example, and pattern B is from set Q. In each case the true pattern is shown first, then in dotted lines superimposed on the backtraced patterns, with the difference in area shaded. As with the other measurement of local learning, the improvement for both patterns is substantial, with the system performing better on pattern A.

Original
Pattern

Backtrace
Pattern
Before
Learning

Difference in Area = .12302

Backtrace
Pattern
After
Learning

Difference in Area = .05000
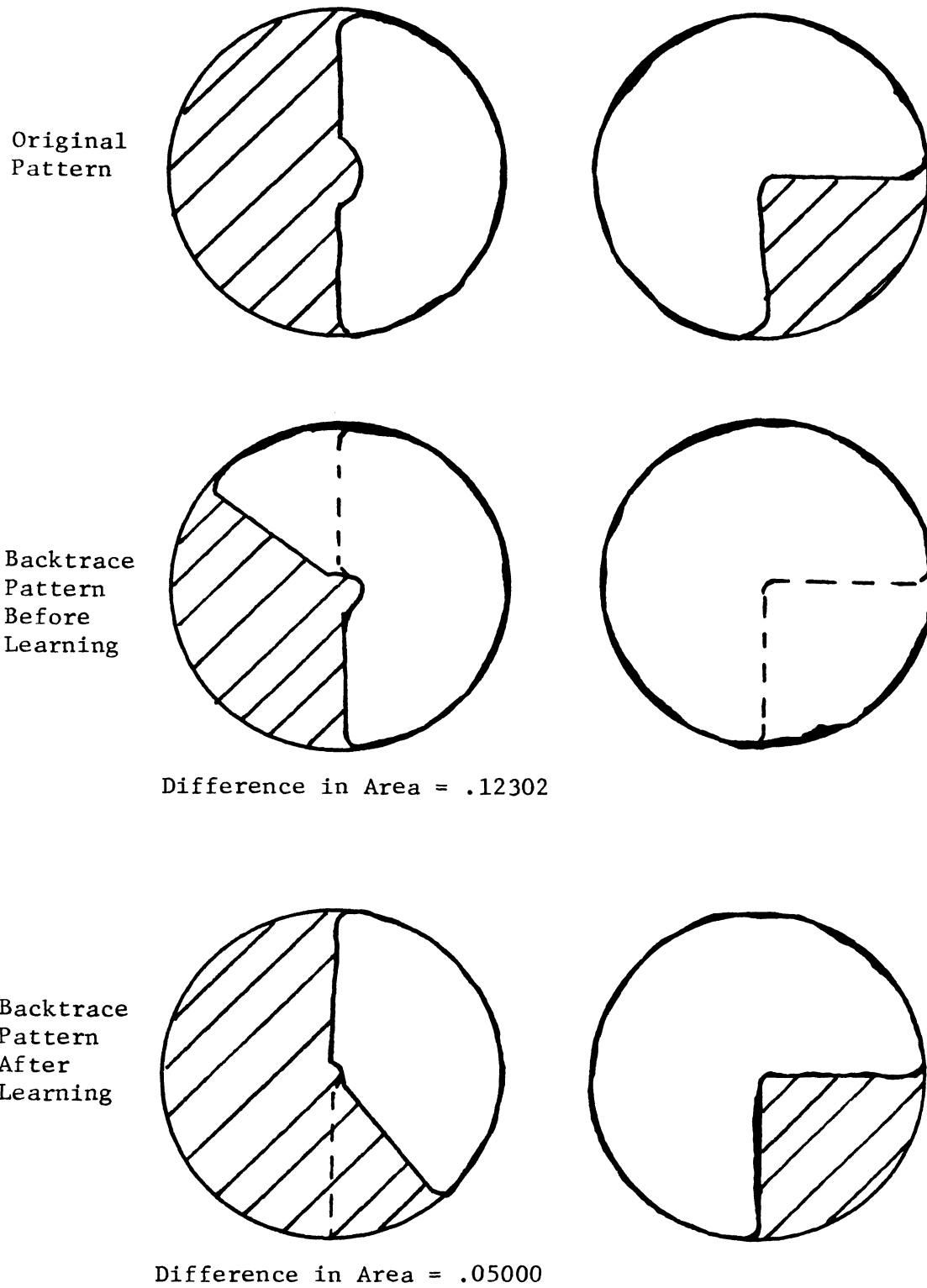
Figure 50. Difference of Area for Pattern A.

Original
Pattern

Backtrace
Pattern
Before
Learning

Difference in Area = .26984

Backtrace
Pattern
After
Learning

Difference in Area = .22006

Figure 51. Difference of Area for Pattern B.

D. Types of Model Development

There are two criteria on which the development of the model is measured: 1) Does the model perform diagnosis well under simulation? 2) Does the model accurately represent it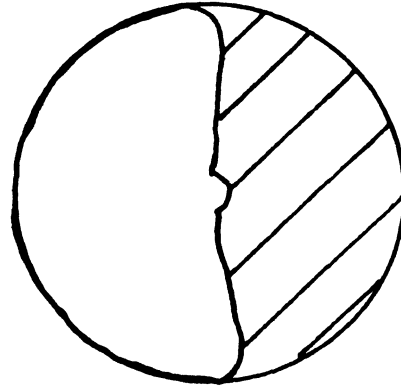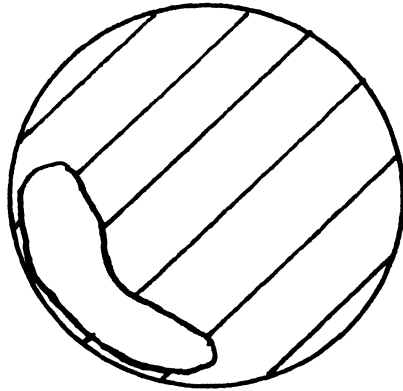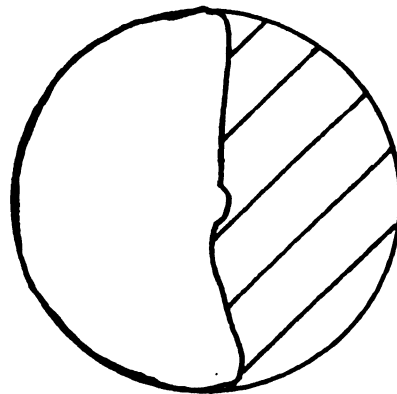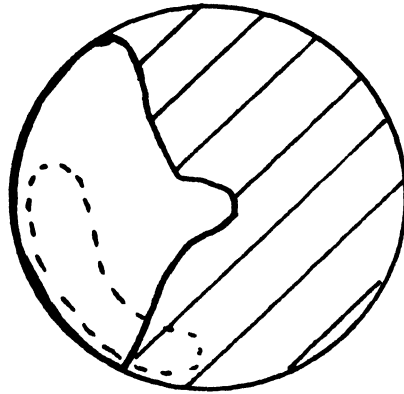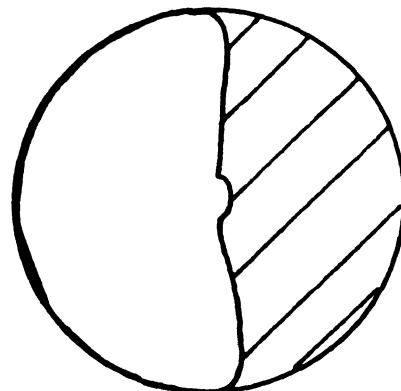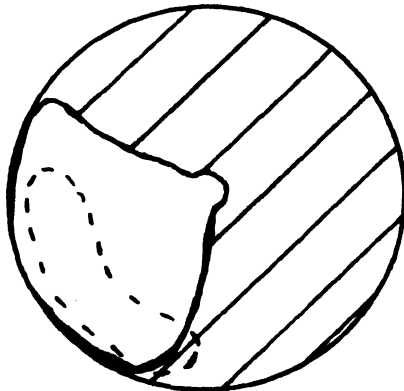s physical counterpart, the anatomy of the visual pathway? Four types of models may result. The model sought is one which satisfies both criteria. It is conceivable that a model may be developed which satisfies one of the criteria but not the other, or perhaps the model will not satisfy either. Before I can definitely assert that I have developed a model of the first type, I feel that the model and the system must undergo more field testing by ophthalmological experts. The evidence thus far supports such an optimistic conjecture. First, the global learning, of which Example 3 is typical, shows criteria 1 is satisfied. Secondly, the graphical display of the model enables the ophthalmologist to compare the model to the actual anatomy of the visual pathway as he knows it. Also, the results of the difference of area measurements as shown in Example 4 tend to support the conjecture that the model closely represents its physical counterpart.

E. Conclusions

The system presented shows that through interactive graphics, the tasks of learning and diagnosis can be merged. In fact, in this research the tasks are interconnected and complimentary — faulty diagnosis yields learning, learning improves the model, and the improved model improves the diagnosis.

Furthermore, the learning algorithms, because of their originality, are a contribution in learning machines. I hope and feel that this push-pull learning algorithm will have broad applications. Finally,

the system shows that utilizing path sensitizing and clustering con-
cepts together is a practical and efficient means of topologically
diagnosing networks.

The present system has two notable shortcomings. First, in diag-
nostic mode, the system considers only one isopter at a time. There
is often some information that can be gained by comparing several
isopters of a patient's visual fields.[7] When the system is expanded
to include the optic nerve and the post-chiasmal pathway, the ability
of the system to consider more than one isopter in its diagnosis pro-
cedure will become important. Secondly, a few diseases result in
multiple pressure locations in the visual pathway. These diseases
cannot be handled properly by VISUAL, , although intuitively it seems
that they could be dealt with by extending some of the concepts
developed in this research.

BIBLIOGRAPHY

1. Shelman, C. B. "A Proposal for an Automated Visual Field System." Argonne National Laboratory, Applied Mathematics Division, Technical Memorandum No. 124, October 1968. (Unpublished)

2. Goldbogen, Geof. "A Preliminary Report on the Research of the Visual Fields Problem." Argonne National Laboratory, Applied Mathematics Division, Technical Memorandum No. 151, August 1969. (Unpublished)

3. Szygenda, S. A. and Geof Goldbogen. "Implementation and Extension of Multidimensional Path Sensitizing in a Simulation and Diagnostic System," Proceedings of the Seventh Annual Allerton Conference on Circuit and System Theory, University of Illinois (October 1969), 284-292.

4. Goldbogen, Geof. "Interactive Diagnosis of Visual Fields Utilizing Path Sensitizing," IEEE Conference Record of the Symposium on Feature Extraction and Selection in Pattern Recognition, Argonne National Laboratory (October 1970), 192-199.

5. Goldbogen, Geof. "Modeling the Visual Pathway for Interactive Diagnosis of Visual Fields," Proceedings of Second Annual Pittsburg Conference on Modeling and Simulation, University of Pittsburg (March 1971), 32-40.

6. Nilsson, Nils J. Learning Machines, Foundations of Trainable Pattern Classifying Systems. New York: McGraw Hill Book Company, 1965.

7. Harrington, David O. The Visual Fields (A Textbook and Atlas of Clinical Perimetry). St. Louis: C. V. Mosby, 1964 (Second Edition).

8. Lynn, Dr. John R. Personal interview. December 1970.

9. Kolers, Paul A. "Some Psychological Aspects of Pattern Recognition," Recognizing Patterns, Studies in Living and Automatic Systems. Cambridge, Mass.: MIT Press, 1968. 4-61.

10. Chung, Shin-Ho. "Neurophysiology of the Visual System," Recognizing Patterns, Studies in Living and Automatic Systems. Cambridge, Mass.: MIT Press, 1968. 82-101.

11. Lyle, Willoughby. Applied Physiology of the Eye. Bath, England: Pitman Press, 1958.

12. Bartley, S. Howard. Vision, A Study of Its Basis. New York: Hafner Publishing Company, 1963.

13. Lettvin, J. Y., H. R. Maturana, W. S. McCullough, W. H. Pitts. "What the Frog's Eye Tells the Frog's Brain," Proceedings of the IRE (November 1959), 1940-1951.

14. Eldred, R. D. "Test Routine Based on Symbolic Logic Statements," Journal of the ACM, Vol. 6(1), 1959. 33-36.

15. Armstrong, D. B. "On Finding a Nearly Minimal Set of Fault Detection Tests for Combinational Logic Nets," IEEE Transactions on Electronic Computers, Vol. EC 15(1), 1966. 66-73.

16. Schneider, P. R. "On the Necessity to Examine D-Chains in Diagnostic Test Generation - An Example," IBM Journal of Research and Development, Vol. 11(1), 1967. 114.

17. Roth, J. P. "Diagnosis of Automatic Failures: A Calculus and a Method," IBM Journal of Research and Development, Vol. 10, 1966. 278-291.

18. Baron, Robert J. "A Model of the Elementary Visual Networks of the Human Brain," International Journal of Man-Machine Studies, Vol. 12(3), July 1970. 267-290.

19. Blum, Harry. "An Associative Machine for Dealing with the Visual Field and Some of Its Biological Implications," Biological Prototypes and Synthetic Systems, Vol. 1. New York: Plenum Press, 1962. (E. E. Bernard, editor)

20. Chambers, Roger A. and Donald Michie. "Man-Machine Cooperation on a Learning Task," Computer Graphics. London: Plenum Press, 1969. 179-189. (R. D. Parslow, editor)

21. Sammon, John W. "Interactive Pattern Analysis and Classification," IEEE Transactions on Computers, Vol. C-19(7), July 1970. 594-616.

22. Licklider, J. C. R. and W. E. Clark. "On-Line Man-Computer Communication," Proceedings 1962 Spring Joint Computer Conference, Palo Alto, California: National Press, 1962. 113-128.

23. Bush, Vannorav. "As We May Think," Atlantic Monthly. July 1945. 101-108.

24. Smith, Lyle B. "A Survey of Interactive Graphical Systems for Mathematics," Computing Surveys, Vol. 2(4), December 1970. 261-301.

25. Licklider, J. C. R. "Graphic Input - A Survey of Techniques," Computer Graphics Utility/Art/Production. Washington, D.C.: Thompson Book Co., 1967. 39-70. (Fred Gruenberger, editor)

26. Luxenberg, H. R. "Survey and History of the State-of-the-Art." Computer Graphics Utility/Art/Production. Washington, D.C.: Thompson Book Co., 1967. 23-38. (Fred Gruenberger, editor)

27. Nievergelt, J. (Editor). Emerging Concepts in Computer Graphics. New York: W. A. Benjamin, Inc. 1968.

28. Nievergelt, J. (Editor). Pertinent Concepts of Computer Graphics. Chicago: University of Illinois Press, 1969.

29. Sutherland, I.E. "Sketchpad: A Man-Machine Graphical Communication System." Proceedings 1963 Spring Joint Computer Conference. Washington, D.C.: Spartan Books, 1963. 329-346.

30. Sutherland, I.E. "The Ultimate Display," Proceedings of the IFIPS Congress, Part II. Washington, D.C.: Spartan Books, 1965. 506-508.

31. Hansen, W. J. Creation of Hierarchic Text with a Computer Display. Ph.D. Dissertation. Presented to Stanford University. May 1971.

32. Cohen, Stan. "An Introduction to Speakeasy," Argonne National Laboratory, Physics Division, Information Report PHY-1968E. December 1968.

33. IBM System/360 Operating System. Graphic Subroutine Package (GSP) for Fortran IV COBOL, and PL/I. File No. S360-24,-25,-29 Form C27-6932-3 IBM Corp. Programming Publications, Kingston, N.Y. 1970.

34. Feigenbaum, Edward A. and Julian Feldman. Computer and Thought. New York: McGraw-Hill Book Company, 1963.

35. Stewart, D. J. Automation Theory and Learning Systems. New York: Academic Press, 1967.

36. Mendel, J. M. and K. S. Fu. Adaptive, Learning, and Pattern Recognition Systems. New York: Academic Press, 1967.

37. Nagy, George. "State of the Art in Pattern Recognition." _Proceedings of the IEEE_, Vol. 56(5), May 1968. 836-862.

38. Yu-Chi Ho. "On Pattern Classification Algorithms - Introduction and Survey." _Proceedings of the IEEE_, Vol. 56(12), December 1968. 2101-2114.

39. Sebestyen, George S. _Decision Making Processes in Pattern Recognition_. New York: Macmillan Company, 1962.

40. Bongard, M. _Pattern Recognition_. New York: Spartan Books, 1970.

41. Fu, K. S. _Sequential Methods in Pattern Recognition and Learning Machines_. New York: Academic Press, 1968.

42. Yau, S. S. "Forward." _IEEE Conference Record of the Symposium on Feature Extraction and Selection in Pattern Recognition_, Argonne National Laboratory (October 1970).

43. Neely, Peter M. "Comparison of Several Algorithms for Computation of Means, Standard Deviation, and Correlation Coefficients," _Communications of the ACM_, Vol. 9(7), July 1966. 496-499.

VITA

The life of Chiam Geoffrey Goldbogen began on January 3, 1944, in Chicago, Illinois. He achieved his primary and secondary education in Lake Zurich, Illinois. He received a Bachelor of Science degree in Mathematics from the University of Missouri at Rolla in 1965, and went to Adelphi University in New York for graduate study under a National Aeronautics and Space Administration Predoctoral Training Grant. In 1967 he received a Master of Science degree in Mathematics from Adelphi University. While in New York, he also worked as a systems analyst for Lockwood, Kessler, and Bartlett for a brief time before returning to Rolla, Missouri to pursue his graduate studies and teach mathematics as a half-time instructor. During these graduate studies, he has performed pattern recognition research at Argonne National Laboratory as a summer student in residence and currently as an Argonne Universities Association Fellow.

Appendix A

MATHEMATICAL FOUNDATIONS

In this appendix, the ingredients of a classical learning machine are introduced, and the fundamental training theorem is stated. The classical learning machine may use many types of learning processes. An intuitive and geometric introduction of three of these processes — fixed, fractional, and absolute movement — is given. Each of these processes uses a weight vector, $\bar{w}$, and each process requires certain conditions and constraints for the learning to be successful. Since these conditions are not thoroughly developed in this review, the fundamental training theorem is presented in the form of an axiom; that is, without proof and without the conditions of the hypothesis being specifically stated. The interested reader will find Nilsson[6] a source for a more complete development of the fundamental training theorem. After the classical learning machine structure is discussed in Section 1 the corresponding mathematical structure for the adapted learning machine is presented. The learning process in the adapted machine utilizes a model, M, as opposed to the weight vector, $\bar{w}$. This discussion of the adapted machine is incomplete in that the conditions under which the learning processes are successful are not presented. Although the conditions have not yet been formally developed and in fact are not completely understood, the mathematical framework for the adapted machine is presented, and corresponding fundamental training theorems are stated. It is hoped that future research will complete the mathematical structure by supplying the conditions under which the processes succeed in their learning task.

Note that the important terms in this appendix are underscored as they are presented. Also note that the discussion of the adapted machine is divided into a section on local learning and a section on global learning. Before proceeding, the reader should refer to Figure 31 on page 59, which diagrammatically compares the structure of the classical machine and the adapted machine.
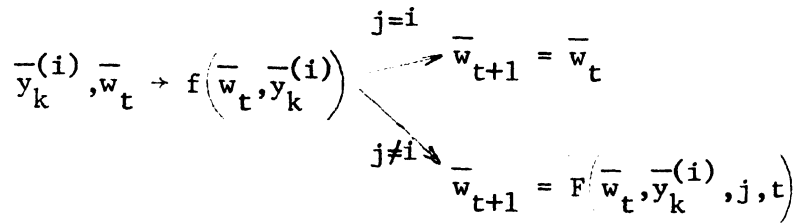
1. The Classical Machine

The classical learning machine as presented by Nilsson[6] has the task of identifying an input as a member of one of R categories. The inputs are elements of a pattern space from which f features are extracted. The vector representation of an element is $\bar{y} = [y_1, y_2, \ldots, y_n]^T$. To train the machine, a training set Y is available. Y is partitioned into subsets $V_1, V_2, \ldots, V_R$, where $Y = V_1 \cup V_2 \cup \ldots \cup V_R$ and $V_i \cap V_j = \phi$ when $i \neq j$. Each V_i is a collection of training patterns,

$$V_i = \left\{ \bar{y}_1^{(i)}, \bar{y}_2^{(i)}, \ldots, \bar{y}_{m_i}^{(i)} \right\} \quad .$$

An element $\bar{y} \in Y$ belongs to V_i if its true label is category i. The training set is finite and has $N_0 = m_1 + m_2 + \ldots + m_R$ training patterns. The machine learns on the training patterns utilizing some learning process and some weight vector, $\bar{w}_t$, which (as a result of the learning process) changes with time. $\bar{w}_t$ is an element of a vector space W. The learning process consists of two phases, evaluation and learning. At time t an input test pattern, $\bar{y}_k^{(i)}$ is presented to the learning machine for evaluation, denoted by $f(\bar{y}_k^{(i)}, \bar{w}_t) = j$. Where f is a function mapping $W \times Y \rightarrow \{1, 2, \ldots, R\}$, j is the machine's choice of category for $\bar{y}_k^{(i)}$. When $j = i$, the machine has diagnosed $\bar{y}_k^{(i)}$ correctly; however, when $j \neq i$, the learning process changes $\bar{w}_t$ to a new weight vector,

$$w_{t+1} = F(\overline{w}_t, \overline{y}_k^{(i)}, j, t).$$ Diagrammatically:

$$\overline{y}_k^{(i)}, \overline{w}_t \rightarrow f\left(\overline{w}_t, \overline{y}_k^{(i)}\right) \begin{cases} j=i & \overline{w}_{t+1} = \overline{w}_t \\ \\ j \neq i & \overline{w}_{t+1} = F\left(\overline{w}_t, \overline{y}_k^{(i)}, j, t\right) \end{cases}$$

When $j = i$, $\overline{w}_t$ _satisfies_ $\overline{y}_k^{(i)}$. It is convenient to think of the transition of $\overline{w}_t$ to $\overline{w}_{t+1}$ as movement in the W space. If a $\overline{w} \in W$ exist which satisfies all $y \in Y$, it is a _solution_. The set of all such $\overline{w}$'s is the solution set, $S \subseteq W$.

Geometrically, the movement of the weight vector in W is of three types. In fixed (-error-correction) movement, $\left|\overline{w}_{t+1} - \overline{w}_t\right| = \left|F\left(\overline{w}_t, \overline{y}_k^{(i)}, j, t\right)\right| = d$. That is, the distance moved in the W space is a constant. In fractional (-error-correction) movement, $\left|\overline{w}_{t+1} - \overline{w}_t\right| = d(t)$; that is, the distance moved in the W space depends on time. These two types of movement are diagrammed in Figures 52 and 53. In absolute (-error-correction) movement, $\overline{w}_t$ moves a sufficient distance so that $\overline{w}_{t+1}$ satisfies the input test pattern, $\overline{y}_k^{(i)}$. Denote the set of $\overline{w}$ which satisfies the test pattern $y_k^{(i)}$ by $h_k^{(i)}$. Then the learning process at time, t, with input pattern, $y_k^{(i)}$, and $\overline{w}_t$, insures that $\overline{w}_{t+1} \in h_k^{(i)}$. In Figure 54, absolute movement is diagrammed for three training patterns $\overline{y}_1$, $\overline{y}_2$, $\overline{y}_3$ applied in the sequence $\overline{y}_1, \overline{y}_3, \overline{y}_1, \overline{y}_2, \overline{y}_1$, showing the movement of $\overline{w}$ into h_1, h_3, h_1, h_2, h_1.

Let $C_y' = \overline{y}_1', \overline{y}_2', \overline{y}_3', \ldots$ be a sequence of elements of Y such that every element in Y occurs infinitely often in C_y'. C_y' is a _complete training sequence_.

To begin the learning process, an initial weight vector $\overline{w}_0$, is selected. Then apply C_y' and form a new sequence $C_y = y_1, y_2, y_3, \ldots$ by
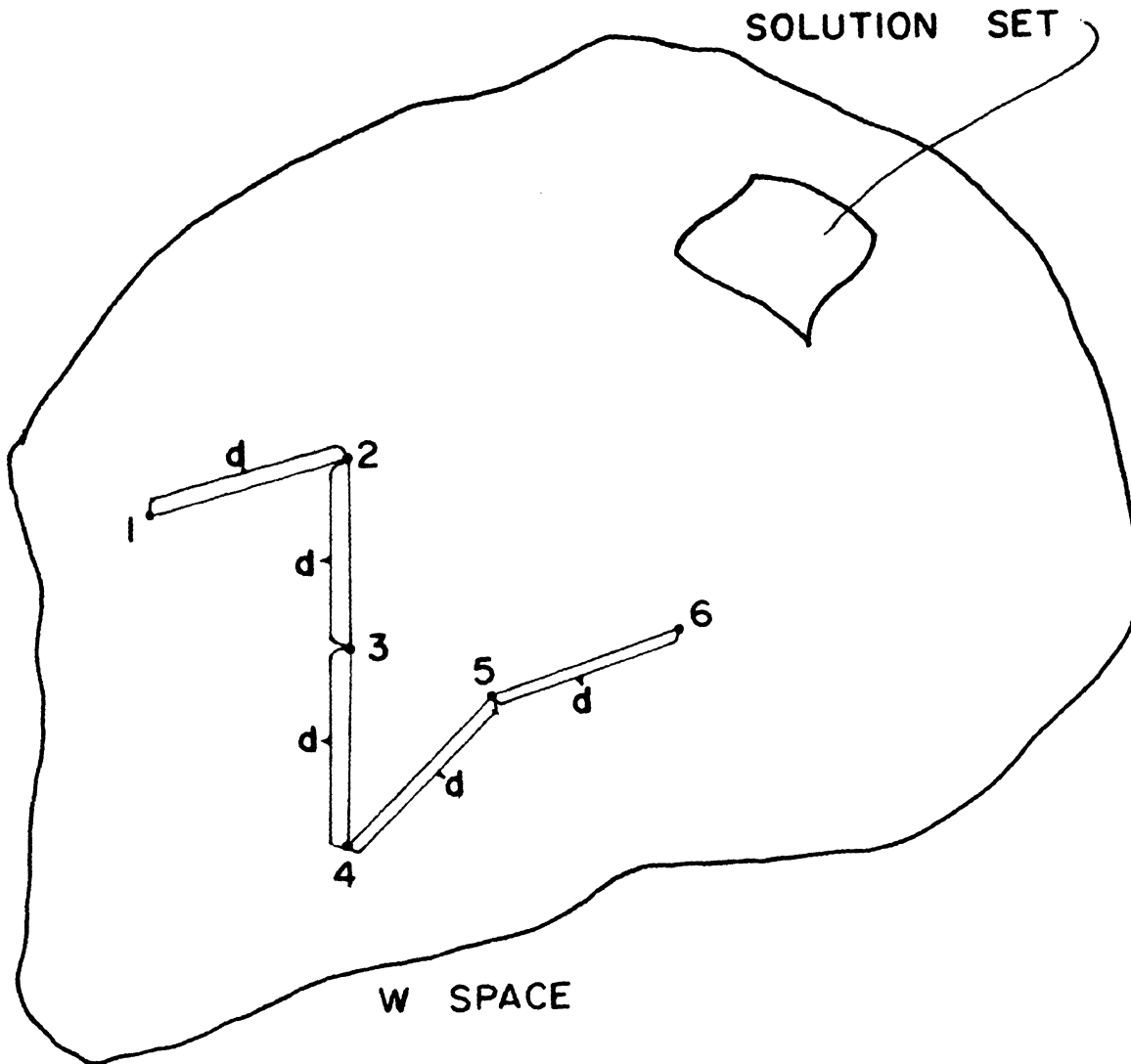
Figure 52. Fixed Movement.

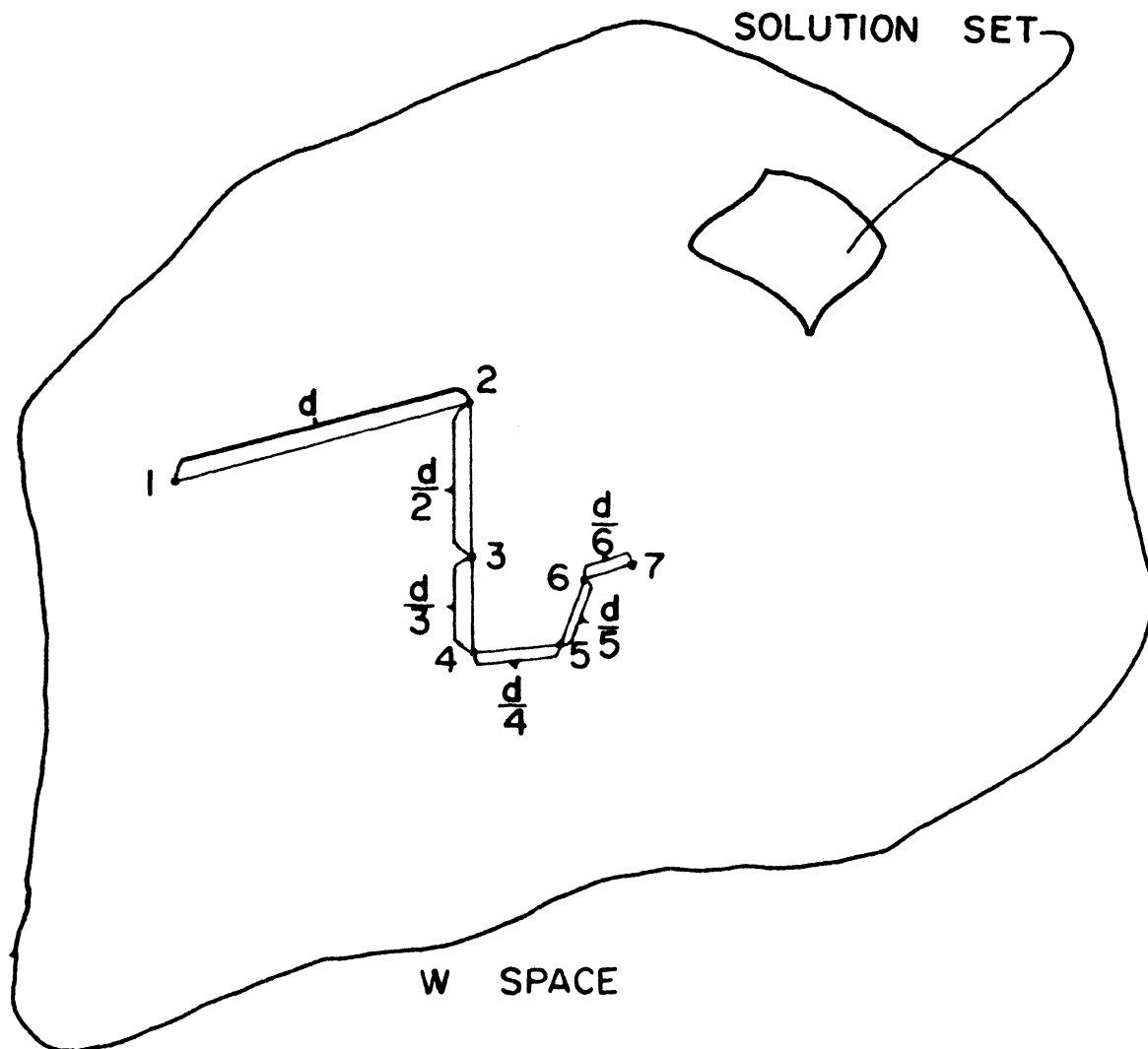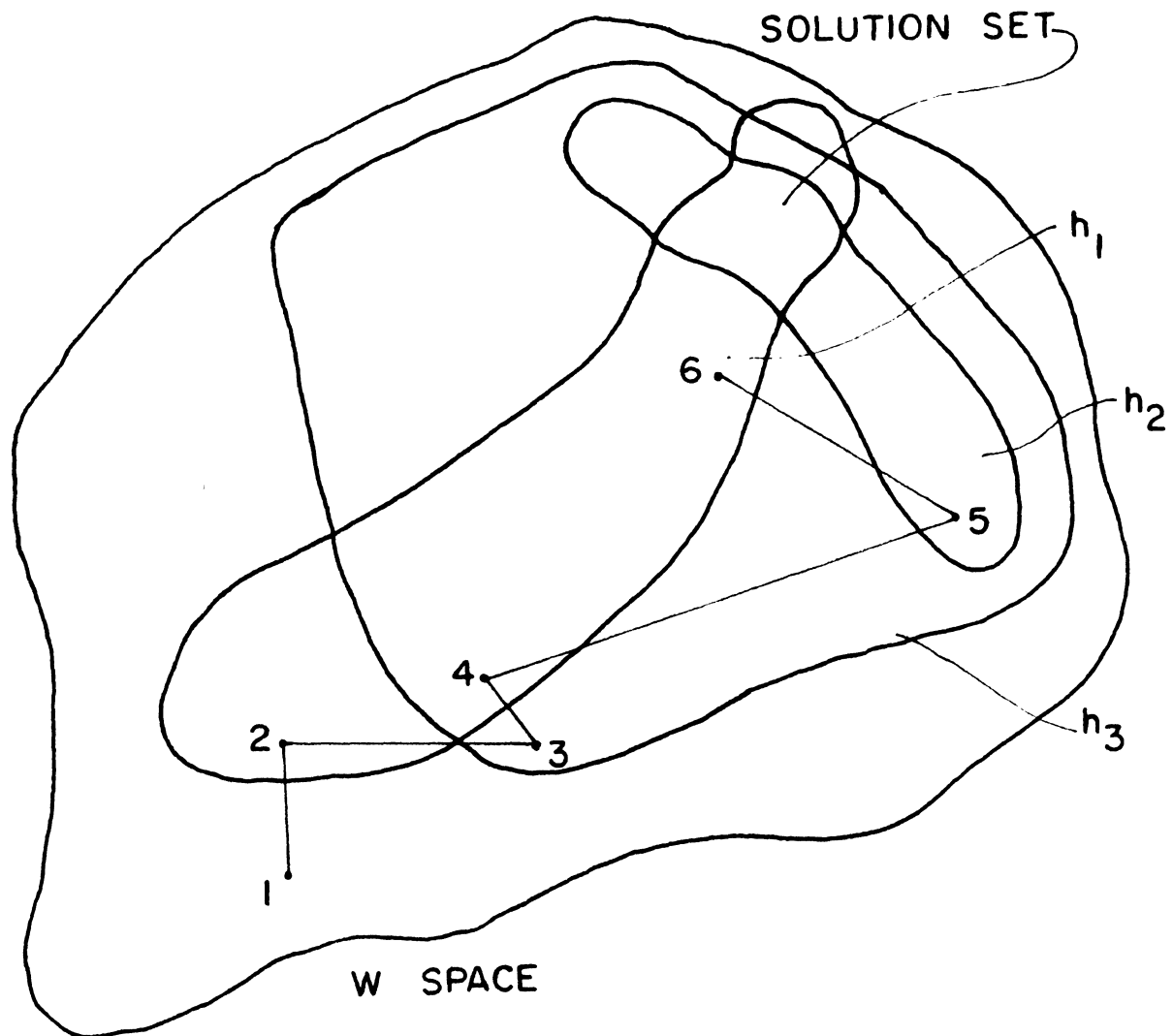Figure 53.   Fractional Movement.

Figure 54.   Absolute Movement.

deleting those elements of C_y' which do not move the weight vector $\bar{w}_t$.

C_y is the (reduced) training sequence and $W_c = \bar{w}_0, \bar{w}_1, \ldots$ is the corresponding reduced weight sequence. When $\bar{w}$ moves into a point in the solution set, S, application of any element of Y will not move $\bar{w}$ and hence the learning process halts. The task of the classical learning machine is to take an initial $\bar{w}_0$ and use the training set Y to move $\bar{w}$ into the solution set. When certain conditions are satisfied, the classical learning machine behaves so as to satisfy what Nilsson calls the **general fundamental training** theorem.

Let L be a learning machine, using $\begin{Bmatrix} fixed \\ fractional \\ absolute \end{Bmatrix}$ movement in its learning process, whose task is to identify patterns of R types. Let Y be a set of training patterns, and suppose a solution set exists for Y. Then given any initial weight vector, $\bar{w}_0$, and applying any complete training sequence C_y will cause $\bar{w}$ to move to a point in the solution set in a finite number of steps.

2. The Adapted Machine (Local Learning).

In Chapter III, the model was presented as a sequence of 252 lists. Then in Chapter V, the model was partitioned into P submodels, M_i, $i=1,2,\ldots,P$, where P is the number of c-sects. Recalling that local learning involves one c-sect at a time, we facilitate the discussion here by fixing i and calling the **submodel**, M_i, simply "the model."

The **model** M_i is concerned with N_i elements, corresponding to the fiber bundles that intersect c-sect i. Each element has an x and y coordinate corresponding to the lattice point intersected by the element. Thus M_i may be described by $2N_i$ discrete variables, $x_1, y_1, x_2, y_2, \ldots, x_{N_i}, y_{N_i}$ where the range of each variable z is $1 \leq z \leq 96$

as determined by the lattice size 96 x 96. Hence, M_i may be considered a point in $2N_i$-dimensional __model space__.

Similarly, the training set T can be partitioned into P subsets, the subset T_i consisting of training patterns that affect c-sect i.

The local learning __task__ of the adapted machine is to arrange the N_i elements within the c-sect so that the model M_i satisfies the training patterns in T_i. A training pattern $t_j \varepsilon T_i$ consists of a visual field D_j, a c-sect $C_j = i$, a center (α_j, β_j) and a radius of damage extent r_j. D_j is processed to assign to each of the N_i elements the value of one, zero, or x ('don't care'), depending on whether the fiber bundles are affected, unaffected, or outside normal vision. Therefore, the processed D_j is represented by the two lists $\Theta_j = (\Theta_1^j, \Theta_2^j, \ldots, \Theta_{q_j}^j)$, $Z_j = (z_1^j, z_2^j, \ldots, z_{\ell_j}^j)$ omitting the list of 'don't care' fiber bundles which consists of the $N_i - q_j - \ell_j$ bundles not listed in Θ_j or Z_j. The training pattern t_j has the following information for the local learning $t_j = \{\Theta_j, Z_j, (\alpha_j, \beta_j), r_j\}$. M_i __satisfies__ t_j if: 1) elements of Θ_j are within a distance of r_j from (α_j, β_j); 2) elements of Z_j are at least a distance of $f_j = r_j$*FRAC from (α_j, β_j).

When M_i does not satisfy t_j, the elements that violate (1) are in the set $\hat{\Theta}_j$, while those that violate (2) are in set $\hat{Z}_j$. The application of M_i to t_j and the result is denoted by:

$$f(M_i, t_j) = \begin{cases} \phi & \text{when } M_i \text{ satisfies } t_j \\ (\hat{\Theta}_j, \hat{Z}_j) & \text{when } M_i \text{ does not satisfy } t_j. \end{cases}$$

In the learning process, each element in $\hat{\Theta}_j$ is moved toward (α_j, β_j) while the elements listed in $\hat{Z}_j$ are moved away from (α_j, β_j). Considering the movement of these elements one at a time, a movement

of an element a distance d in the c-sect causes the model M_i to move distance d through the model space. These elements (and hence the model) may move in three modes as the weights do in the classical system.

In the adapted system, a <u>complete</u> <u>training</u> <u>sequence</u> C_i is a sequence of patterns from T_i such that every element of T_i appears infinitely often in C_i. When applying C_i, the model M_i moves through the model space every time a pattern is encountered which the model does not satisfy. The (reduced) <u>model</u> <u>sequence</u> is a record of the path traversed by the model M_i from the initial point M_i^0 forward. $[M_i^0, M_i^1, M_i^2, \ldots,]$. The set of points in the model space that simultaneously satisfy every member of T_i is the <u>solution</u> <u>set</u> S_i, and is a subset of the model space. Well-behaved means that the exit β in the movement algorithm was never used (see Figure 62).

The <u>general</u> <u>fundamental</u> <u>training</u> theorem for a local-learning model-building system:

Let M_i be a submodel partitioned from M and let T_i be the corresponding partition of the training set T. If a solution set S_i exists and if, when applying a complete training set C_i, the system is well-behaved, then given any starting configuration M_i^0, the model travels through the model space to a point

in S_i for a learning machine using fixed absolute fractional movement.

Corollary.

Let M *be a model partitioned into submodels* $M_1, M_2, \ldots, M_p$ *and let* T *be a training set partitioned into* $T_1, T_2, \ldots, T_p$. *Then if each* M_i *converges to a point in* S_i *(defined by* T_i*) then the model* M *converges to a point in* S *defined by* T.

The total model space in which M is found is the concatenation of $M_1, M_2, \ldots,$ and M_p. S is a subset of M formed from the concatenation of $S_1, S_2, \ldots, S_p$. The corollary follows immediately from the fundamental theorem.

In practice, it seems that the given set of patterns T_i has inconsistencies and application of a <u>complete</u> <u>training</u> <u>sequence</u> C_i results in oscillation of the fibers as shown in Figures 37, 38, and 39. Using a ping-pong strategy of applying a finite segment of C_i and then revising the radii according to the oscillation appears to eliminate the inconsistencies in T_i and allows M_i to have a solution set to which it can converge.

3. The Adapted Machine (Global Learning)

The local learning in the c-sects has a profound effect on the ability of the machine to perform diagnosis using the diagnosis algorithm detailed in Appendix B. The results in Chapter VI show that local learning improves the diagnosis percentage. In this regard, the training set T is said to be <u>sufficient</u> if, after the set is partitioned and local learning converges for each c-sect, the diagnosis of each member of T is performed correctly. It is hoped that a large <u>sufficient</u> training set applied to the model contains enough information for the machine to formulate a model capable of diagnosing most patterns.

Appendix B

TWO HEURISTICS

1. <u>Radii Selection</u>

Function - The radii selection algorithm selects and changes a radius use in a pattern sequence. The radius selection will most probably contribute greatly to alleviation of an inconsistency condition. Data from the last learning cycle has been retained for each pattern. MOVI(I) and MOVO(I) is the number of elements of $\hat{\Theta}_i$ and $\hat{Z}_i$ (see Appendix A), respectively If some patterns have either $\hat{\Theta}_i$ or $\hat{Z}_i$ equal to zero but not both, then among these patterns, I is selected which has the maximum MOVI(I) or MOVO(I). If no $\hat{\Theta}_i$ or $\hat{Z}_i$ is null, the greatest $\frac{MOVI(I)}{MOVO(I)}$ ratios determine I. For clarity, the flowchart of this process is included as Figure 55.

2. <u>Diagnosis</u>

Function - Select c-sect and determine center and radius of damaged area. Select preference is to c-sect with approximately highest count and best clustering. Approximate highest count is used since c-sect 5 always has highest count because all nerve bundles pass through it, and because of looping some affected fibers may nearly miss the affected c-sect (see Figure 56).
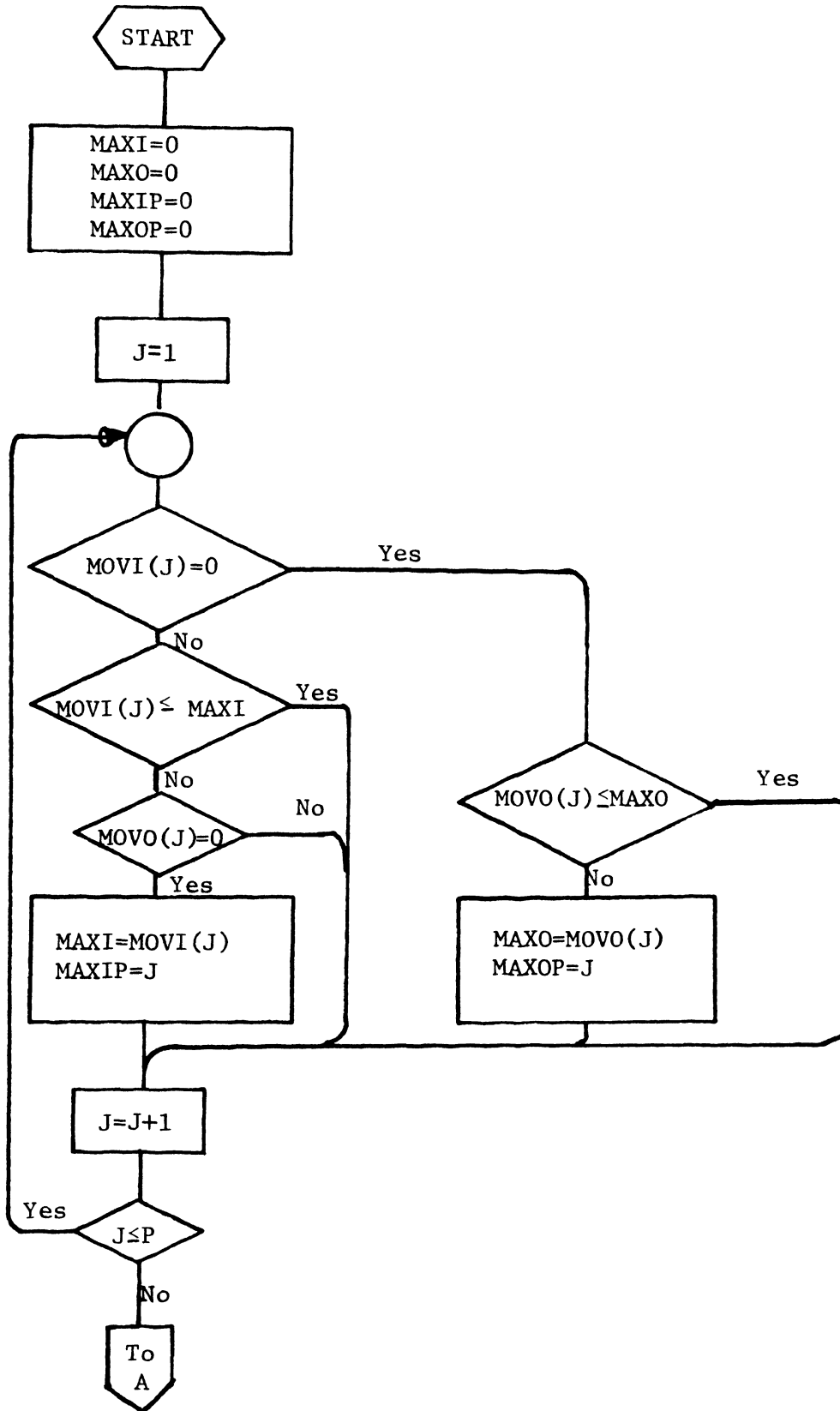
Figure 55. Select and Change Radius.
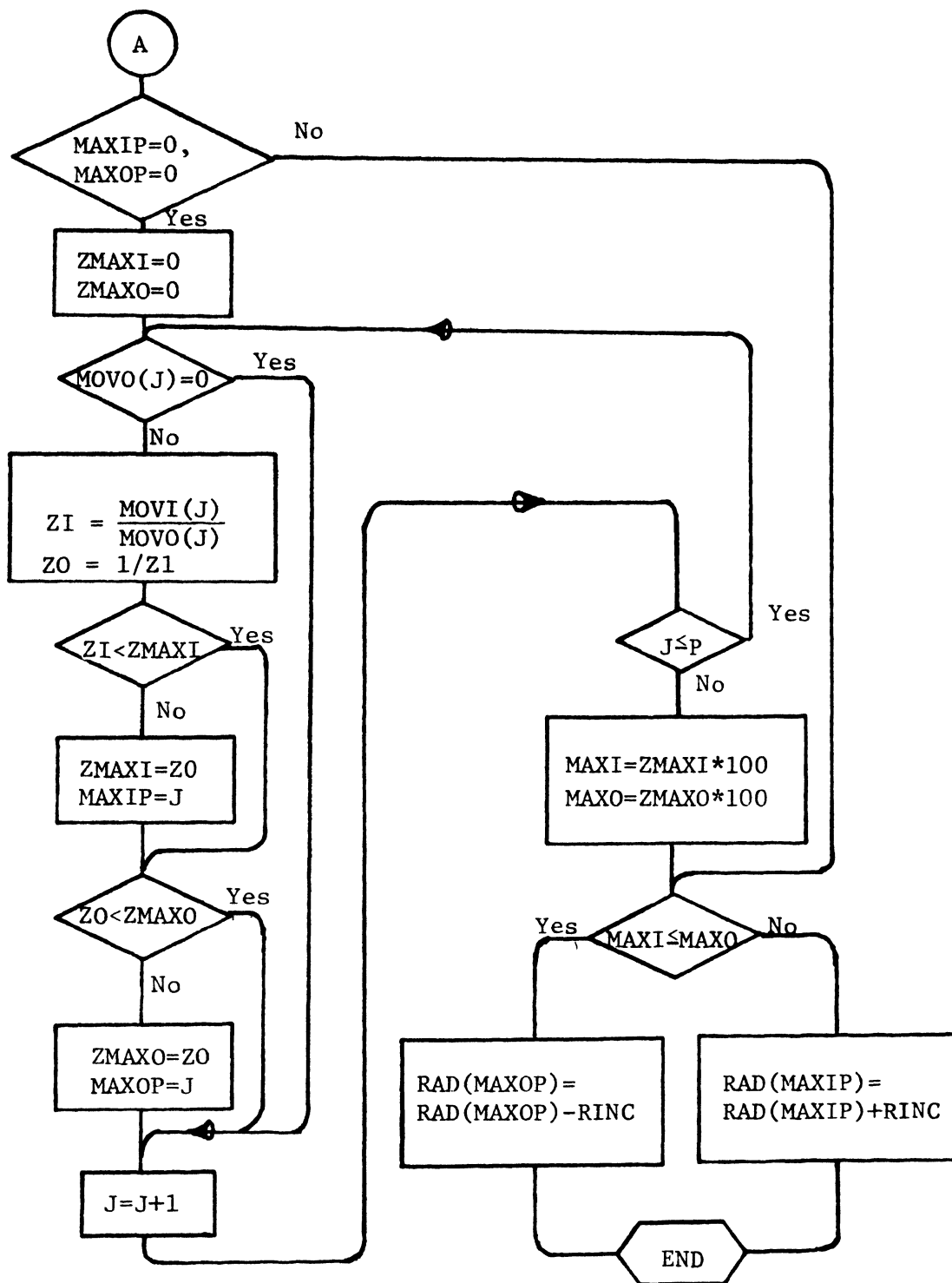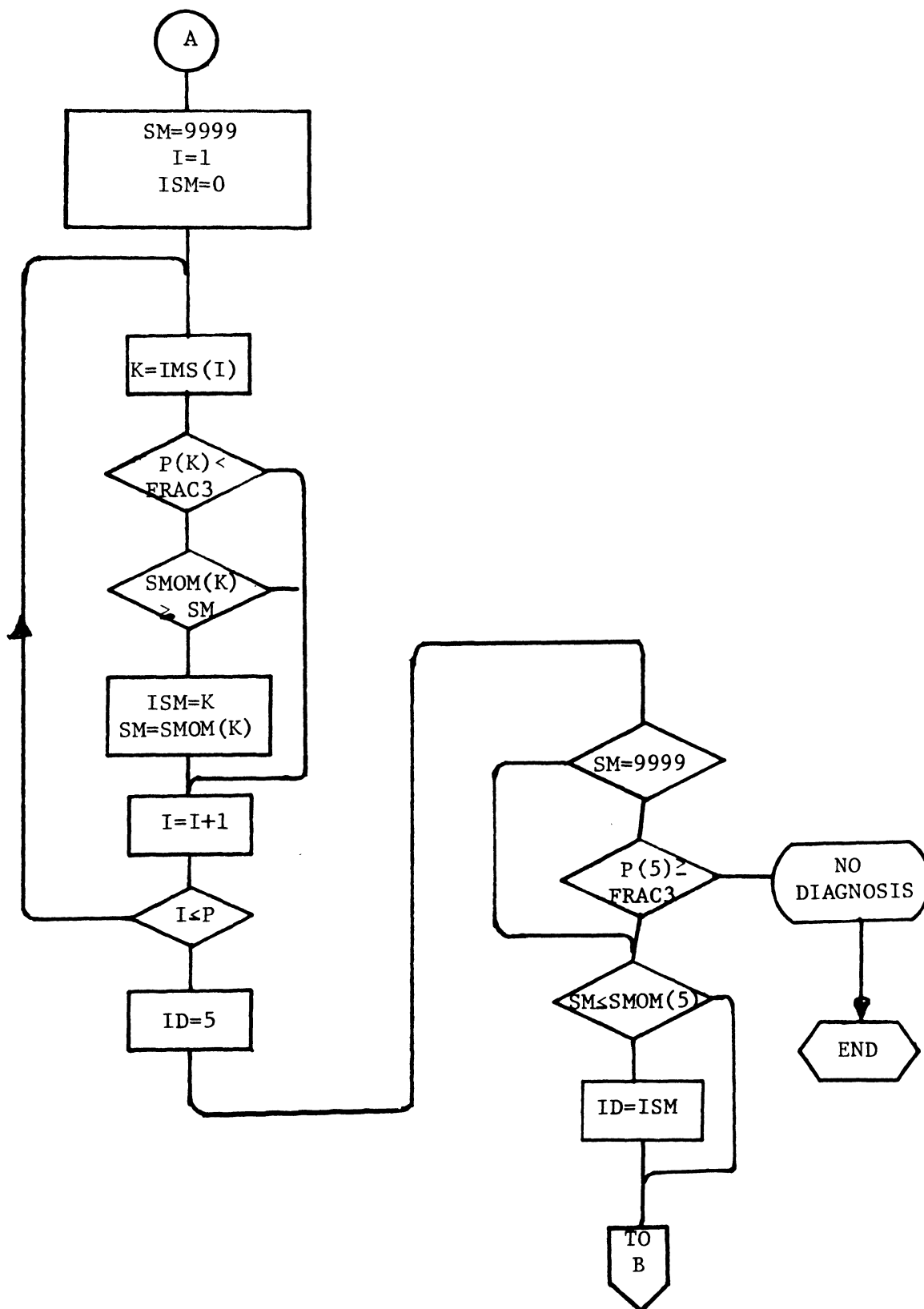
Figure 55 (contd.). Select and Change Radius.
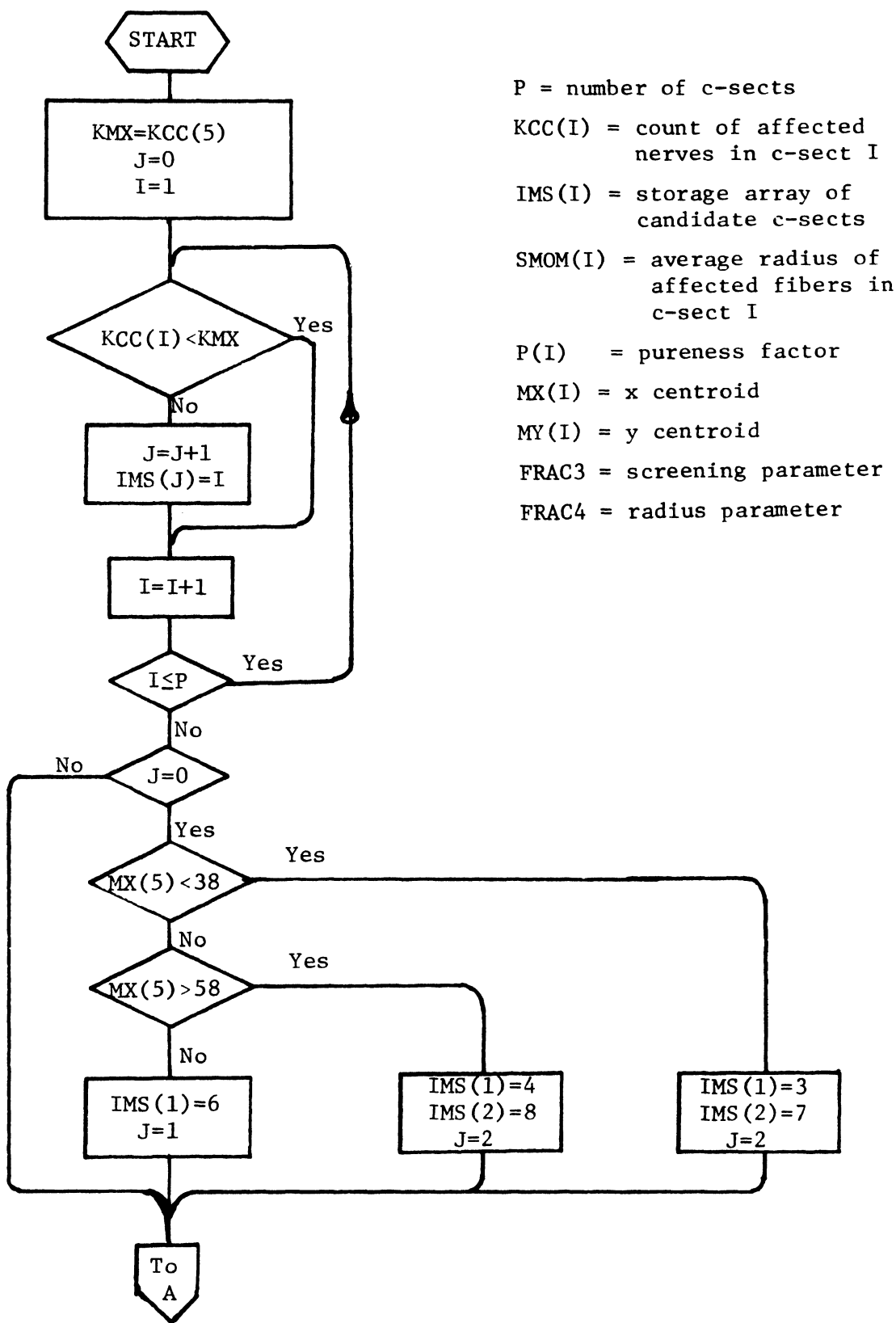
Figure 56. Diagnosis Algorithm.

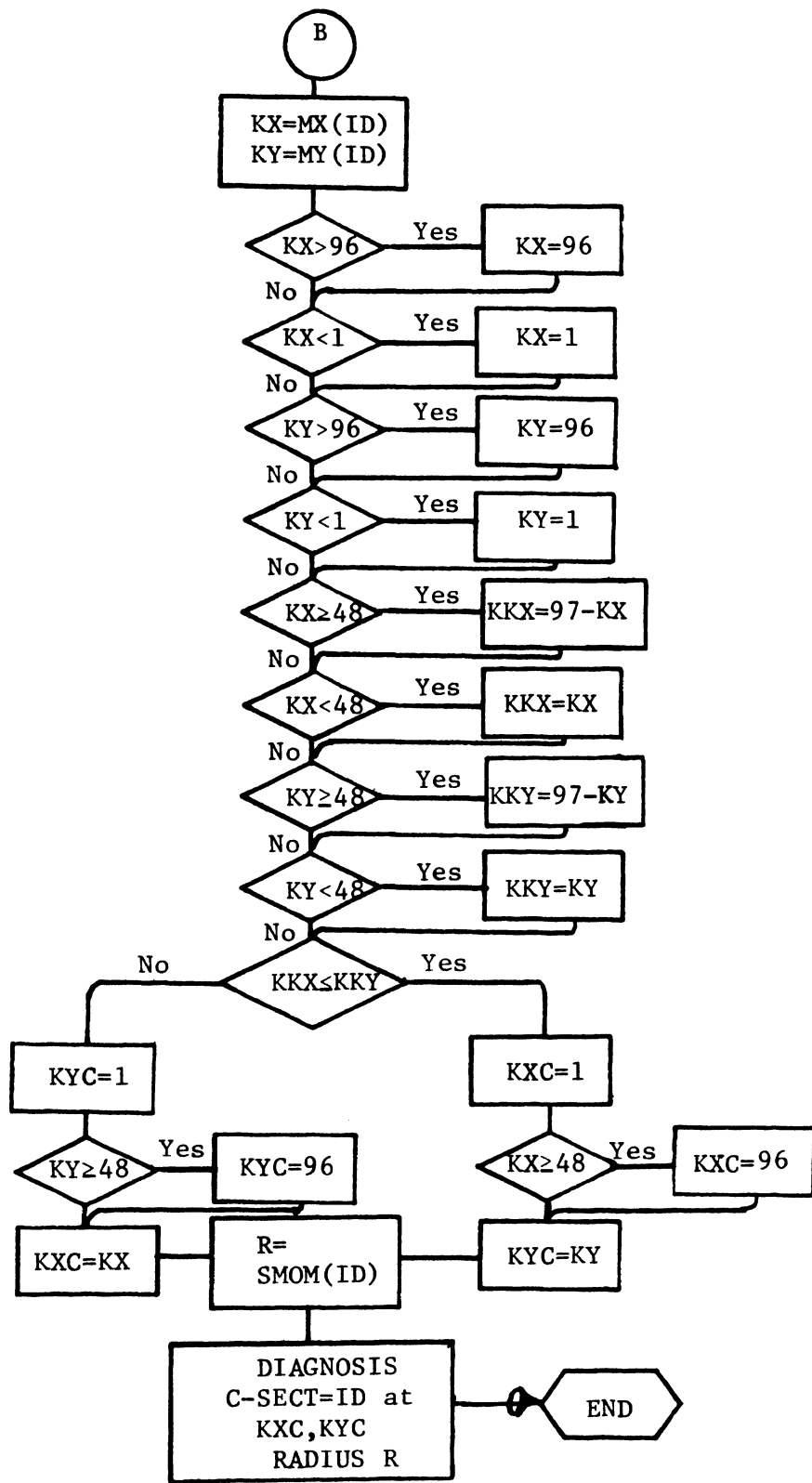Figure 56 (contd.). Diagnosis Algorithm.

Figure 56 (contd.). Diagnosis Algorithm.

Appendix C

OTHER APPROACHES TO LEARNING

1. Moments

Initially, a method was proposed using moments of fiber bundles

within a c-sect relative to the centroid of the fibers. Calculation of

the centroid involved first x and y moments. In this method the faulty

sense points are set and the fiber bundles sensitized, and as before,

this results in a fault cluster of 1's. When the centroid of the 1's

is taken, the distance between the centroid to the (x,y) coordinate of

the center given in the test pattern data is calculated. If this dis-

tance is greater than a threshold, a fixed increment moving process is

called. Figure 57 illustrates this process.

To some extent, this method worked fairly well, but it had several

disadvantages. The threshold value is difficult to estimate, hence the

learning process may stop too soon or not soon enough. It is possible

to have the centroid within the threshold distance of the center and

yet have some fiber bundles far away from the main cluster. No provi-

sion is made to move unsensitized bundles away from the center. Move-

ment tends to draw too many points near the center. This latter dis-

advantage can be corrected by moving fiber bundles in a fractional

increment mode rather than fixed increment mode. For example, the

incremental distance moved by the fiber could be made proportional to

the distance between the fiber and the center.
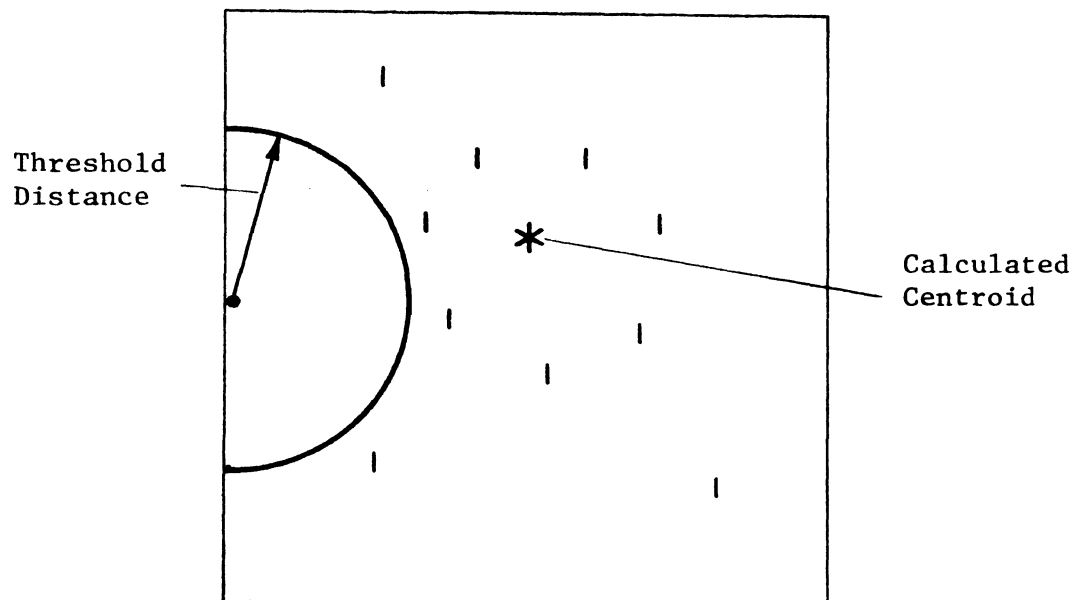
(A)
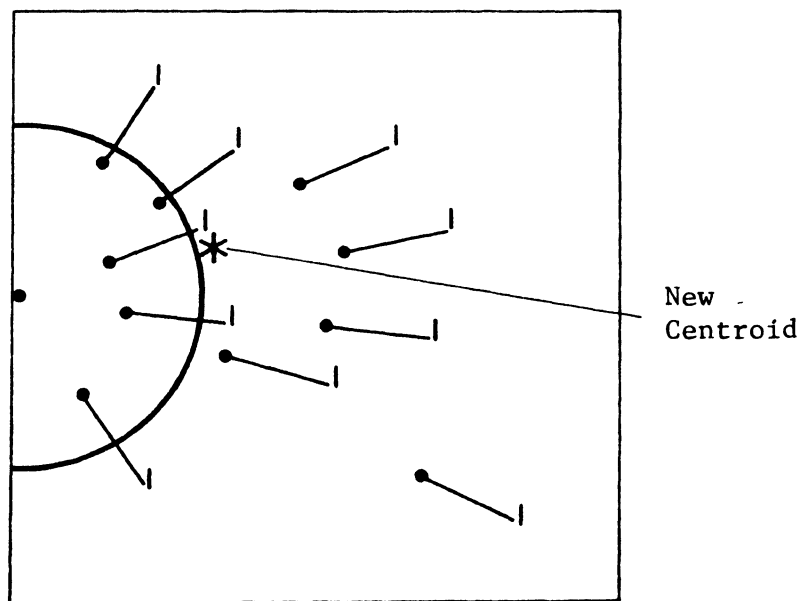
(B)

Figure 57. Moment Learning.

2. Interpolation

A form of interpolation may be used to supplement the arrangement of fiber bundles in a c-sect where not many patterns are available for learning. If this c-sect lies between two c-sects which have experienced much learning, the position of the fiber bundles in these c-sects can be used to influence the position of the c-sect in question. Again, a diagram aids the discussion.
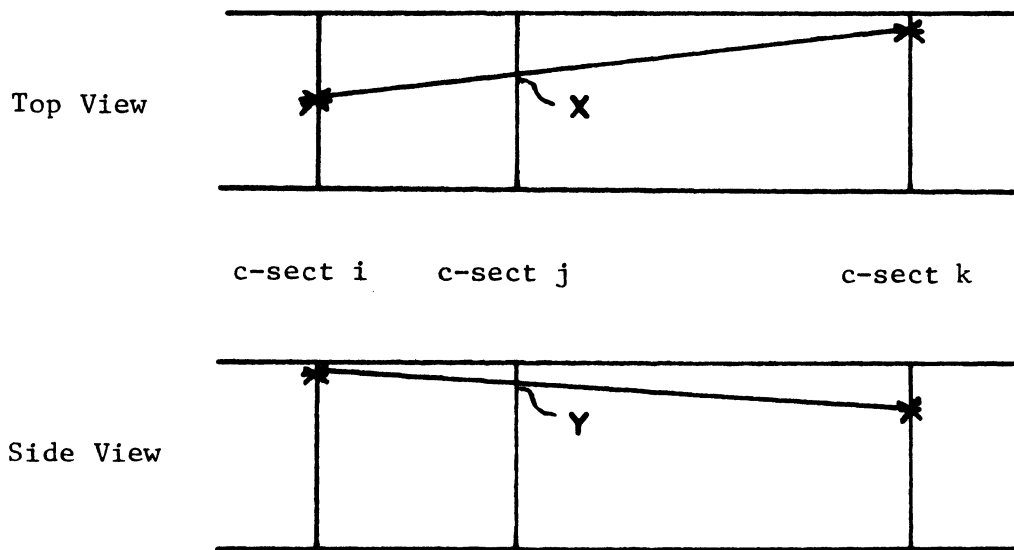
Top View

c-sect i   c-sect j   c-sect k

Side View

Figure 58.   Interpolation.

A fiber bundle passes through c-sect i, j, and k. The position of the fiber bundle in c-sect i and k has been established through local learning, and is indicated by *'s. Linear interpolation could be used to establish the (x,y) coordinates in c-sect j.

More elaborate interpolation systems can be envisioned. Higher order interpolation can be used if the fiber position in more c-sects is considered in the interpolation. Technical problems arise in actual implementation of this algorithm. For example, interpolating for c-sect 3 using c-sect 1 and 5 presents a problem, since in the top view
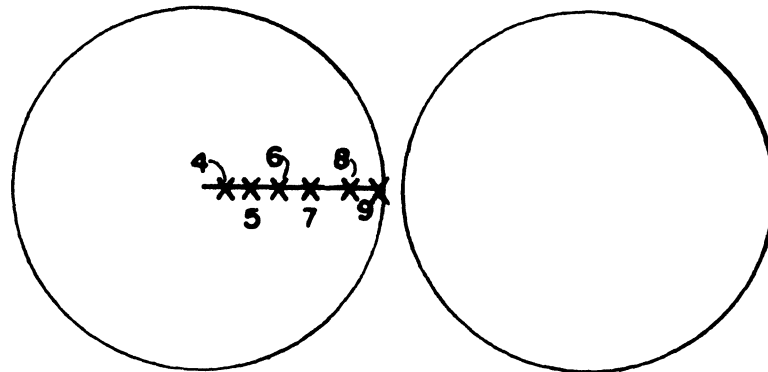
(see Figure 11) linear interpolation could result in selecting a point off of c-sect 3. To see this, connect the upper edge of c-sect 1 and c-sect 5 in Figure 11 with a straight line.

3. Reordering Curves

Another form of learning which has been considered but not implemented or tested is called reordering curves. The basic assumption for this heuristic is that the sequence of the sense points on a ray carries over to the new positions of the fibers in the c-sects. To illustrate this, consider the sequence of sense points 4, 5, 6, 7, 8, 9 on ray 1 of the left retina. As the fibers pass through the c-sects, it is probable that along some simple curve in the c-sect connecting the fiber bundles, the sequence is preserved. This assumption leads to a reordering phase in the learning process. Figure 59(A) shows the arrangement of the sense points 4, 5, 6, 7, 8, 9 in the left retina. Figure 59(B) shows the corresponding fiber locations in c-sect 1 after a hypothetical local learning process. A simple (second degree) curve approximately fitting these points is drawn. Notice that fibers 7 and 8 are "out of sequence." This would be repaired by reordering the fibers in locations along the curve as shown in Figure 59(C).
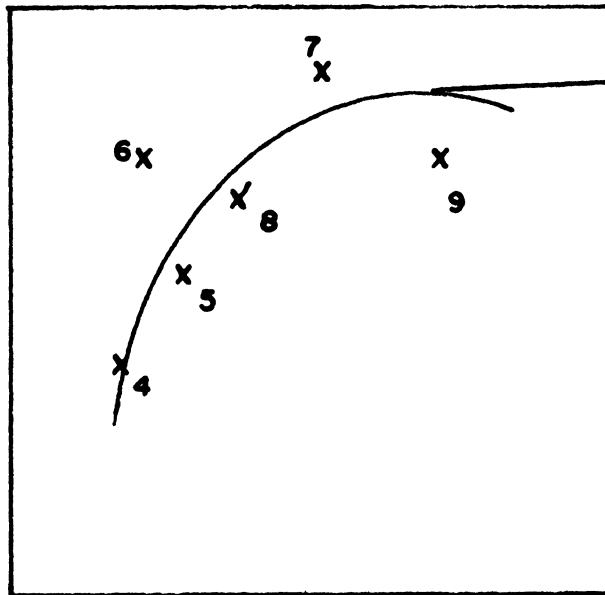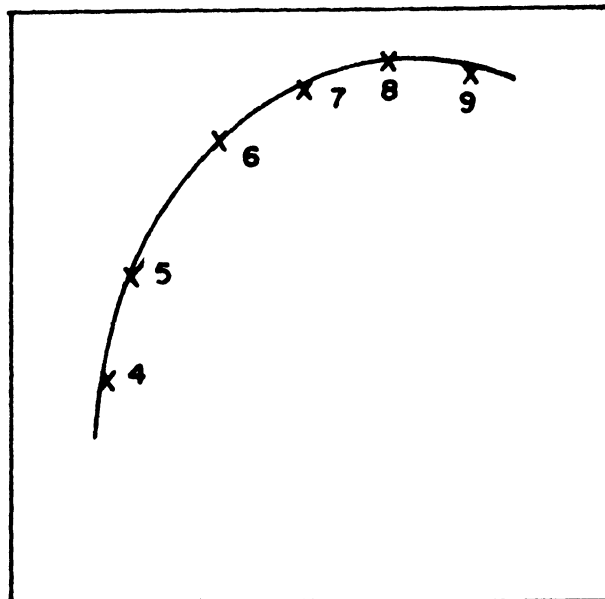
Figure 59. Reordering.

4. Directional Learning

This type of learning is similar to the local incremental learning discussed in Chapter V. However, it represents an immediate feedback of correction for the feedback loop of Figure 31(B). The learning process proceeds in this manner: In diagnostic mode, a pattern is simulated and diagnosed. If the diagnosis is incorrect, the user inputs the corrected diagnosis. Immediately the system performs directional learning by moving the sensitized fibers toward the center of the correction and moving the unsensitized fibers away from the center. In this process no limiting radii are used, and hence the process suffers from the same "overlearning" problem that the moment-generated learning has.

5. Addition Learning

The present model seems adequate enough to make this type of learning unnecessary. This is because the *a priori* knowledge of the model contained sufficient information concerning through which c-sects the different fiber bundles pass (although exactly where is not known). Until this degree of knowledge proved sufficient, it was necessary to consider this type of learning.

Consider a lesion in c-sect 8. Fiber bundle 171 passes through c-sects 2, 4, 5, 6, 7, 9, but is affected by the lesion. Therefore, a rethreading of the fiber bundle to pass through c-sect 8 is performed. This is the addition process and results in a change in the list structure. Figure 60 illustrates this.
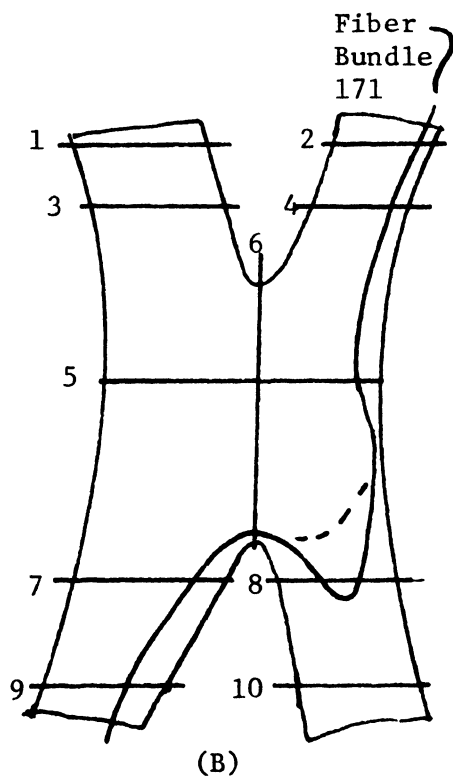
Fiber Bundle 171

Before Addition Learning

(A)
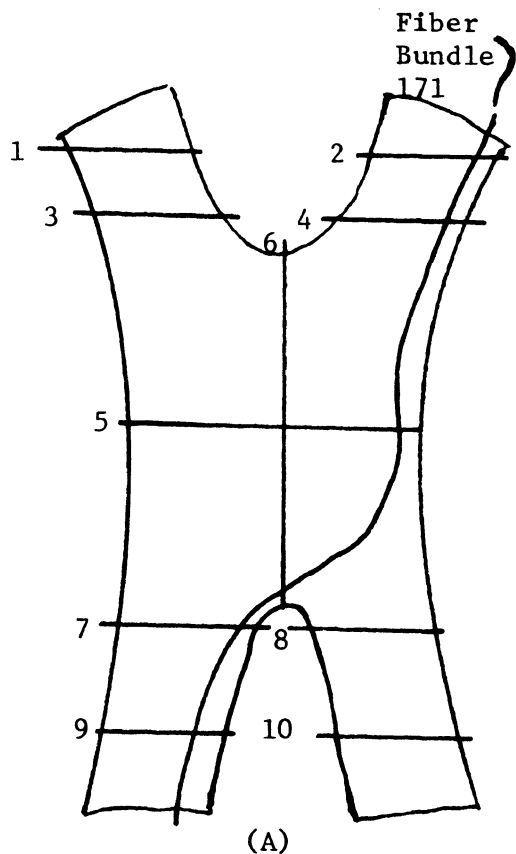
List Structure

-2	5	9
2	x_2	y_2
4	x_4	y_4
5	x_5	y_5
6	x_6	y_6
7	x_7	y_7
7	x_9	y_9
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

Fiber Bundle 171

After Addition Learning

(B)

List Structure

-2	5	9
2	x_2	y_2
4	x_4	y_4
5	x_5	y_5
8	x_8	y_8
8	x_8'	y_8'
6	x_6	y_6
7	x_7	y_7
9	x_9	y_9
0	0	0
0	0	0
0	0	0

Figure 60. Addition Learning

6.   Symmetry

Because of the symmetry of the visual pathway, the local learning process can be extended through use of symmetry. There are three types of c-sects in regard to symmetry. (Refer to Figures 11, 12, and 14.) Consider a mid-line drawn in Figure 11. The anatomy of the visual pathway can be considered symmetric to this line for our application. C-sects are of three types with respect to this mid-line. Type I c-sects have a companion c-sect symmetric to the line. This type includes c-sects 1, 2, 3, 4, 7, 8, 9, 10. Type II c-sects straddle the mid-line. C-sect 5 is the only type II c-sect. Type III c-sects (c-sect 6) lie on the mid-line. To understand how symmetry is used, it is necessary to first define symmetric sense points. A pair of sense points are symmetric sense points if one is in the right retina and one is in the left retina, they are both on the same circle, and they are symmetric with respect to a vertical line drawn through the center of each retina. Figure 61(A) shows two symmetric pairs of points: sense points A and B, and sense points C and D.

When learning has occurred in a type I c-sect, the new positions of the fibers in the c-sect are noted. Then the sense point and hence the fiber bundle corresponding to the other member of the sense point pair is found. This partner is moved to a point in the companion c-sect which is opposite (relative to the mid-line) the location in the original c-sect. Inspection of Figure 61(B) will clarify this concept. C-sects 1 and 2 are companion c-sects. Points A and B, and points C and D are symmetric pairs. Therefore, the location of B in c-sect 2 is opposite A in c-sect 1.

When learning has occurred in a type II c-sect, the symmetry is basically the same except the partner of a moved fiber is moved to a point opposite relative to the center line in the same c-sect. Figure 61(C) illustrates this.

Type III c-sect presents a conflict between the structures of the model and the principle of symmetry. For symmetry to be precise, fiber of symmetric pairs that intersect with this type of c-sect should occupy the same point in the c-sect. This would violate the construction principle that one fiber only occupy a lattice point in a c-sect. To circumvent this conflict, symmetric pairs are located in adjacent lattice points, the fiber from the left retina positioned above the fiber from the right. Figure 61(D) illustrates this concept.
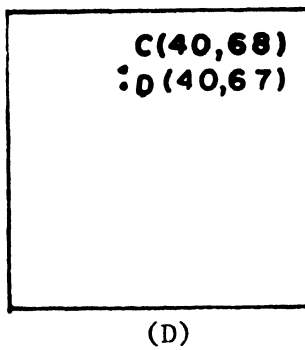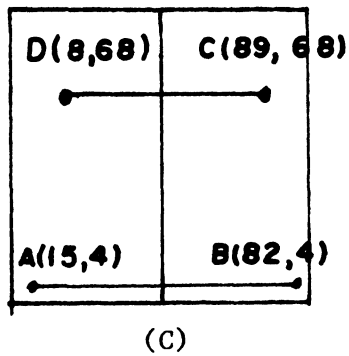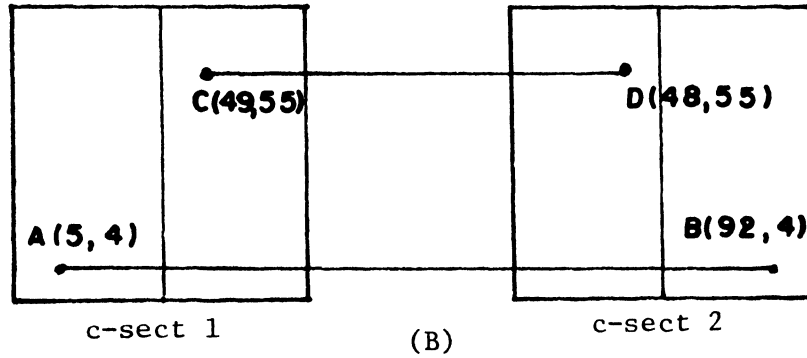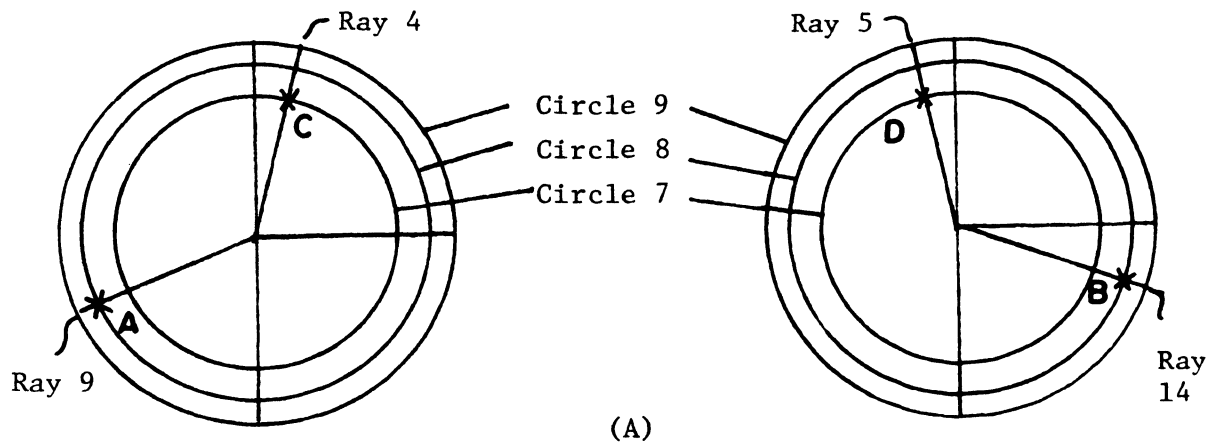
Figure 61. Symmetry.

Appendix D

MOVEMENT

The movement of fibers in the learning process depends on four factors. The starting location (a,b), the distance moved, d, the direction moved, and the location of the other fibers in the c-sect. The distance moved depends on the mode of movement. In fixed increment movement, the distance is always a constant. In fractional increment movement, the distance depends on some measure of time. In absolute movement, the distance is sufficiently great to move the fiber bundle into the desired region. Once this distance d is determined, the algorithm seeks a vacant lattice point a distance d from the starting location (a,b) toward (or away) from the center (α,β). When occupied lattice points are encountered, the search for unoccupied points continues in approximately the same direction at very small increments. This process for fixed or fractional movement is flowcharted in Figure 62.
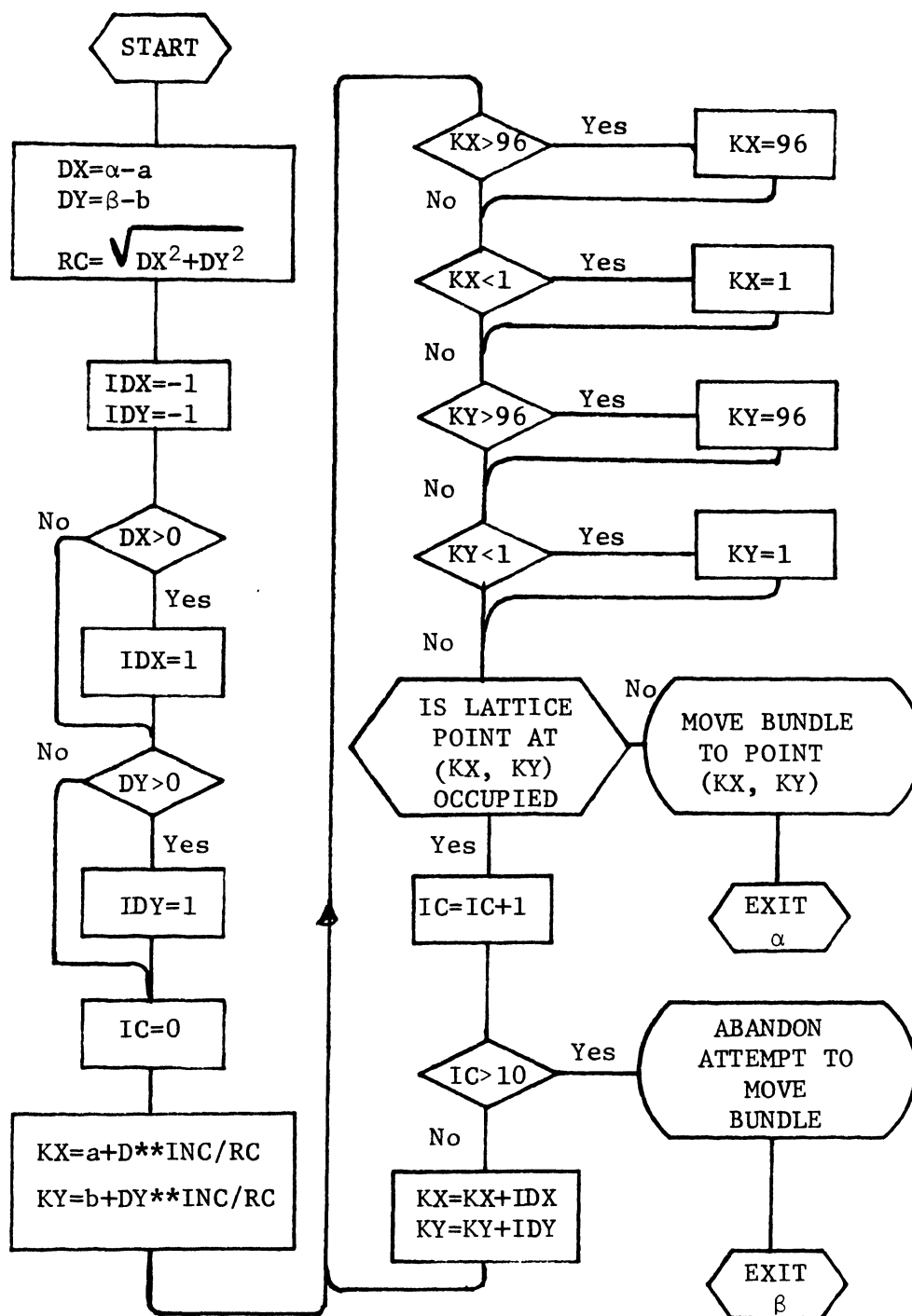
Figure 62. Movement of Fibers.