Description and Classification of Neuronal Structure in the Frog Retina

thesis by

Michael Joe Shantz

In Partial Fulfillment of the Requirements
for the degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California

1976
(submitted November 20, 1975)

DEDICATION

To all those frogs who sacrificed their eyes to science, my humble thanks.

ACKNOWLEDGEMENTS

I am very grateful to my advisor Professor G. D. McCann for his continual encouragement and insight throughout my graduate years. I am grateful to Ken-Ichi Naka for suggesting the frog retina and for foresight and guidance in all aspects of the work.

I thank Thomas Ogden and Bill Stell for advice on the Golgi method and on the general problem of morphological analysis. I thank Derek Fender, Nick George, and Ken Castleman for helpful advice on optics and image processing.

I thank Per Brinch Hansen for enlightening discussions on the class concept and on the advantages of high level, well structured programs as conceptual tools in biology.

I thank Mona Hsu for preparing many of the camera lucida drawings, and Nancy Carroway for help in the Golgi work as well as manuscript preparation.

I thank Dale Knutsen, Jerry Pabst, Howard Frieden and Greg Jirak for programming assistance. I thank Ben Ellert, Al Broers, and Dan Aranovich for technical assistance.

I especially thank Vera Tesluk Shantz for help in manuscript preparation and for keeping me in partial touch with reality.

I am very grateful to the California Institute of Technology for providing financial aid throughout my

graduate years.

Table of Contents

ABSTRACT

The intent of this study is to develop and apply an objective method for the description and classification of structure in a biological system for use in the correlation of structure and function. The biological system of interest is the frog retina, in particular, the neuronal structures which ramify in the inner plexiform layer of the retina.

A tree grammar is developed for the description of the 3-dimensional structure of Golgi impregnated retinal neurons seen in light microscope examination. This descriptive method includes 1) an optical sectioning procedure for automatic digitization of 3-dimensional light microscope images, 2) various methods for extracting the neuronal image from noise, 3) the extraction from stereo pairs of the primitive elements of the neuronal tree grammar, 4) the use of the grammar as a tool for investigating the occurrence of simple or complex substructures in a neuronal structure, and 5) the use of the grammar as a tool for the quantitative classification of neurons.

The Weiner - Lee method for characterizing nonlinear systems is applied to the various cells of the frog retina using a two input, spot-annulus white noise stimulus. These functional studies include Procion dye injection of the

neurons whose kernels are obtained. Procion dye injection provides a way of linking the functional descriptions to the Golgi-obtained structural descriptors. Correlation of structure and function is approached through comparisons of structural reconstructions and functional identifications.

In short this study employs quantitative descriptors and classifiers to correlate structure and function in the frog retina.

"It is an especially characteristic feature of all living organisms that their structures and functions are predominantly purposive."

Bernhard Rensch

INTRODUCTION

Seventy years have passed since Cajal yet his work remains as a paragon of structural description and investigation. Cajal modeled the structural units that he observed in a natural language, employing high level descriptive terms such as varicose, twisted, arborization, stratification, etc. His descriptive tools were side view drawings and natural language description. In the case of the frog retinal neurons Cajal left us with no flat mount or "en face" drawings and his tools provided no objective or algorithmic description of the retinal neurons.

New tools have appeared since Cajal and the task of this thesis is to outline an objective descriptive method for the analysis of the 3-dimensional structure of frog retinal neurons. This method, like Cajal's, is based on Golgi impregnated neurons. However, the flatmount retinal preparation is used and some techniques from language theory and image processing are employed in a computer assisted

approach to structural description and classification. Also, we now have the Procion dye injection method which allows direct correlations to be made between structure and function.

The questions for which answers are sought using these tools are: 1. What flat mount or top view structural characteristics do the frog retinal neurons have and to which of Cajal's side view neurons do these correspond. 2. What 3-dimensional objective characterizations can be made of these neurons. 3. To what extent do neurons fall into stereotyped classes and to what extent do they form a structural continuum. 4. Which structural types perform which functions as evaluated by white-noise functional tests. This thesis will answer number 4 in only a very preliminary way due to the fact that our data on this point is, as yet, inadequate.

The central motivation for this work is the need for a solid description of the structure of the retina for use in functional-structural correlations. It is felt that in order to understand this system the structure as well as the function should be rigorously and precisely described.

Whereas the human visual system is very good at classifying objects according to abstract, high level descriptors, it is also relatively poor at defining exactly what its algorithm for abstraction is and which descriptors

are being used. It therefore is not well suited to a quantitative evaluation of structure. I think that it is quite difficult to specify an algorithm which will more appropriately separate the retinal neurons into classes than Cajal's has done but rather easy to design an algorithm which will point out the quantitative statistical nature of the groupings. If we wish to find correlates between aspects of function and structural subunits we need an objective description of structure. If we wish to find the statistical variations in a structural characteristic throughout a population of cells we need such a description. This objective information, I think, will be useful in the central problem of correlating function and structure.

The general problem therefore is one of establishing a methodology for structuring information about a biological system, in our case the frog retina. What sort of data type is useful for accumulating information about a biological structure and for allowing an investigator to interrogate the data in a facilitated manner? Much of the information which has been gleaned from experimentation over the years lies entombed in small bits and pieces of natural language description which defy incorporation into an inclusive framework of retinal knowledge. In part, this situation is a reflection of the complexity of neural systems and of the fact that many investigators concentrate on an in-depth

analysis of a single aspect of structure or function in a variety of preparations without incorporating their information into a larger picture. We have as a result a depth first rather than a breadth first search for knowledge.

This depth first approach is appropriate I think for an excruciating examination of specific problems. However there is also a place for a global approach which places more emphasis on the overall picture. This latter approach I believe can make full use of modern computer science methods for structuring the data into an overall model.

Biological systems exhibit structural and functional characteristics and these two are inextricably intertwined. It seems therefore that a framework for representing these systems should incorporate both structural and functional information. This thesis is mainly concerned with structural descriptors but a formal approach to the larger problem is set forth in 5.3, and its application to a simple case is described in 8.2.

A canonical form or global data type for describing these neurons should have the capability of representing data from Golgi, methylene blue, or Procion stained neurons seen in light microscope images, data from ultrastructural studies of synaptic types and interconnection details and data from functional studies on functional pathways and

transfer characteristics. The experimental state of the art is not quite to the point of being able to collect and correlate all these various pieces of information but it is getting there through the multiavenued approach advocated by such researchers as Ken Naka.

The thesis can be divided into two main issues, namely the methodology of obtaining and structuring information about the retina, especially light microscope, morphological information and the application of these methods to our data on the frog retina.

Chapter 2 covers the history of **3-dimensional** reconstruction in neuronal morphology and discusses work on objective description of neurons. It also includes a brief summary of retinal structure especially that of the frog.

Chapter 3 describes the digital image processing software system which we developed to handle 3-dimensional reconstruction, description and classification. This chapter describes our hardware configuration and discusses the problems and advantages associated with a mini-computer based image processing system.

Chapter 4 deals with the problem of obtaining an accurate 3-dimensional digital representation of light microscope objects. The **3-dimensional defocus system** transfer characteristics of the microscope are discussed and an algorithm for removal of defocused information is

evaluated. Examples of enhancing filters and interactive noise removal operations on images are given. A method is presented for computing stereo pairs from optical sections.

Chapter 5 outlines a grammatical approach to the representation and description of neuronal structure. An interactive procedure is used for the extraction of primitive descriptors from stereo pairs. A simple grammar is defined on these primitives for the description of neurons. The class concept is employed to form a framework for modeling structure and function.

Chapter 6 discusses the use of the structures presented in Chapter 5 for the classification of neurons. Two methods for classification are discussed, finding clusters in a vector space and partitioning a cell population by occurrences of structural subunits.

Chapter 7 and 8 present our results on frog retinal morphology and function. Chapter 7 covers the results of our classical morphology with emphasis on the ganglions and amacrine cells of the frog retina. Chapter 8 deals with structural-functional correlates and speculates on functional models based on structural descriptors. Biologists may be more interested in these chapters than in the preceding ones.

"There are schools in the sciences, communities, that is, which approach the same subject from incompatible viewpoints."

Thomas S. Kuhn (20)

2. HISTORICAL PERSPECTIVE

In this chapter we will establish a background of previous work in the area from which to view the material in later chapters. We divide this historical review into four parts, namely classical morphology, quantitative approaches, 3-dimensional reconstruction and structure-functional correlates **in the frog retina.** These necessarily overlap but the distinction may be helpful.

2.1 Classical Morphology

Classification of neuron types is the first step in the understanding of a part of the central nervous system. In the retina, Ramón y Cajal classified the neurons into five main types, based on a comparative study of many types of retinas from many species, and his classification scheme, which was extended by Polyak, has remained unchanged. Cells seen in either Golgi or methylene blue preparations were classified according to their global subjective appearances

and their relative position in that part of the central nervous system. Cajal viewed these neurons from saggital sections and hence his classification scheme is based primarily on the side view features. Figure (1) shows a graphical representation of Cajal's classification scheme, and illustrates the fact that his descriptions are couched in high level, natural language structural terms. Figure (2) shows the side view drawings which Cajal left us in the case of the frog retina. His classification has stood the test of time and provides a strong example of the power of the human visual system in grouping complex structures into similar types. It is unfortunate that Cajal did not perform flat mount studies and thus give us top view data as well. This method however did not appear until later when pioneered by such researchers as Richardson, Stell, Witkovsky, Boycott and others (44,52,3).

Ramon Moliner's (41,42) attempt to categorize neurons on the basis of their dendritic patterns is a recent example of the same general method. He dealt with neurons from various mammals and from diverse regions of the central nervous system and confined his quantitative analysis to such terms as rich, poor, average, long and short. The major terms which form the substance of his classificatory scheme were radiate, tufted, wavy and terms referring general overall orientation of dendrites.

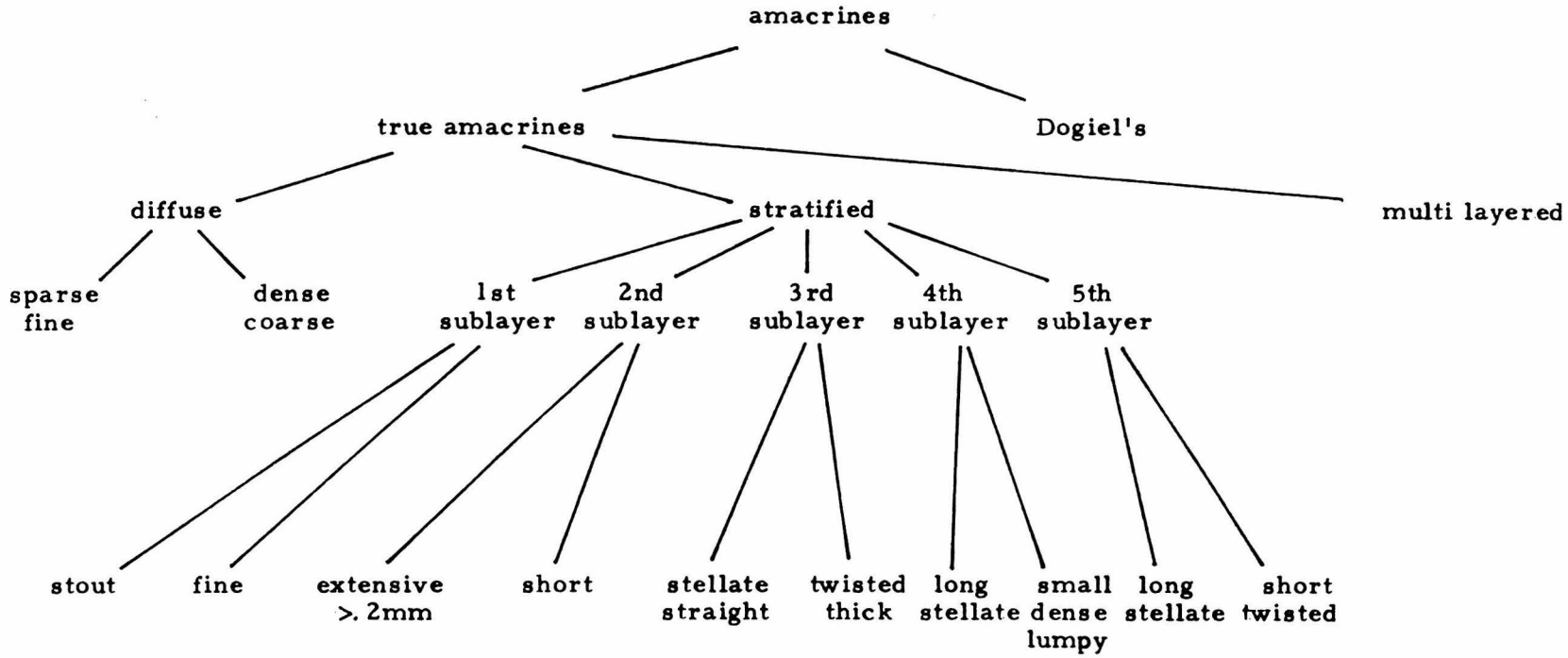


Figure 1a. Basic discriminating features used by Cajal to subdivide amacrines into types.

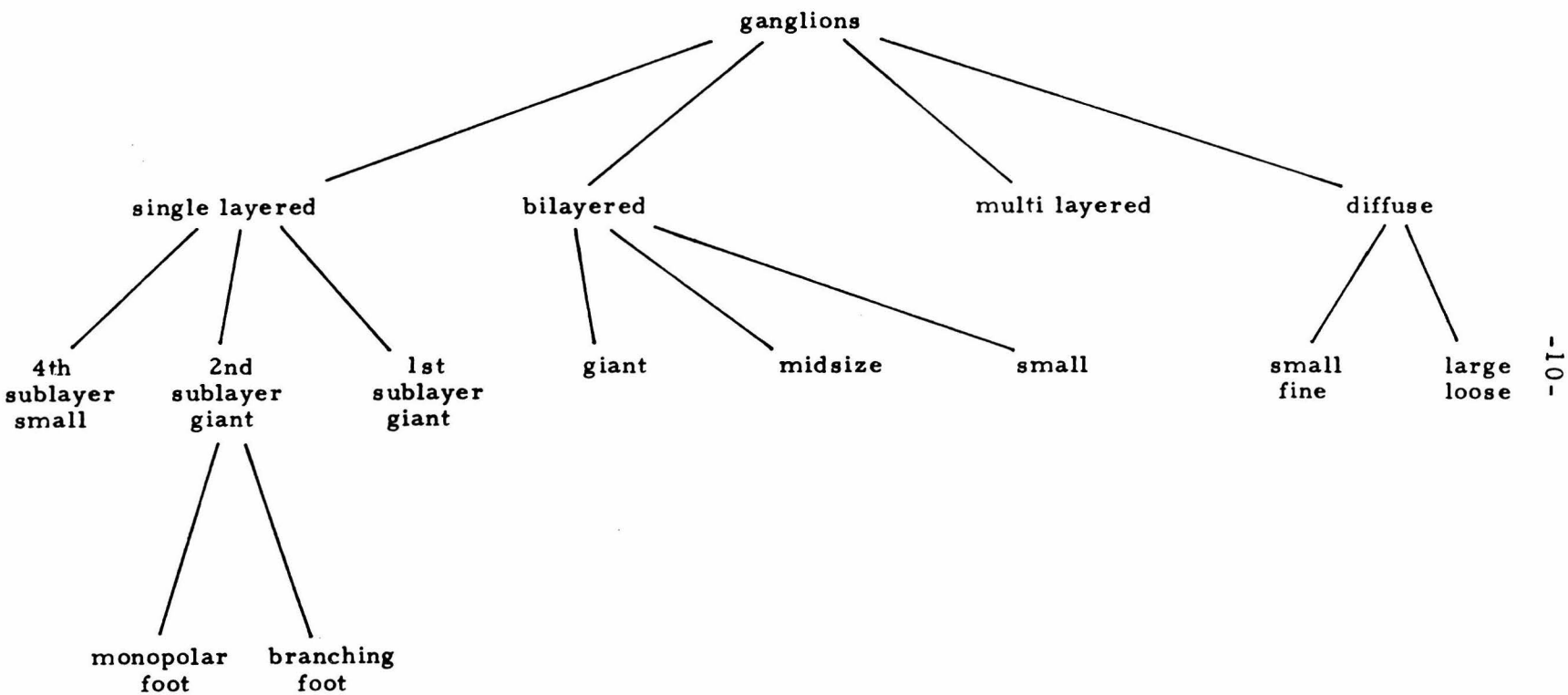


Figure 1b. Basic features used by Cajal to subdivide frog ganglion cells into types.

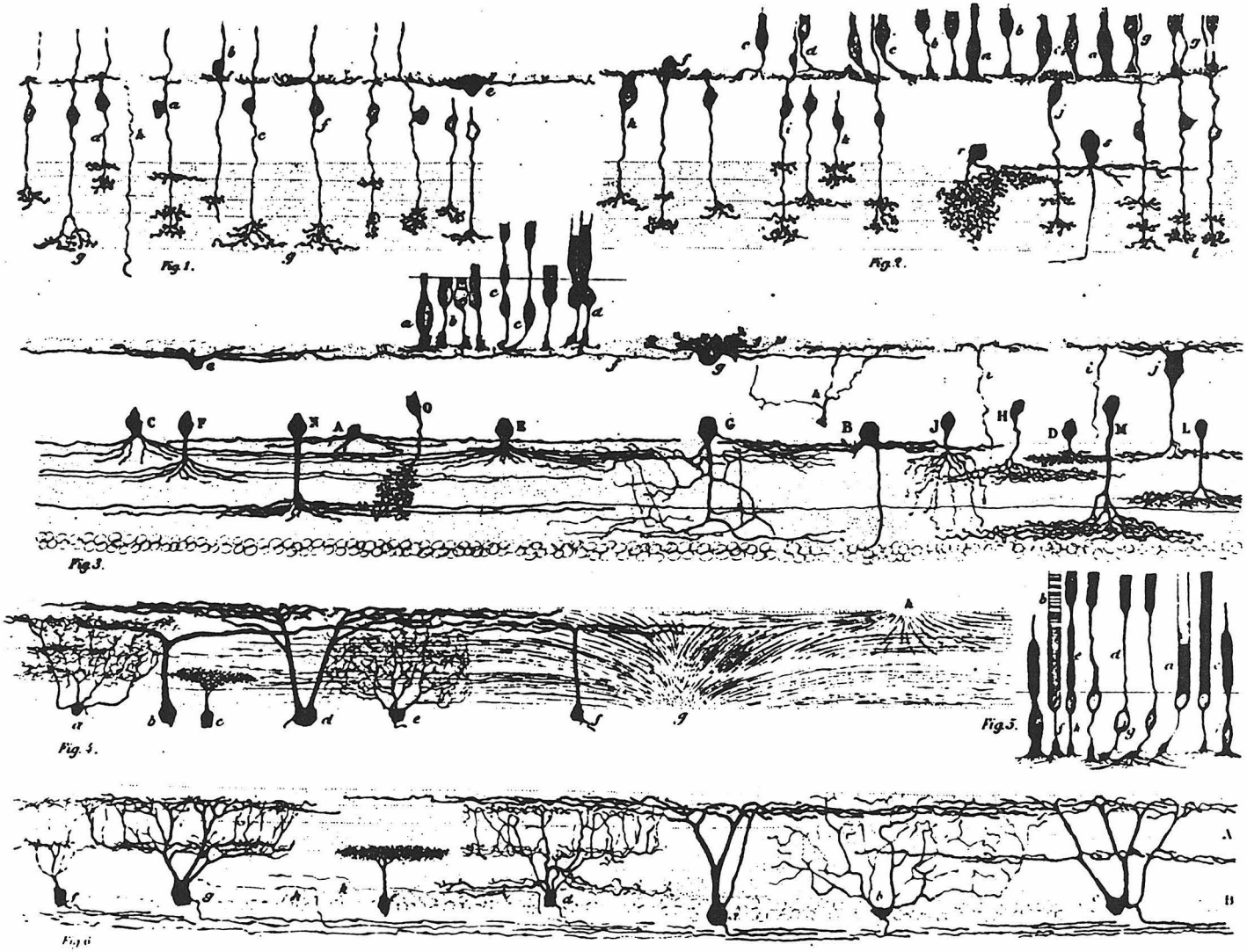


Figure 2. Cajal's side view drawings from frog retina.

These features were subjectively evaluated for a large population of neurons. He was able to propose a prototype dendritic pattern and to correlate certain shapes with locations in the central nervous system. This line of work climaxed in the categories of isodendritic, allodendritic, and idiodendritic neurons, which Molinear establishes as points in a continuum of shapes and associates them with regions in the brain. In the frog retina the situation may be somewhat better defined because of the very organized nature of the retina as compared to the much more complex array of structures occurring throughout the CNS, hence a more quantitative approach applies better to the retina initially.

2.2 Quantitative approaches

Until recently most methods for analyzing the structure of cells were based on subjective evaluation and produced qualitative results. However several people have seen a need for precision and have attempted to obtain quantitative descriptions of structure. Palkovits "et al."(33,34) have used this approach productively in their analysis of the cerebellar cortex in the cat, obtaining simple quantitative information such as numbers of cells, extent and length of dendrites. Bok and Sholl (1,2,48,49,50) also were early pioneers in quantitative structural methods. They sought to

obtain mathematical laws governing neuronal structure in the cortex. Stell (52), Colonnier (6), Rall (38,39,40), Dowling (9), Fox and Barnard (11) and others have all made progress in methods for quantifying various aspects of neuronal structure. Stell and Witkovsky used a graphical method for obtaining size parameters of ganglion cell dendritic trees in dogfish retinas.

Several methods have emerged for quantitative measurement of cell body and nuclear sizes from tissue sectons. Workers have traced section images onto thick uniform paper, cut out the outlines and weighed them to obtain information on the cross sectional area distribution of cells (45,17). Others have traced cell profiles on graph paper and then by counting the squares or using a planimeter have obtained statistical data on cell body sizes (29,37,55,19). All of these methods are laborious and none deal with the more difficult problem of characterizing complex cell branching structures. Wann "et al" (53,54) have extended this work with a computerized system for measuring neuroanatomical data, and the general interest in a quantitative approach is evident from many workers (7,21,37,29,12,26,16).

2.3 3-Dimensional Reconstruction

Two aspects of 3-D reconstruction efforts will be

discussed, namely, reconstructions of light microscope images of dye injected or silver stained cells, and reconstructions of ultrastructure from electron microscope data.

2.3.1 Light Microscope Reconstruction

Early efforts in the developement of computing techniques for reconstructing and describing neurons include the works of several. Glaser and Van der Loos (15) developed a small computer neuronal measurement system and dealt with the problem of analyzing thick light microscope sections. Fox and Barnard (11) developed quantitative aspects of dendritic branching in Purkinje cells. Manner (27), Colon (5) and Ledley (22) reconstructed the branching structure of neurons.

An especially successful recent approach is that taken by Cowan "et al."(53,54). They wished to obtain measurements of neuronal processes in Golgi preparation and to do this they used a small computer which could control and sense the stage and focus positions of a microscope. The researcher moved a microscope cross-hair around on the neuron, signaling various points to the computer thus building dendritic origins, branch points and process terminating points. When a particular branch of a neuron has been recorded to its end the computer returns the stage and focus

position to the last uncompleted branch and the user continues recording. In this way the 3-dimensional skeleton of the cell is obtained with good accuracy.

Advantages of this approach are: 1. The human eye is viewing the neuron directly by looking into a microscope thus using the best resolution available for light microscope images. 2. The computer is doing what it does very well, that is, performing a bookkeeping task. 3. This data allows the computer to compute quantitative features of neurons to rotate and display them and thus facilitate objective structural description.

Disadvantages of this approach are: 1. The small depth of focus of the microscope, especially at high power prevents the researcher from resolving ambiguities in structure which are due to incomplete or broken Golgi impregnation and other sorts of noise. It is remarkable how easily these types of ambiguities can be resolved if the worker can use high level structural cues obtainable when viewing the entire 3-D structure.

If a neuronal process is composed of only a string of silver blobs, as is often the case with Golgi, this method suffers due to limited depth of field. 2. In a complex structure when the computer returns the user to the last incomplete branch it is sometimes difficult to remember which path was last taken and the researcher may tend to get

lost. 3. The resulting structure is represented in the computer as a graph structure. This forms a good basis for the imposition of formal structure but these workers have not taken the next important step of developing a grammatical representation of these structures in which it would be easy to define and extract more high level or abstract features. A syntactic representation would put the power of language theory and parsing techniques at the disposal of these workers. This thesis presents a solution to the above difficulties in chapters 5 and 6.

A second major approach to the 3-dimensional reconstruction and description problem is exemplified by the work of Reddy et al.(43). In this method a dye such as procion or cobalt chloride is injected into a neuron, in this case into a motorneuron controlling swimmerett motion in lobster. The cell is then fixed, embedded and serially sectioned. Photographs are made of each section followed by digitization of the photographs. Boundaries of objects are located in the sections, either manually or automatically. Attempts to automatically locate the object have not as yet been successful and the manual method of tracing section images has been used. The computer assists in assembling the 3-D structure from the sectional outlines then displays, rotates and computes such quantitative features as number of dendritic branches, dendrite diameter, length, volume and

an approximate surface area. In brief the method involves: 1. staining 2. sectioning 3. photographing and digitizing 4. **Aligning** adjacent sections 5. finding the objects in each section 6. assembling sectionals into 3-D description and 7. extracting numerical features.

The disadvantages of this method I think are overwhelming. 1. The process of serial sectioning is time consuming, tedious, and often results in distorted sections as well as ruined sections. 2. Photographing each section image is time consuming. 3. **Alignment** of adjacent digitized section images is a problem compounded by distortions of sections induced by the physical sectioning procedure. 4. Despite attempts at automatically searching for the dendritic profiles, success has normally relied on the manual method of tracing the various cell elements. **Thresholding** and edge detection techniques were applied but served only to accent the fact that a rather amazing pattern recognition system is needed even to find dendritic profiles that to the human observer seem clearly separated from the background. The manual method is tedious to the point of being impossible. It seems that this approach will be more productive in solving pattern recognition problems than in providing useful biological data. However, a possible advantage of this method is that by sectioning the object the out of focus information is minimized. The difficulties

inherent in this technique are revealed by the paucity of 3-dimensionally reconstructed cells so far obtained.

A third approach worthy of mention is that masterminded by Selverston et al. (46,47). This method is very similar to Reddy's in that the cell is injected with Procion dye (crustacean ganglia), then physically sectioned, photographed, digitized, **aligned,** traced and analyzed. This method suffers from most of the disadvantages mentioned above but shares the powerful advantage that analysis of structure from dye injection provides for direct correlation of structure and function.

We believe that correlation of structure and function may be most easily accomplished by a multiavenued approach. Golgi provides perhaps the best representation of structure but dye injection while not usually as complete in structural definition provides direct correlation to function. These observations suggest that 3-dimensional analysis should be performed on Golgi impregnated cells and that this analysis should be linked through Procion dye injection to functional properties of cells by performing functional electrophysiology on neurons then injecting dye into those same neurons.

2.3.2 Electron Microcope Reconstructions

Several notable efforts have been made to employ

computer assistance in building 3-dimensional representations of objects from EM sections. Computer enhancement of electron micrographs especially to identify periodic objects in noise using Fourier filtering techniques has been worked on theoretically and practically by Moody (31), Nathan (32), Huang (18), and others. Three-dimensional reconstruction from projections obtained from transmission electron micrographs taken from several angles is worked out by Crowther "et al."(6) and Gilbert (14).

Leventhal and Ware (25) studied neuronal regeneration and specificity in genetically identical clones of the rotifer Asplanchna brightwelli and the crustacean Daphia magna, through use of **3-dimensional** reconstruction of serial sections from low magnification electron micrographs. These micrographs were aligned and photographed with a 35 mm movie camera to make a cine film which was used for interactive recording of data. The computer was used as a notebook for recording such information as location of pathways, cell and fiber positions, branches and synapses. This information was extracted by the researcher manually using a computer sensed 'mouse' and was maintained in the machine as several list structures. Using these lists it was possible to compare neurons from several animals in studies of regeneration and specificity. These lists formed a binary tree data structure which allowed additional

ultrastructural information to be collected.

To summarize the various work being done on 3-dimensional reconstruction and morphological characterization in nervous system study we will point out what we consider to be some strengths and weaknesses of the field. 1. We are not aware to date of any successful, fully automated procedures for extracting structural descriptions from either light microscope or electron microscope images. The pattern recognition problems are great and it will probably be a few more years until such fully automated systems can compete with the human eye. Until then the time consuming manual method seems to be the best way to go. The human observer can quickly bring a huge amount of information to bear on each structural ambiguity or decision and this power is severely lacking in current machine implementations. For Golgi impregnated neurons this problem is compounded by the occurrence of noise in the form of random silver stained particles and cell pieces lying around the object of interest. The ability to tell what is the neuron of interest and what is not is a very high level visual task. 2. The computer's real strength in this field lies in its ability to function as a data structuring device. However, as yet this strength has not been well explored. Binary tree structures have been used to advantage by Leventhal (25), Reddy (43) and others but we

see the need for a more powerful conceptual tool to operate on this sort of data structure for the extraction of abstract descriptions. What is needed is a specialized language for describing and classifying structure. Such a language construct should provide both a data structure for morphological information and a set of operations on this data structure. This approach to morphological and functional analysis will be developed in chapters 5,6 and 8. The particular operations will vary with the research problem but could include classification, comparison and the construction of functional models based on structural data.

2.4 Correlation of Structure and Function

We turn now to our central problem of correlating structure and function in the frog retina. In this review chapter we will mention only the work done on the frog retina.

In a landmark paper Lettvin et al. (23,24) correlated certain spatio-temporal visual patterns with spiking activity of fibers in the frog optic nerve and thus classified the functional activities of the fibers into four operations; 1. sustained contrast detection (boundary detectors), 2. net convexity detection (movement gated, dark convex boundary detectors), 3. moving edge detection,

and 4. net dimming detection. The first two operations were displayed by unmyelinated fibers and 3 and 4 by myelinated fibers. The functional groupings were reportedly quite distinct but many intermediate functions were observed.

In a later paper Lettvin "et al."(24) suggested a provisional form-function relation in the frog retina. They divided Cajal's ganglion cells into the following 5 types based on their layering and size characteristics. 1. one-level constricted sustained contrast detectors, 2. one-level broad field, net dimming 3. many- level H distribution, net convexity detection 4. many- level E distribution, moving edge detector and 5. diffuse trees, average light levels . These correlations are based on the assumption that functional receptive field is determined by dendritic field size and that soma size is proportional to axon size which is matched with conduction velocity. Furthermore these people speculate that the inner plexiform layer is organized around two major layers, one the outermost layer where the group 4 cells arborize and the other middle layer where the group 1 cells arborize. The group 2 and the group 3 cells arborize in both of these layers forming bistratified structures. Since group 4 is net dimming and group 1 is boundary detection, they speculate that groups 2 and 3 may form different combinations of information about

boundaries and dimming. It remains for a more detailed examination of structure and function to prove or disprove these speculations.

Pomeranz and Chung (36) have provided additional indirect evidence in support of the above function-structure correlates. They performed functional experiments on tadpoles by recording extracellularly from the optic nerve while presenting various spatio-temporal patterns. They also performed Golgi structural studies on tadpole ganglion cells. In both structure and function they identified neuron types corresponding to groups 2, 3, and 4 above. Group 1 appeared to be absent both structurally (Golgi) and functionally. Also group 2 cells reportedly were structurally absent from the periphery of the tadpole retina and functionally absent from the same area. From these results it was concluded that the form-function scheme of Lettvin "et al" had been strengthened.

In the above studies all structural-functional correlates are either speculative or concluded from indirect data. To fully resolve these questions dye injection techniques should be used to obtain direct form-function links. Also, in our studies of Golgi impregnated tadpole retinal neurons, we observed cells which we identified as corresponding to group 1 frog neurons. These structural identifications should be made using both side and top views

of these cells because in the case of the single level restricted cell the top view is singularly characteristic in both frog and tadpole. For these reasons, and despite the fine work of the above researchers we feel that not only the description and classification of structure but also the establishment of structural functional correlates is still an open problem.

Probably the most direct and well conceived approach to the correlation of structure and function in the frog retina is represented by the work of Matsumoto and Naka (28). These workers performed Golgi studies which they correlated with Procion dye injected cells which were functionally studied. With this multistratified approach direct form-function correlates could be made. This work is of a preliminary nature but demonstrates a powerful method. The horizontal cells gave rise to s-potential responses. Bipolar cells gave rise to slow potential changes of center-surround interaction and presumed amacrines showed complex center-surround interaction.

With this brief summary of the work that has been done in 3-dimensional reconstruction, morphological description and classification, and the correlation of structure and function in frog retinal neurons we now turn to the methods developed and applied in this thesis.

Readers not interested in computers may wish to skip

chapter 3 which describes our minicomputer based image processing system and chapter 4 which describes various image processing operations.

References

1. Bok, S. T., (1934), Messungen an den Ganglionzellen der Grosshirnrinde. Die Eintheitlichkeit der einzelnen Hauptzonen. Zeitschr. f. mikr. anat. Forsch, 36: 645-650.

2. -----, (1936), A quantitative analysis of the structure of the cerebral cortex. Proc. Acad. Sci. Amst. , 35: 1-55.

3. Boycott, B. B., and J. E. Dowling, (1969), Organization of the primate retina: Light microscopy. Phil. Trans. Roy. Soc. Lond., B, 255: 109-184.

4. Cajal, S. Ramon Y, (1911), Histologie du Systime Nerveux de l'Homme it des Verte'bre's, Maloine, Paris.

5. Colon E. J., and G. J. Smit, (1970), Quantitative analysis of the cerebral cortex. II. A method for analyzing basal dendritic plexuses, Brain Res., 22: 363-380.

6. Colonnier, M., (1964), The tangential organization of the visual cortex , J. Anat., London. 98,3: 327-344.

7. Cook, W. H., J. H. Walker, and M. L. Barr, (1951), A cytological study of transneuronal atrophy in the cat and rabbit. J. Comp. Neur. 94: 267.

8. Crowther, R. A., D. J. DeRosier, and A. Klug, (1970), The reconstruction of a three-dimensional

structure from projections and its application to electron microscopy , Proc. Roy. Soc. Lond., A, 317: 319-340.

9. Dowling, J. E., and B. B. Boycott, (1969), Retinal ganglion cells: a correlation of anatomical and physiological approaches. , In: The Retina: Morphology, Function and Clinical Characteristics, UCLA Forum in Medical Sciences No. 8. B. R. Straatsma Ed., Univ. of Calif. Press, Berkeley and Los Angeles, 31-62.

10. Elias, H., (1971), Three-dimensional Structure Identified from Single Sections, Science, 174: 993-1000.

11. Fox, C. A., and J. D. Barnard, (1957), A quantitative study of the Purkinje cell dendritic branches and their relationship to afferent fibres. , J. Anat. , Lond., 91:299-308.

12. Fry, F. J. and W. M. Cowan, (1972), A study of retrograde cell degeneration in the lateral mammillary nucleus of the cat, with special reference to the role of axonal branching in the preservation of the cell. J. Comp. Neur. 144:1.

13. Garvey, C. F., "et al", (1973), Automated Three-dimensional Dendrite Tracking System, , Electroencephalography and Clinical Neurophysiology,

35: 199-204.

14. Gilbert P. F. C., (1972), The reconstruction of a three-dimensional structure from projections and its application to electron microscopy. II. Direct Methods, Proc. Roy. Soc. Lond., B. 182: 89-102.

15. Glaser, E. M., and H. Van der Loos, (1965), A semi-automatic computer microscope for the analysis of neuronal morphology, IEEE Trans. BME, 12: 22-31.

16. Globus, A., and A. B. Scheibel, (1967), Pattern and field in cortical structure: the rabbit. J. Comp. Neurol. 131: 155-172.

17. Hamlyn, L. H., (1954), The effect of preganglionic section on the neurons of the superior cervical ganglion in rabbits. J. Anat., Lond., 88: 184.

18. Huang, T. S., (1967), Combined use of Digital Computers and Coherent Optics in Image Processing, SPIE Computerized Imaging Techniques Symp., Washington D. C.

19. Kelly, J. P. and W. M. Cowan, (1972) Studies on the development of the chick optic tectum. III Affects of early eye removal. Brain Research, 42:263.

20. Kuhn, T. S., (1970), The Structure of Scientific Revolutions, , 2nd Ed. International Encyclopedia of Unified Science Otto Neurath Ed., U. of Chicago Press, Chicago, Ill.

21. Kupfer, C., (1965), The distribution of cell size in the lateral geniculate nucleus of man following transneuronal cell atrophy. J. Neuropath. exp. Neurol. 24: 653.

22. Ledley, R. S, (1964), High-speed automatic analysis of biomedical pictures. Science 146: 216-222.

23. Lettvin, J. Y., H. R. Maturana, W. S. McCulloch , and W. H. Pitts, (1959), What the frog's eye tells the frog's brain , Proc. Inst. Radio Engr. 47: 1940-1951.

24. Lettvin, J. Y., H. R. Maturana, W. H. Pitts and W. S. McCulloch, (1961), Two remarks on the visual system of the frog. In: Sensory Communication Ed. A. Rosenblith Wiley, New York p. 757.

25. Leventhal, C. and R. Ware, (1972), Three-Dimensional reconstruction from serial sections , Nature , 236: 5344, 207-210.

26. Loewy, A. D., (1972), The effects of dorsal root lesions on Clarke neurons in cats of different ages. J. Comp. Neur. 145: 141.

27. Mannen, H., (1964), Arborization dendritiques etude topographique et quantitative dans le noyau vestibulaire du chat. Arch. Ital Biol. 103: 197-219.

28. Matsumoto, N., and K.-I. Naka, (1972), Identification

of Intracellular Responses in the frog retina. Brain Research, 42: 59-71.

29. Matthews, M. R., W. M. Cowan, and T. P. S. Powell, (1960), Transneuronal cell degeneration in the lateral geniculate nucleus of the Macaque monkey. J. Anat., Lond., 94: 145.

30. Maturana, H. R., J. Y Lettvin, W. H. Pitts, and W. S. McCulloch, (1960), Physiology and anatomy of vision in the frog. J. Gen. Physiol., 43 suppl. 129-175.

31. Moody, M. F., (1968), Method for eliminating spherical aberration and preserving phase contrast information in electron microscopy, Internal publication, The Rockefeller University.

32. Nathan, R., (1968), Image processing for electron microscopy: I. Enhancement procedures, In: Ed. Karl-Joseph Hanszen.

33. Palkovits, M., P. Magyar and J. Szentagothai, (1971) Quantitative histological analysis of the cerebellar cortex in the cat. I. Number and arrangement in space of Purkinje cells, Brain Res., 32: 1-15.

34. -------, (1971), II. Cell numbers and densities in the granular layer, Brain Res., 32: 15-30.

35. Polyak, S. L., (1941), The Retina, U. of Chicago

Press, Chicago, Ill.

36. Pomeranz, B. and S. H. Chung, (1970), Dendritic tree anatomy codes form-vision physiology in tadpole retina, Science, 170: 983.

37. Powell, T. P. S. and S. D. Erulkar, (1962), Transneuronal cell degeneration in the auditory relay nuclei of the cat. J. Anat., Lond., 96: 249.

38. Rall, W., (1967), Distinguishing theoretical synaptic potentials computed for different soma-dendritic distributions of synaptic inputs. J. Neurophysiol. , 30: 1138-1168.

39. -----, and G. M. Shepherd, (1968), Theoretical reconstruction of field potentials and dendrodendritic synaptic interactions in olfactory bulb. J. Neurophysiol. , 31: 884-915.

40. -----, R. E. Burke, T. G. Smith, P. G. Nelson and K. Frank, (1967), Dendritic location of synapses and possible mechanisms for the monosynaptic EPSP in motoneurons. J. Neurophysiol. 30: 1169-1193.

41. Ramon-Moliner E., (1962), An attempt at classifying nerve cells on the basis of their dendritic patterns, J. Comp. Neurol. 119: 211-227.

42. -----, (1968), The morphology of dendrites. In: The structure and function of nervous tissue. vol. 1, Ed. G. H. Baurne, New York, Academic Press. 205-267.

43. Reddy, D. R., W. J. Davis, R. E. Ohlander and D. J. Bihary, (1973), Computer analysis of neuronal structure, in: Intracellular staining in neurobiology Eds. S. B. Kater and C. Nicholson, Springer-Verlag, NY p. 227.

44. Richardson, K. C., (1960), Studies on the structure of autonomic nerves in the intestine, correlating the silver-impregnated image in light microscopy with the permanganate fixed ultrastructure in electron microscopy. J. Anat., 94: 457-472.

45. Scammon, R. E. and G. H. Scott, (1927), The technique of determining irregular areas in morphological studies , Anat. Rec., 35: 269.

46. Selverston, A. I., (1973), Using dye injection to study the functional architecture of crustacean neurons. In: Intracellular Staining in Neurobiology. Ed. S. B. Kater and C. Nicholson, New York: Springer Verlag.

47. Selverston, A. J., and D. Kennedy, (1969), Structure and function of identified nerve cells in the crayfish, Endeavor, 28: 107-113.

48. Sholl, D. A., (1953), Dendritic organization in the neurons of the visual and motor cortices of the cat, J. Anat., Lond., 87: 387-406.

49. -----, (1955), The organization of the visual cortex

in the cat, J. Anat., 89: 33-46.

50. -----, (1956), The organization of the cerebral cortex, Methuen and Company, Ltd. London; Wiley and Sons, Inc., New York.

51. Stein P. G., "et al.", (1969), Spectre II: General-Purpose microscope input for a computer, Science , 166: 328-333.

52. Stell, W. K. and P. Witkovsky, (1973), Retinal structure in the smooth dogfish, Mustelus canis: general description and light microscopy of giant ganglion cells. J. Comp. Neur., 148: 1-32.

53. Wann, D. F. and W. M. Cowan, (1972), An image processing system for the analysis of neuroanatomical data, Proc. Computer Image Processing Symp., Univ. of Missouri, Columbia, pp. 411-419.

54. -----, T. A. Woolsey and W. M. Cowan, (1973), An on-line digital computer system for semi-automatic analysis of Golgi impregnated neurons. IEEE Trans. Biomed. Eng., Jul. 233-247.

55. Wiesel, T. N. and D. H. Hubel, (1963), Effects of visual deprivation on morphology and physiology of cells in the cat's lateral geniculate body, J. Neurophysiol., 26: 978.

56. Willey, T. J., R. L. Schultz and A. H. Gott, (1973), Computer graphics in three dimensions for

perspective reconstruction of brain ultrastructure, IEEE Trans. on Biomed. Eng., July 288-291.

"Each order of magnitude increase in speed and memory size of computers requires a reexamination of computational techniques and an assessment of the new problems which may be brought within the realm of solution."

Robert Kalaba (2)

3. A MINI-COMPUTER BASED IMAGE ANALYSIS SYSTEM

Image processing is a computer activity which requires the ability to handle huge amounts of data. A typical digital image is made up of 512 lines with 512 samples each. If each picture element (pixel) is one byte, this amounts to over a quarter of a million bytes. If we wish to represent a third dimension by using a stack of images, the data space required may exceed ten million bytes. This situation would be quite discouraging for the mini-computer user if it were not for the rising power and decreasing costs of the mini plus the availability of large, high-speed secondary storage. This chapter describes a picture processing hardware and software system which runs on a mini-computer. The cost of such a basic hardware system could be kept under 80K.

3.1 Hardware

3.1.1 Image Acquisition

The basic parts of an image processing system are image acquisition, input-output, image computations, image storage, and image display.

Until relatively recently, a major obstacle to the development of image processing systems for biology was the unavailability or high cost of input devices for grey level images. High resolution multiple grey level devices were required in order to handle the complex biological image data . In the late sixties and early seventies the cost has dropped precipitously and the quality has improved dramatically.

For our work with Golgi impregnated neuronal images, we use an automated light microscope system (ALMS) figure (3), which was developed at Caltech's Jet Propulsion Laboratory (1,2), to digitize our cell images. This digitization system uses computer driven galvanometers to move mirrors which scan the image seen through the microscope across a small hole behind which is positioned a photo-multiplier tube. A TV camera with computer interface and digitization hardware could also be used for data acquisition, with some sacrifice of resolution. The slower scan rates of the image plane scanner allow for better resolution especially at low light levels with a considerable loss of speed, whereas the Vidicon TV camera tubes or silicon diode array

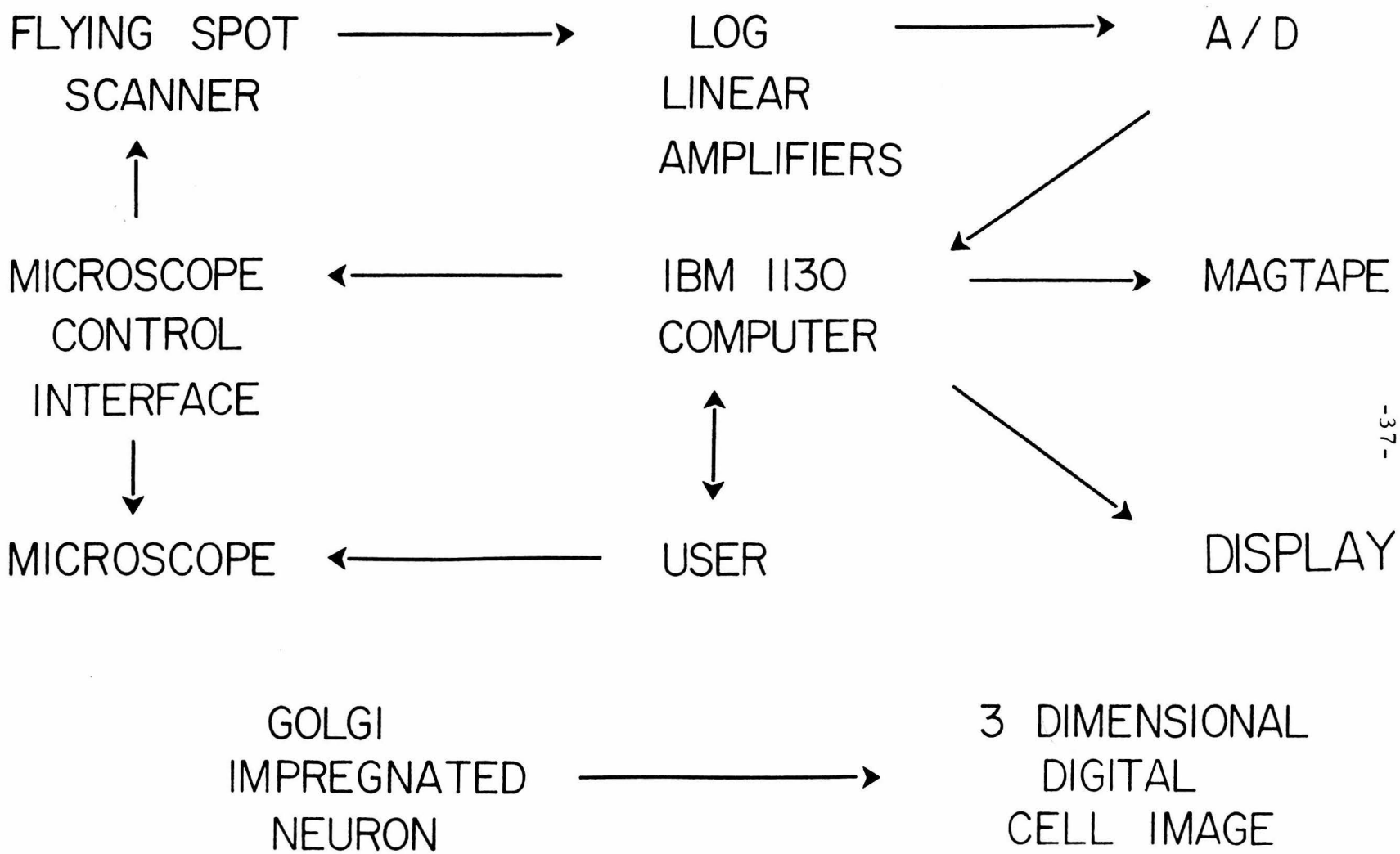


Figure 3. Schematic of the JPL automated light microscope system (ALMS).

tubes scan much faster. The scan speed is usually limited by the computer's ability to handle the incoming data but their resolution suffers due to bandwidth limitations of video signals. The diode array tubes are rapidly increasing in resolution, decreasing in cost, and are thus becoming more appealing.

3.1.2 Image Storage Hardware

Two types of storage are normally required for the efficient handling of image data. For long term storage of images, a magnetic tape device is the obvious choice due to its great capacity and low cost. However, long access time and inability to do random access write operations renders this medium unsuitable for secondary storage from which computations are performed. For high speed random access 'computational' secondary storage a disk is the obvious choice. Our experience suggests that all operations on images should be performed using large-buffer I/O from disk and only occasionally should one take the time to input from or output to tape. This of course depends on the particular task.

3.1.3 Image Computation Hardware

The particular computer used for image computations is not very important. The main considerations are cpu speed

and memory size. Many image operations tend to be I/O bound and small memories, which preclude large I/O buffers, will intensify this problem thus reducing speed. For operations involving floating point computations it is, of course, advantageous to have a computer with a floating point processor, especially of the sort that performs the floating point instructions in parallel with normal instructions. The PDP 11/45 for example has a floating point processor which hangs on the unibus just like another device and has its own set of registers. This processor acts in parallel with the cpu and if floating point instructions are interleaved with regular instructions, speed can be increased.

Another important consideration is software support. A machine for which an operating system is provided offers a great advantage. A file management system, editor, compilers for high level languages, and library facilities all combine to make the difference between efficient software development and absurdity.

3.1.4 Image Display Hardware

A wide variety of digital image display devices are currently available. They consist of a CRT display screen, a refresh memory, and an interface to the computer. The display screen may be high resolution (1000 lines), standard

TV, or color. The refresh memory may be in the form of a storage tube [Hughes], a fixed head disk [Comtal], or solid state memory [Ramtek]. For many applications requiring interactive operations an x, y cursor or light pen is needed; this may be a separate device mounted on the display [Grafpen] or may be incorporated into the display hardware.

An image display is an essential experimental tool for immediate inspection of the result of some operation, for debugging, and for interactive operations on images. With a color display system having appropriate double image refresh capacity, it is possible to display full screen, red-green, stereo pairs for viewing 3-dimensional structures.

3.1.5 One Configuration

Figure (4) shows our present system for picture processing. The system hardware consists of a 28K PDP-11/45 computer from Digital Equipment Corporation, two TM11 tape decks, a Comtal color image display with a .5M byte refresh storage capacity, interactive cursor and pseudocolor hardware, a 20M byte DIVA disk system, a DEC 1.2M word disk cartridge and a Tektronics 4010 terminal. This system costs in the neighborhood of 150K dollars but a minimal configuration consisting of a PDP-11/45 with a floating point processor, an RK05 disk cartridge for system storage

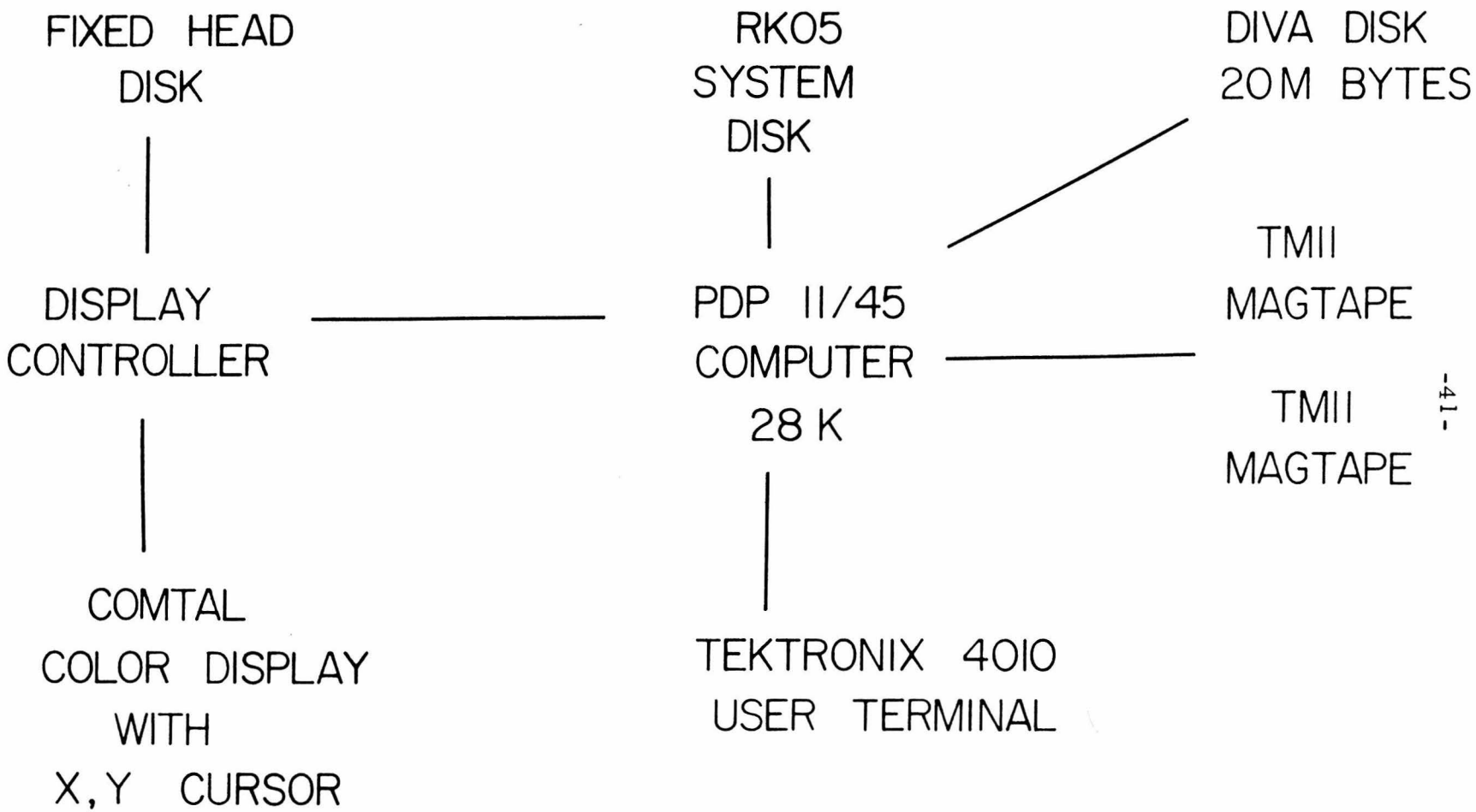


Figure 4. Schematic of image analysis system hardware.

and high speed secondary data storage, a TM11 tape deck for long term large scale storage, and a scan converter display system with light pen could be purchased for under 80K. This is, we think, a minimal system. Long term storage on magtape is a must, and high-speed secondary storage on disk is necessary due to the fact that an entire picture will not normally fit in core. A light pen or cursor is invaluable for many of the interactive procedures that enrich a picture processing system. This configuration does not include a method for digitizing images via TV camera, etc., or for hard copy output. These two items would add another 10-20K dollars to the cost of a system, depending on the quality desired.

3.2 Software

Our Image Analysis System (IAS) runs under the disk operating system (DOS) provided by Digital Equipment Corporation. DOS provides an editor for creating source files, an assembler for assembling macro-coded routines, a Fortran compiler, and library facilities in addition to the DOS monitor which handles system programs and utilities, input/output transfer, device handlers, file management, and various commands for control of program execution and modification. We can therefore write Fortran callable subroutines in Macro and can call Fortran subroutines from

Macro programs by using a CALL macro. This enables one to encode the inner loops, or those portions of the program which need to run fast, in Macro and to use Fortran elsewhere for ease of coding and debugging. We find that little can be gained by macro encoding anything other than innermost loops of a program. This system would benefit greatly by inclusion of a compiler for a more powerful high level language such as PL1, PASCAL, or LISP, but is very useful as it stands.

The core of IAS is composed of the basic input-output routines which provide buffered transfer of image lines to and from core. These routines were developed by Howard Frieden (1) and Kenneth Castleman (3) at JPL to run on a PDP-11/40 computer. They provide double or single buffering so that a maximum of speed can be realized during operations on images too large to fit into core. These routines are used to get a line of an image from disk into core, to put a line from core to disk, to get an image label record, to put a label, and to transfer image lines from tape to core and core to tape. I have used this basic input/output package, modified to run on our configuration, as the underlying input/output for IAS.

Our basic application routines consist of the following:

1) FLT - This digital filtering routine inputs an image

and a user specified two-dimensional filter array and computes the two-dimensional image which is the cross-correlation of the input image and the specified filter. It allows for user specification both of scaling parameters and of the area of the input image to be filtered.

2) FFT - This routine computes the image (whose elements are complex numbers) which is the complex Fourier transform of the input image. This routine can input either byte or complex images, and computes either the direct or inverse transform. The algorithm uses a one-dimensional, Fortran-coded, Cooley-Tukey algorithm which we obtained from IBM to transform each line of the image in place. The resulting image array is transposed and the lines are again transformed in place producing the final image.

3) DIF - This routine subtracts one image from another to yield a difference image.

4) STR - This routine performs a user specified linear transformation on each element of the image.

5) MAX - This routine computes the element maximum of two input images.

6) TOP - This routine computes the top view through a stack of images representing a 3-dimensional image of an object. The output image is obtained by selecting

the maximum element in each vertical column down through the stack.

7) SVU - This routine computes an image which is the side view of an object which is represented by a stack of images.

8) FGEN - This program is a set of function generating routines. It provides facilities for the creation of grey level ramps, Gaussian intensity humps, Gaussian distributed random images, line drawings interactively created, sinusoidal images, and variable text size on images.

9) STEREO - This routine provides for the creation of stereo pairs from a stack of images representing a series of optical sections through an object.

10) NODES - This routine provides a 3-dimensional cursor for extracting features from stereo pairs. It implements the construction of a tree-grammar sentence for describing a neuron in 3 dimensions. The output is a tree whose nodes are; soma (x, y, z, size 1, size 2), process (x1, y1, z1, x2, y2, z2, startsize, endsize), branch (x, y, z), and terminal (x, y, z). See chapter 5 for further explanation.

11) SCRUB - This routine provides a 'paint brush' for creating or modifying images using the interactive cursor. The user specifies a grey level and a brush

size then moves the cursor around on the display screen as if it were a paint brush. In this way selected portions of the image can be erased to remove noise, or corrections can be performed on the image.

The above routines provide some basic analytical tools for analyzing images. Images can be analyzed in the frequency domain by Fourier transforming and performing operations such as multiplication or squared absolute value in the spatial frequency domain. The filtering facility allows great flexibility in linear filtering of images to detect edges, simulate optical defocus or to model retinal function. Syntactic descriptions of 3-dimensional structures can be constructed interactively through use of stereo pairs and the 3-dimensional software cursor. In the following chapters we will describe our application of this system to the problem of 3-dimensional reconstruction, description, and classification of frog retinal neurons.

REFERENCES

1. Frieden, H., Caltech Jet Propulsion Laboratory (unpublished results).

2. Kalaba, R., (1974), Editor's foreword in Pattern recognition principles, Addison-Wesley, Reading, Mass.

3. Weinstein, M. and K. R. Castleman, (1971), 'Reconstructing three-D specimens from two-D section images,' Proc. of the SPIE, 26: 131-138.

"...to distinguish an object from the afflictions of its appearances is an awe-inspiring cognitive accomplishment....the more complex the shape of the object, the harder the perceptual task of extracting it."

Rudolf Arnheim (1)

4. IMAGE COLLECTION AND RESUSCITATION

In this chapter we deal with techniques for the acquisition of digital 3-dimensional light microscope images, methods for removal of various types of noise, and the generation of stereo pairs from light microscope images. We first outline our procedure for the optical sectioning of neuronal objects and the digitization of section images. The second part discusses the removal of out of focus information in optical section images. The third part discusses filtering for object enhancement and interactive noise removal, and the fourth gives a simple method for stereo pair construction. These procedures represent initial stages in the structural description of retinal neurons.

4.1 Optical Sectioning

In section 2.3 we discussed the disadvantages of

methods involving a mechanical sectioning of the tissue, namely, time consumption, tissue distortion, necessitation of section image realignment and the need for photographing each section. In this section we present an optical sectioning procedure (12) which avoids these problems.

Figure (3) is a block diagram of the automated light microscope system (ALMS) developed at JPL which we use for our optical sectioning. The IBM 1130 minicomputer controls the microscope stage position by stepping motors yielding a minimum stage step size of 5 microns in the x and y directions and 0.05 micron in the z direction (axial). The computer also controls the image plane scanner (IPS), mounted on top of the microscope, through user specified programs which can digitize either transmittance or optical density to 256 levels of grey. The computer can be instructed to scan an image from the microscope, generating a digitized image on magtape of a specified resolution - up to a 1024 x 1024 array of eight bit picture elements (pixels). With the 100x microscope objective this maximum x, y resolution is about eleven lines per micron.

The computer can be instructed to scan a series of images separated by a given focus change in microns. Thus, in an automatic optical sectioning procedure the computer scans an image, focuses deeper, scans another image and so on until a stack of digital images has been created and

stored on magtape representing the 3-dimensional structure of the neuron. Each image in this stack represents one optical section through the tissue. In the case of Golgi impregnated neurons the object is semi-opaque and the background is light. Figure (5) shows a schematic of the optical sectioning procedure and figure (6) shows an example of a digitized optical section through a Golgi impregnated neuron from the frog retina. Each optical section contains both in focus and out of focus or defocused information.

With high power objectives the depth of field is shallow and hence objects lying outside of the plane of focus become highly defocused at small distances from the plane. In this case the defocused information in an optical section image is not particularly objectionable and the image is a reasonably accurate representation of the 2-dimensional distribution of optical density at a given level in the tissue.

4.2 Removing Defocused Information

One drawback of the optical sectioning procedure lies in the fact that defocused information in each section image impairs resolution in the z direction. In other words 3-dimensional structures are smeared along the z axis. We have experimented with methods to solve this problem.

The microscope can be viewed as a system which takes as

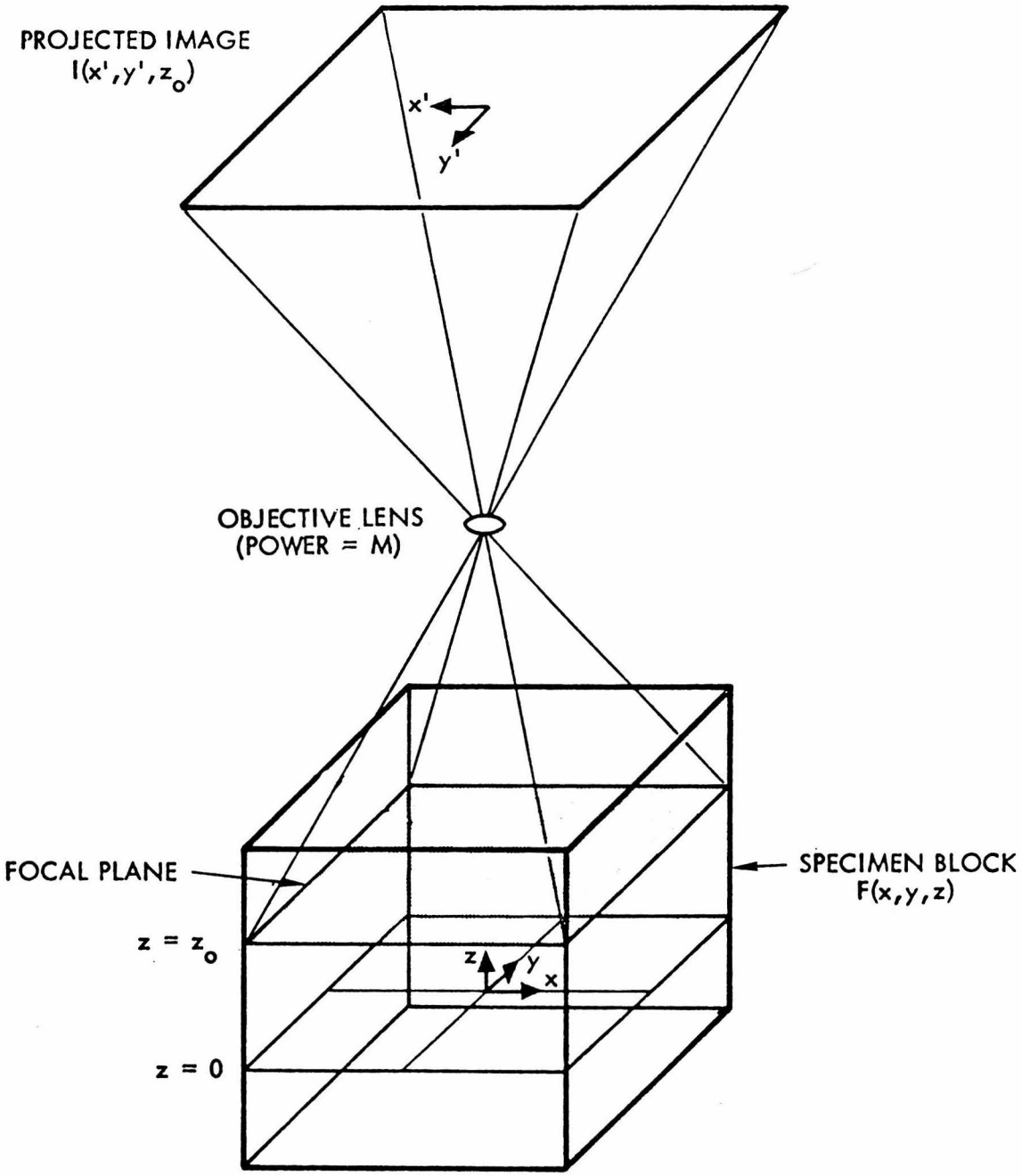
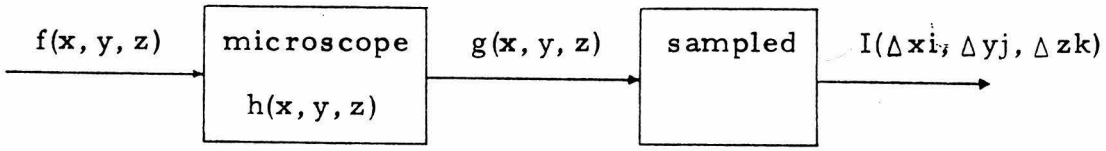


Figure 5. Diagram of optical sectioning geometry.

Figure 6. Example of an optical section of Golgi impregnated tissue, scanned through 100x objective. Note out of focus information. Image is 45 microns wide.

input a 3- dimensional function f(x,y,z) representing the distribution of optical density in the tissue preparation, and produces an output g(x,y,z) which is sampled or digitized to obtain the 3-D digital image $I(\Delta xi, \Delta yj, \Delta zk)$.



The microscope is a linear system, and if we assume that it is also a space invariant system (isoplanarity holds) then the output can be written as follows:

$$4.2.a \quad g(x,y,z) = \int\int\int_V h(\sigma_1,\sigma_2,\sigma_3) f(x-\sigma_1, y-\sigma_2, z-\sigma_3)\, d\sigma_1\, d\sigma_2\, d\sigma_3$$

This is just the convolution integral for linear systems. Isoplanarity does not strictly hold for the microscope optical system but our space invariant assumption does not introduce significant error. The point spread function of the microscope is very nearly the same if the input point of light is moved around in the input space.

The problem of removing defocused information is basically that of finding f(x,y,z), given a sampled version of g(x,y,z) and the 3-dimensional microscope defocus point spread function. From equation 4.2.a we see that this is a

deconvolution problem.

4.2.1 Difficulties in Numerical Deconvolution

Franklin (3) noted that all integral equations of the form of 4.2.a in which f is a continuous function, are ill posed linear problems. He based his argument on Riemann's lemma which states that a small change

4.2.1.a. $\delta h(\sigma 1,\sigma 2,\sigma 3) = \sin(w1\,\sigma 1) + \sin(w2\,\sigma 2) + \sin(w3\,\sigma 3)$

where w1, w2, w3 are high frequencies, produces a very small change in g. However, a small change in g produces a very large variation in h. Thus if we wish to find the 3-dimensional defocus point spread function $h(\sigma 1,\sigma 2,\sigma 3)$ we must be aware of the fact that deconvolution of a linear system is an ill-posed problem and if we have noise in our measurements of f and g we will likely obtain very unreliable estimates of h. The problem does not end here. Even if we can find h, the sheer magnitude of the computational problem of applying equation 4.2.a to large image data sets is prohibitively costly. We have tried the following short cuts to a solution of the problem of obtaining the defocus point spread function and to the problem of finding f(x,y,z) given g(x,y,z) and $h(\sigma 1,\sigma 2,\sigma 3)$.

4.2.2 Theoretical Defocus Transfer Functions

The problem is that of describing and correcting for

the out of focus information in an image which contains both out of focus and in focus components. For example, a photograph taken with a finite depth of field contains both in focus and out of focus information. What is the mathematical nature of the out of focus or defocused images and how can they be removed?

A deblurring algorithm has been described (12) which first takes the original scanned section images and blurs each image by convolution with a Gaussian function. The algorithm then takes the original images and the blurred images and produces a set of deblurred images. If Ii(x,y) are the original data images and Bi(x,y) are their blurred counterparts then deblurred images, 'clean images', Di(x,y) are produced by the following algorithm:

$$4.2.2.a.\quad bD_i(x,y) = I_i(x,y) - a\left(B_{i-1}(x,y) + B_{i+1}(x,y)\right)$$

Where a and b are weights for proper grey level scaling, typically 0.8 and 0.2 respectively.

The major problems with the above algorithm are: 1. The blurring filter is not rigorously derived, it does not consider the amount of defocus dz, it ignores the microscope magnification being used, it ignores the light wavelength, and the numerical aperture of the objective. 2. The

mathematics of the deblurring process had not been previously worked out. In order to put the deblurring process on a firm theoretical basis, two improvements have been made. a) The mathematics of the defocusing process have been examined and b) an experimental determination of the point spread function for defocus has been performed.

Our first task was to obtain the defocus transfer characteristics of our microscope for various degrees of defocus and for our several microscope objectives. The defocus transfer characteristics were obtained in two ways. 1) Stokseth's approximation was used to obtain the theoretical defocus transfer functions. 2) An experimental method was used which involved scanning a very small spot at various degrees of defocus.

Hopkins "et al" (5) have shown that if the aperture is circular and all aberrations are rotationally symmetric then the OTF is given by

$$4.2.2.b. \quad T(s) = \frac{\iint_{-\infty}^{\infty} f(x+\frac{s}{2}, y) f^*(x-\frac{s}{2}, y)\, dxdy}{\iint_{-\infty}^{\infty} |f(x,y)|^2\, dxdy} \qquad x^2 + y^2 \leq 1$$

F(x,y) is the pupil function of an aberration free, defocused optical system, s is related to the spatial frequency f measured in cycles/length by

4.2.2.c. $s = (\lambda/n \sin\alpha) f$

where λ is the wavelength of light, n is the refractive index and α is the aperature angle. The pupil function of an aberration free defocused optical system is of the form

$$4.2.2.d. \quad f(x,y) = \begin{cases} \exp\left[i(2\pi/\lambda)\, w(x^2 + y^2)\right] & , x^2 + y^2 \leq 1 \\ 0 & , x^2 + y^2 > 1 \end{cases}$$

Where w is shown in figure (7) and can be expressed in terms of the defocus distance z by the following expression:

$$4.2.2.e. \quad w = -r - z\cos\alpha + \left(r^2 + 2rz + z^2\cos^2\alpha\right)^{\frac{1}{2}}$$

By substituting (4.2.2.d) in (4.2.2.b) and solving for T(s), Hopkins obtained a slowly converging series of Bessel functions for the defocus transfer function. Stokseth (10) derived an approximation to Hopkin's exact transfer function of the following form:

$$4.2.2.f. \quad T_A(s) = \begin{cases} z\left(1 - 0.69s + 0.0076s^2 + 0.043s^3\right. \\ \quad \left. x\,[J_1(a - 0.5as)/(a - 0.5as)]\right. & , |s| < 2 \\ 0 & , |s| \geq 2 \end{cases}$$

$A = 4\pi ws/\lambda$, TA(s) is the approximate defocus transfer function, and J1 is a Bessel function of the first kind of order one.

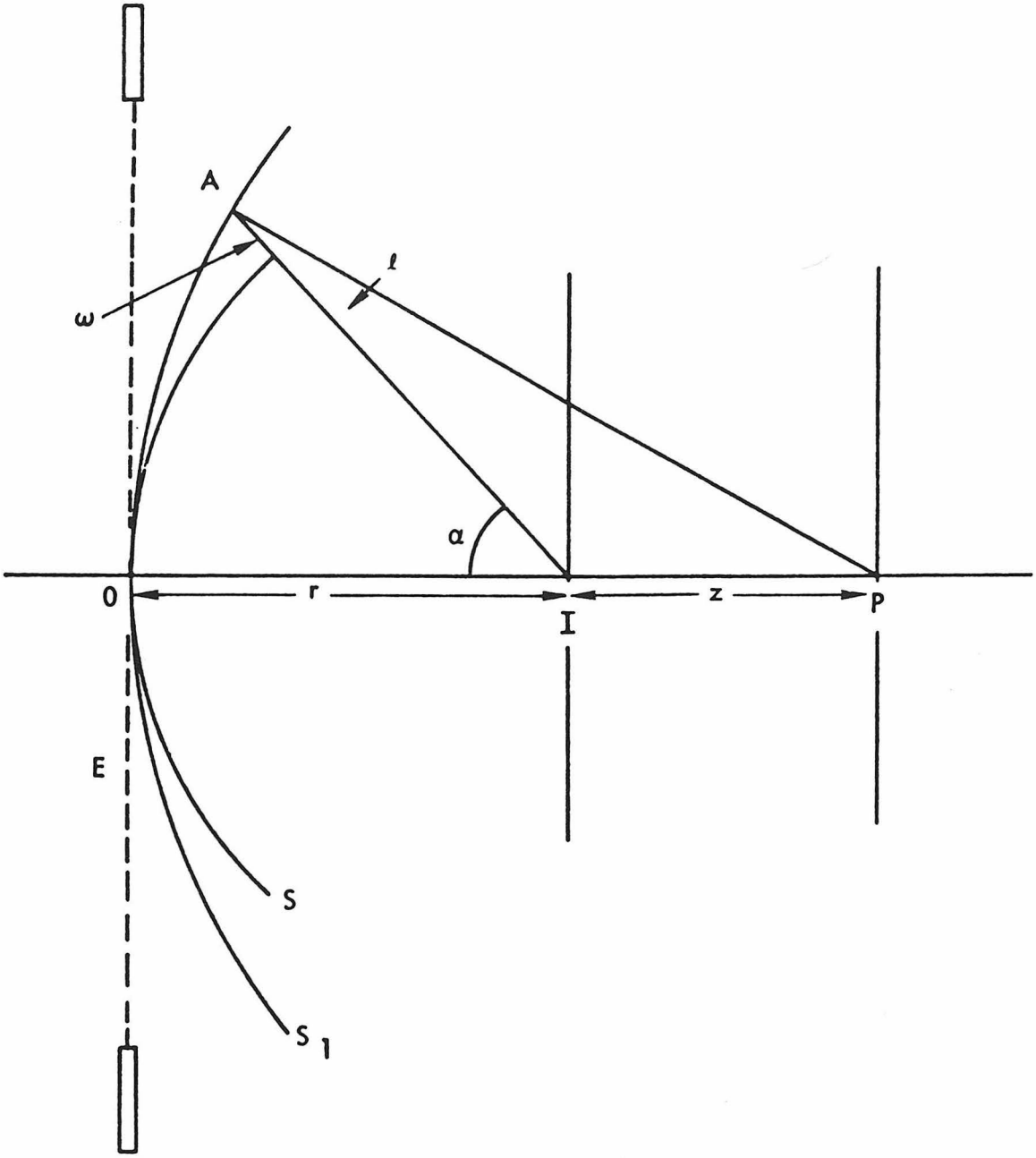


Figure 7. Illustration of parameters dealing with exit pupil of a defocused optical system.

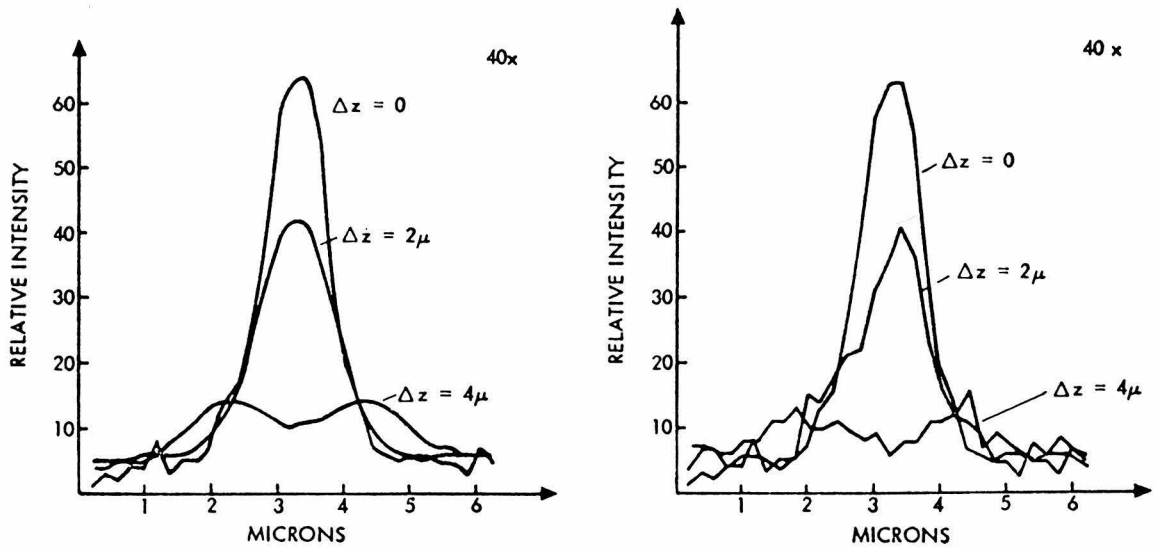
We have applied the Stokseth approximation to determine the defocus transfer functions of the ALMS microscope for the 40x and the 100x objectives. As pointed out by Stokseth the defocus transfer function (DTF) is not symmetrical about the in focus plane. Figure (8) shows the PSFs for various amounts of defect of focus where negative defect of focus is in front of the image plane. The PSFs are convolved with the scanned image of an in focus spot and the results are shown in figure (8 a and c). These functions were computed for the purposes of comparison with our experimentally obtained PSFs.

4.2.3 Experimental Determination of Defocus PSF

We have devised an experimental method for obtaining approximations to the defocus point spread function. This is done in the following way. A sample of Golgi impregnated tissue was mechanically sectioned at .1 micron and a section placed on a slide. A very small round silver granule was located on the slide and was taken to be an approximation to a delta function. This spot was used to obtain approximate experimental defocus PSFs by scanning the spot and digitizing the resultant image at various amounts of defocus. Profiles of the images obtained in this manner are shown in figure (8 b and d) for our 40x and 100x objectives

A. Theoretical Defocus

B. Experimental Defocus



C. Theoretical Defocus

D. Experimental Defocus

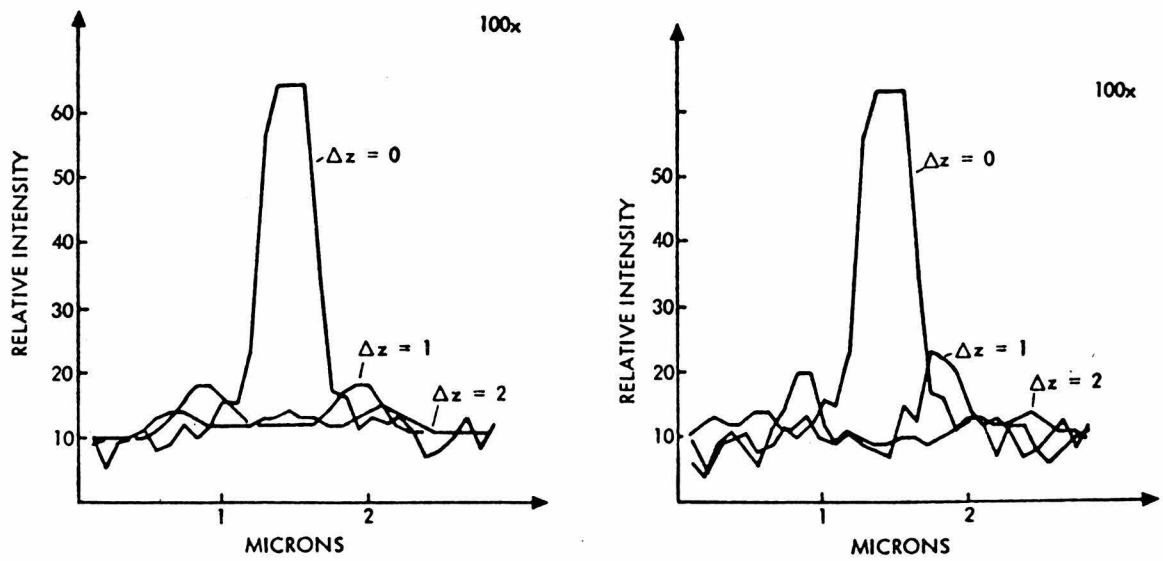


Figure 8. Defocus point spread functions. A and C are the result of convolving the theoretically obtained defocus PSFs with the image of an infocus spot. Δz indicates microns of defocus.

defocused various amounts. It should be noted that the image at 0 defocus distance is the convolution of the spot with the infocus PSF of the microscope which includes aberrations or,

$$4.2.3.a.\quad W_{od}(x,y) = \frac{1}{d + s * PSF_o * PSF_d}$$

where I is the in focus image, S is the spot, PSFo is the PSF of the in focus microscope and PSFd is the defocus PSF, see figure (8). 1/d is the background level and the inverse form of the equation is due to the fact that our grey levels are a measure of inverse transmittance. Figure (8d) the experimentally obtained result for the expression

$$4.2.3.b.\quad W_o(x,y) = \frac{1}{d + S * PSF_o}$$

ie. the in focus spot.

In order to check the validity of our experimentally obtained PSFds we convolved the PSFds obtained from the Stokseth approximation with the scanned image of our in focus spot. The results of this computation were compared with the defocus scanned images and are shown in figure (8ac).

The similarity of these defocus point spread functions indicated that our experimental scanning method was adequate

for generating defocusing filters for our deblurring routines. We have assumed here that the spot which we scanned was a good approximation to a delta function. Although the defocus transfer functions do not obey isoplanarity for finite defocus distance, we shall ignore these deviations and use a spatially invariant filter in the deblurring process.

4.2.4 Deblurring Optical Sections

The purpose of our deblurring algorithm is to transform our original set of scanned optical section images into a new set of section images each of which accurately represents the 2-dimensional distribution of optical density at its given level. Thus we wish to remove all information from each section image which does not represent **in focus** material. The resultant set of 'clean' sections then gives a good 3-dimensional representation of the light microscope specimen, and forms the basis for our 3-D reconstruction procedures. We seek to answer the question ; 'How can knowledge of the defocus transfer functions be used to satisfactorily deblur our section images?'.

If we discretize z in equation 4.2.a we have for the ith optical section

$$4.2.4.a.\quad g(x, y, z_i) = \sum_{j=1}^{n} \int_{-\infty}^{\infty}\int h(\sigma_1, \sigma_2, \Delta z_{i-1}) \, f(x-\sigma_1, y-\sigma_2, z_j)\, d\sigma_1 d\sigma_2$$

Each scanned optical section can be considered to be the summation of an in focus image and a series of defocused images . Since we assume that $h(\sigma 1, \sigma 2, 0) = \delta(\sigma 1, \sigma 2, 0)$ we have

$$4.2.4.b. \quad g(x, y, z_i) = f(x, y, z_i) + \sum_{\substack{j=1 \\ j \neq i}}^{n} \int\int_{-\infty}^{\infty} h(\sigma_1, \sigma_2, \Delta z_{i-j}) f(x-\sigma_1, y-\sigma_2, z_j) \, d\sigma_1 \, d\sigma_2$$

Where g(x,y,zi) is the ith scanned section image containing both focused and defocused information, f(x,y,xi) is the ith 'clean' section image, (ie. the two dimensional distribution of optical density at the ith level with all defocused information removed), and $h(\sigma_1, \sigma_2, \Delta z_{i-j})$ is the defocus point spread function for wavelength and defocus distance equal to the distance between sections g(x,y,zi) and g(x,y,zj). This expression is, of course, only an approximation to the exact expression due to the discrete nature of the data. Each term in the summation represents the contribution of the defocused jth section in the image g(x,y,zi). Expression 4.2.4.a may be rewritten as

$$4.2.4.c. \quad g(x, y, z_i) = f(x, y, z_i) + \sum_{j=i}^{i-1} h(\sigma_1, \sigma_2, \Delta z_{j-1}) * f(x, y, z_j) + \sum_{j=i+1}^{n} h(\sigma_1, \sigma_2, \Delta z_{j-1}) * f(x, y, z_j)$$

where * represents convolution and the summation is separated into two parts. This equation is valid for $1<i<n$.

If we convolve both sides of 4.2.4.c by $h(\sigma 1, \sigma 2, \Delta z_{-1})$ and substitute i-1 for i we have

$$g\left(x, y, z_{i-1}\right) * h\left(\sigma_1, \sigma_2, \Delta z_{-1}\right) = f\left(x, y, z_{i-1}\right) * h\left(\sigma_1, \sigma_2, \Delta z_{-1}\right)$$

4.2.4.d.
$$+ \sum_{j=1}^{i-2}\left[f\left(x, y, z_j\right) * h\left(\sigma_1, \sigma_2, \Delta z_{j-i+1}\right)\right] * h\left(\sigma_1, \sigma_2, \Delta z_{-1}\right)$$
$$+ \sum_{j=i}^{n}\left[f\left(x, y, z_j\right) * h\left(\sigma_1, \sigma_2, \Delta z_{j-i+1}\right)\right] * h\left(\sigma_1, \sigma_2, \Delta z_{-1}\right)$$

Substituting i+1 for i-1 in 4.2.4.d we have

$$g\left(x, y, z_{i+1}\right) * h\left(\sigma_1, \sigma_2, \Delta z_{+1}\right) = f\left(x, y, z_{i+1}\right) * h\left(\sigma_1, \sigma_2, \Delta z_{+1}\right)$$

4.2.4.e.
$$+ \sum_{j=1}^{i}\left[f\left(x, y, z_j\right) * h\left(\sigma_1, \sigma_2, \Delta z_{j-i-1}\right)\right] * h\left(\sigma_1, \sigma_2, \Delta z_{+1}\right)$$
$$+ \sum_{j=i+2}^{n}\left[f\left(x, y, z_j\right) * h\left(\sigma_1, \sigma_2, \Delta z_{j-i-1}\right)\right] * h\left(\sigma_1, \sigma_2, \Delta z_{+1}\right)$$

By extracting terms containing $f(x,y,z_{i+1})$ and $f(x,y,z_{i-1})$ from the summations in equation 4.2.4.c we have

$$g\left(x, y, z_i\right) = f\left(x, y, z_i\right) + f\left(x, y, z_{i-1}\right) * h\left(\sigma_1, \sigma_2, \Delta z_{-1}\right)$$

4.2.4.f.
$$+ f\left(x, y, z_{i+1}\right) * h\left(\sigma_1, \sigma_2, \Delta z_{+1}\right)$$
$$+ \sum_{j=1}^{i-2} f\left(x, y, z_j\right) * h\left(\sigma_1, \sigma_2, \Delta z_{j-i}\right)$$
$$+ \sum_{j=i+2}^{n} f\left(x, y, z_j\right) * h\left(\sigma_1, \sigma_2, \Delta z_{j-i}\right)$$

Subtracting 4.2.4.d and 4.2.4.e from 4.2.4.f we obtain 4.2.4.g.

$$
\begin{aligned}
& g(x, y, z_i) - g(x, y, z_{i-1}) * h(\sigma_1, \sigma_2, \Delta z_{-1}) \\
& \quad - g(x, y, z_{i+1}) * h(\sigma_1, \sigma_2, \Delta z_{+1}) \\
= & \; f(x, y, z_i) - \sum_{j=1}^{i} f(x, y, z_j) * h(\sigma_1, \sigma_2, \Delta z_{j-i-1}) * h(\sigma_1, \sigma_2, \Delta z_{+1}) \\
& \quad - \sum_{j=i}^{i} f(x, y, z_j) * h(\sigma_1, \sigma_2, \Delta z_{j-i+1}) * h(\sigma_1, \sigma_2, \Delta z_{-1}) \\
& \quad - \sum_{j=i}^{n} f(x, y, z_j) * h(\sigma_1, \sigma_2, \Delta z_{j-i+1}) * h(\sigma_1, \sigma_2, \Delta z_{-1}) \\
& \quad - f(x, y, z_j) * h(\sigma_1, \sigma_2, \Delta z_{j-i}) \\
& \quad - \sum_{j=i+z}^{n} f(x, y, z_j) * h(\sigma_1, \sigma_2, \Delta z_{j-i-1}) * h(\sigma_1, \sigma_2, \Delta z_{+1}) \\
& \quad - f(x, y, z_j) * h(\sigma_1, \sigma_2, \Delta z_{j-i})
\end{aligned}
$$

The right hand side of equation 4.2.4.g contains the 'clean' image f(x,y,zi) minus 4 terms which represent greatly defocused information. The subtraction of highly defocused information from our desired image is a form of high pass filtering which, in our case, is not particularly undesirable.

From the above discussion we have an algorithm for removing defocused information when high pass side effects are tolerable. We take the left hand side of 4.2.4.g as our approximation of the distribution of optical density at the ith level in our Golgi object. With appropriate scaling our algorithm becomes

$$4.2.4.h. \qquad f^*(x, y, z_i) = .8\Big[g(x, y, z_i) - \Big(g(x, y, z_{i-1}) * h(\sigma_1, \sigma_2, \Delta z_{-1}) + g(x, y, z_{i+1}) * h(\sigma_1, \sigma_2, \Delta z_{+1})\Big)/2\Big]$$

An example of this deblurring process is shown in figure (9), where A is the g(x,y,zi) and B is the corresponding f*(x,y,zi). Notice that we have removed defocused information and high pass filtering has occurred.

The set of processed images f*(x,y,zi) can now be used in 3-dimensional reconstruction. Each image represents a high pass filtered version of the 2-D optical density distribution at a given level of focus, and the entire set of optically sectioned and 'cleaned' section images represents an image of the 3-D distribution of optical density in our specimen. An example of the top and side views thus created is shown in figure (11).

The above deblurring algorithm has some rather serious limitations. Digital filtering in the spatial domain is

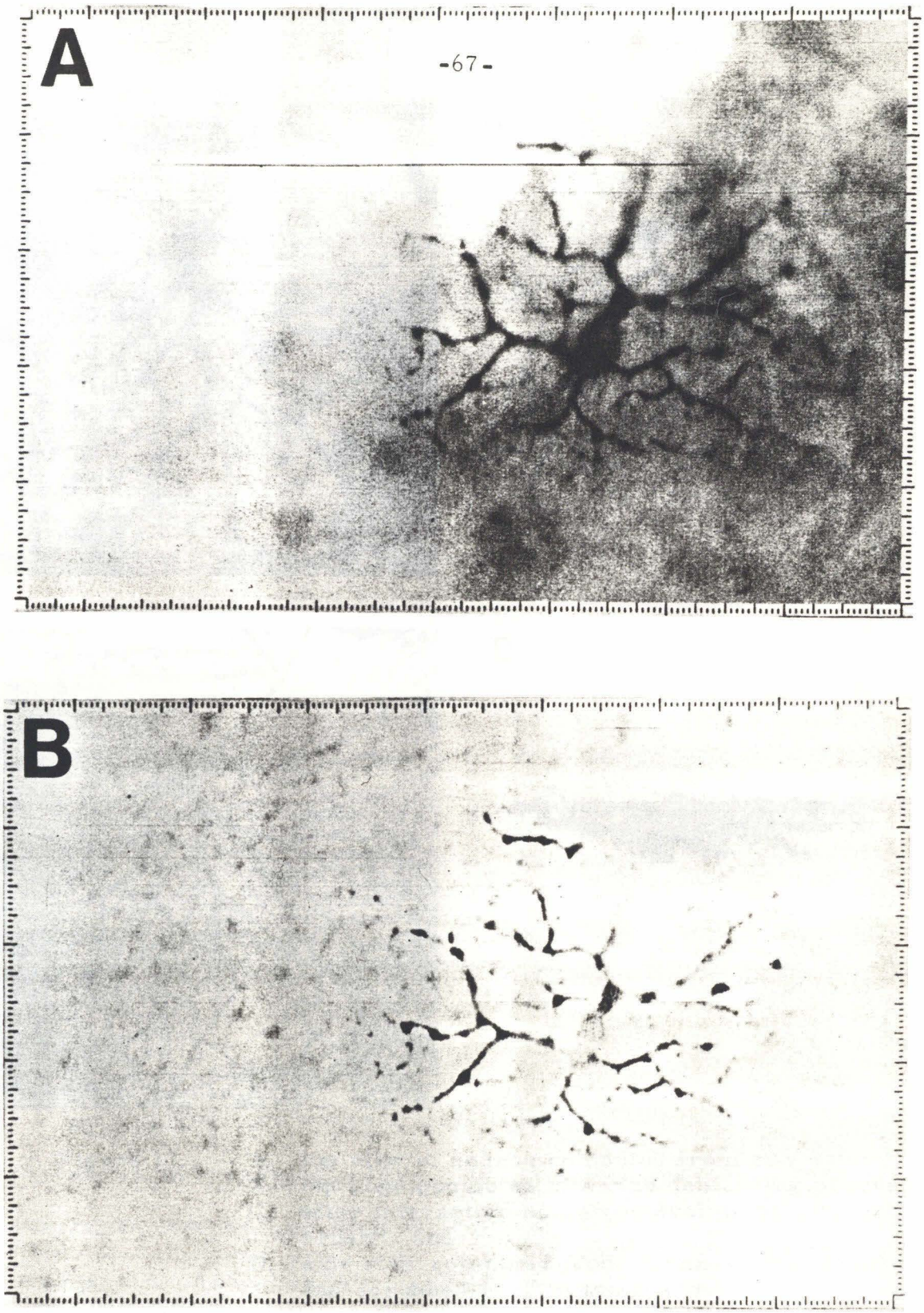


Figure 9. A, an optical section image; B, deblurred version.

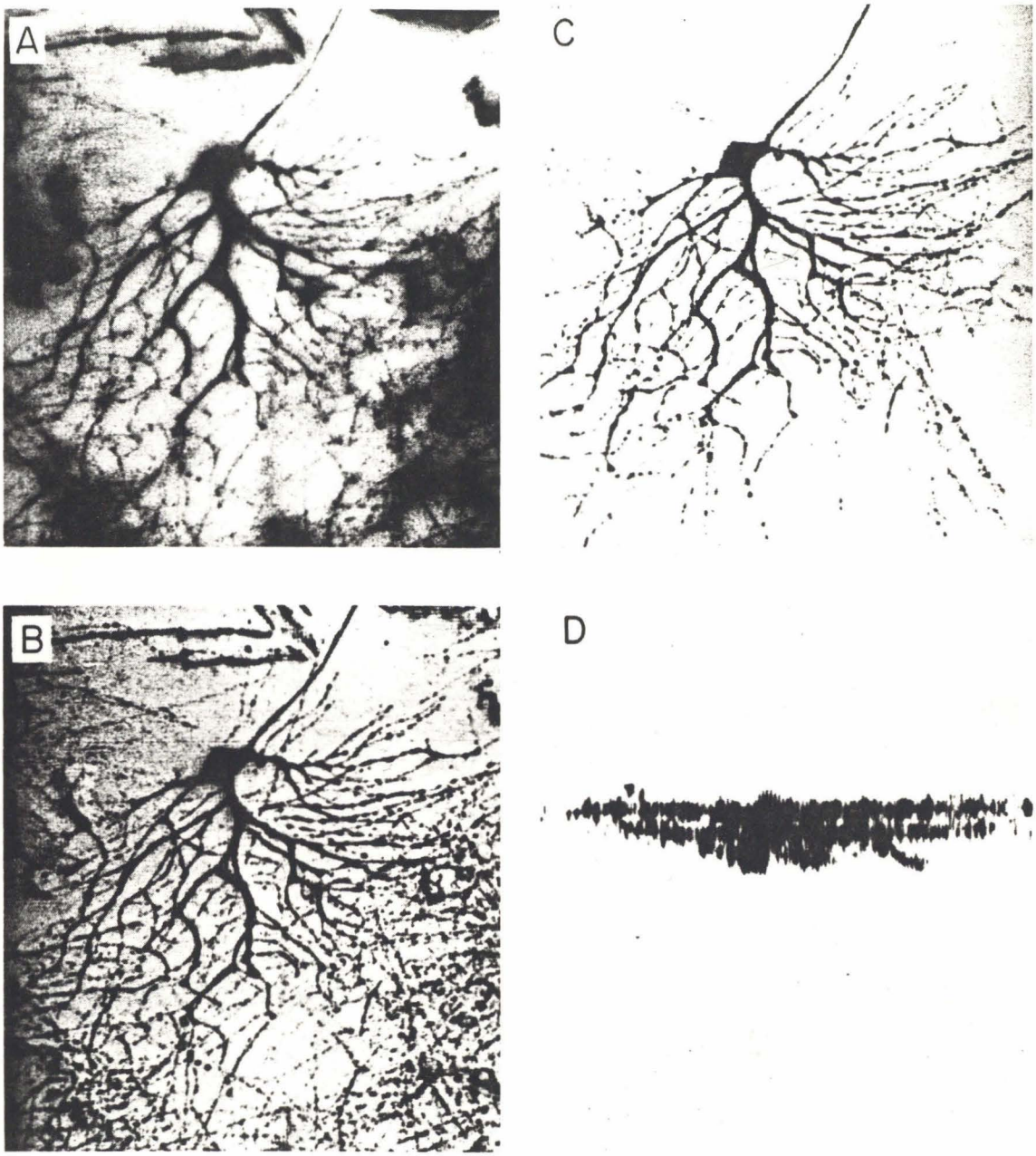


Figure 11.
A. top view of neuron computed from raw data.
B. top view computed after the deblurring process.
C. after interactive noise removal using a "paint brush", and
D. side view computed from the resulting cleaned and scrubbed section images.

computationaly costly when applied to images containing 250,000 pixels. For a neuron which is represented by 15 images of 512 lines each and 512 pixels per line this deblurring algorithm is a computational disaster. This is especially true in view of the fact that neuronal images scanned at high power such that the microscope depth of focus is small, are not seriously encumbered by defocused information. In short, until the 3-dimensional deconvolution operation can be hardware implemented to operate quickly and efficiently on huge data sets, it is probably better to solve the problem by going to a high power objective with short depth of focus and tolerating a certain amount of defocused information in the result. The later approach may necessitate obtaining several sets of images to fully cover an object too large to fit in the field of view at high power, but the additional computation of fitting these sets together to form a single 3-D representation may still be less than that of deblurring.

4.3 Pattern Matching Filters

The ability to extract or recognize certain simple patterns in an image is useful in many applications (9). We may wish to extract an object from noise in situations where we know certain spatial frequency information about the object. In this case we may directly apply the matched

filter approach described below. We may wish to extract patterns to be used as basic symbols in a low level grammar for vision (7). In this case we must decide at which point in the initial processes of vision does the image processing change from transformational operations to symbolic manipulation. We may wish to implement a low level image processor which utilizes certain linear filters to produce the primitives in a grammar for vision. The construction of such grammars is discussed in chapter 5.

4.3.1 Mathematical Details

Papoulis (8) uses an application of the Schwarz' inequality to show how a linear spatial system can be built to respond maximally to certain specified patterns. He considers the band limited image f(x,y) having Fourier transform F(w1,w2) where F(w1,w2)=0 for w1 >wo or w2 >wo. with energy E given by

$$4.3.1.a. \quad E = \int_{-\infty}^{\infty}\int |f(x,y)|^2 \, dx\, dy = \frac{1}{4\pi^2} \iint_R |F\ \omega_1, \omega_2|^2 \, d\omega_1\, d\omega_2$$

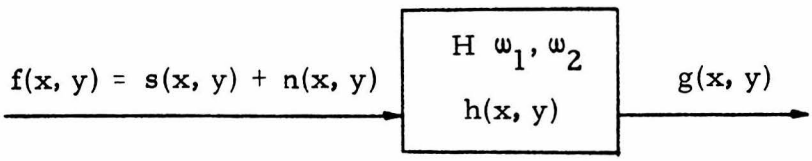
and a linear system with impulse response h(σ1,σ2) and system function H(w1,w2). The output from system h(σ1,σ2) given input f(x,y) is

$$4.3.1.b. \quad g(x,y) = \frac{1}{4\pi^2} \iint_R F\left(\omega_1, \omega_2\right) H\left(\omega_1, \omega_2\right) e^{j\left(\omega_1 x + \omega_2 y\right)} \, d\omega_1\, d\omega_2$$

The problem is to find that system H which maximizes the output g for a given input pattern f(x,y). Using the Schwarz' inequality he finds the desired system is given by

$$4.3.1.c. \quad H^*(\omega_1, \omega_2) = F_o(\omega_1, \omega_2)/Ae^{-i(\omega_1 x_1 + \omega_2 y_1)}$$

where A is a constant and * indicates complex conjugate



Davenport and Root (2) describe the use of matched filters to maximize signal to noise ratio in signal detection problems. We may use the same argument for detecting patterns in an image containing noise. The input image f(x,y) is composed of a signal s(x,y) plus noise n(x,y)

$$4.3.1.d. \quad f(x, y) = s(x, y) + n(x, y)$$

We wish to specify a linear filter which will act on f(x,y), where n(x,y) is a sample function from a wide sense stationary random process, so that the output signal to noise ratio

4.3.1.e.
$$\left(\frac{S}{N}\right)_o = \frac{s_o^2(t)}{E\left[n_o^2(t)\right]}$$

is a maximum at some position x1,y1. The linear filter

4.3.1.f. $$H_1(\omega_1, \omega_2) = K_2 S^*(\omega_1, \omega_2)$$

is shown (2) to satisfy this condition if the noise is white noise and K2 is a proportionality constant. Thus the optimum filter is given by the conjugate of the signal transform.

For the case of nonwhite noise Urkowitz (11) the optimum filter, in the sense of maximizing the signal to noise ratio, is called a 'nonwhite' noise (NWN) matched filter. For this case our optimum filter is given by

4.3.1.g. $$H(\omega_1, \omega_2) = \frac{KS^*(\omega_1, \omega_2)\exp\left(-i(\omega_1 x_1 + \omega_2 y_1)\right)}{N(\omega_1, \omega_2)}$$

the factor exp(-i(w1x1+w2y1)) is a spatial shift factor hence, except for this shift, the optimum filter has a transfer function given by the conjugate of the signal spectrum divided by the power density spectrum of the noise.

All of these considerations have assumed that we know the position where the signal or pattern appears. If this is not known it is necessary to search the entire image

looking for the location of the pattern. The problem of deciding at each point of shift whether we have an occurrence of a pattern or not is a problem of statistical decision theory. In our case we build a filter to detect a particular pattern, shift it around on the image producing an output at each shift given by

$$4.3.1.h. \quad g(x, y) = K \iint_{\text{image}} s(\sigma_1, \sigma_2) h(x-\sigma_1, y-\sigma_2)\, d\sigma_1\, d\sigma_2$$

This representation treats the nonwhite noise matched filter as a linear functional and is sometimes referred to as an integrate and dump operation or a correlator embodiment of the matched filter.

Hence, given a filter designed to detect a certain pattern or feature, we cross correlate the image and the filter producing a number for each degree of shift. We must decide on the basis of this number whether the feature is present at this location. This is a problem in statistical decision theory. If we know the probability density function of $g(x,y)$, given the absence of the feature and the density function in the presence of the feature, then we can choose a decision surface or a threshold which minimizes our risk of error.

We will not diverge here to explain the standard methods of statistical decision theory. Essentially, for

any given detection problem the probability of false detection and the probability of false dismissal are computed and the total error is taken as the weighted sum of these. We then pick a decision surface such that this error is minimized.

4.3.2 Application to Neuronal Image Enhancement

In our case we have images of neurons and we wish to enhance the neuronal object while subduing the noise. We notice that a given neuron has processes which are of a more or less consistent size; in other words, the neuron is made of units which are similar. We may detect and enhance these units through use of filters. Circular filters with positive weighting in the center and negative weighting in the surround are correlated with the image as in equation 4.3.1.h. The filter is designed to react maximally when it is located over a neuronal process. Figure (12) shows the result of this operation. Since these filters are bandpass in nature and are tuned to react to neuronal processes, the larger cell body shows the effect of high pass filtering.

We used this process to obtain enhanced top and side views of neurons in the following way. First we compute the top view (TOP) of the original data by taking the pixel by pixel maximum down through a stack of images. Then we apply a

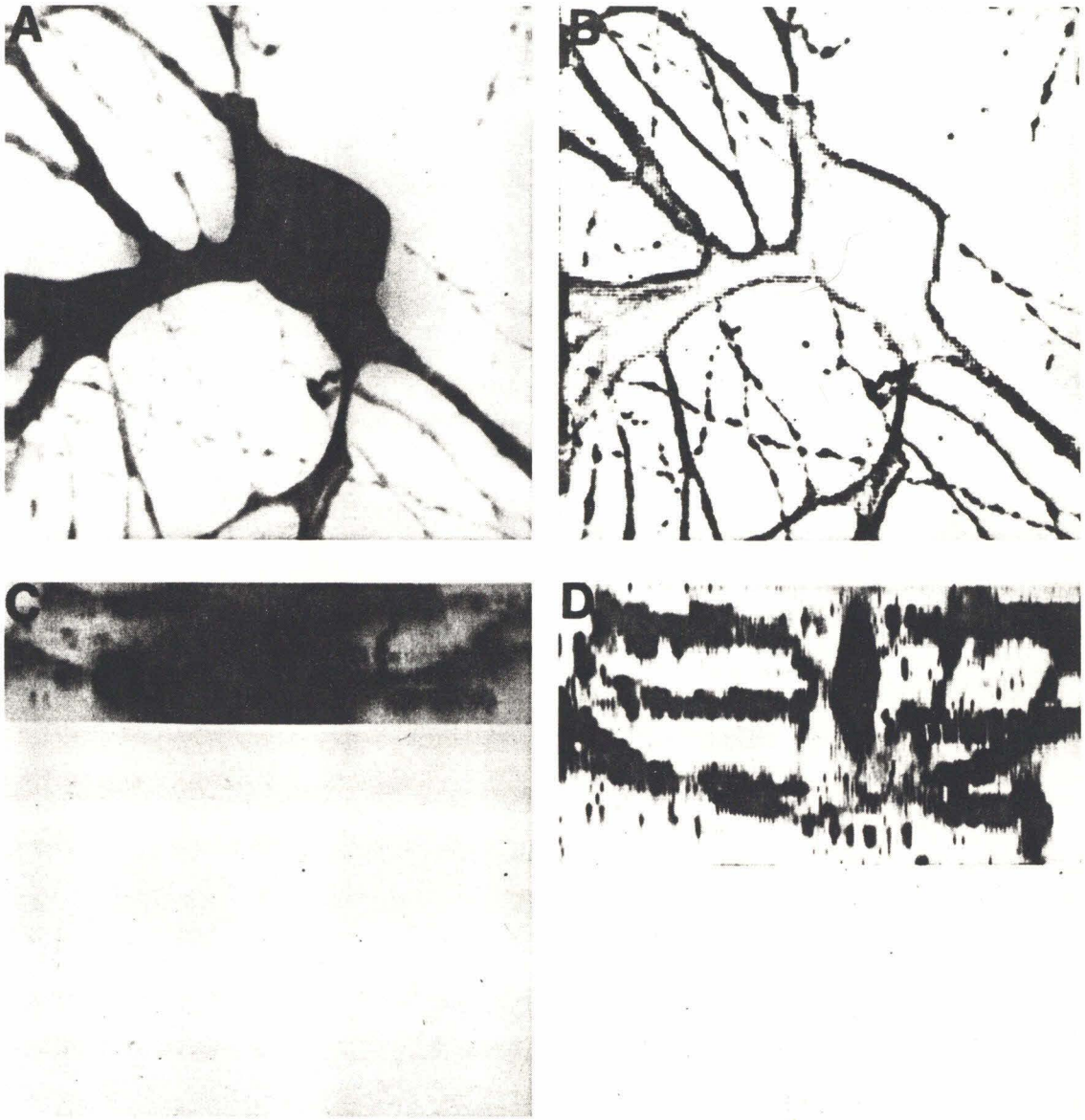


Figure 12. A, top, and C, side views, computed from optical sections. B and D show top and side views after application of pattern matching filters; see text.

pattern matching filter to this top view and using an interactive paint brush (see chapter 3) routine we selectively erase the remaining noise from the enhanced top view (ETOP). We subtracted ETOP from TOP producing an image of pure noise (PN). This PN image is subtracted from each image in the original stack which removes all noise except that lying directly above or below a part of the neuron. Side views computed from this clean stack are much improved (figure 12).

Although the above procedures are adequate for producing side and top views for the extraction of certain structural parameters they do not provide the kind of complete structural description which is needed for quantifying a more complete structural characterization. Also the variety of noise in neuronal images, namely, random Golgi noise, intruding processes from other neurons, variations in background intensity, make it imperative that the human view the data and make decisions. It has been our experience that all the above methods of image enhancement are unnecessary if the human can view the original data in a 3-dimensional representation such as stereo pairs. The human has a marvelous ability to ignore defocused information and to decide which pieces belong to the neuron of interest and which do not. For these reasons we have pursued a completely interactive route to get from the data

to a complete structural description. This method is based on stereo pairs computed from the original data.

4.4 Stereo Pair Construction

As described in 4.1 our basic structural data from Golgi impregnated neurons seen in light microscope preparation is on magnetic tape as a stack of digital images. This stack of images represents the 3-dimensional raw data. We wish now to compute stereo pairs from this data, display them on our interactive image display device and use a 3-dimensional cursor to interactively extract a structural description of the neuron.

We need a simple stereo pair algorithm which does not produce distance perspective because this would introduce errors in measurements taken from the 3-dimensional space defined by the stereo pair. For the left eye image we 'tilt' the stack of images to the right and compute the maximum pixel by pixel value through the shifted stack. The ith image in the tilted stack is given by

$$\text{tiltPi}(x,y)=\text{originalPi}(x-i\Delta z\tan\theta,y)$$

where Δz is the distance between optical section images in microns, θ is the angle between the normal to the original stack and the line connecting the left eye to the center of the original pic0(x,y). A more complete description of stereo pair algorithms is given in (12).

Having computed the stereo pair for a neuron we are in a position to describe an interactive method for extracting a syntactic description of neuronal morphology.

REFERENCES

1. Arnheim, R., (1969), Visual Thinking, Faber and Faber Limited, London p.52.

2. Davenport, W. B., and W. L. Root, (1958), An Introduction to the Theory of Random Signals and Noise, McGraw-Hill, New York, p. 244ff.

3. Franklin, J. N., (1970), 'Well-posed stochastic extensions of ill-posed linear problems,' J. Math. Anal. and Appl., 31: 682-716.

4. Goodman, J. W., (1968), Introduction to Fourier Optics, McGraw-Hill Publishers, N. Y.

5. Hopkins, H. H., (1955), 'The frequency response of a defocused optical system,' Proc. Roy. Soc., Lond. a 231: 91-103.

6. Hunt, B. R., (1972), 'A theorem on the difficulty of numerical deconvolution,' IEEE Trans. Audio Electroacoustics, AU-20: 94-95.

7. Marr, D., (1974), 'The low-level symbolic representation of intensity changes in an image,' Artificial Intelligence Laboratory Memo, Mass. Inst. of Technology, Cambridge, Mass.

8. Papoulis, A., (1968), Systems and Transforms with Applications in Optics, McGraw-Hill, New York p. 189ff.

9. Rosenfeld, A., (1969), Picture processing by computer,

Academic Press, New York.

10. Stokseth, P. A., (1969), 'Properties of a defocused optical system,' J. Optical Soc. Amer., 59: 1314-1321.

11. Urkowitz, H., (1969), An introduction to signal processing, (in press), General Atronics Corp., Phila., Pa. p.8.15ff.

12. Weinstein, M. and K. R. Castleman, (1971), 'Reconstructing 3-D specimens from 2-D section images,' Proc. SPIE 26: 131-138.

"The process of science is conceived as the finding of that formal language that best expresses the available experimental evidence"

D. Randall (8).

5. DESCRIPTIONS OF NEURONAL STRUCTURE

In order to explicate the relationship between structure and function in a biological system one must first obtain a description of structure and function in an appropriate language. This descriptive language should facilitate 1) the classification of structural and functional units into types, 2) the mapping of structural types to functional types, 3) the examination of the functional significance of substructures and 4) construction of a unified theory which incorporates both structural and functional information. The aim of this chapter is to develop a specialized formalism which satisfies the above criteria. It is thought that a tool for incorporating varied information into a unified model of the retina will allow the investigator to induce constraints on the set of possible form-function models. With the scattered data available on complex neuronal systems the number of possible models is very great. As Randall points out in his thesis (8):

"Each new observation he (the scientist) makes, which is consistent with his previous experience, delimits a smaller set of models or possible configurations of the universe."

Here, "model" is used in the formal set theoretic sense. The point to be made, in our case, is that one should have a language in which observations of light microscope structure, EM ultrastructure, and functional synaptic properties, can be stated and used to reduce the set of form-function models to the "true" model. This of course is a massive task and we can only hope to make some progress in the right direction. We begin this chapter by describing neuronal structure and later show how functional information can be included.

5.1 A Grammar for Describing Neurons

We wish to find a concise, complete descriptor for the **structure** of a neuron. This descriptor should be generative in the sense that we should be able to reconstruct a reasonably accurate neuron from this descriptor, and it should be concise in the sense that the amount of information required to represent the descriptor is small. These considerations indicate that we should use a specialized language for neuronal structure. We notice that neurons are branching tree-like structures having roundish

cell bodies and more or less cylindrical processes. These observations suggest a tree grammar.

Tree grammars have been found to be useful tools in syntactic pattern recognition work (2,3,4). The sentences generated or recognized by a tree grammar are tree structures. To parse such a tree sentence one uses a set of production rules which are themselves written in terms of tree structures. We will follow the description given by Tou and Gonzalez (9). A tree grammar is defined as a quintuple

$$G=(VN,VT,P,R,S)$$

where VN is the set of nonterminals and VT the set of terminals in the grammar. S is the start symbol which is normally a tree; in our case it represents a neuron. P is a set of productions of the form a->b, where a and b are trees. R is a ranking function which denotes the number of direct descendants of a node which is an element of VT.

For example we can construct a simple grammar for neurons. G1=(VN,VT,P,R,S)

VN = (<subtree>)

VT = (s,b,t,p)

where

s = [soma, location, fatsize, thinsize]

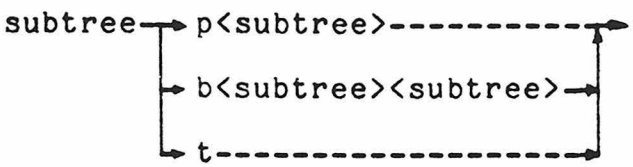
b = [branch, location]

t = [terminator, location]

p = [process, start location, end location,
start diameter, end diameter]

The production rules P are:

S→s<subtree>→



R(s) = 1

R(b) = 2

R(t) = 0

R(p) = 1

Here we have specified the various syntactic pieces of a neuron to be soma, process, branch and terminator. From these pieces we can build neurons using the production rules. We shall refer to the nonterminals and terminals in the grammar as nodes. The soma node contains information about the location of the neuron's soma in 3-dimensional coordinates (x,y,z), and also contains two distances - fatsize and thinsize. Fatsize is the maximum distance across the soma and thinsize is the minimum distance across

the soma. A branch node contains the coordinates of the branch location. A process node contains a truncated cone approximation to a segment of a neuronal process. It specifies the starting and ending diameters. A terminator node has three coordinates specifying its location.

The first production rule should be read: S is a soma connected to a subtree. The second rule reads: a subtree is either a process connected to a subtree, or a branch connected to two separate subtrees or a terminator. The fact that this grammar generates binary trees is seen from the ranking function. A terminator node has no direct descendants in the tree. A soma and a process have one direct descendant, and a branch has two. Since no node has more than two direct descendants, the tree is a binary tree. The simple schematic neuron of figure (13a) is represented by the binary tree of figure (13b).

The above grammar is clearly capable of recognizing any neuron in the frog retina for which the primitives can be collected and in this sense it provides an adequate descriptive tool for our task. To extract features from this tree representation of a neuron it is necessary simply to traverse the tree using a standard algorithm (5) and to perform some operation at each node to accumulate a parameter representing some global feature.

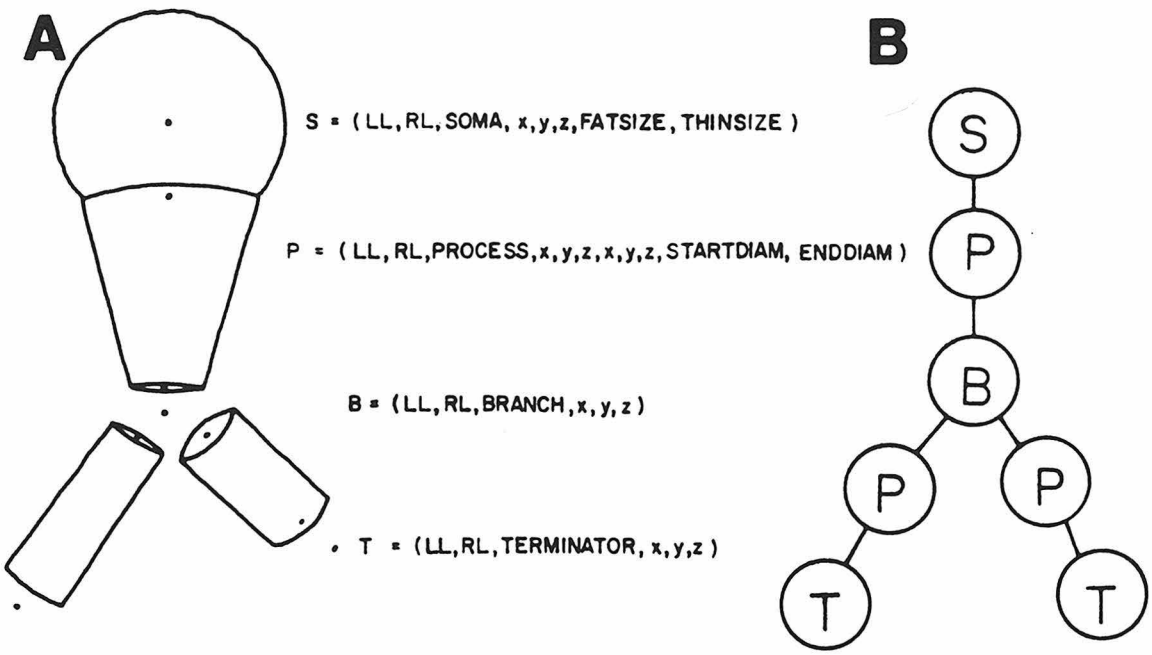


Figure 13. A is a simple schematic neuron with a soma, three processes, one branch and two terminators. B is the binary tree representation. For nodes of type s and p only the left link is used in connecting the tree; for nodes of type b both the left link and the right link point to other nodes and for type t the left and right links are not used.

A number of such parameters can be computed and used in a cluster analysis scheme for classification purposes.

If we wish to define substructures of neurons and to partition a population of neurons on the basis of occurrences of substructural units we may require a more flexible grammar. We introduce into our grammar an additional set of nonterminals to give the following grammar.

$$G2 = (VN, VT, P, R, S)$$

where

VN = (bitree, monotree, diffusetree, subtree,
bistrat, monostrat)

VT = (s,sp,dp,b,t)

where

s = [soma, x, y, z, fatsize, thinsize, synapse]

sp = [process such that $z1 \cong z2$, location,
startsize, endsize, synaptic information]

dp = [process such that $z1 \ncong z2$, location,
startsize, endsize, synaptic information]

b = [branch, x, y, z]

t = [terminator, x, y, z]

R(s) = 1

R(p) = 1

R(b) = 2

R(t) = 0


P1 neuron→s<subtree>→
P2 subtree→→<bitree>
→<monotree>
→<diffusetree>
→b<subtree><subtree>
→dp<subtree>
→sp<subtree>
P3 bitree →→dp<bistrat>
→<bistrat>
P4 monotree→→dp<monostrat>
→<monostrat>
P5 diffusetree→dp-t
→b<diffusetree><diffusetree>
→dp<diffusetree>
P6 bistrat→→b<dp<monostrat>><monostrat>
→b<bistrat><bistrat>
→sp<bistrat>
P7 monostrat→→sp-t
→b<monostrat><monostrat>

This is an ambiguous grammar in that a given neuron may have no unique parse. Our main interest here is not so much to parse neurons but rather to provide a tool for partitioning our population of neurons by occurrences of substructures. This can be done with a partial parse as described in 6.2. Notice that the terminals in the language can carry information about synapses. It is thus possible to use this representation to accumulate ultrastructural information from EM studies. Similarly one could attach functional information associated with the synaptic attributes.

We have specified a grammatical tool which can be used to conveniently represent neuronal structure, to form a basis for feature extraction and classification and to allow a partitioning of our set of neurons by occurrence of substructures. We shall now treat our method for obtaining the low level, or primitive descriptors of neuronal parts from our stereo pair data.

5.2 Interactive Extraction of Primitives

Ideally we would like to have an algorithm which could look at the stereo pair of a neuron and present us with a binary tree representation of the neuronal structure. With

current state of the art in pattern recognition, the only existing such algorithm is in the human visual system. It has a high resolution scanner and low level transformer-parser, and can apply a large body of knowledge about what neurons are like to the resolution of structural ambiguities and to the separation of neuron from noise. It is good at using binocular disparity depth cues to specify 3-dimensional structure; it can put together a neuronal process, even if the Golgi stain is spotty and broken, by applying information from a large region of 3-dimensional space.

5.2.1 3-Dimensional Cursor

Our interactive extraction technique is as follows. We display the stereo pair computed by the algorithm of 4.4 on the color display so that the left image is red and the right image is superimposed in green. Each image thus covers the entire 512x512 element display face. This display is viewed through green-red glasses so that the right eye sees the left image and the left eye sees the right image - this is done so that the 3-dimensional space appears in front of the screen instead of behind the screen. A joy stick is available for moving a spot of light around in the space. Moving the joystick left-right and forward-backward controls the x,y position of the spot and

sliding the joystick in and out controls the z position; see figure (14).

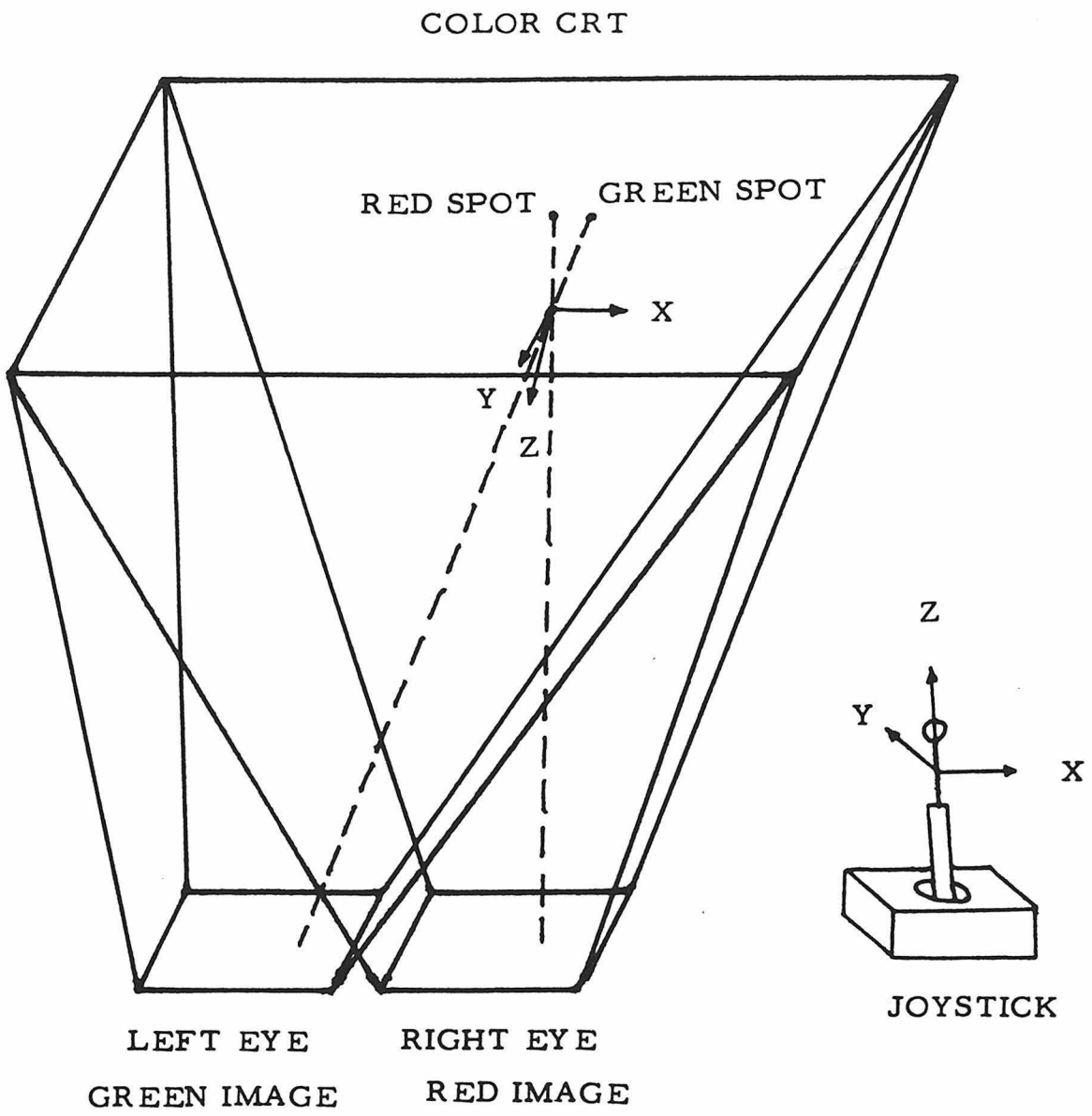


Figure 14. Joystick movement causes the red and green spots to move so that the apparent position of the cursor moves about in the 3-D space defined by a stereo pair. Neurons can thus be displayed and recorded in three dimensions.

The cursor is implemented in software by placing a spot at x-kz,y in the left image and at x+kz,y in the right image. This causes the spot to appear at x, y, z in the space defined by the stereo pair. We calibrate the cursor for each neuron so that the x,y coordinates read from the cursor are in microns and the z coordinate is in normalized microns. The z dimension is normalized so that for all neurons the distal side of the inner plexiform layer (DIPL) is at 100 microns in normalized coordinates and the proximal side is at 150.

Using the joystick and a set of computer sensed toggle switches we record a list of nodes in the following way. First a name node is built for the purpose of cell identification. Then a soma node is made by using the cursor to point to the position of the soma and to measure the fatsize and thinsize parameters as in figure (13). Process nodes, branch nodes and terminal nodes are added to the nodelist by using the cursor to specify their position and to take measurements. When each node is completed it is indicated by the computer on the stereo pair in 3-dimensions. Thus one can always see which part of the neuron has been recorded.

Figure (15a) shows an original stereo pair, B shows the stereo pair after recording, illustrating how the program indicates which nodes have been recorded, and figure (16)

shows the skeletons of two cells plotted from the resulting syntactic descriptor.

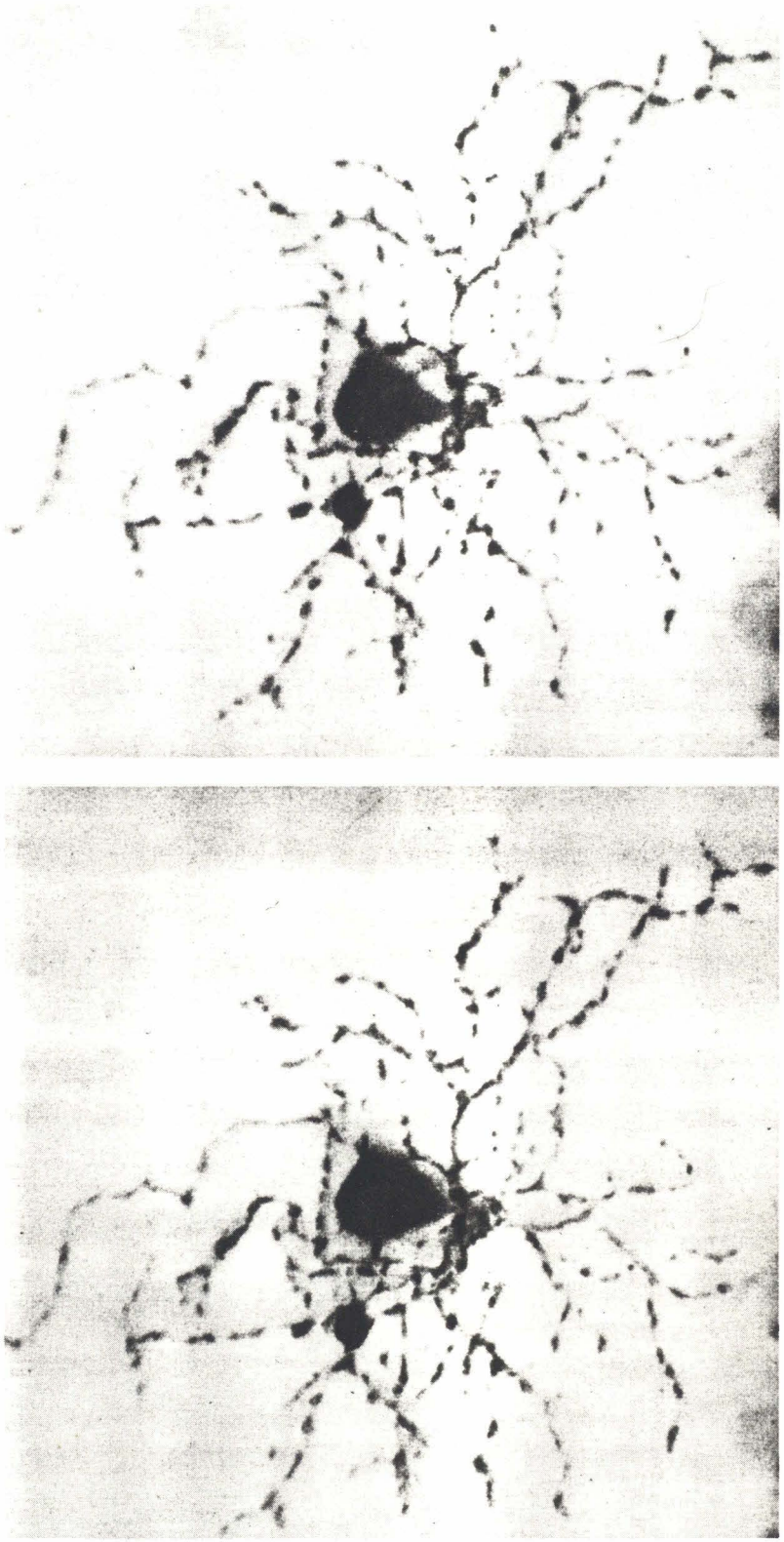


Figure 15a. A stereo pair of a neuron.

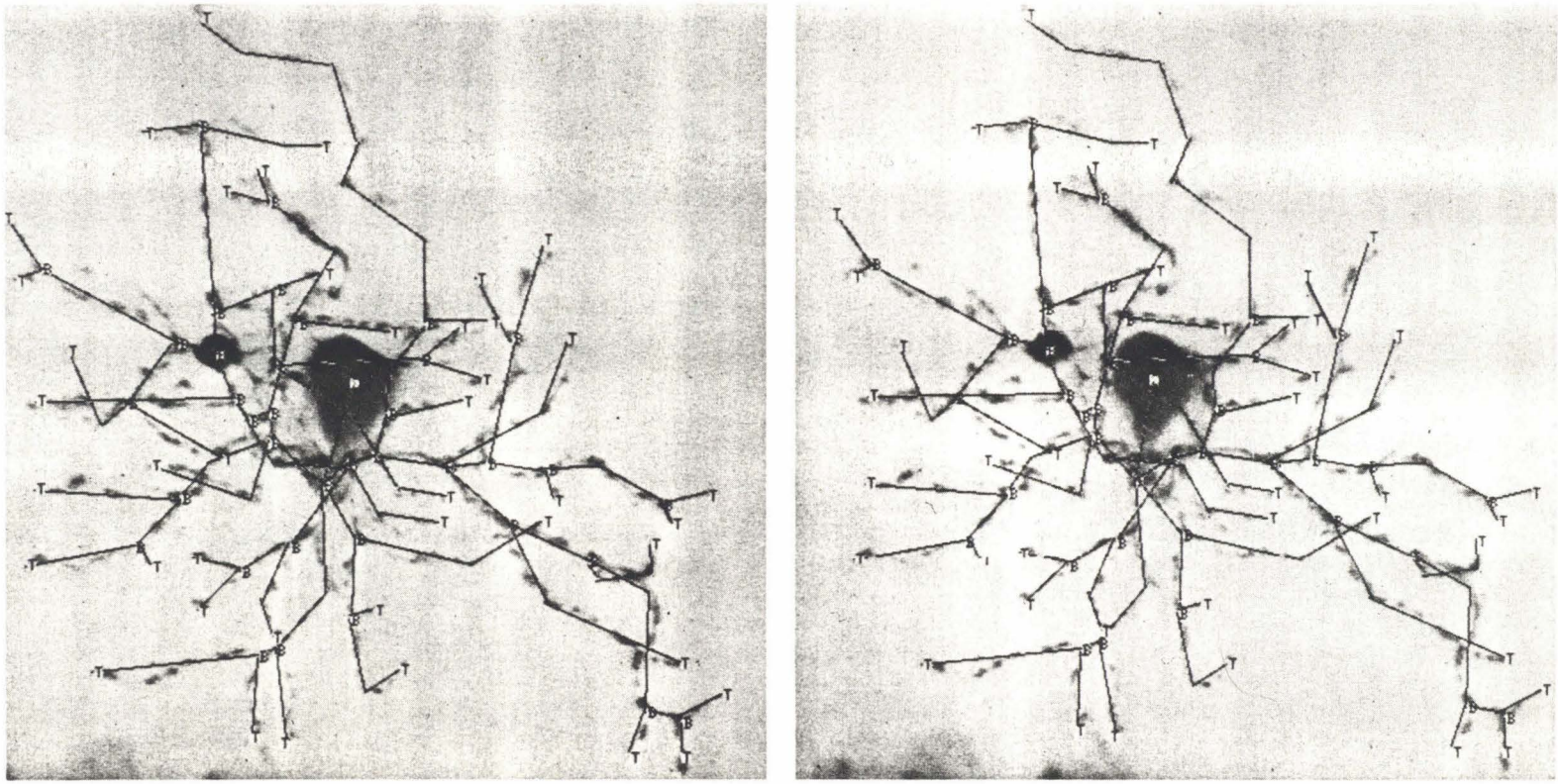


Figure 15b. Stereo pair of the neuron of 15a. After recording of the primitive elements showing how the computer indicates which parts have been recorded by placing lines and letters in three-space.

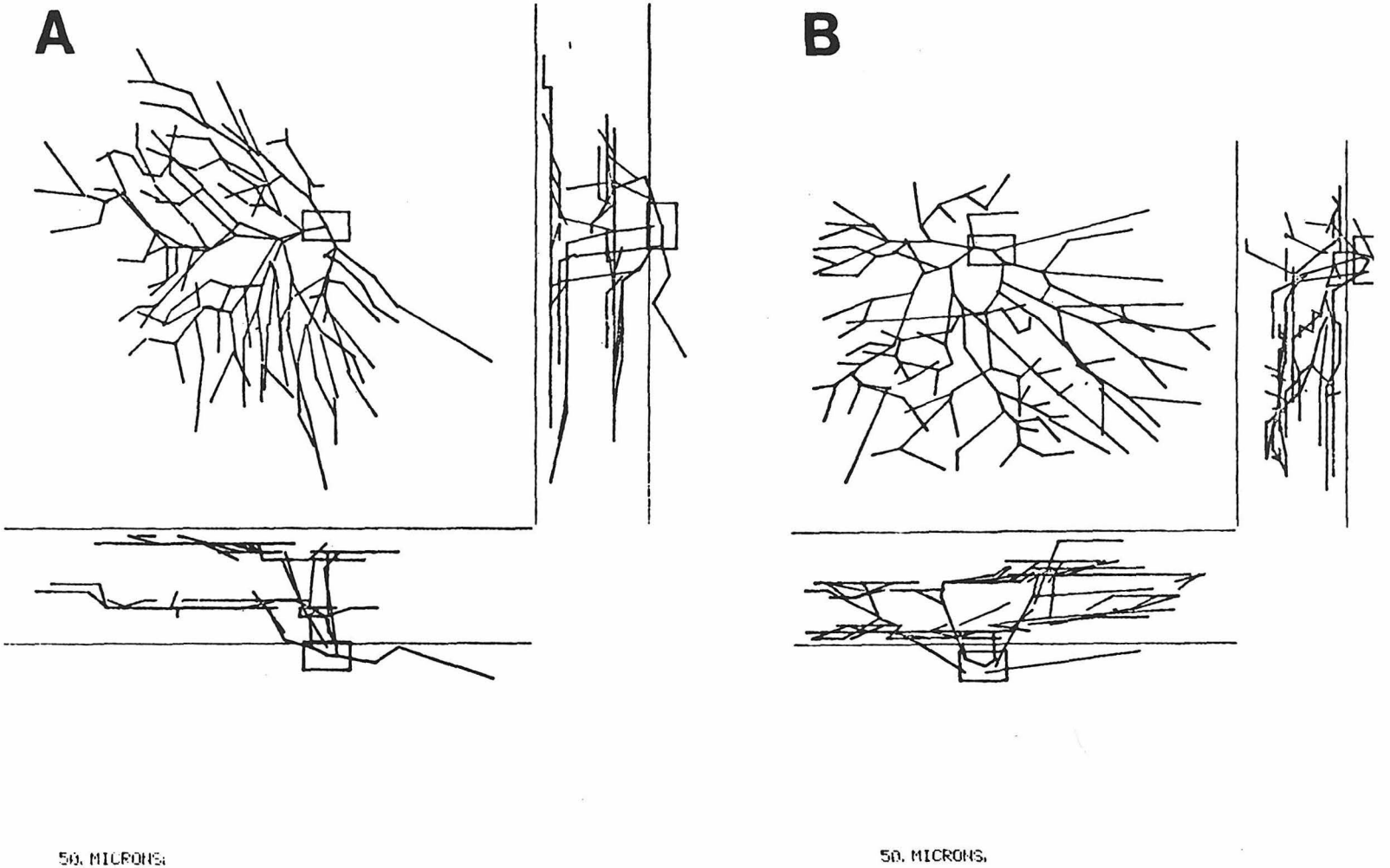


Figure 16. Computer drawings from syntactic descriptors of two neurons. INL is indicated on side views.

5.2.2 Measurement Errors

We have examined the resolution of the 3-dimensional cursor system by testing the repeatability of recording a particular location on a neuron. The error, expressed in standard deviation, is given in table I for various microscope objectives assuming that optical sections are taken at 2 micron intervals, and that the stereo pair construction involves a shift per section of 2 pixels.

objective	100x	63x	40x	20x	10x
x,y error(SD)	0.4m	0.7m	1.0m	2.0	4.5m
z error(SD)	1m	1.2m	1.8m	5.0m	10.0m

TABLE I

It is perhaps surprising that the z axis repetition error is less than the distance between optical sections. We discovered that when viewing our structures in 3-dimensions the eye has a remarkable ability to interpolate reliably between sections. This discovery prompted us to improve the

z axis resolution of the cursor, and to use larger amounts of binocular disparity in the stereo pairs. Some of our early neuronal recordings, therefore, reflect poorer z axis resolution than that represented above.

There are many other forms of error in our measurements which are difficult to estimate. It is not possible for example to ascertain the degree to which the cell is filled with opaque silver granules. Also, there are ambiguities in the stereo pair representation which cannot always be resolved. The number of ambiguities depends largely on the complexity and the density of the particular neuronal arborization. In the extreme case of the small amacrines, we can only say that our representation preserves the form of the arbor but deviates in many details. In larger amacrines and in ganglions there are very few ambiguities which require guessing. Occasionally it is difficult to tell whether a given process belongs to the neuron in question, but here we are fortunate in that our Golgi technique frequently produces a light background and isolated neurons.

5.3 Data Structures for Neurons, the Class Concept

We now wish to employ some conceptual modelling tools from the structured programming branch of computer science in order to unify our treatment of retinal structure and

function. The interesting notion is that of a class. A class can be defined as an abstract data type which "defines a data structure and the operations that can be performed on it..." (1). In our case a form-function model of a neuron is a class which defines a binary tree data structure to whose nodes are attached ultrastructural and functional attributes, as described in 5.1, and a collection of operations on the data structure.

We may wish the class to have the following operations:

1) A procedure for extracting n structural parameters so that each neuron can be represented as a point in an n-dimensional space for the purposes of classification. (see 6.1)

2) A procedure for detecting occurrences of substructures within a neuron (see 6.2).

3) A procedure for inserting information about synaptic structure and synaptic function.

4) A procedure for functional modelling which would examine the neural structure, abstract linear spatial filters and apply these filters to specified spatial input.

The fourth procedure is rather speculative but I think it indicates a method of unifying structural and functional data in a single model.

The high level programming language PASCAL (1) provides

many data structuring facilities which make this sort of modelling quick and easy. For example our neuronal data structures are represented using the facilities for defining complex data types. The data types for nodes of the tree structure are defined as follows:

```
Type nodekind = (endnode, namenode, somanode, procnode,
   branchnode, termnode);

Type node = Record
    leftlink, rightlink:  integer;
    case kind:  nodekind of
    endnode:  (endval:  enddesc);
    namenode:  (nameval:  namedesc);
    somanode:  (somaval:  somadesc);
    procnode:  (procval:  procdesc);
    branchnode:  (branchval:  branchdesc);
    termnode:  (termval:  termdesc)
    end;

Type enddesc = array (.1..64.) of char;

Type namedesc = Record
    name:  cellname;
    pad:  namepad
    end;

Type branchdesc= Record
    location:  point3D;
    pad:  branchpad
    end;

Type term desc = Record
    location:  point3D;
    pad:  branchpad
    end;

Type somadesc = Record
    location:  point3D;
    fatsize, thinsize:  real;
    pad:  somapad
    end;

Type procdesc = Record
    location:  vector 3D;
```

```
    startsize, endsize:  real;
    end;
```

Once the data type 'node' has been defined, variables of type node can be created and the different kinds of nodes can be easily distinguished and accessed. For example, if nodex is a variable of type node and vector a variable of type vector3D the statement

```
With nodex do
    if kind = procnode then with procval do
    vector := location;
```

checks what kind of node nodex is and if it is a process node then the location is accessed.

The programming language CONCURRENT PASCAL **implements** classes as an abstract data type in the language. Such classes are a powerful conceptual aid in both modelling and thinking about biological systems. The outline of a form-function model of the retina can be represented by the following data definitions in CONCURRENT PASCAL. This retina has only one kind of receptor, one kind of horizontal, and one kind of bipolar.

```
type layertype = (R, H, B);
```

"A retina is a class having data types, receptor layer, horizontal layer, and bipolar layer, the retina has operations on these data types for initializing their structure, updating the synaptic information on the structure and for defining spatial filters from the structure."

```
type retina = class
```

```
var receptors, horizontals, bipolars:  layer;

procedure entry init(rfile, hfile, bfile:  diskfile);
   begin
   receptor.initialize(rfile);
   horizontals.initialize(hfile);
   bipolars.initialize(bfile);
   receptors.putsynapse;  receptors.definespatial;
   horizontals.putsynapse;  horizontals.definespatial;
   bipolars.putsynapse;  bipolars.definespatial;
   end;
```

"Given a stimulus image, and a layer of cells to test, probe models retinal function"

```
procedure entry probe(input:  image;  var output:  image;
      outlayer:  layertype);
   var image1, image2, dummy:  image;
   begin
   if outlayer = R then receptors.model(input, image
      receptors.model(input, image1, output);
   else if outlayer = H then
      begin
      receptors.model(input, image1, image2);
      horizontals.model(image2, image1, output);
      end;
   else if outlayer = B then
      begin
      receptors.model(input, dummy, image1);
      horizontals.model(image1, dummy, image2);
      bipolars.model(image1, image2, output);
      end;
   end;
begin end;
```

"A layer is a class which embodies a form-function model of a cell layer. Each layer has a form (a tree grammar representation), a function (a set of spatial filters operating on the input light image or on the output from other layers) , and an output image representing the cell potential at each point in the layer."

```
type layer = class;
   var   form:  structure;
      function:  linearspatialfilters;
      output:  image;
```

"Initialize finds a given tree structure on disk of the sort whose nodes are defined above and initializes the variables 'form' to this tree structure."

```
   procedure entry initialize(structurefile:  diskfile);
      begin
      initializeform(form, structurefile);
      end;
```

"Putsynapse traverses form, tells the user where it is on the structure and allows synaptic information to be added."

```
   procedure entry putsynapse;
      begin
      traverseandmodify(form);
      end;
```

"Definespatial looks at form with its synaptic information and builds a linear spatial filter using synaptic values and locations to specify filter weights."

```
   procedure entry definespatial;  begin
      traverseforfunction(form, function);
      end;

   procedure entry model(input1, input2:  image;
         var out:  image);
      begin
      filter(input1, input2, function, output);
      out := output;
      end;
   begin end;
```

Here we assume for the purposes of a simple example that receptor phototransduction can be modeled as an array of synapses from the outside world, that receptors only receive input from the input image, that horizontals only receive input from receptors and that bipolars receive input from both receptors and horizontals. Having defined the above abstract data types we can write a main routine which sets up the structure of these three layers, introduces functional synaptic data, builds the spatial filters from this information and executes the form- function model.

```
var file1, file2, file3:  diskfile;
begin
specifyfiles(file1, file2, file3);
specifyi/o(inputimage, outputimage);
specifycell(probetype);
retina.probe(inputimage, outputimage, probetype);
end;
```

The above example is intended to illustrate the data structuring power of PASCAL as applied to organizing information about biological neural structures. Complex organizations of information can quickly be set up and easily accessed. Once the data types have been defined, the development of procedures which operate on these types is greatly facilitated. Using these tools we can hang functional data on a structural framework and produce a unified model. In the above use of abstract data types to model a simple structural functional system we have only specified the highest level procedures. None of the algorithms for actually doing the work have been specified such as the implementation of a digital filter, the construction of digital spatial filters from neuronal structure descriptors and synaptic information and the traversal of the tree representation of a neuron. The intent of this example is to present a conceptual aid to structured thinking about biological form-function systems. This sort of approach helps pseudobiologists such as myself to mentally organize the many details of retinal lore into some semblance of order and structure.

After looking at representations of neuronal structure and seeking functional correlates it seems that the functional identification should be of a spatial nature. This is not to say that temporal activity is unimportant, but that perhaps spatial function is more important. Historically, temporal characteristics of the retina were studied simply due to the fact that the experimental state of the art did not allow the easy acquisition of detailed spatial information. We suggest in section 8.2 a method of generating spatio-temporal white-noise stimuli and an approach to characterizing the spatio- functional attributes of neurons. Thorough form-function models await the results of this attempt.

Much recent emphasis has been placed on the use of Wiener's non-linear system identification techniques for functional investigations in the central nervous system (6,7). This work so far has dealt almost exclusively with the nonlinear characteristics of the temporal activity of neurons. However, we now believe that perhaps the temporal activity, especially in the retina, is somewhat artifactual to the 'true' function of the retina which is to extract spatial information from the environment. We also suspect that the nonlinearities present may be primarily of a thresholding nature and, from a spatial function point of view, may be important but secondary to linear spatial

filtering functions.

The point for the present discussion is that the spatial properties of function are probably more readily correlated with morphological models but our ability to measure spatial properties is limited. The full impact of efforts to correlate structure and function awaits good spatial function characterizations.

With the above conceptual framework as background we turn now to some details of classification methods and then present our results.

REFERENCES

1. Brinch Hansen, P., (1975), 'The programming language Concurrent Pascal,' IEEE Trans. Software Engr. , 1: 2.

2. Fu, K. S., and B. K. Bhargava, (1973), 'Tree systems for syntactic pattern recognition,' IEEE Trans. Computers, c-22: 12, 1087-1099.

3. Gonzalez, R. C., and M. G. Thomason, (1974), 'Tree grammars and their application to pattern recognition,' Tech. Rept. TR-EE/ CS-74-10, EE Department, U. of Tenn., Knoxville.

4. -----, (1974), 'Inference of tree grammars for syntactic pattern recognition,' Tech. Rept. TR-EE/CS-74-20, EE. Department, U. of Tenn., Knoxville.

5. Knuth, D. E., (1968), The art of computer programming-fundamental algorithms, vol. 1, Addison-Wesley Publishing Co., Reading, Mass.

6. Marmarelis, P. Z., and K.-I. Naka, (1972), 'White-Noise analysis of a neuron chain: an application of the Wiener theory,' Science, 175: 1276-1278.

7. Marmarelis, P. Z., and G. D. McCann, (1973), 'Development and application of white - noise modeling techniques for studies of insect visual nervous

system,' Kybernetik, 12: 74-89.

8. Randall, D., (1970), 'Formal methods in the foundations of science,' Thesis in Information Science, Caltech, p. iii and p. 48.

9. Tou, J. T., and R. C. Gonzalez, (1974), 'Pattern Recognition Principles, Addison - Wesley Publishing Co., Reading, Mass.

"Thus a new sacred cow of mathematical machinery is created - its priesthood will probably make a good academic living regardless of whether the cow gives any milk."

Bremermann (1)

6. CLASSIFICATION OF NEURONS

Previously the classification of neurons into morphological types has been performed on the basis of subjectively evaluated characteristics. We wish to present here our approach to a quantitative classification. The data structure of chapter 5 forms the basis for the extraction of numerical parameters for representing global morphological features. We start with a population of 57 neurons each represented by a tree data type. Two ways in which this population can be partitioned into types are: 1) to extract n parameters from each neuron and perform cluster analysis operations on the resulting n-dimensional space, and 2) to specify a syntactic substructure and to partition the population by occurrence of this substructure.

6.1 Clustering Via Numerical Parameters

The feature vector we use for classification is

$$x = (x1, x2, \ldots , x21)$$

where

x1 = maximum distance through soma (0.1 micron)

x2 = minimum distance through soma (o.1 micron)

x3 = z or axial position of soma in normalized coordinates where proximal inner nuclear layer is at z=150 microns and distal INL is at z=100 micron

x4 = total neuronal volume (microns

x5 = total neuronal surface area (micron 2/10)

x6 = average diameter of processes (microns/100)

x7 = maximum distance from soma to terminator (micron/10)

x8 = average distance from soma to terminator (microns/10)

x9 = average distance from center of gravity of terminators to terminator (micron/10)

x10 = average distance from center of gravity of branch point (micron/10)

x11 = average deviation from x10 (micron)

x12 = average deviation from x9 (micron)

x13 = average length of circular component of process (micron/10)

x14 = average length of radial component of process (micron/10)

x15 = percent of total surface at level 1, see figure (17)

x16 ... x19 = percent surface at level 2 ... level 5

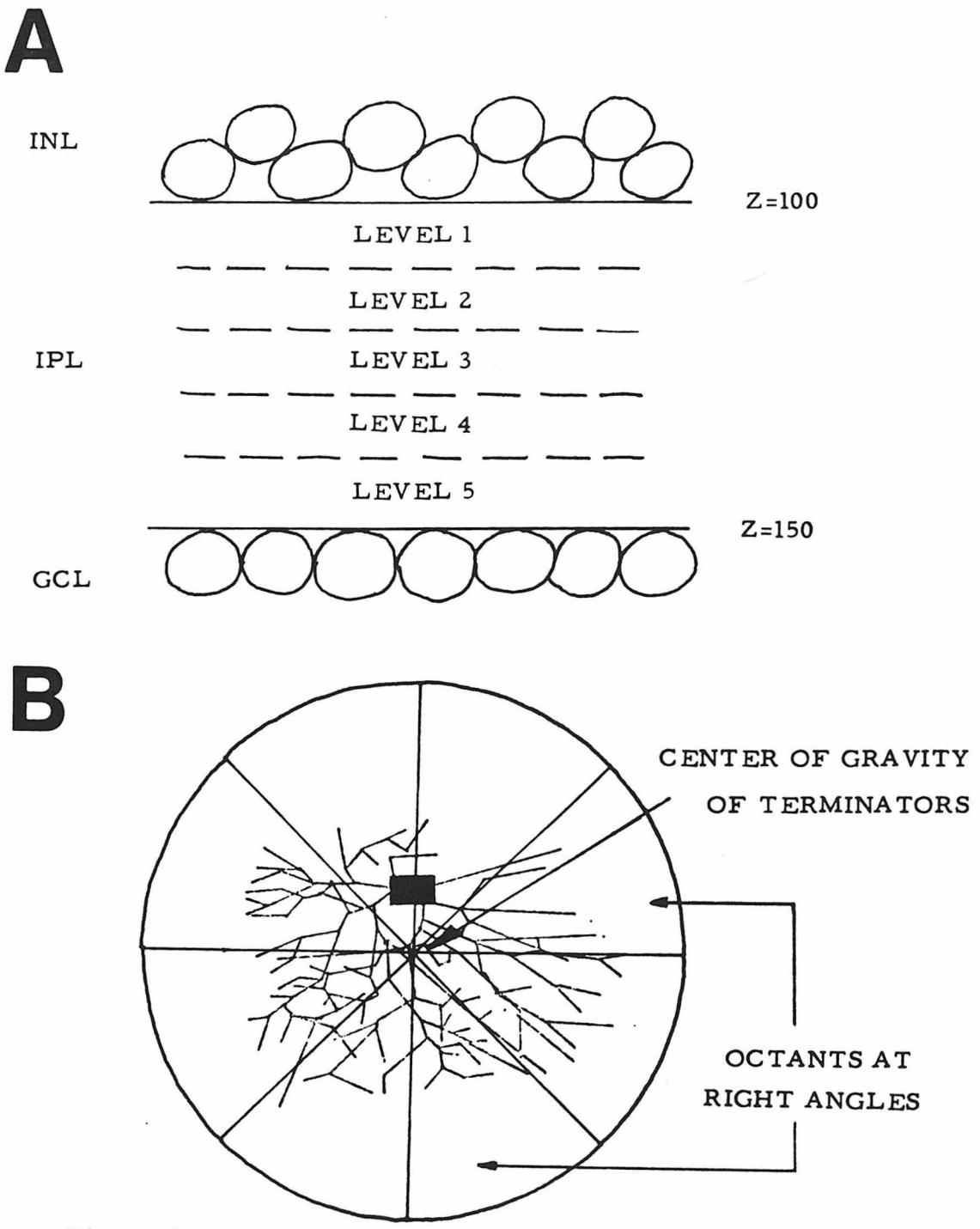


Figure 17. A. Side view of retina showing levels of IPL used in computation of parameters 15-19.
B. Top view of neuron showing division into octants for computing directionality parameters 20 and 21.

x20 = maximum percent surface in opposite octants

x21 = maximum percent surface at right angles.

This feature vector is computed for each neuron and stored in a cell tally data set. Now we must decide how to examine the clustering of points within the 21 dimensional space represented by this tally.

The situation with which we are now faced is implied by the quote that begins this chapter. Most of the established formalisms in pattern recognition are not applicable to our situation. In the simplest case the number of underlying classes w1...wn or types into which we will classify the objects are known and, given a pattern or feature vector $\vec{x}$ to use as the basis for classification, the a priori probabilities $P(w_j)$ and the class conditional densities $P(x|w_j)$ are also known. The a priori probability $P(w_j)$ is the probability that we will get an object of type j if we pick a random object from our population. The class conditional density function $P(x|w_j)$ gives the statistics of class w_j, that is, $P(x|w_j)$ is a function which gives for each x_i the probability that an object which is of class w_j will have parameter $x=x_i$.

If $w_1....w_n$, $P(w_j)$, and $P(x|w_j)$ are known then we can apply the Bayes Rule (5):

$$P(w_j \mid \vec{x}) = P(\vec{x} \mid w_j)P(w_j) / \sum_{j=1}^{n} P(\vec{x} \mid w_j)P(w_j)$$

This expression gives the probability that we have an object of type w_j if it has a feature vector = $\vec{x}$. This information can be used in a multitude of ways to decide of which type an object might be, and thus to decide how to separate a population into given groups.

In our case we do not know $P(x \mid w_j)$. A vast body of literature applies to this unfortunate situation under the general heading of Bayesian learning. Basically one takes a training set of objects St from nature and tries to estimate $P(x \mid w_j)$. In the case of supervised Bayesian learning the type or class of each object of this training set St is known and assuming a form (eg. normal distribution) of the function $P(x \mid w_j)$ The parameters of this algebraic form can be estimated. In the unsupervised case the number of classes is known, $P(w_j)$ for each class w_j is known, and the form of $P(x \mid w, \theta j)$ are known, but the parameters θj are unknown. Since the class of each object of the training set St is now known the probability density functions are of a form called mixture densities (5).

$$P(\vec{x} \mid \vec{\theta}) = \sum_{j=1}^{c} P(\vec{x} \mid w_j, \theta_j)\, P(w_j)$$

where $\vec{x}$ is the feature vector, and $\vec{\theta}$ is the unknown parameter vector for the class conditional probability

density function $P(\vec{x}|w_j,\theta_j)$ (now called component densities). The approach taken in this unsupervised case is to take a set $\hat{X}=(x_1...x_n)$ of sample object feature vectors, compute the likelihood of drawing this sample

$$P(\hat{x}|\vec{\theta}) = \prod_{k=1}^{n} P(x_k|\vec{\theta})$$

and find the value of θ which maximizes this likelihood (5). Knowing $\vec{\theta}$ we can find the component densities $P(\vec{x}|w_j,\theta_j)$ and can use the Baysian classifier.

This machinery is very nice for certain cases but in our case we know neither the number of classes n, nor $P(w_j)$, nor the form of $P(\vec{x}|w_j,\theta_j)$. Our goal is not so much to develop a classifier which will assign a neuron to one of a given set of classes, as to find the underlying classes and subclasses.

The existing approaches to finding the underlying classes in a population are more heuristic and less formal in nature than the above classification methods. They can be included under the general heading of cluster analysis. These numerical cluster analysis methods employ a measure of object similarity, often a simple distance measure in the n-dimensional feature space. All of the above methods are plagued by the curse of dimensionality. That is, the number of samples required for the algorithms to be valid, grows

exponentially with the dimensionality of the feature space. The feature space which we have defined for neuronal structure is 21 dimensional. The required number of samples far exceeds the sample number we are easily able to accumulate. For these reasons we have followed an interactive approach designed to employ the human observer's ability to obtain instantaneous density measurements on a 2-dimensional scattergram.

Our procedure is as follows. Two dimensional projections of the points in our n-dimensional space are systematically plotted on the computer terminal screen. The scattering of points in this subspace is viewed and decision surfaces can be specified which partition the points into subgroups. These subgroupings can then be viewed in 2-dimensional projections of the n-space until a space is found where they can be further partitioned. In this way the population can be quickly broken into subgroups. Examples of this method are shown in figure (28).

The element of subjectivity in this approach is somewhat undesirable. However, we feel that the return from a more rigorous procedure would not justify the time spent in programming and data processing. We would rather spend future time obtaining functional information and developing form-function models.

6.2 Partitioning by Occurrance of Syntactic Subunits

We conclude this chapter with a brief look at an alternative method for partitioning a population of objects. Consider a population of objects (neurons) each represented as a sentence P_i in some language L. For a given object P_i and a given string S in L, the assertion S occurs in P_i is either true or false depending on whether the string S is found in P. For a language generated by a tree grammar of the sort we described in chapter V our object P_i and substructure S are tree stuctures. Assertions such as the one given above can thus be used to separate the set of objects into groups.

In general any operation on a set of sentences M in a language, which imposes a partial ordering on M, can be used to partition M. The task is that of classifying M into disjoint subsets. For example we may wish to specify a substructure S of a particular sort, say a stratified portion of a neuron, and search for its occurrence in all of our neurons. The algorithm would traverse tree T_i and for each node of T_i try to match S with the descendants of this current node in the tree. A very restricted sort of matching process could be performed by matching all attributes of each node of S with the corresponding nodes in T_i. In this case a match would occur only if the locations and sizes of all nodes in S were exactly the same as in some

portions of T_i. A more abstract matching procedure which checked only the type of node would find occurrences of a generalized form of S. Various abstractions could be extracted by allowing S to be a substructure containing nonterminals. The matching algorithm in this case uses the tree S to guide a recursive descent parsing algorithm applied at each node of T_i.

In order to explicate a procedure for abstracting generalized substructure from a data type which is a sentence in a language L generated by a tree grammar G3, we take the following example. Consider the tree grammar

G3 = (VN,VT,P,R,S)

where

VN = (<subtree>,<strat>)

VT = (sp,dp,s,b,t)

P1 neuron→s<subtree>--→

P2 <subtree>→dp<subtree>
→sp<subtree>
→b<subtree><subtree>
→<strat>
→t

P3 <strat>--→b<sp><sp>

here sp is a process which starts and ends at approximately the same retinal z level, that is, z1-z2 <=e, and dp is a process for which z1-z2 >e.

Using the class concept (2,3,4) we specify a class C which has data types and a set of operations on these data types. The data types of C are TTREE and NTREE. TTREE is a sentence in L, that is, TTREE is a binary tree each of whose nodes are in Vt and whose root is a soma node. NTREE is a binary tree whose nodes are in Vt union VN.

The operation of interest is one which takes a variable of type NTREE and counts its occurrences in a variable of type TTREE. The following algorithm demonstrates types definitions and operations for our class C.

"data type definitions TTREE"

```
type tnode = (endnode, namenode, somanode,
      dprocnode, sprocnode, branchnode
      termnode);

type tnodepage = array (.1..nodesperpage.) of tnode;

type tnode = Record
   leftlink, rightlink:  integer;
   case kind:  tnodekind of
      endnode:  (endval:  enddesc);
      namenode:  (nameval:  namedesc);
      somanode:  (somaval:  somadesc);
      dprocnode:  (dprocval:  dprocdesc);
      sprocnode:  (sprocval:  sprocdesc);
      branchnode:  (branchval:  branchdesc);
      termnode:  (termval:  termdesc);
   end;
```

"descriptors for the various types of terminals in grammar"

```
type enddesc = array (.1..64.) of char;
```

```
type namedesc = Record
   name:  cellname;
   pad:  namepad
   end;

type branchdesc = Record
   location:  point3D;
   pad:  branchpad
   end;

type termdesc = Record
   location:  point3D;
   pad:  termpad
   end;

type somadesc = Record
   location:  point3D;
   fatsize,thinsize:  real;
   charge:  real;
   synapse:  synapdescr;
   pad:  somapad
   end;

type procdesc = Record
   location:  vector3D;
   startsize, endsize:  real;
   synapse:  syndescr;
   end;
```

"data type definition for NTREE"

```
type nnodekind = (somanode,dprocnode, sprocnode,
      branchnode, termnode, subtreenode, startnode)

type nnodepage = array (.1..nodesperpage.) of nnode;

type nnode = Record
   leftlink, rightlink:  integer;
   case kind:  nnodekind of
      somanode:  (somaval:  somadesc);
      dprocnode:  (dprocval:  dprocdesc);
      sprocnode:  (sprocval:  sprocdesc);
      branchnode:  (branchval:  branchdesc);
      termnode:  (termval:  termdesc);
      subtreenode:  (subtreeval:  subtreedesc);
      stratnode:  (stratval:  startdesc)
   end;
```

"descriptors for nonterminals"

```
type subtreedesc = Record
   name:  subtree name;
   pad:  subtree pad
   end;

type stratdesc = Record
   name:  stratname;
   pad:  stratpad
   end;
```

"Variables of type ttree are stored on disk and accessed one tnodepage at a time. The rightlink and leftlink point to other nodes thus forming the binary tree structure. The following operation returns the node of number nodeno."

```
Procedure getnode(nodeno:  integer;  var elem:  tnode);
   var index:  integer;
   begin
      index:=(nodeno-1) div nodesperpage +1;
      if pageno <> index then
      begin
      pageno:= index;
      get(1, pageno, block);
      end;
      elem:=block.nodelist(.(nodeno-1)
            mod nodesperpage + 1.)
   end;
```

"The following procedure traverses a variable of type ttree and searches for an **occurrence** of substructure ntree. The structure of these procedures mirrors the production rules of our grammar."

```
Procedure traverse(tnodeno:  integer);
   begin
   getnode(tnodeno, nowtnode);
   with nowtnode do
      begin
      case kind of
         endnode:;
         namenode:;
         somanode:  with somaval do
            begin
            match(1, tnodeno);
            if ok then occurrence=occurrence+1
            else ok:= true;
            traverse(leftlink);
            end;
         branchnode:  with branchval do
            begin
```

```
            match(1, tnodeno);
            if ok then occurrence=occurrence+1
            else ok:=true;
            traverse(leftlink);
            traverse(rightlink);
            end;
         sprocnode:  with procval do
            begin
            match(1, tnodeno);
            if ok thenoccurrence=occurrence+1
            else ok:=true;
            traverse(leftlink);
            end;
         dprocnode:  with procval do
            begin
            match(1, tnodeno);
            if okthen occurrence=occurrence +1
            else ok:=true;
            traverse(leftlink);
            end;
         termnode:  with termval do
            begin
            match(1,tnodeno);
            if ok then occurrence=occurrence +1
            else ok:=true;
            end;
      end;
   end;
```

"At each node of ttree match is called to see if S occurs starting at this node. If a mismatch occurs then ok is set to false."

```
Procedure match(nnodeno, tnodeno:  integer);
   var temptnode:  tnode;
   begin
   if ok then begin
   getnode(tnodeno, temptnode);
   getnode(nnodeno, nownode);
   with nownode do
      case kind of
         endnode:;
         somanode:  if temptnode.kind = somanode then
            match(nownnode.leftlink, temptnode.leftlink)
            else ok:=false;
         dprocnode:  if temptnode.kind= dprocnode then
            match(nownnode.leftlink, temptnode.leftlink)
            else ok:=false;
         sprocnode:  if temptnode.kind = sprocnode then
            match(nownnode.leftlink, temptnode.leftlink)
            else ok:=false;
```

```
            branchnode:  if temptnode.kind= branchnode then
               begin
               match(nownode.leftlink, temptnode.leftlink);
               match(nownode.rightlink,temptnode.rightlink);
               end
               else ok:=false;
            termnode:  if temptnode.kind <> termnode then
               ok:=false;
            subtree:  subtreematch(temptnode);
            strat:  stratmatch(temptnode)
         end;
      end;
      end;
```

"Subtreematch implements a recursive descent parse using production rule P2 of our grammar"

```
   Procedure subtreematch(anode:  tnode);
      var bnode:  tnode;
      begin
      with anode do
      if kind = dprocnode then begin
         getnode(anode.leftlink, bnode);
         subtreematch(bnode);  end
      else if kind = sprocnode then begin
         gettnode(anode.leftlink, bnode);
         subtreematch(bnode) end
      else if kind = branchnode then begin
         getnode(anode.leftlink, bnode)
         subtreematch(bnode);
         gettnode(anode.rightlink, bnode);
         subtreematch(bnode) end
      else if kind = termnode then
      else stratmatch(bnode);
         end;
```

"Stratmatch parses for strats using production rule P3 of our grammar."

```
   Procedure stratmatch(anode:  tnode);
      var bnode:  tnode;
      begin
      if anode.kind = branchnode then begin
         gettnode(anode.leftlink, bnode);
         if bnode.kind = sprocnode then begin
            getnode(bnode.rightlink, bnode);
            if bnode.kind <> sprocnode then ok:=false;
            end
         else ok:=false;
         end
      else ok:=false;
```

```
    end;
```

Here we are assuming the following global variables.

```
var ok:  boolean;
  occurrence:  integer;
   nowtnode:  tnode;
```

Procedures subtreematch and stratmatch implement production rules in the grammar, that is they parse a portion of the neuron to detect their respective substructures. A main routine can now be written to search a neuronfor occurrences of a given substructure.

```
begin
   findstartnodenumber(tnodeno);
   traverse(tnodeno);
   writeout(occurrence) ;
end;
```

With these recursive routines we can easily collect all neurons exhibiting greater than n occurrences of a particular substructure and thus partition our population of objects. We do not currently have an implementation of this method. We believe that a classification scheme based on the above method would simulate some of the abstract techniques used by humans to classify structures into types.

REFERENCES

1. Bremermann, H. J., (1974), Review of An Introduction to Mathematical Techniques in Pattern Recognition, by H. C. Andrews, Amer. Scientist, vol. 62, pp. 244-245.

2. Brinch-Hansen, P., (1975), The Programming Language Concurrent Pascal. IEEE Trans. on Software Engr. 1,2, June.

3. Brinch-Hansen, P., (1975), The Solo Operating System: Processes, Monitors, and Classes. Information Science, California Institute of Technology, July 1975.

4. Dahl, O.-J., and C. A. R. Hoare, (1972), Hierarchical Program Structures. , In O. J. Dahl, E. W. Dijkstra, and C. A. R. Hoare, Structured Programming. Academic Press.

5. Duda, R. O., and P. E. Hart, (1973), Pattern Classification and Scene Analysis., Wiley, New York.

6. Wirth, N., (1971), The Programming Language PASCAL , Acta Informatica 1,1, 35-63.

"The pieces of a biological pattern recognition system are themselves patterns which require description and classification. The eye must parse itself."

Alvin Botosch (1)

7. RESULTS FROM FROG RETINA

This chapter presents the data and results from our morphological studies of the frog retina. A typology is suggested based on, 1) camera lucida drawings of Golgi impregnated neurons seen in flat mount, 2) comparisons of computer reconstructions with Cajal's side view drawings, and 3) classification of cells via numerical parameters.

7.1 Classical Morphology

Adult frogs (Rana Cateisbiana), 4 to 5 inches long purchased from Mogul Ed. Inc., Oshkosh, Wisconsin were used in our Golgi procedure. The frogs were killed by injecting fixative (25ml. of 25% gluteraldehyde in 100 ml. of 5% potassium dichromate, unbuffered) into the brain cavity. After a few minutes the eyes were excised and the anterior part of the eye was cut away from the eye cup. The retina

was slithered out onto a glass slide with receptor side up. A piece of dry lens tissue was placed over the retina followed by a piece of Telfa bandage (plastic only, no gauze). This was tied gently in place with thread and soaked in fixative for 5-8 days then washed with distilled water. The slide was then placed in a 0.5 to 1.0% silver nitrate solution for 1 or 2 days. The staining process was monitored during this period by looking through the back of the slide at the neurons. If the process was unsatisfactory the slide was placed back in the fixative for a while longer. After staining the Telfa and lens paper were removed and the preparation was washed then dehydrated in cold methanol (10 minutes in 50%, 70%, 85%, 95%, and 20 minutes in 100%). This was followed by embedding in Spurr plastic. The retina in plastic was then sliced carefully off of the slide and flipped over so that the receptor side was down and an additional smooth layer of Spurr was baked on. This preparation was then ready to be optically sectioned and digitized as explained in section 4.1.

Figure (18) shows camera lucida drawings of the two types of horizontal cells we found in the frog retina. The large horizontal cell (LH) extends nearly 200 microns and has an axon which extends further. All branches of this cell exhibit spines with little claws or knobs at the ends. The small horizontal (SH) is much smaller with clumps of

claws arranged uniformly and a long axon which also possesses claws. In one case the tail or axon had a large soma-like protrusion hanging from it. We do not attach particular significance to this single occurrence of this feature.

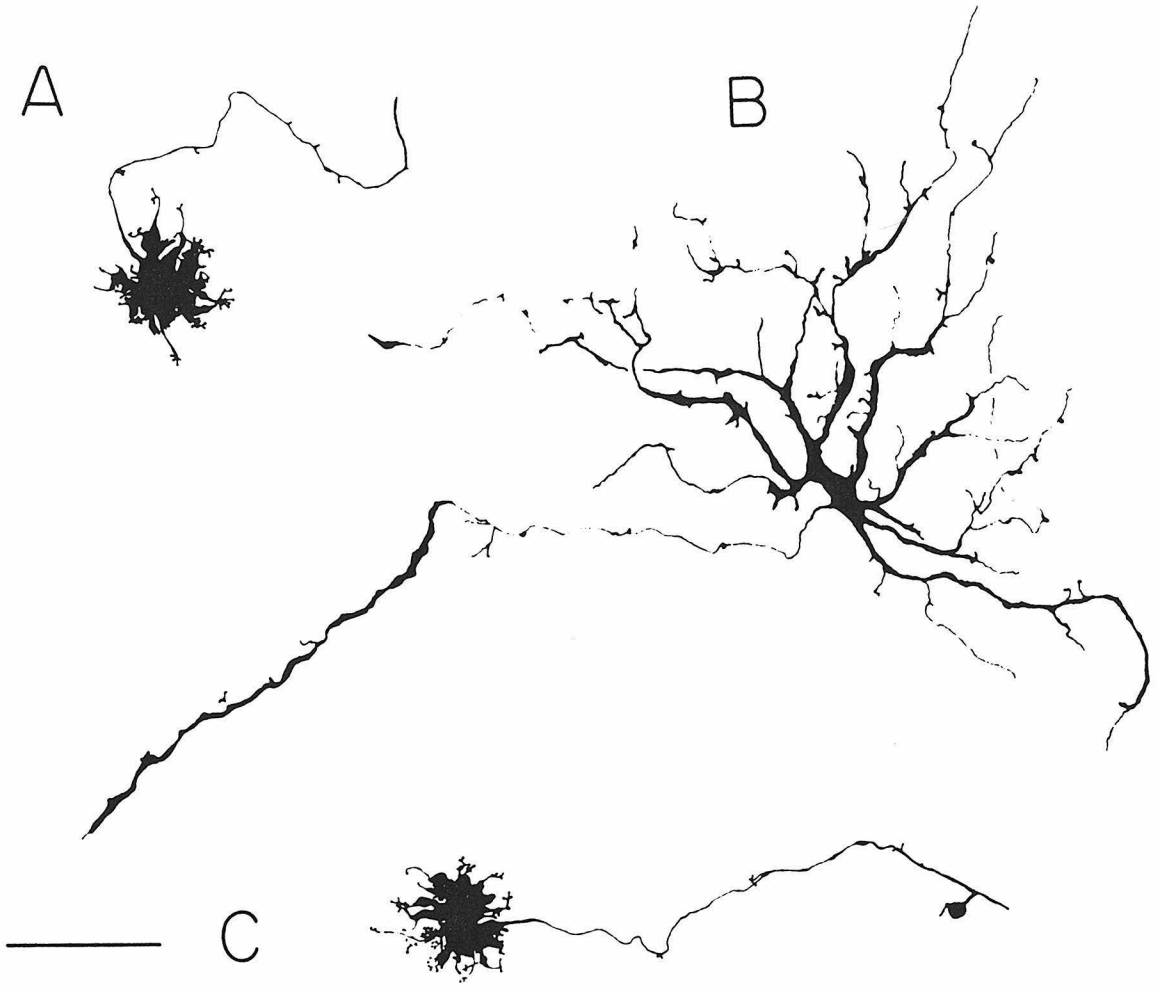


Figure 18. A and C, small horizontal cells; B, large horizontal cell. Calibration is 50 microns.

Figure (19) shows samples of bipolars from our studies. We observed small dendritic field and large dendritic field bipolars, each seen with Landalt's clubs. Since our emphasis has been on classical amacrines and ganglions we have not yet attempted to correlate the dendritic structure of bipolars with their axonal stratifications in the IPL.

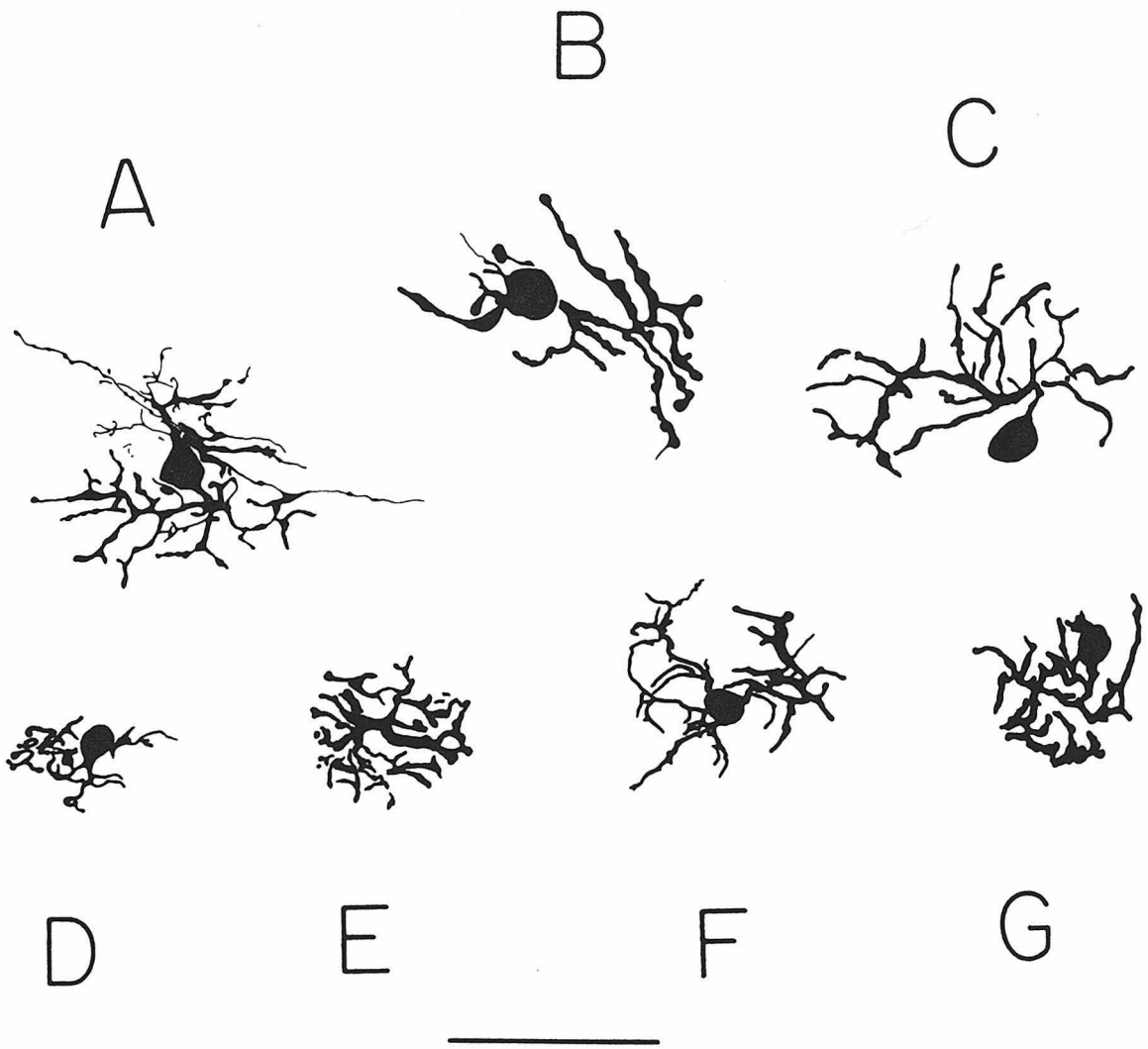


Figure 19. Bipolar cells from frog retina. Calibration is 50 microns.

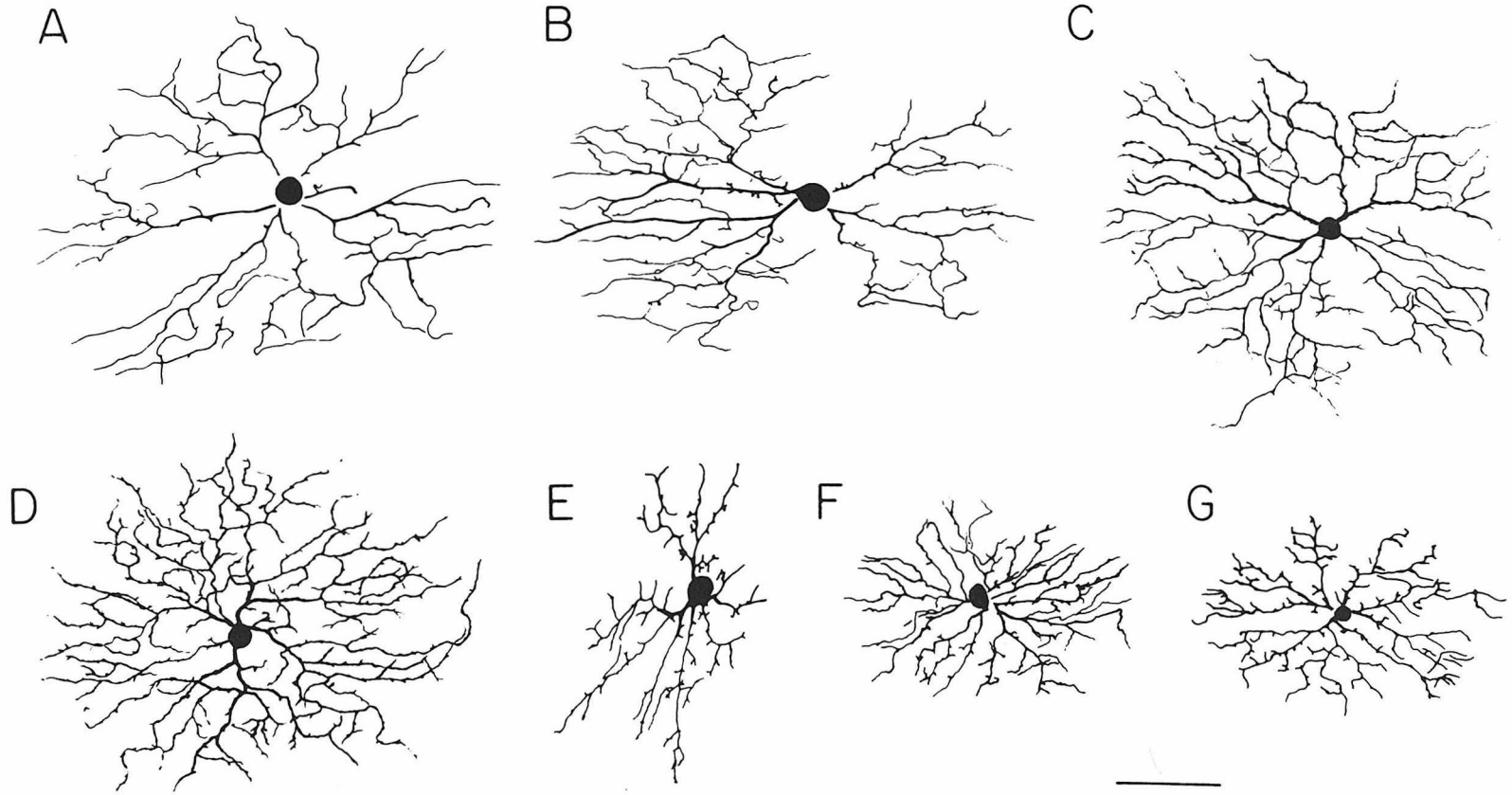


Figure 20. Single level restricted field cells (type GCLSA). Calibration is 50 microns.

7.1.1 Cells with soma in ganglion cell layer (GCL)

Figure (20) shows small cells from the ganglion cell layer which have monostratified restricted fields. These cells which we subjectively group as type GCLSA have the following characteristics.

1) medium size (150 - 250 microns)
2) soma spherical and monopolar
3) no axon seen
4) distinct branching pattern, branching density appears to increase toward the periphery of the arbor.
5) monostratified
6) arborization is round or elliptical, single lobed, symmetric about the soma.
7) spines appear on all parts of the neuron except the soma and initial major processes in the foot.

Figure (21) shows the large ganglions of the frog retina. Cells A - D we group as type GCLLA. These neurons are very large (300 - 600 microns) and are monostratified. Cells E - H are very similar in size and soma shape but they show some evidence of stratification. All cells A - I are placed in the class GCLLA recognizing that a considerable

variation exists within the group and that subgroupings can be made. The subjective distinguishing characteristics are:

1) large size (300-600 micron)
2) soma has uterine shape
3) large axon present
4) webbed branches and heavy processes
5) monostratified
6) three major spatial lobes or regions of arborization

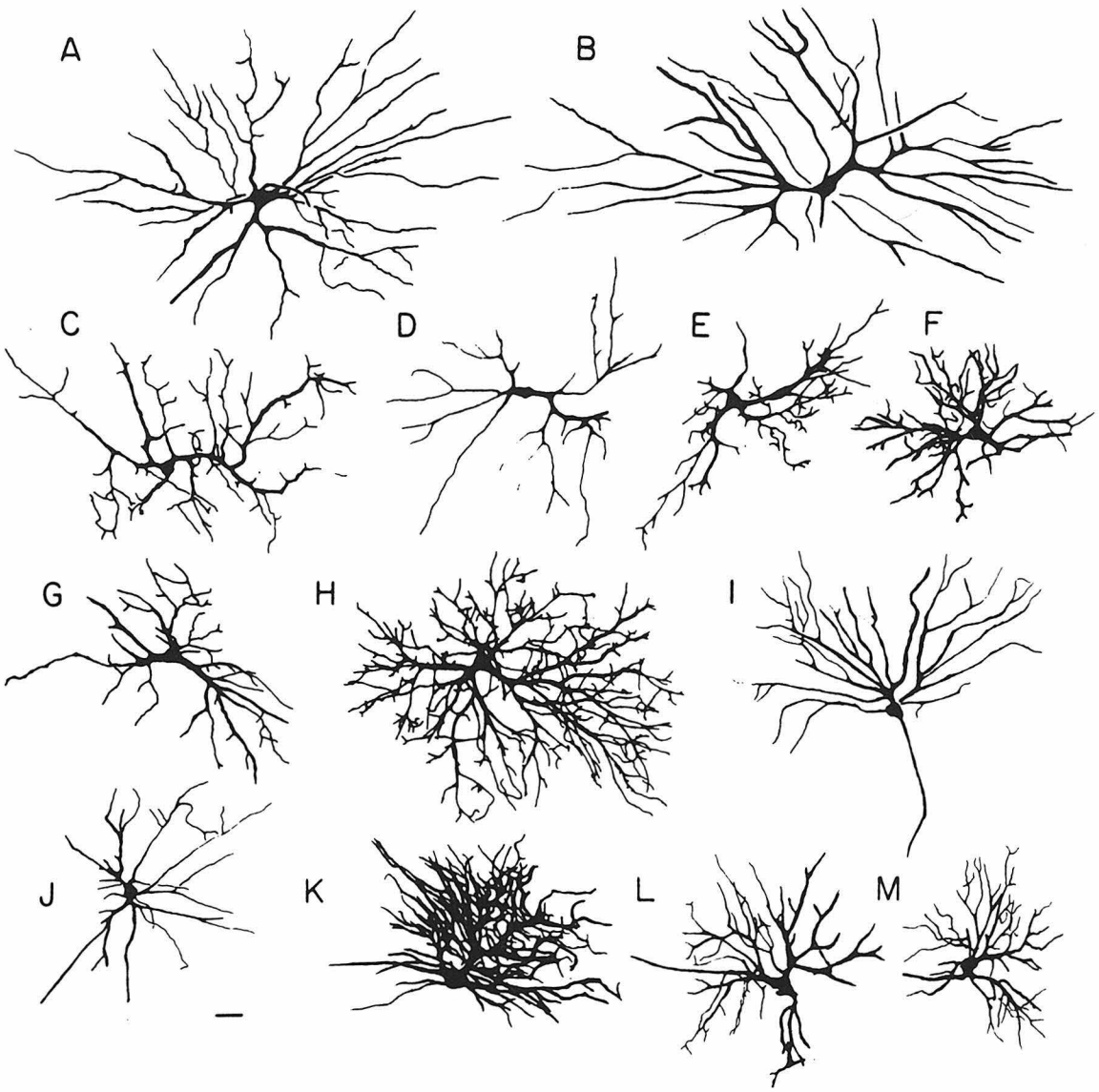


Figure 21. Large neurons from the GCL. A and D are type GCLLA. K and L represent type GCLLB. These are the hypothetical bug catchers. Calibration is 50 microns.

Neurons K and L in figure (21) are large, bistratified cells which we group as type GCLLB. They possess the following characteristics:

1) large size (250-300 microns)
2) soma football shaped, tripolar
3) large axon
4) webbed branches and medium heavy processes
5) bistratified
6) arborization primarily extends from one side of the soma and arises from three major sources on the soma, each of which forms a bistratification.
7) somewhat hemispherical shape of arbor from top view.

Figure (22) shows the small cells having their somas in the ganglion cell layer. Cells A - G are bistratified medium size neurons which we group as type GCLSB. They have the following characteristics.

1) medium size (150 - 200 microns)
2) soma spherical or ellipsoidal and multipolar
3) axon present
4) medium size smooth processes unwebbed branches
5) bistratified

6) arborization varies from round mono-lobed, symmetrical about soma, to hemispherical, mono-lobed on one side of soma, to amorphous, bilobed with one lobe on each side of soma.

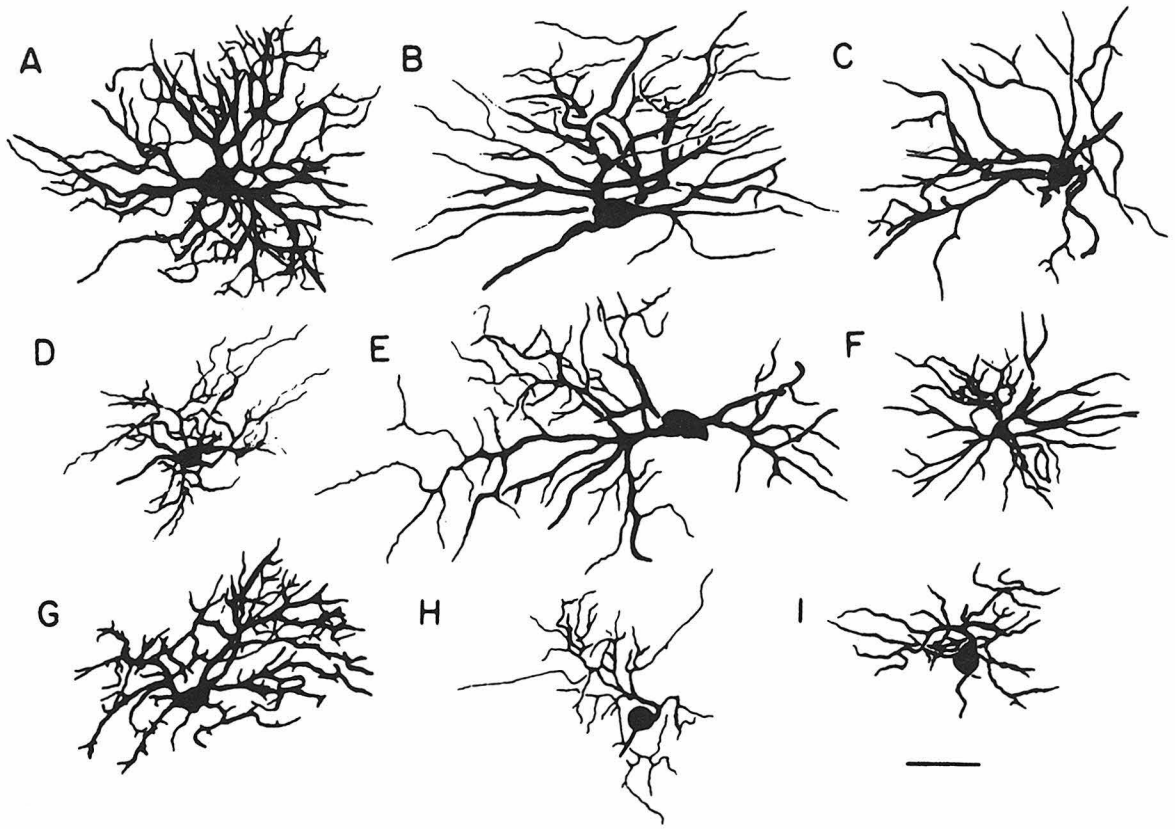


Figure 22. Small cells from the ganglion cell layer. A and G are type GCLSB. H and I represent type GCLSC. Calibration is 50 microns.

Cells H and I are small, diffuse cells having their somas in the ganglion cell layer. We group these as type GCLSC. They have the following characteristics.

1) small size (100 - 150 microns)
2) soma spherical or ellipsoidal, monopolar
3) small axon present
4) fine to medium fine processes
5) diffuse arborization (not stratified)
6) mono-lobed arborization not symmetric about soma

7.1.2 Cells with somas in inner nuclear layer (INL)

Figure (23) shows large cells whose somas lie in the INL. We do not include here the horizontal or bipolar neurons, but only those neurons classically called amacrines. Cells A - I are grouped as type INLLA. They exhibit the following characteristics.

1) large size (500 - 1000 microns)
2) soma halfmoon shaped, bipolar
3) no axon seen
4) long straight, heavy processes extending radially
   few branches and few processes.

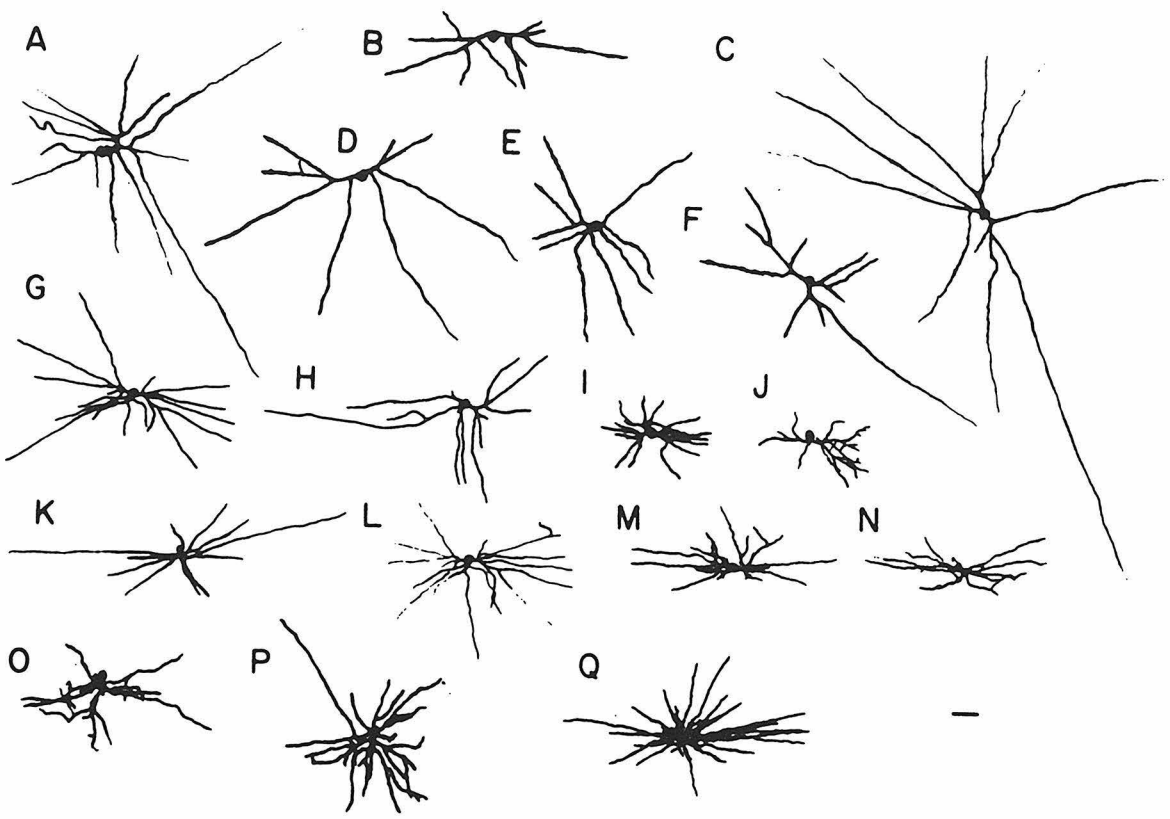


Figure 23. Large neurons from the INL. A-I represent type INLLA. K, L, N, and Q represent type INLLB. O and P represent type INLLC. Calibration is 50 microns.

5) monostratified

6) arborization usually has major axis or orientation. Processes extend occasionally parallel and also at right angles.

Cells K, L, N, and Q are grouped as type INLLB. They exhibit the following characteristics.

1) large size (300 - 400 microns)

2) soma spherical or ellipsoidal and monopolar

3) no axon seen

4) long straight fine processes extending radially.

5) monostratified

6) some have processes extending in preferred directions, some appear radially symmetric.

Cells O and P are heavy monopolar neurons grouped as type INLLC having the following characteristics.

1) large size (300 - 400 microns or more)

2) soma monopolar

3) no axon

4) heavy twisted processes

5) monostratified

6) more or less radially symmetric about the soma.

Figure (24) shows the smaller cells having somas in the INL and arborization in only the inner plexiform layer (IPL). Cells A - J are grouped as type INLSB and have the following characteristics.

1) medium size (100 - 300 microns)
2) soma spherical or pyramidal and monopolar
3) no axon seen
4) fine, highly branching crooked processes
5) monostratified; subtypes may be diffuse
6) radially symmetric about the soma, usually a circular monolobed arbor.

Cell L represents type INLSA. These ubiquitous cells have the following characteristics.

1) very small size (30 - 50 microns)
2) spherical, monopolar soma
3) no axon seen
4) short, highly branched, lumpy processes
5) diffuse branching, non-stratified
6) tightly bunched, round arborization

Cell M represents type INLSC or Dogiel's neuron. These cells are rare and differ from type INLSB only in the

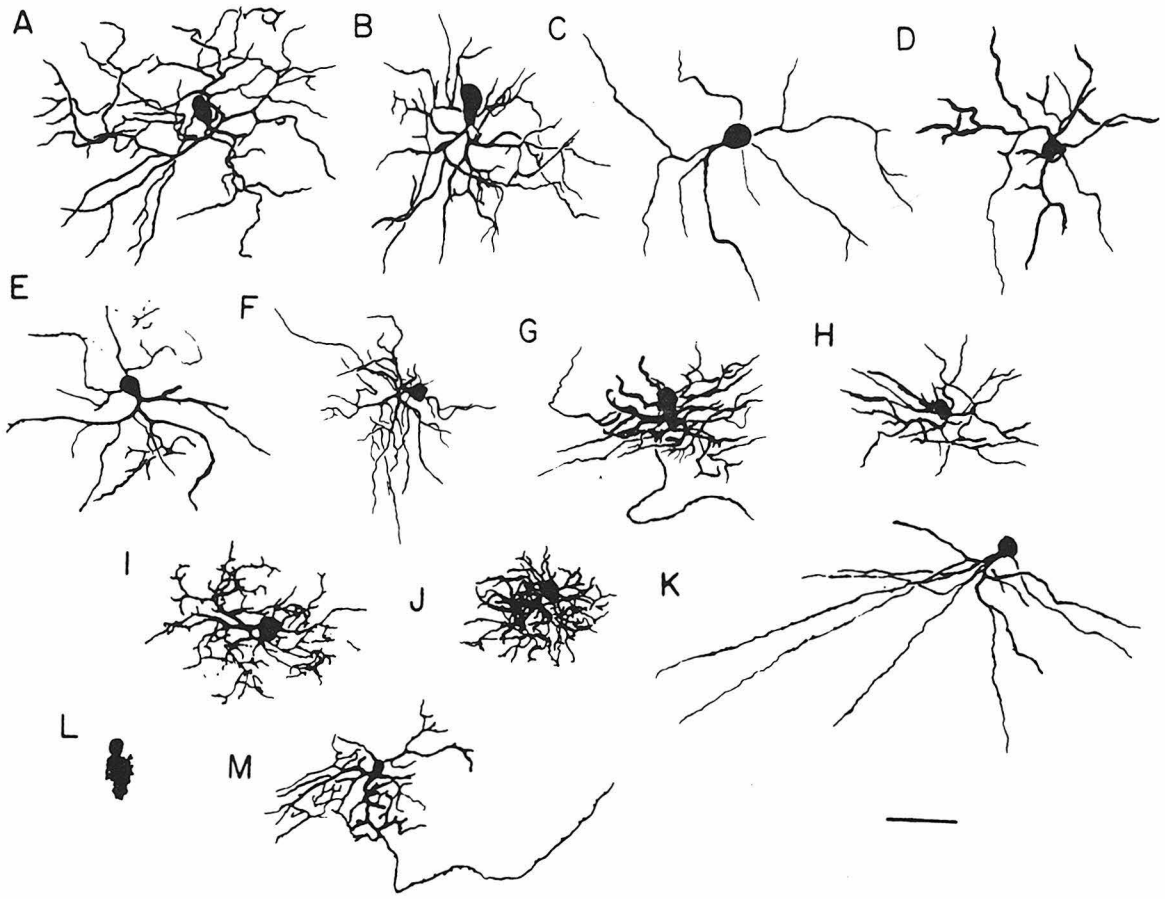


Figure 24. Small cells from the INL. A-J represent type INLSB. L represents type INLSA. M represents a possible Dogiel's neuron or type INLSC. K is unclassified. Calibration is 50 microns.

possession of an axial process which extends far from the remainder of the arborization. We were never able to trace an axial process to the axonal layer of the retina. Saggital sections would make this task easier.

7.2 Computer Morphology

Computer reconstructions of frog retinal neurons were used for several purposes. They provide top views and side views which can be used to correlate flat mount views to Cajal's side view drawings. Reconstructions are also used in the computation of quantitative parameters for objective classification and for compiling statistics on subjectively obtained neuronal categories.

7.2.1 Cajal vs. Computer

Figure (25) shows a comparison of computer drawings with Cajal's side view drawings of frog retinal ganglion cells. A shows that Cajal's single level restricted ganglions match closely with our GCLSA type neurons. The top view and two side views of the skeleton of computer reconstructed neurons are shown.

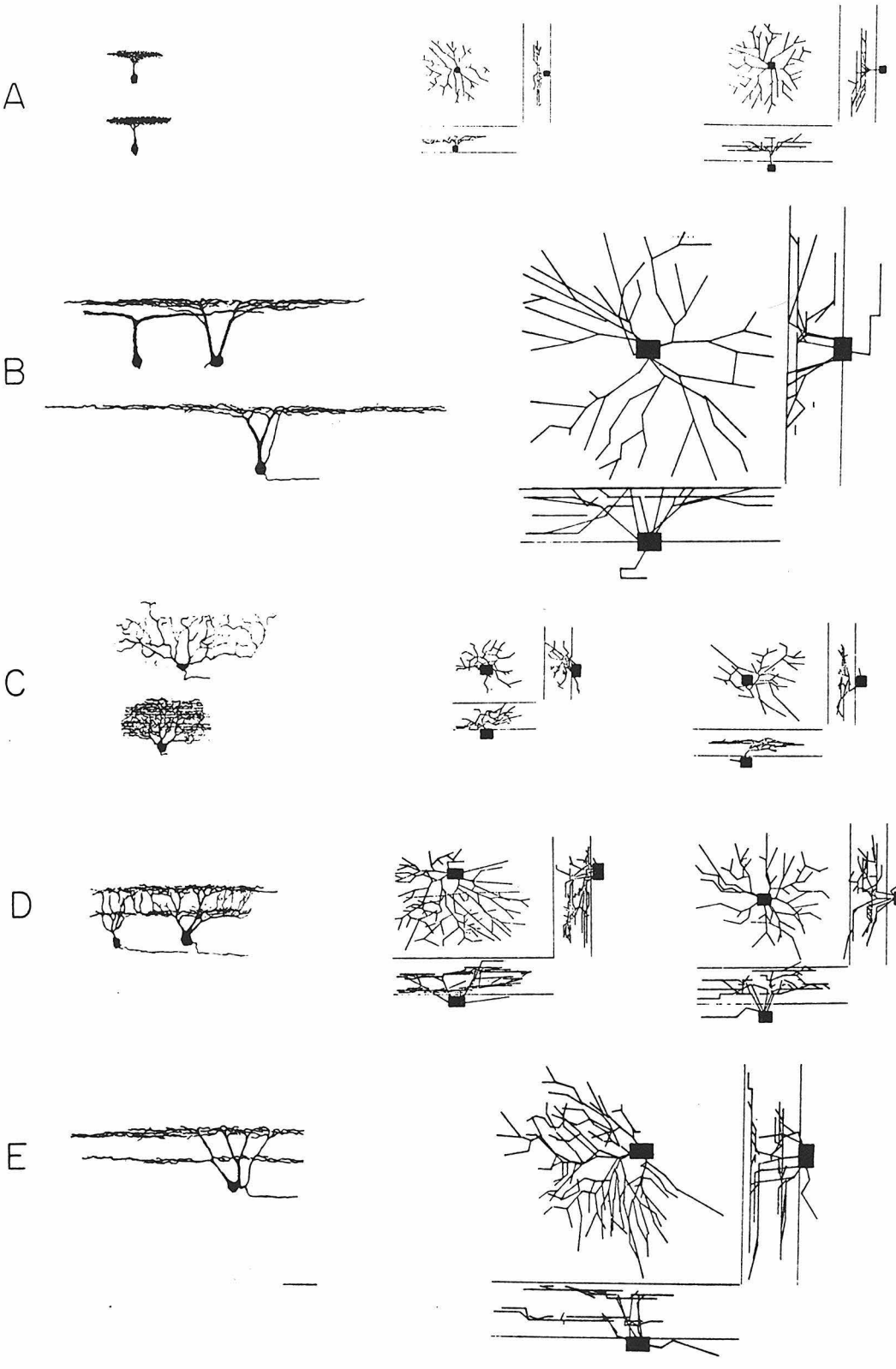


Figure 25. Computer reconstructions of GCL cells compared with Cajal's. Calibration is 50 microns.

These cells should be compared with the camera lucida drawings of figure (20). These have been speculatively termed by Lettvin (3,4)"et al" the boundary detectors.

Figure (25b) compares Cajal's single level broad cells with our type GCLLA neurons. Computer reconstructions of large neurons such as this are computed from images scanned at relatively low power and thus exhibit poor z level resolution and fluctuations in position of the IPL boundaries. For this reason the reconstruction does not show strict monostratification. These neurons may be the 'net dimming detectors' (4).

Figure (25c) shows cells having diffuse trees. Our type GCLSC neurons roughly correspond to Cajal's diffuse tree neurons. These neurons branch throughout much of the IPL showing little stratification. Lettvin and others speculate that these ganglion cells compute average light levels.

Figure (25d) shows a comparison of medium size bistratified ganglion cells. Side views of reconstructed cells show distinct bistratified properties and compare well with Cajal's medium size bistratified ganglions. These cells are our type GCLSB neurons (see figure 22A-G). This cell type has been associated speculatively with the detection of convex boundaries.

Figure (25e) shows our type GCLLB reconstruction. This

large bistratified, hemispherically lobed cell has been tentatively associated with the detection of moving edges (4).

Figure (26) and figure (27) show our mapping of reconstructed amacrines to Cajal's versions. Figure (27) shows large type INLLA, INLLB, and INLLC cells and their Cajal counterparts. The long straight processes of types INLLA and INLLB suggest that these neurons may be involved with the detection of straight edges or bars.

Figure (26a) shows type INLSB neurons. These cells are monostratified neurons which show a great variation in size and in their level of arborization. The top view of their reconstructions exhibits varying extents of unidirectionality and all are rather densely branching structures.

Figure (26b) shows type INLSA neurons. These cells are very small clumpy monopolar units with relatively diffuse branching throughout several layers of the IPL.

The above discussion completes our correlation of computer reconstructions with Cajal's drawings and suggests a subjective classification of cells corresponding to the various types seen by Cajal. Plates I - X at the end of this chapter show examples of stereo pairs of selected neurons as computed from raw optical sections.

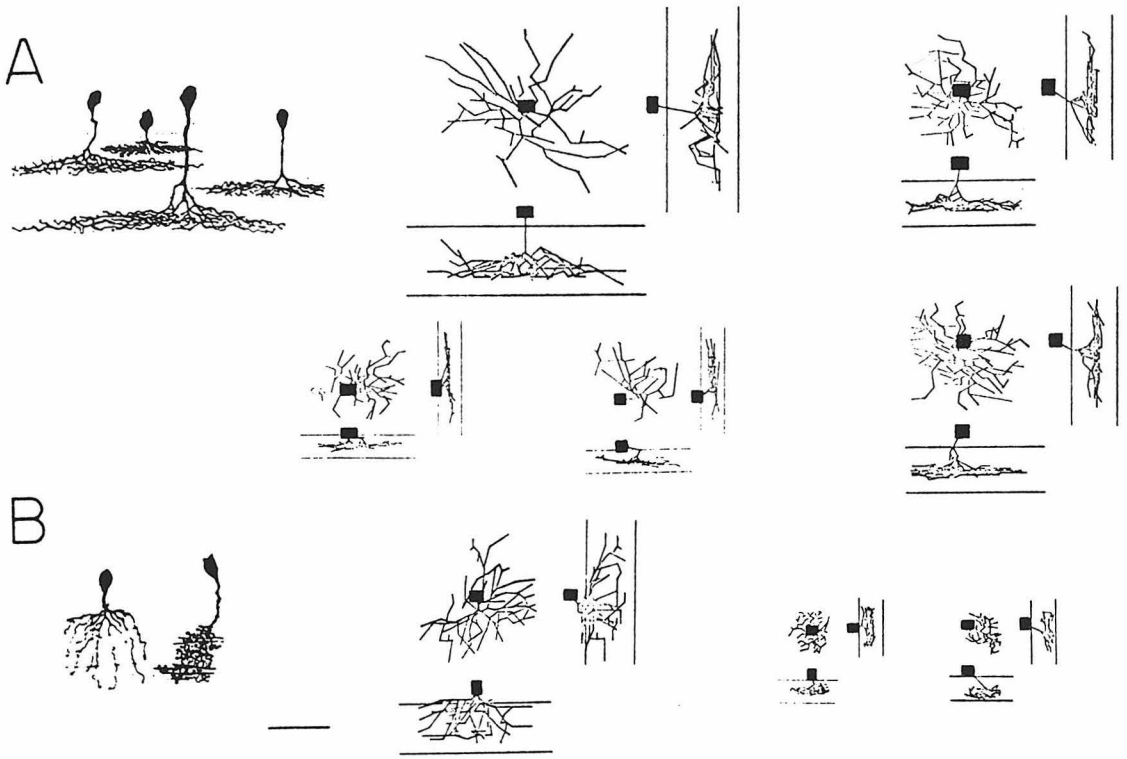


Figure 26. Computer versions of small cells from INL. A represents type INLSB. B represents type INLSA. Calibration is 50 microns.

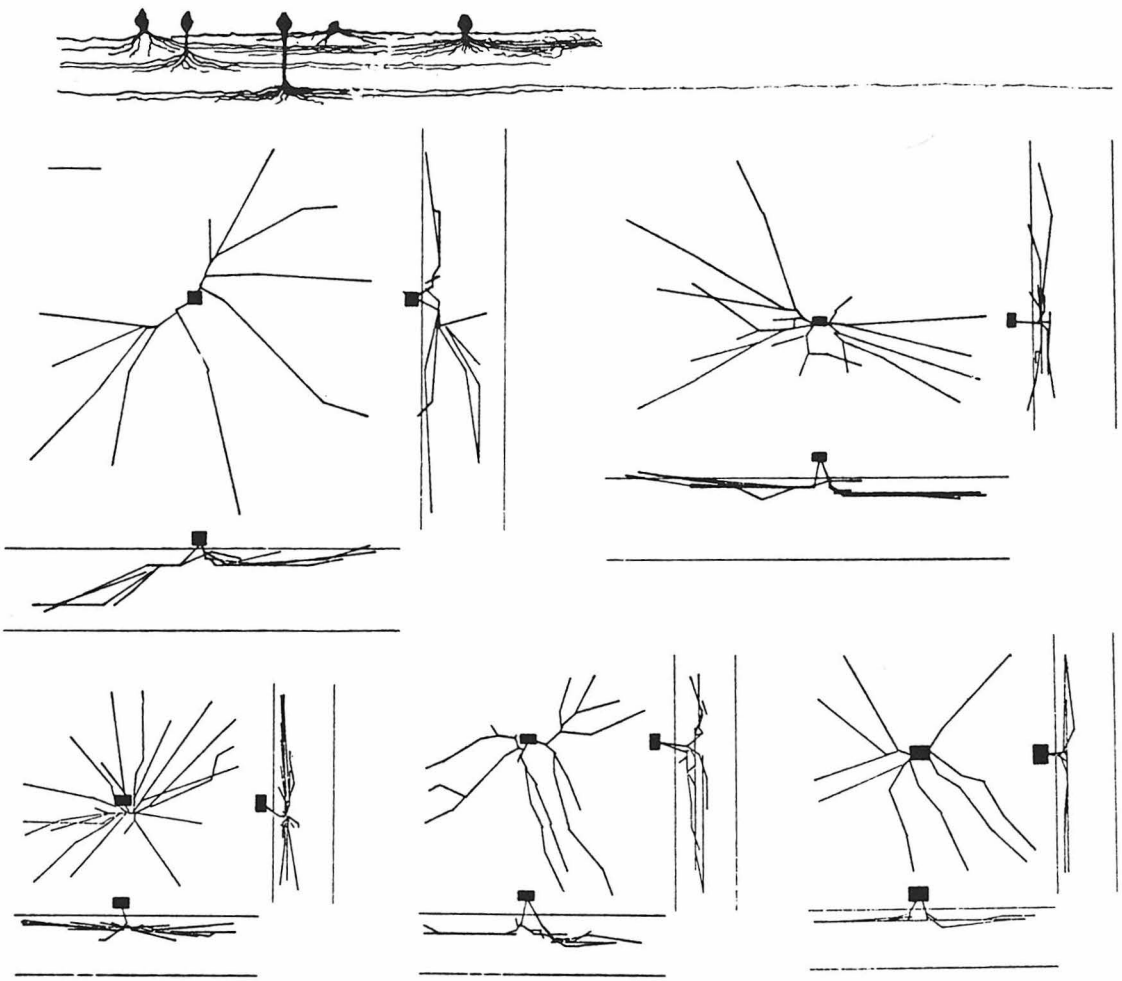


Figure 27. Computer versions of large INL neurons compared with Cajal's. Calibration 50 microns.

We shall now discuss the results of a quantitative classification based on the parameters mentioned in section 6.1.

7.2.2 Quantitative Structural Analysis

We reconstructed 57 neurons from the frog retina and extracted 21 parameters from each cell. Figure (28) summarizes a quantitative breakdown of this cell population into types by viewing the clustering of cell parameters in various spaces. Although a much larger sample of cells is needed to perform a convincing classification we shall briefly describe our results. We first trivially partition the total population according to position of soma yielding a set of cells from the INL and a set from the GCL. The INL cells can be partitioned on the basis of size into two groups, INLBIG and INLSMALL. Group INLSMALL clusters again on the basis of size into the types which we have discussed subjectively above, namely INLSA and INLSB (see section 7.1.2). We can also partition the GCL cells according to size into two groups, GCLBIG and GCLSMALL. GCLBIG contains several examples of type GCLLA (cells 59,96, and 68) and one example of type GCLLB (cell 101). The GCLSMALL neurons can be further partitioned according to stratification properties into monostratified (GCLSA) and a remaining group

including the small bistratified and diffuse ganglion cells.

The mean and variance of all parameters for all cell types have been computed and are included for reference in tables (2-11) at the end of this chapter.

Figure 28. Clustering of neurons by numerical parameters. Par n refers to parameter xn as explained in section 6.1 (see text).

A marked difference clearly arises between the classification by quantitative techniques and the classification by the human observer. In the case of flat mount views of neurons, humans seem to classify the cells according to **occurrences** of structural subunits. A particular form of branch or soma shape is used as a handle for comparison. For example, the monostratified small field type GCLSA neuron has a particularly distinctive top view appearance characterized by symmetry and wide angle branching, the frequency of which appears to increase with distance from the cell body. However, none of our global parameters dealing with distribution of branch points or average orientation of processes was able to capture this apparently distinct feature well enough to cluster these neurons out. The human method of classification by **occurrence** of abstract substructural features seems especially robust against those variations in size, shape and relative orientation to which parametric classification schemes appear quite sensitive. On the basis of this observation it would seem that classification attempts which deal with complex shapes such as neurons should employ both methods. The method described in section 6.2 is an attempt to build a quantitative classification scheme which is analogous to that used by human observers.

REFERENCES

1. Botosch, A., (1975), personal communication.

2. Cajal, R. -y- (1892), The structure of the retina, translated by S. A. Thorpe and M. Glickstein, (1972), C. C. Thomas publisher, Springfield, Ill.

3. Lettvin, J. Y., H. R. Maturana, W. S. McCulloch, and Pitts, (1959), What the frogs eye tells the frogs brain, Proc. IRE, November, p. 1940.

4. -----, (1961), Two remarks on the visual system of the frog , in Sensory communication, W. A. Rosenblith ed., Wiley, N. Y. p. 757.

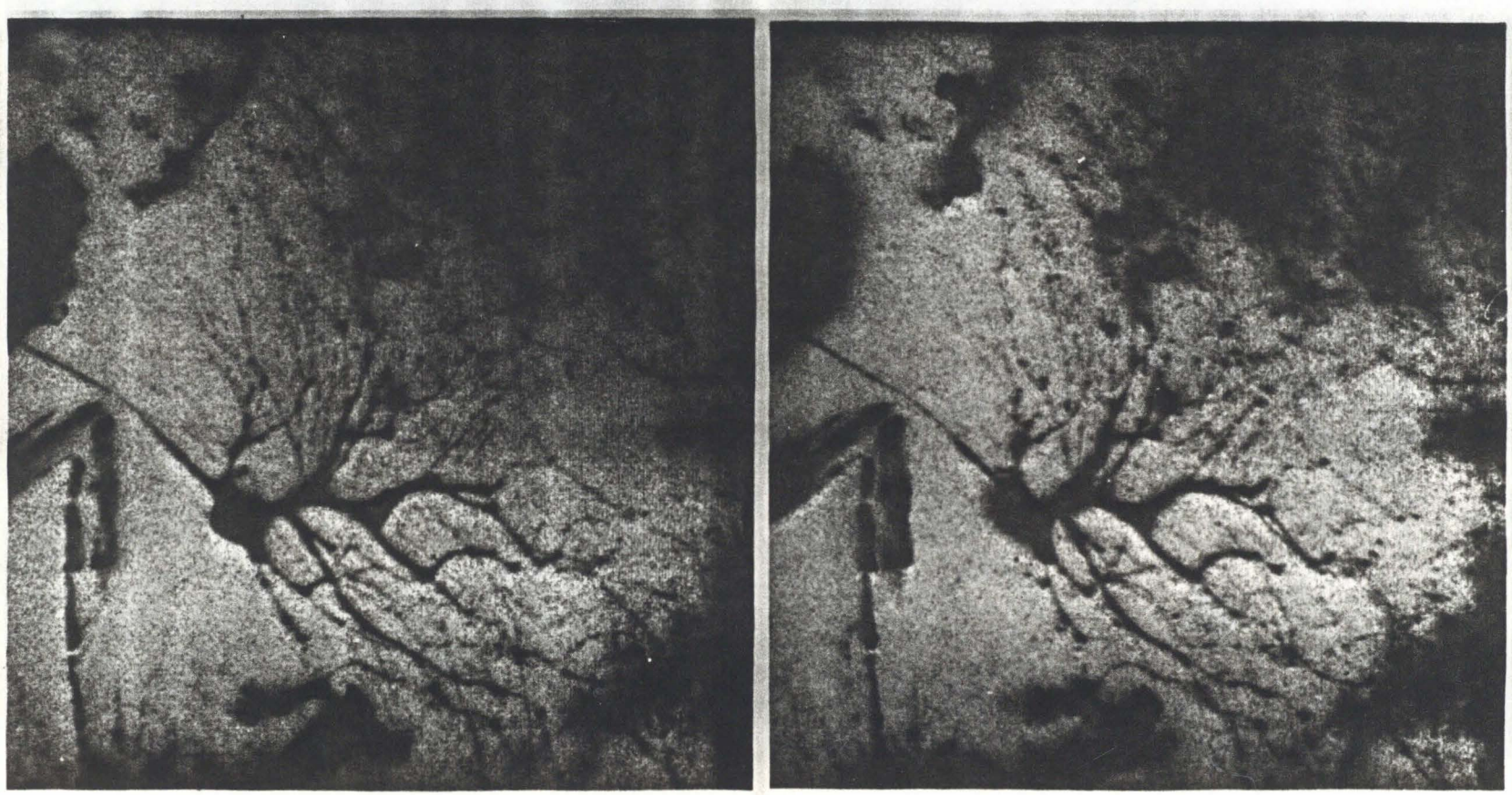


Plate I. Stereo pair of type GCLLB neuron. Full scale is 400 microns.

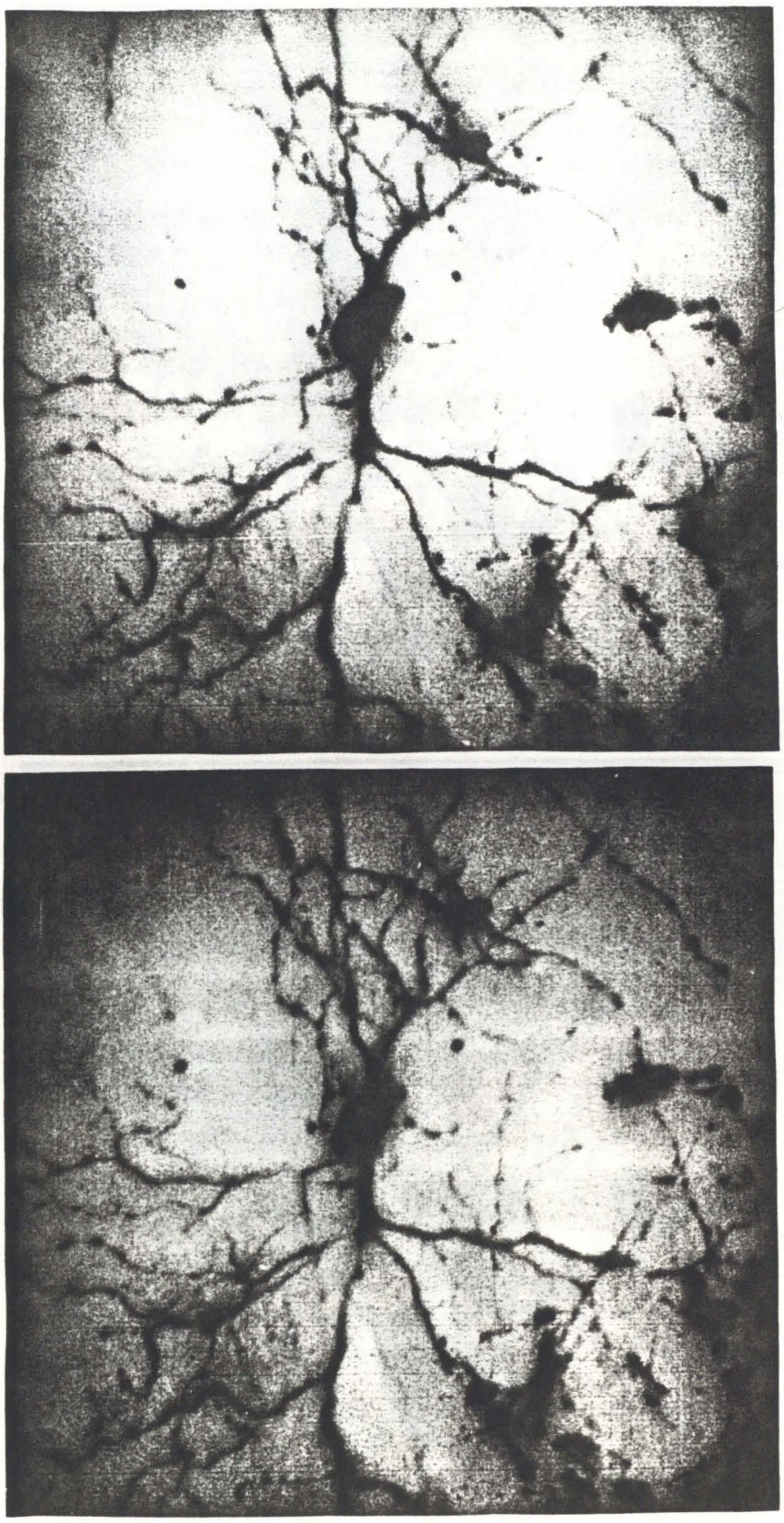


Plate III. Stereo pair of type GCLSB neuron.

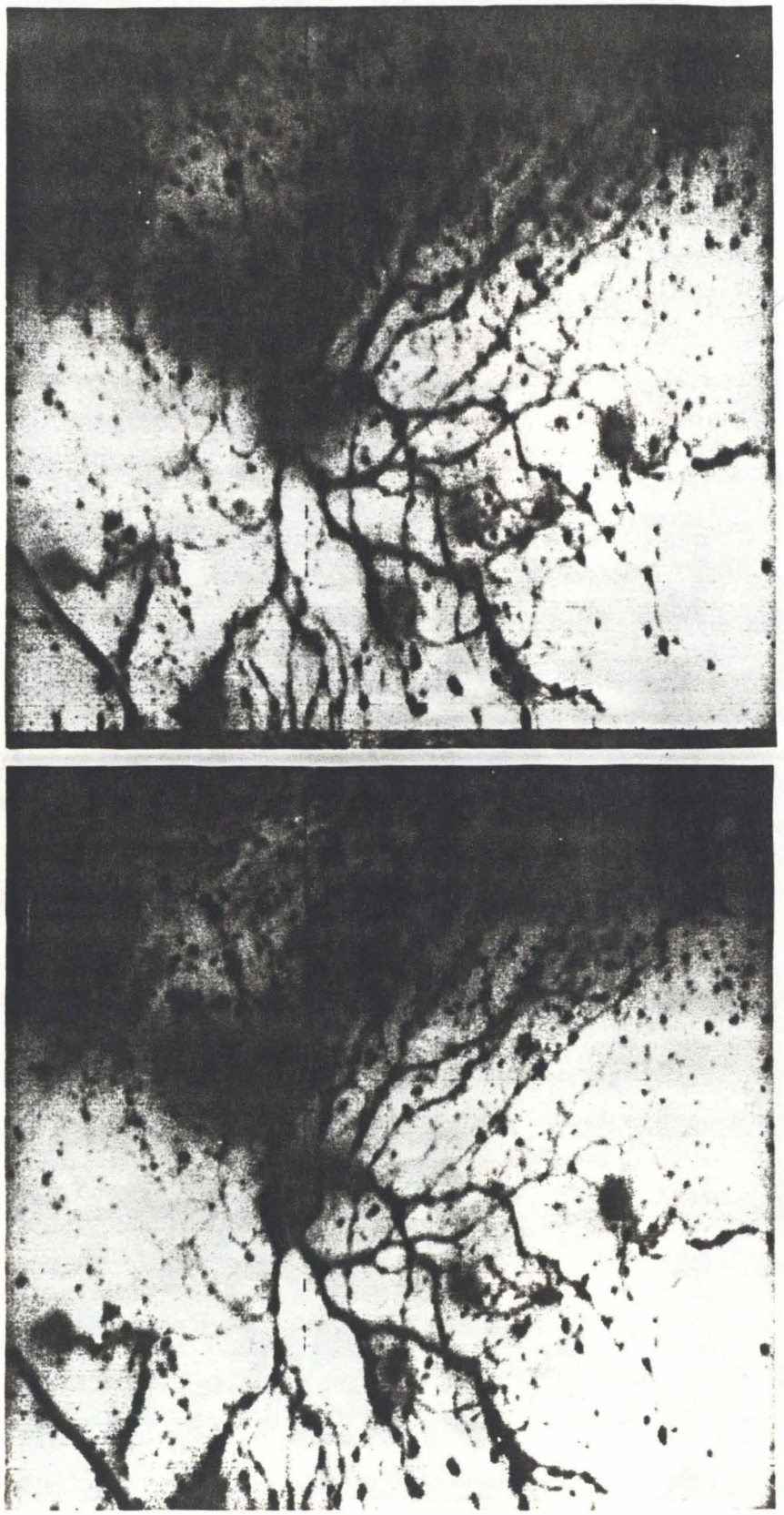


Plate IV. Stereo pair of type GCLSB neuron.

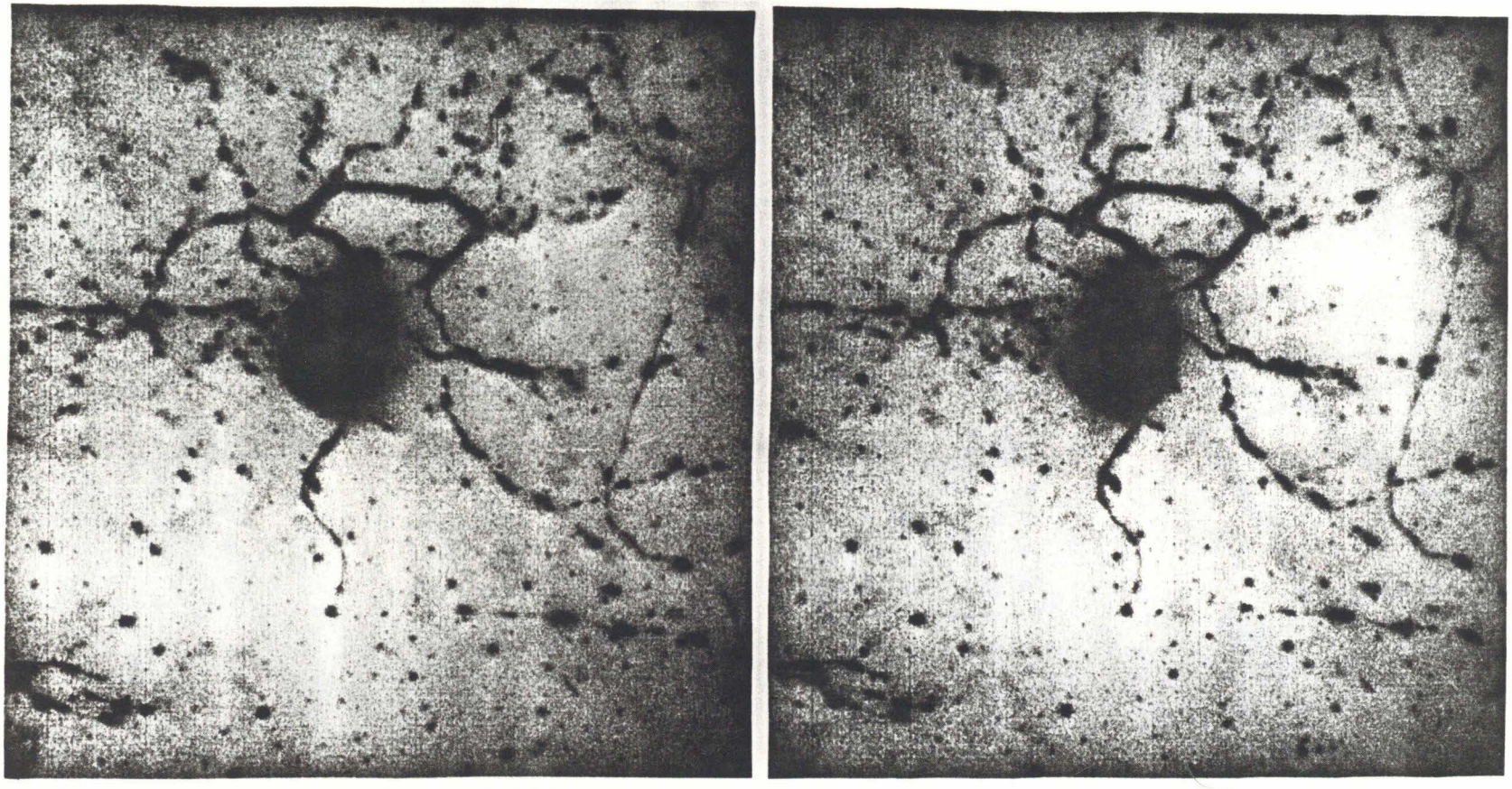


Plate V. Stereo pair of type GCLSC neuron.

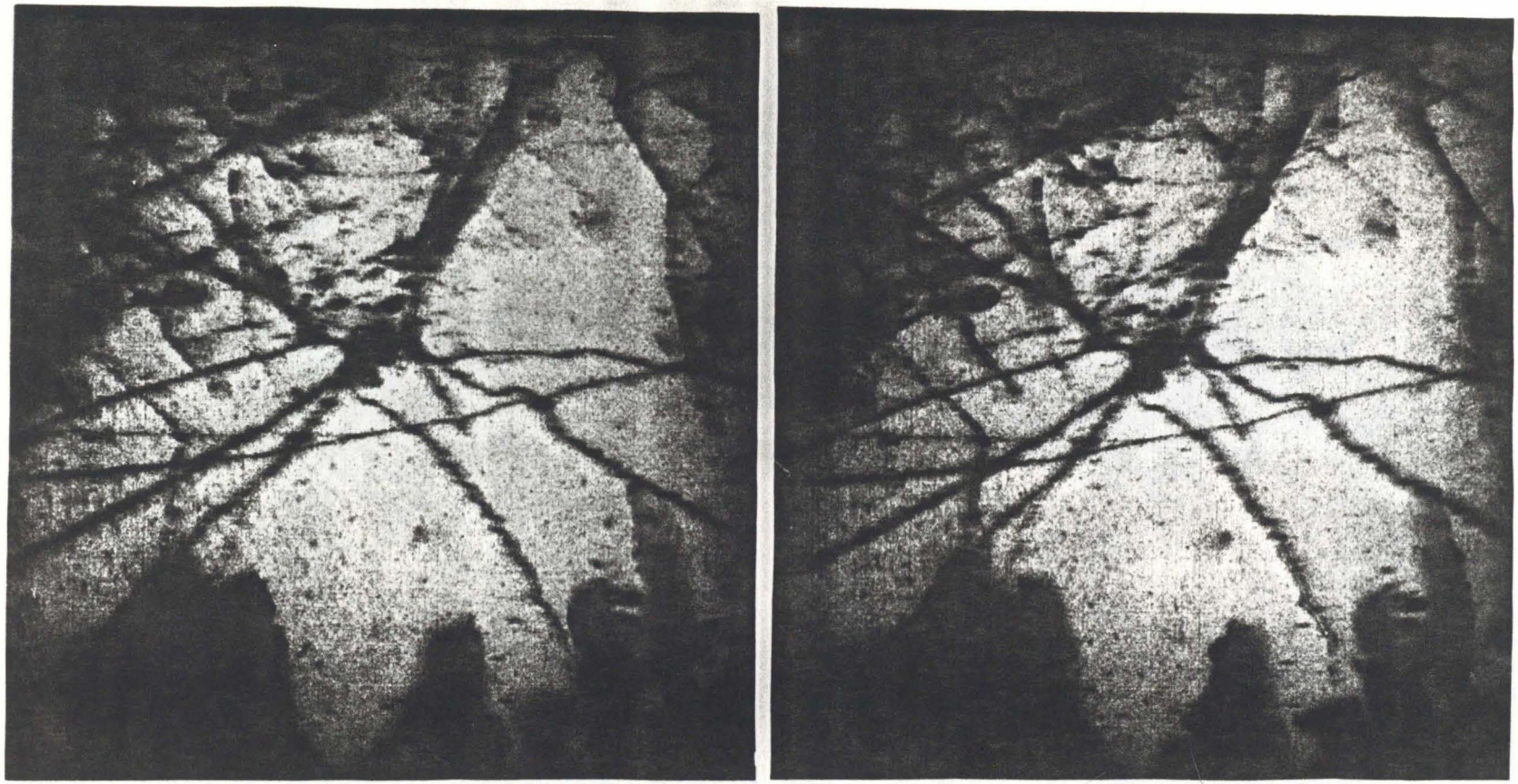


Plate VI. Stereo pair of type INLLA neuron.

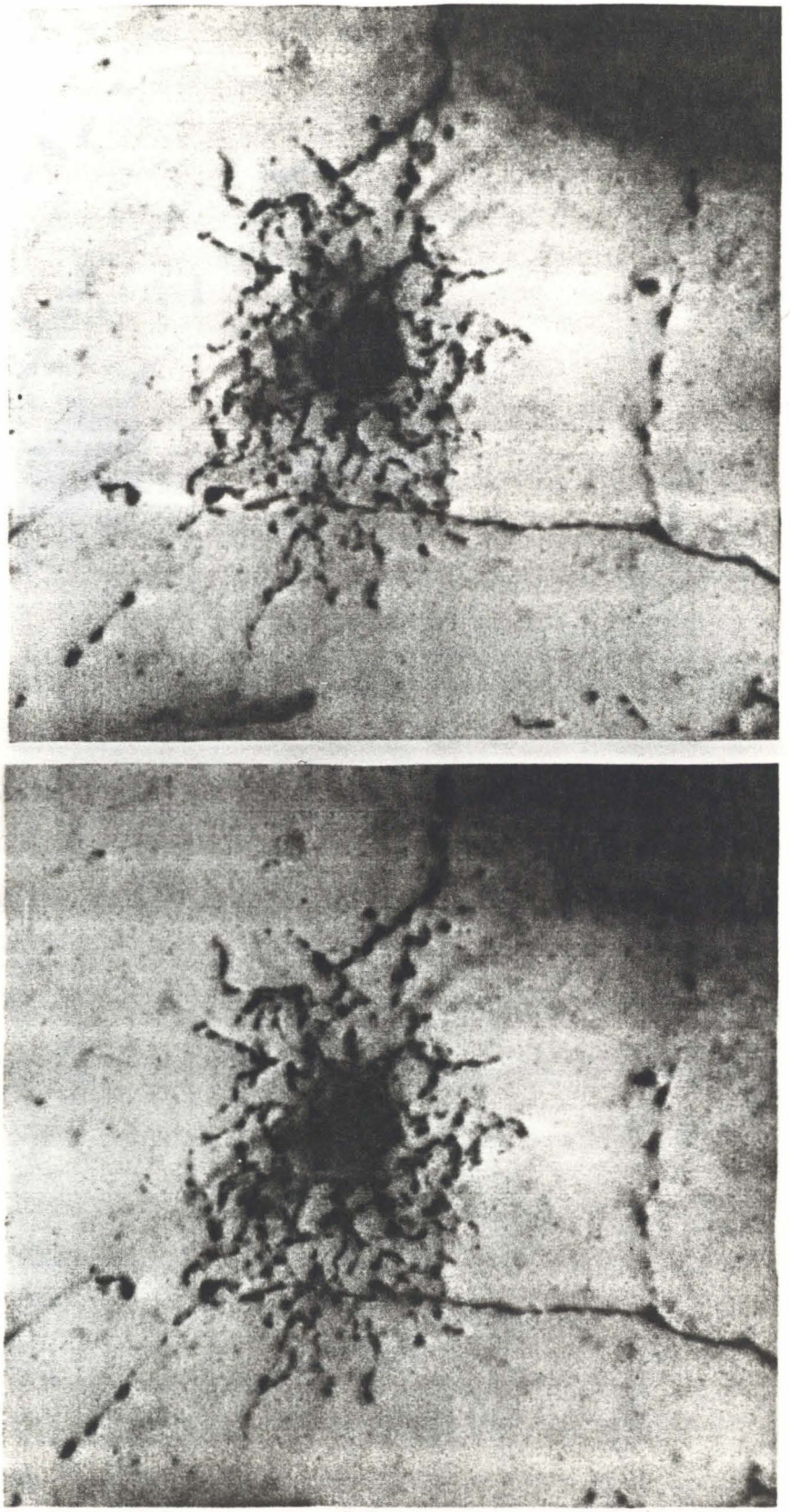


Plate VII. Stereo pair of type INLSB neuron.

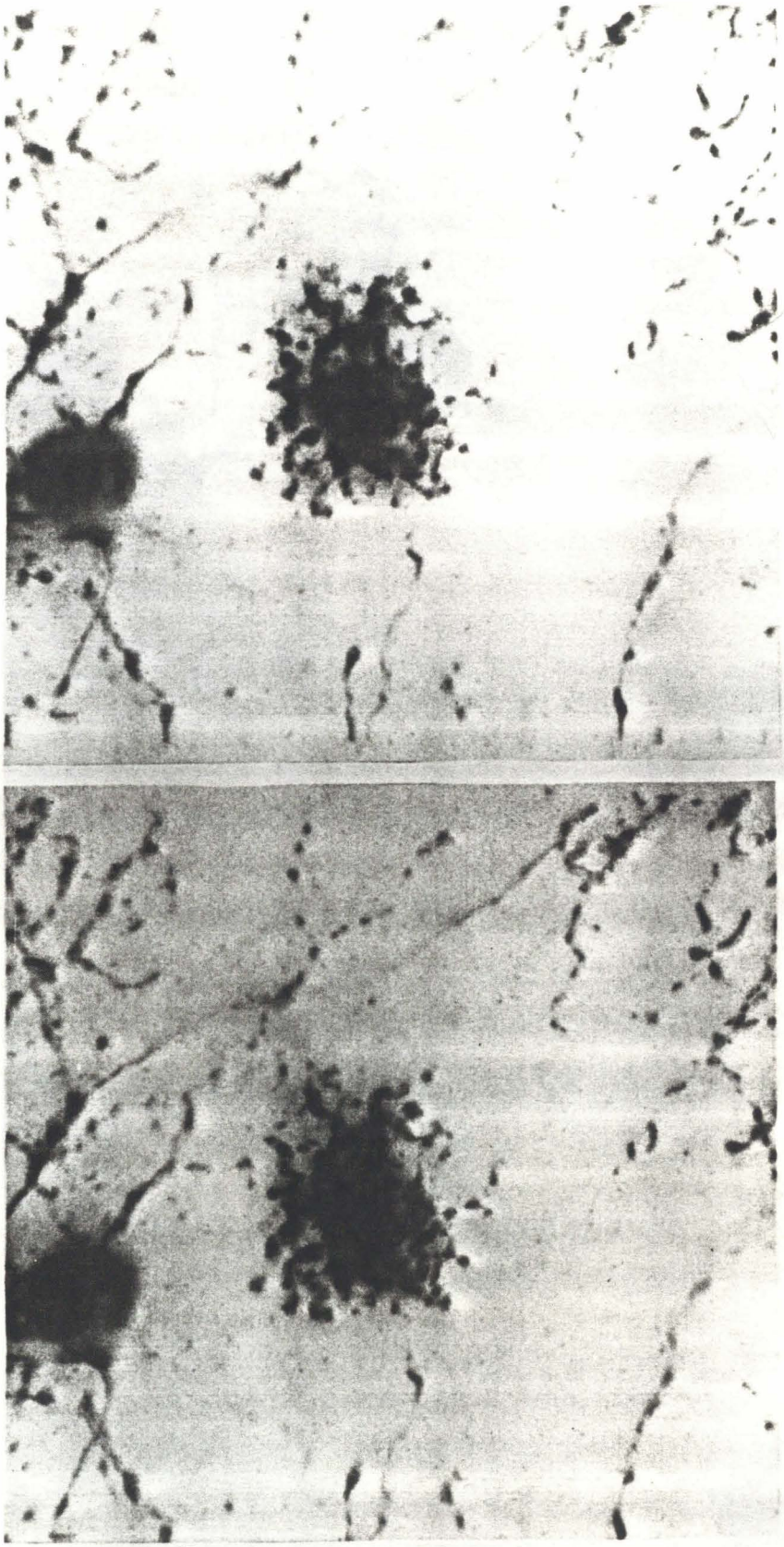


Plate VIII. Stereo pair of type INLSA neuron.

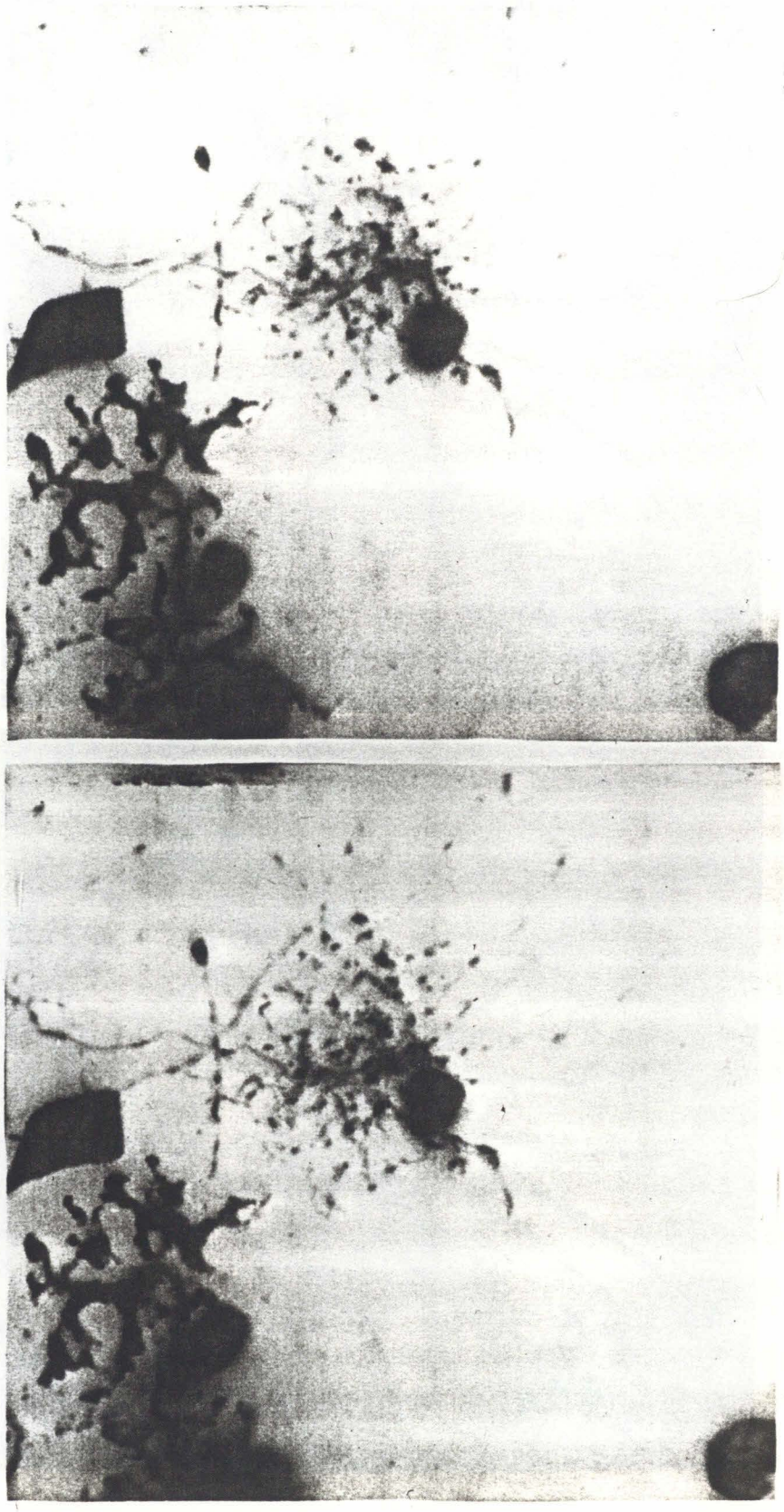


Plate IX. Stereo pair of type INLSB neuron.

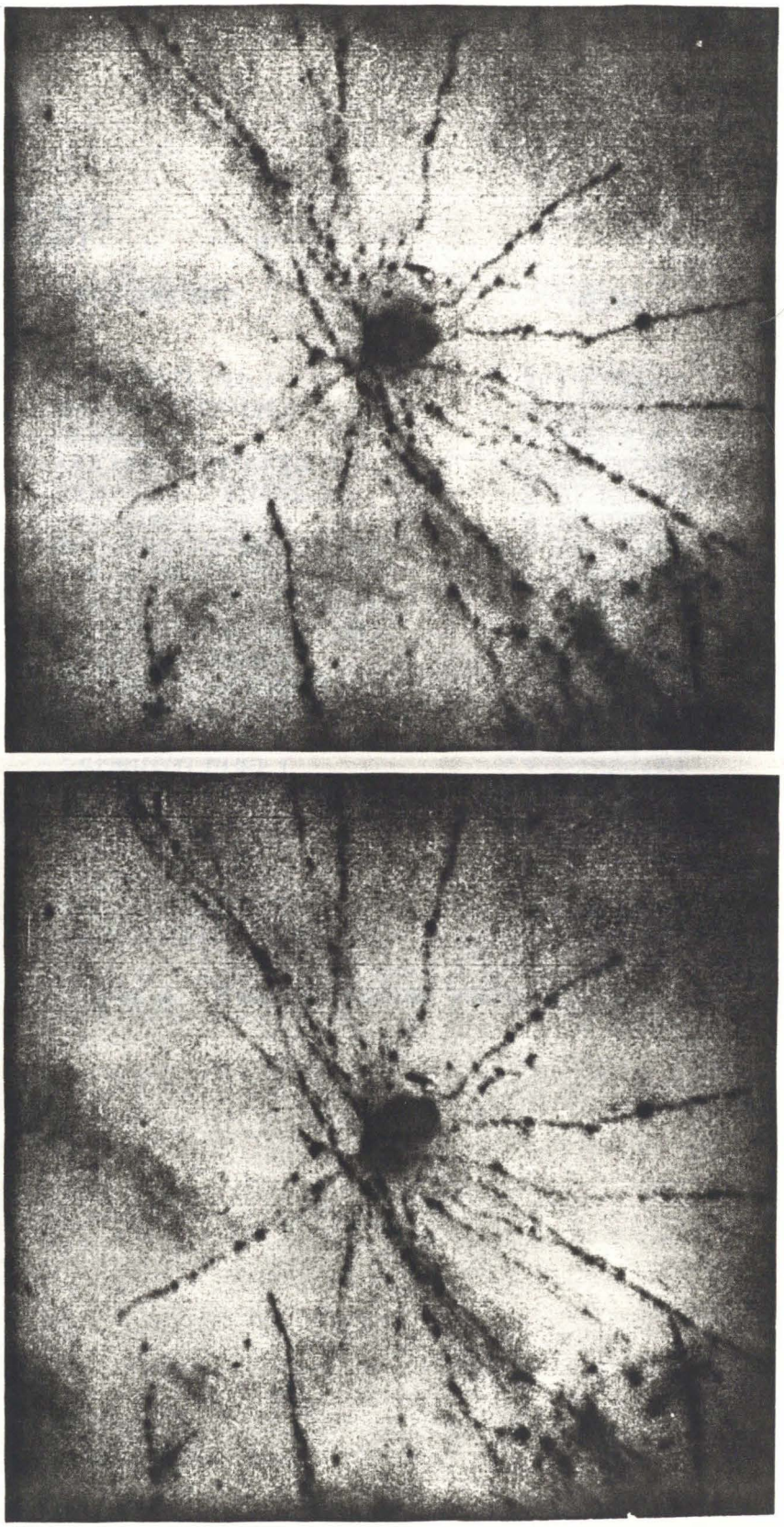


Plate X. Stereo pair of type INLLB neuron.

GOLBIG
COUNT 4

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	3.1525E+02	3.5222E+03
SOMATHIN(.1M)	2.2375E+02	1.1377E+03
SOMAZ(M)	1.4500E+02	7.4500E+01
VOLUME(.1M**3)	2.8510E+03	9.9925E+05
SURFACE(.1M**2)	2.8730E+03	9.6410E+05
AVGDIAM(.01M)	3.8925E+02	5.2382E+03
MAXSOMATOTERM(.1M)	2.2488E+03	4.3386E+04
AVGSOMATOTERM(.1M)	1.4882E+03	7.8425E+04
AVGCENTERTOTERM(.1M)	1.3655E+03	9.6054E+04
AVGCENTERTOBR(.1M)	6.6500E+02	3.3438E+04
BRDEVIATION(M)	3.3250E+01	8.4688E+01
TERMDEVIATION(M)	3.9750E+01	4.5687E+01
AVGCIRCULARCOMP(.1M)	2.3475E+02	3.3192E+03
AVGRADIALCOMP(.1M)	2.8050E+02	4.4147E+03
%SURFATLEVEL1	3.8750E+01	6.2569E+02
%SURFATLEVEL2	1.7250E+01	7.6875E+00
%SURFATLEVEL3	1.0500E+01	1.2875E+02
%SURFATLEVEL4	1.8250E+01	1.4569E+02
%SURFATLEVEL5	6.7500E+00	3.9687E+01
MAX%OFFDIRECT	3.8000E+01	3.0000E+01
MAX%RIGHTANGLES	3.0000E+01	5.0000E-01

Table 2. Feature vector statistics of 4 large ganglions.

GCLSMALL
COUNT: 14

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	1.7643E+02	4.1355E+03
SOMATHIN(.1M)	1.3736E+02	8.8109E+02
SOMAZ(M)	1.5821E+02	1.0760E+02
VOLUME(.1M**3)	6.3614E+02	1.7708E+05
SURFACE(.1M**2),	9.3957E+02	2.5888E+05
AVGDIAM(.01M)	2.4393E+02	7.4894E+03
MAXSOMATOTERM(.1M)	1.2176E+03	1.2607E+05
AVGSOMATOTERM(.1M)	8.1229E+02	5.0559E+04
AVGCENTERTOTERM(.1M)	6.7186E+02	4.5820E+04
AVGCENTERTOBR(.1M)	3.6421E+02	1.2732E+04
BRDEVIATION(M)	1.5857E+01	2.6265E+01
TERMDEVIATION(M)	2.1143E+01	9.9122E+01
AVGCIRCULARCOMP(.1M)	1.4236E+02	1.7491E+03
AVGRADIALCOMP(.1M)	1.0850E+02	3.4265E+03
%SURFATLEVEL1	9.2857E+00	3.5349E+02
%SURFATLEVEL2	1.5786E+01	1.7545E+02
%SURFATLEVEL3	2.7214E+01	4.5017E+02
%SURFATLEVEL4	2.0429E+01	1.3839E+02
%SURFATLEVEL5	1.1000E+01	1.3243E+02
MAX%OPPDIRECT	3.5286E+01	5.4633E+01
MAX%RIGHTANGLES	2.9500E+01	1.9964E+01

Table 3. Feature vector statistics for a group of 14 small ganglions.

GCLSA
COUNT: 4

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	1.1275E+02	1.0222E+03
SOMATHIN(.1M)	1.0950E+02	4.2525E+02
SOMAZ(M)	1.5850E+02	9.6250E+01
VOLUME(.1M**3)	3.1400E+02	2.8713E+04
SURFACE(.1M**2)	6.1650E+02	5.9681E+04
AVGDIAM(.01M)	1.8800E+02	1.4985E+03
MAXSOMATOTERM(.1M)	9.6850E+02	6.5522E+04
AVGSOMATOTERM(.1M)	6.9050E+02	2.9039E+04
AVGCENTERTOTERM(.1M)	5.7175E+02	1.7028E+04
AVGCENTERTOBR(.1M)	3.5050E+02	7.3712E+03
BRDEVIATION(M)	1.5750E+01	6.6875E+00
TERMDEVIATION(M)	1.5250E+01	3.3688E+01
AVGCIRCULARCOMP(.1M)	1.0975E+02	9.4219E+02
AVGRADIALCOMP(.1M)	8.0250E+01	3.5569E+02
%SURFATLEVEL1	0.0	0.0
%SURFATLEVEL2	1.0250E+01	8.3188E+01
%SURFATLEVEL3	5.6500E+01	8.7250E+01
%SURFATLEVEL4	2.3750E+01	1.9687E+01
%SURFATLEVEL5	1.5000E+00	2.5000E-01
MAXOPPDIRECT	3.8500E+01	3.6750E+01
MAXRIGHTANGLES	3.1000E+01	6.5000E+00

Table 4. Feature vector statistics for 4 type GCLSA neurons.

GCLSB
COUNT: 10

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	2.0190E+02	3.1101E+03
SOMATHIN(.1M)	1.4850E+02	6.2885E+02
SOMAZ(M)	1.5810E+02	1.1209E+02
VOLUME(.1M**3)	7.6500E+02	1.7831E+05
SURFACE(.1M**2),	1.0688E+03	2.8011E+05
AVGDIAM(.01M)	2.7330E+02	7.8068E+03
MAXSOMATOTERM(.1M)	1.3172E+03	1.1555E+05
AVGSOMATOTERM(.1M)	8.6100E+02	5.0861E+04
AVGCENTERTOTERM(.1M)	7.1190E+02	5.1725E+04
AVGCENTERTOBR(.1M)	3.6970E+02	1.4771E+04
BRDEVIATION(M)	1.5900E+01	3.4090E+01
TERMDEVIATION(M)	2.3500E+01	1.0585E+02
AVGCIRCULARCOMP(.1M)	1.5540E+02	1.4764E+03
AVGRADIALCOMP(.1M)	1.1980E+02	4.2090E+03
%SURFATLEVEL1	1.3000E+01	4.4660E+02
%SURFATLEVEL2	1.8000E+01	1.9320E+02
%SURFATLEVEL3	1.5500E+01	1.1505E+02
%SURFATLEVEL4	1.9100E+01	1.7969E+02
%SURFATLEVEL5	1.4800E+01	1.3476E+02
MAX%OPPDIRECT	3.4000E+01	5.6000E+01
MAX%RIGHTANGLES	2.8900E+01	2.4090E+01

Table 5. Feature vector statistics for a group of 10 type GCLSB neurons.

INLBIG
COUNT= 15

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	1.6773E+02	6.9993E+02
SOMATHIN(.1M)	1.2760E+02	1.0438E+03
SOMAZ(M)	8.7533E+01	3.9449E+01
VOLUME(.1M**3)	7.2740E+02	1.5574E+05
SURFACE(.1M**2)	1.0005E+03	1.8007E+05
AVGDIAM(.01M)	2.8127E+02	4.0925E+03
MAXSOMATOTERM(.1M)	1.8811E+03	1.6017E+05
AVGSOMATOTERM(.1M)	1.2497E+03	1.2776E+05
AVGCENTERTOTERM(.1M)	1.1579E+03	1.4011E+05
AVGCENTERTOBR(.1M)	2.8367E+02	1.1063E+04
BROEVIATION(M)	1.4467E+01	8.3582E+01
TERMDEVIATION(M)	3.5800E+01	1.5656E+02
AVGCIRCULARCOMP(.1M)	2.9840E+02	1.8047E+04
AVGRADIALCOMP(.1M)	2.7787E+02	1.8083E+04
%SURFATLEVEL1	2.4600E+01	6.7477E+02
%SURFATLEVEL2	2.4067E+01	3.5953E+02
%SURFATLEVEL3	1.7000E+01	3.5293E+02
%SURFATLEVEL4	1.2867E+01	3.4478E+02
%SURFATLEVEL5	4.6667E+00	4.0439E+01
MAX%OPPDIRECT	3.5467E+01	4.3982E+01
MAX%RIGHTANGLES	3.2800E+01	2.2693E+01

Table 6. Feature vector statistics for a group of 15 large INL neurons.

INLSMALL
COUNT: 24

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	1.2254E+02	8.4958E+02
SOMATHIN(.1M)	1.0258E+02	3.1549E+02
SOMAZ(M)	8.4625E+01	5.6068E+01
VOLUME(.1M**3)	1.4408E+02	1.4840E+04
SURFACE(.1M**2)	3.2067E+02	3.6418E+04
AVGDIAM(.01M)	1.6462E+02	2.8831E+03
MAXSOMATOTERM(.1M)	7.7733E+02	4.0236E+04
AVGSOMATOTERM(.1M)	5.6433E+02	1.3616E+04
AVGCENTERTOTERM(.1M)	2.9892E+02	1.6835E+04
AVGCENTERTOBR(.1M)	1.6792E+02	2.6955E+03
BRDEVIATION(M)	7.4583E+00	1.1665E+01
TERMDEVIATION(M)	1.0000E+01	3.5833E+01
AVGCIRCULARCOMP(.1M)	8.5292E+01	1.0786E+03
AVGRADIALCOMP(.1M)	3.3083E+01	1.1939E+03
%SURFATLEVEL1	1.0000E+01	1.7342E+02
%SURFATLEVEL2	2.2042E+01	2.7329E+02
%SURFATLEVEL3	3.3000E+01	3.0117E+02
%SURFATLEVEL4	1.7667E+01	1.7639E+02
%SURFATLEVEL5	2.9167E+00	3.6910E+01
MAX%OPPDIRECT	3.2583E+01	3.0493E+01
MAX%RIGHTANGLES	3.1167E+01	3.6472E+01

Table 7. Feature vector statistics for a group of 24 small INL neurons.

INLSA
COUNT: 7

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	1.0257E+02	5.8767E+02
SOMATHIN(.1M)	9.9286E+01	4.2906E+02
SOMAZ(M)	8.2571E+01	8.5102E+01
VOLUME(.1M**3)	6.5429E+01	1.1925E+03
SURFACE(.1M**2)	1.5371E+02	9.6620E+02
AVGDIAM(.01M)	1.6600E+02	3.7991E+03
MAXSOMATOTERM(.1M)	6.0029E+02	7.6785E+03
AVGSOMATOTERM(.1M)	4.5829E+02	4.0925E+03
AVGCENTERTOTERM(.1M)	1.5786E+02	1.0012E+02
AVGCENTERTOBR(.1M)	1.1729E+02	1.9061E+01
BRDEVIATION(M)	4.0000E+00	5.7143E-01
TERMDEVIATION(M)	4.4286E+00	2.4490E-01
AVGCIRCULARCOMP(.1M)	5.4000E+01	1.0743E+02
AVGRADIALCOMP(.1M)	3.8571E+00	5.2122E+01
%SURFATLEVEL1	8.8571E+00	3.5837E+01
%SURFATLEVEL2	2.8143E+01	5.4384E+02
%SURFATLEVEL3	2.5714E+01	1.2020E+02
%SURFATLEVEL4	1.4857E+01	2.3612E+02
%SURFATLEVEL5	2.8571E+00	3.4408E+01
MAX%OPPDIRECT	2.6571E+01	3.3878E+00
MAX%RIGHTANGLES	2.7857E+01	2.2980E+01

Table 8. Feature vector statistics for a group of 7 type INLSA neurons.

INLSB
COUNT 17

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	1.3076E+02	7.2559E+02
SOMATHIN(.1M)	1.0394E+02	2.6241E+02
SOMAZ(M)	8.5471E+01	4.1661E+01
VOLUME(.1M**3)	1.7647E+02	1.6863E+04
SURFACE(.1M**2)	3.8941E+02	3.4813E+04
AVGDIAM(.01M)	1.6406E+02	2.5048E+03
MAXSOMATOTERM(.1M)	8.5024E+02	3.5420E+04
AVGSOMATOTERM(.1M)	6.0800E+02	1.1000E+04
AVGCENTERTOTERM(.1M)	3.5700E+02	1.2159E+04
AVGCENTERTOBR(.1M)	1.8876E+02	2.3074E+03
BRDEVIATION(M)	8.8824E+00	9.2803E+00
TERMDEVIATION(M)	1.2294E+01	3.2443E+01
AVGCIRCULARCOMP(.1M)	9.8176E+01	9.0932E+02
AVGRADIALCOMP(.1M)	4.5118E+01	1.1675E+03
%SURFATLEVEL1	1.0471E+01	2.2931E+02
%SURFATLEVEL2	1.9529E+01	1.4025E+02
%SURFATLEVEL3	3.6000E+01	3.4482E+02
%SURFATLEVEL4	1.8824E+01	1.4720E+02
%SURFATLEVEL5	2.9412E+00	3.7938E+01
MAX%OPPDIRECT	3.5059E+01	2.0644E+01
MAX%RIGHTANGLES	3.2529E+01	3.5661E+01

Table 9. Feature vector statistics for a group of 17 type INLSB neurons.

INLSB1
COUNT: 6

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	1.3383E+02	8.4314E+02
SOMATHIN(.1M)	1.1033E+02	2.1089E+02
SOMAZ(M)	8.5000E+01	2.2333E+01
VOLUME(.1M**3)	2.2267E+02	1.9323E+04
SURFACE(.1M**2)	4.8650E+02	3.2618E+04
AVGDIAM(.01M)	1.7033E+02	3.4292E+03
MAXSOMATOTERM(.1M)	8.2850E+02	1.7952E+04
AVGSOMATOTERM(.1M)	6.0750E+02	5.2582E+03
AVGCENTERTOTERM(.1M)	3.5200E+02	1.6685E+04
AVGCENTERTOBR(.1M)	2.0400E+02	9.8567E+02
BRDEVIATION(M)	9.5000E+00	6.2500E+00
TERMDEVIATION(M)	1.1500E+01	2.6250E+01
AVGCIRCULARCOMP(.1M)	8.9833E+01	4.7647E+02
AVGRADIALCOMP(.1M)	2.8000E+01	4.6433E+02
%SURFATLEVEL1	1.6667E+00	2.2222E-01
%SURFATLEVEL2	5.3333E+00	7.8889E+00
%SURFATLEVEL3	4.8667E+01	2.8056E+02
%SURFATLEVEL4	2.8833E+01	1.2481E+02
%SURFATLEVEL5	5.1667E+00	8.3472E+01
MAXLOFPOIRECT	3.5833E+01	2.9806E+01
MAXRIGHTANGLES	3.1500E+01	1.1583E+01

Table 10. Feature vector statistics for a subgroup of INLSB.

INLSB2
COUNT: 8

PARAMETER	MEAN	VARIANCE
SOMAFAT(.1M)	1.3838E+02	5.0023E+02
SOMATHIN(.1M)	9.6750E+01	2.7944E+02
SOMAZ(M)	8.4750E+01	6.3938E+01
VOLUME(.1M**3)	1.7700E+02	1.5063E+04
SURFACE(.1M**2)	3.7000E+02	2.6767E+04
AVGDIAM(.01M)	1.7462E+02	1.8532E+03
MAXSOMATOTERM(.1M)	8.7825E+02	5.0800E+04
AVGSOMATOTERM(.1M)	6.0637E+02	1.1379E+04
AVGCENTERTOTERM(.1M)	3.4637E+02	1.2567E+04
AVGCENTERTOBR(.1M)	1.9225E+02	2.3404E+03
BRDEVIATION(M)	8.8750E+00	1.1859E+01
TERMDEVIATION(M)	1.2750E+01	4.6438E+01
AVGCIRCULARCOMP(.1M)	9.7375E+01	1.1145E+03
AVGRADIALCOMP(.1M)	3.8125E+01	6.3511E+02
%SURFATLEVEL1	5.3750E+00	9.4344E-00
%SURFATLEVEL2	2.5375E+01	5.2344E+00
%SURFATLEVEL3	3.6625E+01	1.0998E+02
%SURFATLEVEL4	1.7250E+01	4.0937E+01
%SURFATLEVEL5	2.3750E+00	1.0734E+01
MAX%OPPDIRECT	3.3875E+01	1.4359E+01
MAX%RIGHTANGLES	3.0500E+01	6.5000E+00

Table 11. Feature vector statistics for a subgroup of INLSB.

"In the early days of biology, the biologists had to deal with the purely descriptive problem of finding out what living things there were, and so they just had to count such things as the hairs of the limbs of fleas. After these matters were worked out with a great deal of interest, the biologists went into the machinery inside the living bodies, first from a gross standpoint, naturally, because it takes some effort to get into the finer details."

R. P. Feynman(1)

8. FUNCTIONAL IMPLICATIONS AND CORRELATES

With the above analysis of structure in mind we now turn to functional characterizations of neuronal structures. The **Wiener** white-noise analysis technique is used in conjunction with Procion dye injection thus providing a functional representation and a link to our structural descriptors. A criticism of temporal white-noise as a tool in vision research is given and a spatio-temporal scheme for the characterization of function is described.

8.1 Some comparisons of structure and function; white-noise analysis and Procion dye injection

White-noise functional studies were performed using

intracellular glass electrodes filled with Procion dye. The stimulus pattern was a single spot .2 mm in diameter, or a single annulus 5 mm in diameter, or a spot and annulus together. In the first two cases the spot or annulus was modulated in intensity as a Gaussian random function of time. In the latter case, independant Gaussian functions modulated the spot and annulus. Spatially this pattern has only two regions, the spot and the annulus. Temporally these two regions are independently driven in white noise fashion (3,4). A description of the Wiener- Lee techniques as applied to biological systems can be found in many references (2,3,4,5). Here we will only present the first order kernels obtained for the various neurons in the frog retina and will assume that the reader is familiar with the Wiener-Lee application.

If s(t) is the spot stimulus function, a(t) is the annulus input and r(t) is the intracellular response for a given neuron then we compute the following first order kernels.

$$h1s/a = (1/Ps)\ E[r(t)s(t-tau)]$$

$$h1a/s = (1/Pa)\ E[r(t)a(t-tau)]$$

If spot input alone or annulus input alone is used we

compute

$$h1s = (1/Ps)\ E[r(t)s(t-tau)]$$

$$h1a = (1/Pa)\ E[r(t)a(t-tau)]$$

Figure (29) shows kernels from frog horizontal cells as identified through Procion dye injection. The kernels show variations in peak rise times and in degree of damping. All examples show slow hyperpolarizing responses to annulus and spot alike, although the response to the spot is much less pronounced.

Figure (30) shows the kernels from cells which, from their Procion images appear to be bipolars. These cells produce only slow potentials and demonstrate a biphasic receptive field which means that the spot and annulus give rise to responses of opposite polarity. Both on-center and off-center cells are shown. In the presence of the annulus, the latency and peak response time (h1s/a) of the spot input are reduced. These characteristics are similar to those observed in other retinas (3,4).

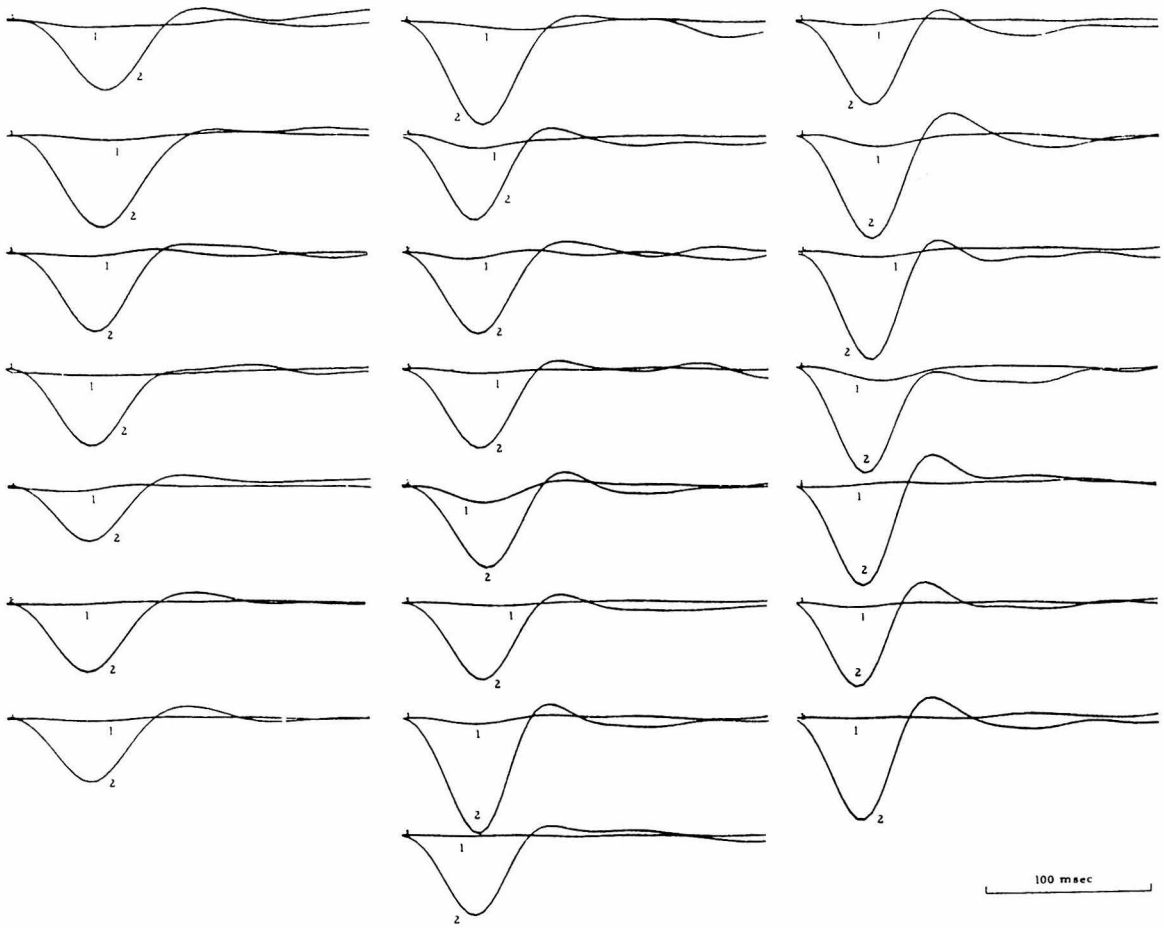


Figure 29. First order kernels from cells identified from Procion as horizontals. Plot 1 is $h1^s$. Plot 2 is $h1^a$.

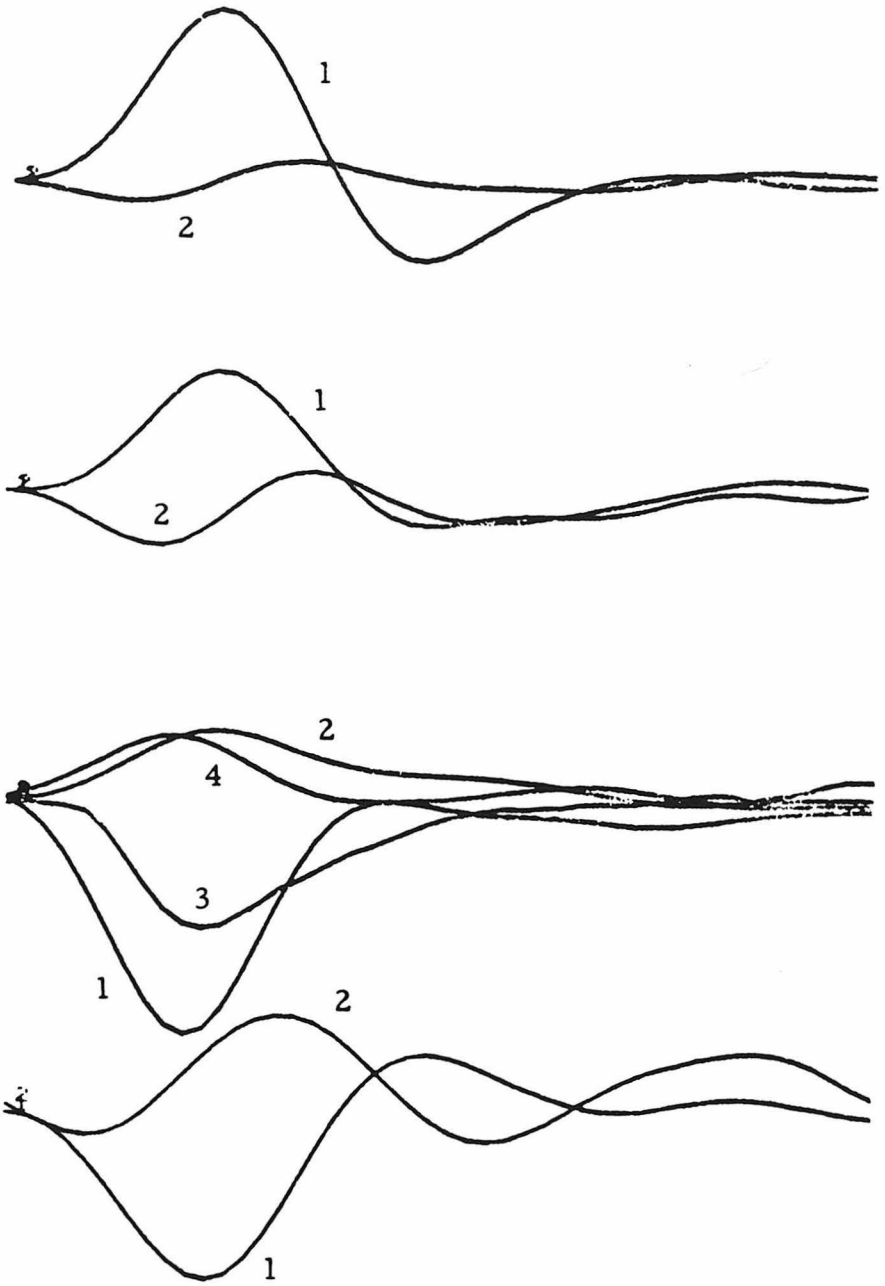


Figure 30. First order kernels from bipolars. 1 is $h1^{a}/s$, 2 is $h1^{s}$, 3. is $h1^{a}$ and 4. is $h1^{s}/a$.

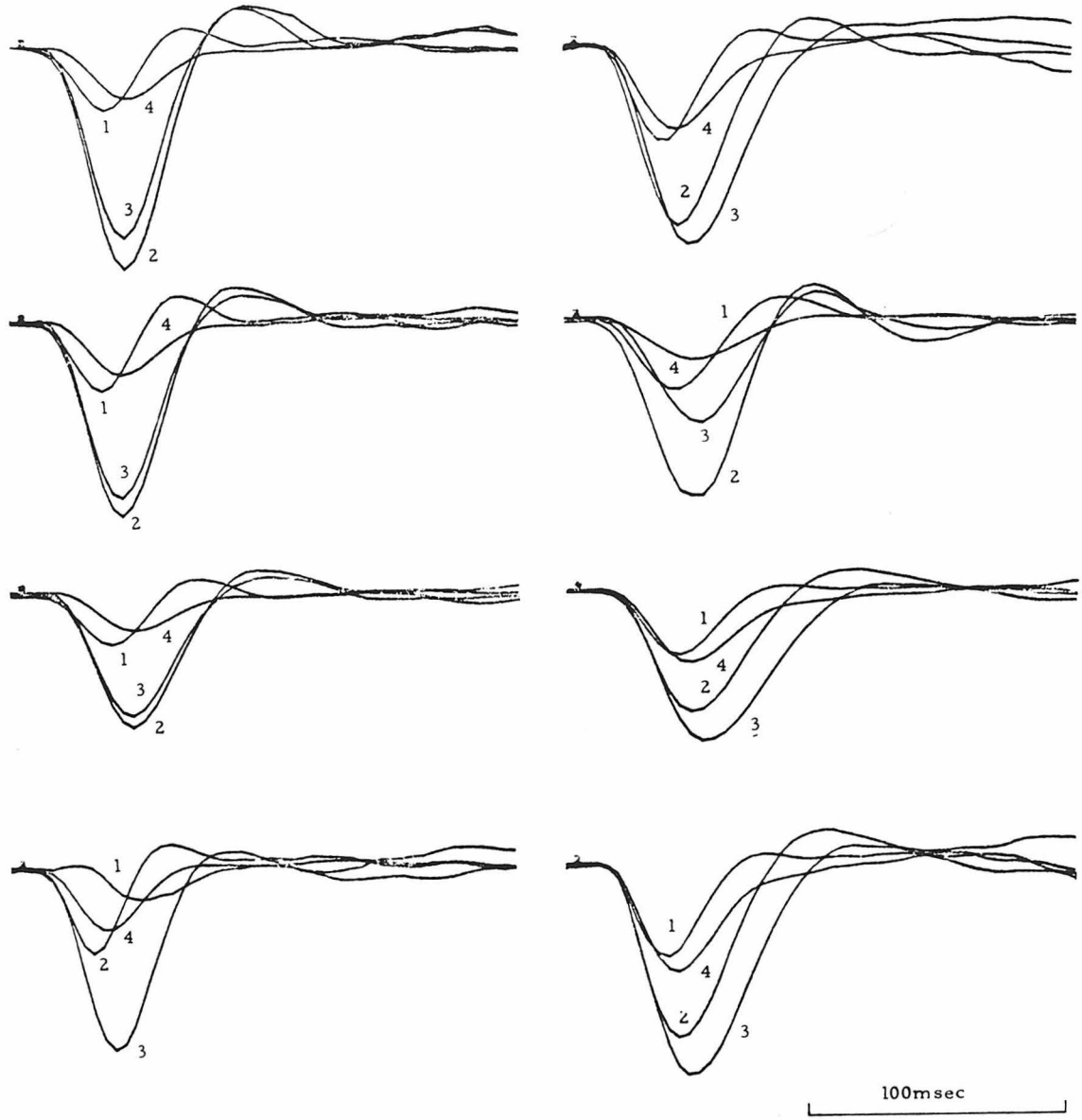


Figure 31. First order kernels from amacrines. 1 is h1s/a, 2 is h1a, 3 is h1a/s, 4 is h1s.

Figure (31) shows the kernels from neurons whose somas lie in the middle to proximal portion of the INL, as determined from Procion images. These kernels show a large spot component producing hyperpolarization. The spot component is smaller when the annulus is not present (h1s) than when the annulus input is present (h1s/a). These cells probably correspond to sustained type amacrines.

Figure (32) shows kernels from neurons having their somas in the GCL. Subjectively evaluated, these kernels range in type from those of figure (31) to very noisy kernels. Some of these cells showed spiking activity and most display a faster rise time and faster peak response than the cells from other layers. The middle column of figure (32) shows examples of spiking ganglions, and the noisier kernels are probably from transient on-off ganglions. Some of these cells are indistinguishable from amacrine responses and may represent displaced amacrines.

We experienced many problems in our attempts thus far to characterize function and to tie it to structure. Frog retinal neurons are small and they all have somas that are very similar. Procion dye typically infiltrates the soma and one or two of the fine processes making the identification of Procion neurons very difficult. Also, at times 2 or 3 adjacent cell bodies will become Procion filled making identification of the functionally

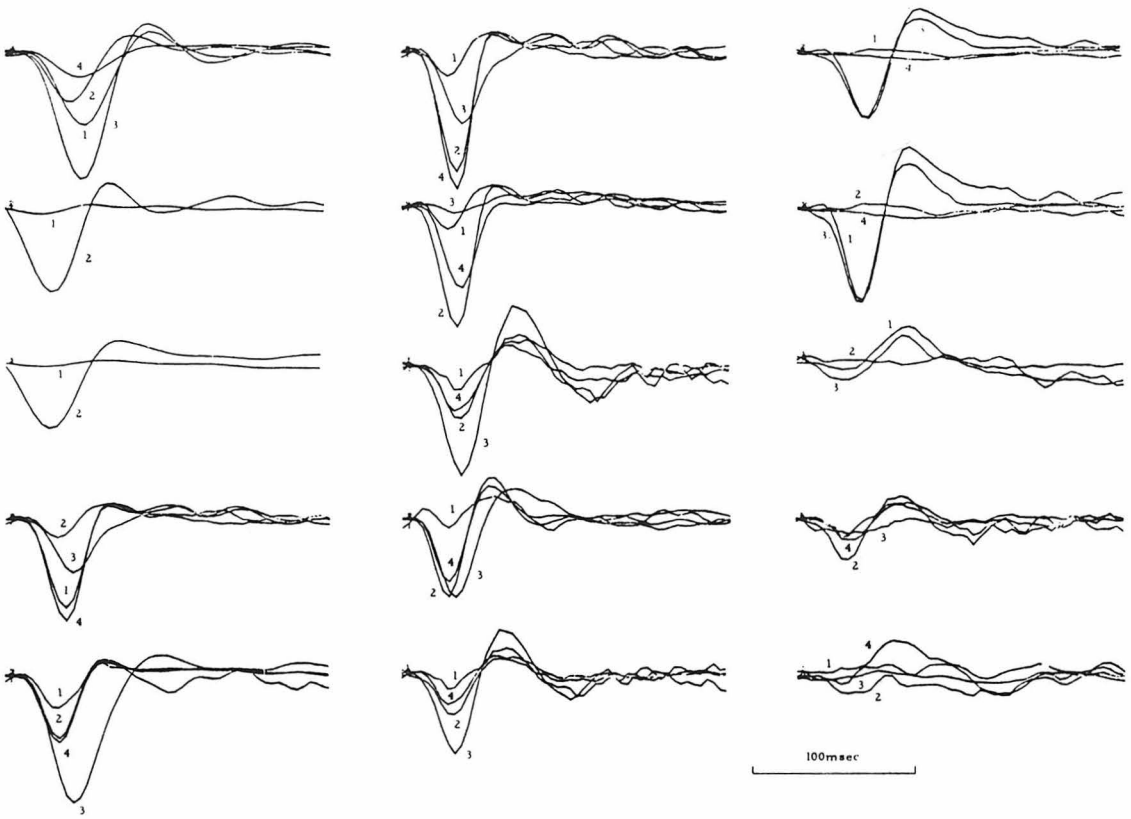


Figure 32. First order kernels from GCL cells. 1 is $h1^{s/a}$, 2 is $h1^{a/s}$, 3 is h1a, 4 is h1s.

tested cell impossible. The frog retina is a complex retina and we might expect many types of functional activity but the kernels obtained from the temporally oriented white-noise stimulus form a continuum of types that defy distinct subdivision. This stimulus is rich temporally but poor spatially. Perhaps a spatially rich stimulus would provide the sort of characterizing tool capable of classifying the functional activity into distinct types.

As we have previously mentioned the retina is a low-level pattern detector and as such must have a spatially oriented function. We have developed and made some initial tests of a spatio-temporal stimulus pattern which looks promising as a tool for resolving the problem of retinal function characterization. This scheme may provide the ability to perform a proper statistical analysis for functional groupings.

8.2 A method for functional characterization in time-space.

We have developed a stimulus consisting of a temporal sequence of random images or frames. Each frame is a 15 x 15 grid of squares. Each square has a grey level whose value is an independent random variable from a Gaussian distribution. The intensity or grey level of each square can take on 64 different values. This stimulus is generated as a TV video signal and is fed to a CRT to form the image

which is projected on the retina. Each point in time and space of this stimulus s(x,y,t) is an independent random variable with a given mean and standard deviation. The frames can be changed at a maximum rate of 30 times per second thus the maximum temporal frequency is 15 hertz. If we project this image on the retina so that the entire grid covers an area 400 microns on a side then the maximum spatial frequency represented is about 20 cycles/mm. Figure (33) shows one frame of this stimulus using a 10 x 10 grid, a mean of 32 and a standard deviation of 10 grey level units.

We have performed an initial test of this stimulus by projecting it on the retina using a 10 hertz temporal frequency and recording extracellularly from spiking neurons. We obtain an analog of the first order spatio-temporal kernel by performing the following operation.

$$h(x,y,t1) = 1/N \quad r(t)s(x,y,t-t1)$$

for $0 <= t1 <= M$, $1 <= x <=$ gridsize, and $1 <= y <=$ gridsize. where h is our resulting function, N is the number of time increments in the response, and s is the stimulus function. The function h is the time-space pattern which is most effective in getting a positive response out of the neuron

and thus represents the essence of the feature which the neuron reports.

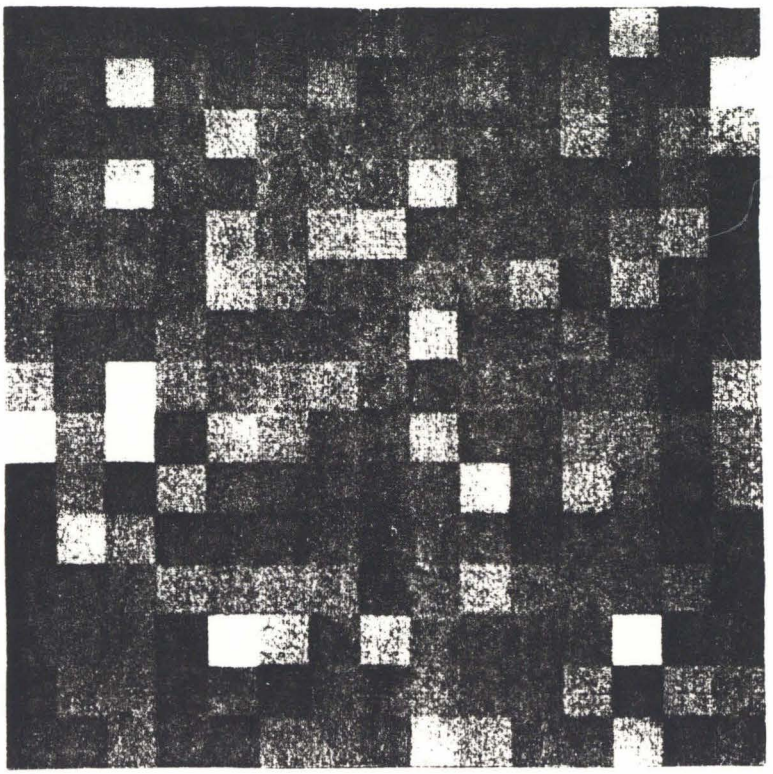


Figure 33. One frame or time instant of a spatio-temporal white noise-stimulus.

This assumes that the system, whose output is the layer of cells (one of which is being probed), is a spatially linear, spatial shift invariant system. An assumption must also be made about the discrete nature of the system elements. If a neuron fires when a particular pattern occurs at any of many positions on the neuron's receptive field then the above procedure will not likely work. That is, for the above method to work well the neuron must respond only to a specific spatio-temporal pattern which occurs at a fixed location. This situation views the neuron layer as a correlator embodiment of a matched filter and therefore the spatio-temporal first order kernel is essentially a reproduction of the feature to be detected.

From our results on bipolar cells we noticed the typical biphasic response. It has been suggested (3) that the bipolar cell computes its response by subtracting the horizontal response from that of a small field of receptors. Using the model which views the layers of the retina as spatial filters we can construct a spatial filter for the system light to bipolars. This spatial filter has a positively weighted area in the center representing the receptor input and a large negative surrounding area representing the subtraction of the S-space or horizontals. Figure (34) shows the action of such a spatial filter model of the bipolar layer on an input image. Such a filter

destroys low spatial frequency information in the image, responds weakly to edges, more strongly to corners and responds maximally to a small round spot. Thus a simple linear spatial filter can model the sorts of low-level pattern recognition activities one might expect retinal neurons to perform.

Hopefully the above rich spatial stimulus will enable us to build representations of function which provide a better tool for a statistical correlation of structure and function. If, as suggested in 5.3, we can construct a data structure for including both light microscope and ultrastructural data together with assumptions on synaptic function then we can make functional predictions from structural models and compare this form-function model with functional experiments based on rich spatial stimuli.

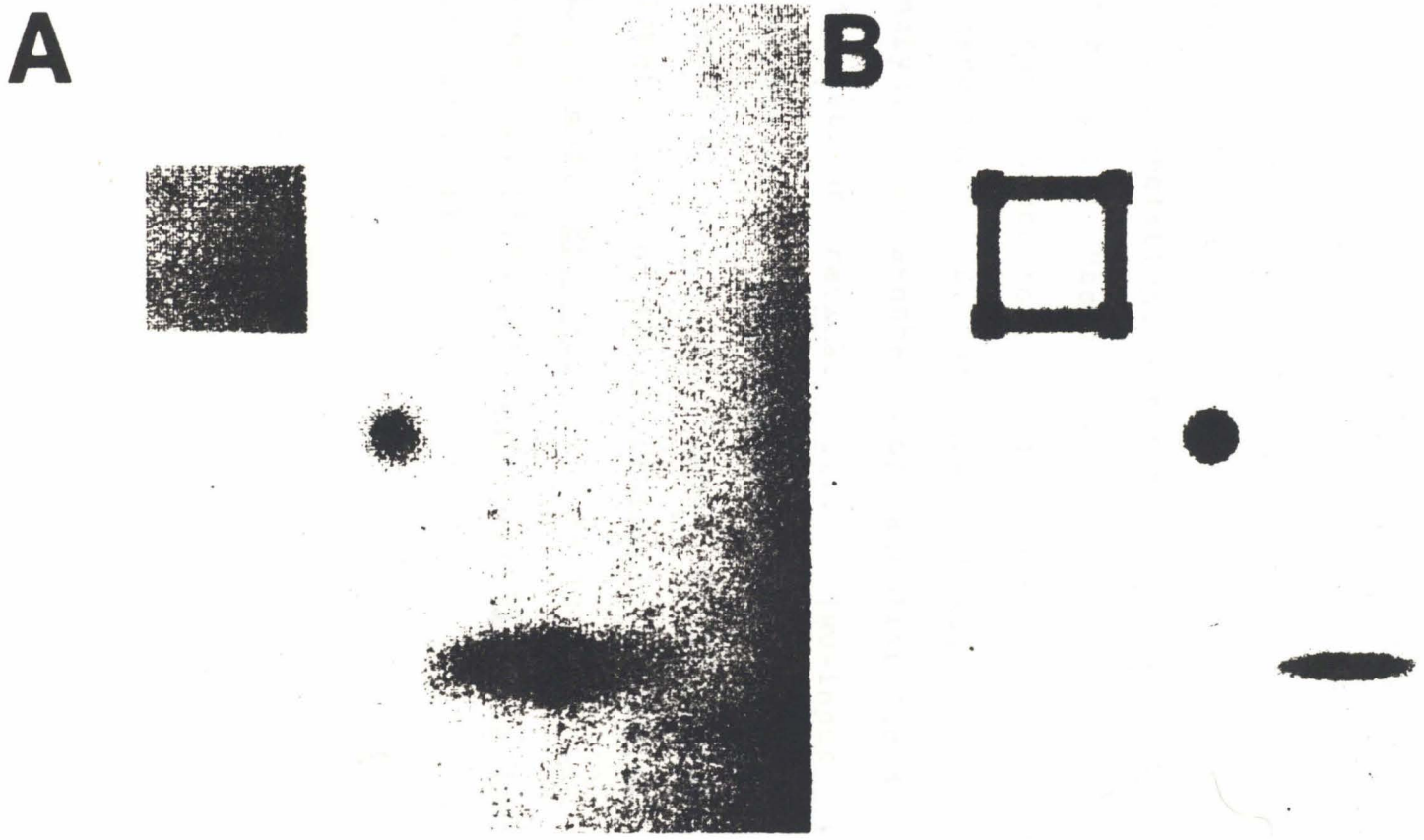


Figure 34. A. input image; B. bipolar layer model output.

REFERENCES

1. Feynman, R. P., R. B. Leighton, and M. Sands, (1963), The Feynman Lectures on Physics, Addison-Wesley, New York, vol. I, p3-2.

2. Lee, Y. W. and Schetzen, . ,(1961), Measurement of the kernels of a nonlinear system by cross-correlation. Quarterly Progress Report No. 60., Cambridge: Massachusetts Institute of Technology Research Laboratory of Electronics.

3. Marmarelis, P. Z. and K.-I. Naka, (1973), Nonlinear analysis and synthesis of receptive field responses in the catfish retina. III. Two-input white-noise analysis. , J. Neurophysiol, 36: 619-633.

4. Marmarelis, P. Z. and K.-I. Naka, (1974), Identification of multi-input biological systems. IEEE Trans. Bio-Med. 21: 88-101.

5. Wiener, N., (1958), Nonlinear Problems in Random Theory, New York, Wiley.

'This ultimately will provide a picture of a single neuron based on electrophysiological recordings, overall morphological type, and characterization of the types and distribution of synapses on its dendritic tree.'

R. W. Ware (1)

9. CONCLUSION

The task of this thesis has been to build quantitative descriptors and reconstructions of neurons for use in the correlation of structure and function. As a structural data base we have performed Golgi studies on frog retinal neurons. Golgi staining was used because it provides a relatively complete delineation of cell structure and its opaque quality facilitates digitization and computer image processing.

We employed an optical sectioning procedure first suggested by Castleman "et al" (2) to obtain 3-dimensional digital images of Golgi stained neurons from the light microscope. This optical sectioning process has several advantages over mechanical sectioning methods. No mechanical sectioning of tissue is required, no photography on tissue sections is required and realignment of adjacent section images is unnecessary.

From the optical sections we compute stereo pairs which

are displayed by the computer so that the 3-dimensional structure of a neuron can be viewed. This method overcomes the problem of limited depth of field encountered when viewing objects through the light microscope. It also allows the human to use a 3-dimensional view of the entire neuron to help resolve structural ambiguities.

We have developed an interactive procedure for the collection from stereo pairs of the primitives in a language for describing neuronal structure. This procedure makes use of a 3-dimensional cursor to allow the human to enter into the computer the syntactic primitives in a tree grammar. This method employs the human pattern recognition system and associated specialized knowledge about the retina to distinguish neuron from noise and to interactively build the descriptor. Thus we use the best of interactive human capabilities coupled with the best of computer capabilities, that of implementing data type definitions and operations on these data types which facilitate the analysis of structure.

Our tree grammar defines structures which are binary trees, each of whose nodes describes a soma, a branch, a process or a termination point. Each node carries information such as position of cell part, size of cell part and has the ability to include information on synaptic properties of the cell part. Each cell is thus a sentence in the language generated by this grammar. We have obtained

such a syntactic representation for 57 neurons from the frog retina. We use the high level language PASCAL which provides facilities for user definition of complex data types to implement this representation. The method allows for the use of high level methods such as searching for the occurrences of generalized substructures in neuron descriptors, and it provides convenient data organization for computation of structural parameters for classification purposes.

We have computed 21 structural parameters for each of 57 neurons and have used these parameters in an interactive cluster analysis. Although the sample of neurons is small, marked clustering of cell types is seen based on these quantitative parameters. These groups or types compare well with subjective groupings and with Cajal's side view drawings.

Three-dimensional reconstructions from syntactic descriptors were used to enhance the visual examination of structure and for the purpose of adding 3-dimensional information to Cajal's drawings.

We show results from preliminary functional experiments and correlate function as identified by Wiener kernels from 2 input (spot- annulus) white-noise experiments with our 3-dimensional reconstruction of various cell types. Procion dye was injected into cells whose function was identified

providing a means of mapping kernels to structural types. Although the information obtainable from temporally oriented stimulus patterns is not great when studying a system which is essentially a spatial feature extractor we were able to make some observations concerning form - function correlates. Type INLLA neurons demonstrate linear kernels similar to horizontal cells in that both spot and annulus kernels show depolarization. First order kernels of types INLSB show underdamped characteristics. Nonlinearities increase in the types from the GCL but conclusive statements are hindered by the uncertainty in mapping the poorly impregnated Procion dyed neurons with our Golgi reconstructions.

In section 8.2 a method is described for the generation of a spatio - temporal stimulus pattern each point in time and space of which is an independent random variable from a Gaussian distribution. Spatially this pattern is a 15 x 15 array of squares each having an intensity drawn from a Gaussian distribution with 64 possible grey levels. Temporally this pattern can be randomly changed at a maximum frequency of 30 Hz. This stimulus eats random numbers at the rate of 6750 per second and is possible through special hardware. We are currently using this stimulus to obtain spatio - temporal kernels from retinal neurons. This information should greatly facilitate the correlation of

structure and function in the vertebrate retina.

We conclude that grammatical tools and data structuring techniques are extremely useful in the analysis of neuroanatomy. Interactive 3 - dimensional analysis techniques facilitate a global approach to the correlation of structure and function in the central nervous system.

REFERENCES

1. Ware, R. W., (1975), Three - dimensional reconstruction from serial sections, Intl. Rev. of Cytol., 40: 325-440.

2. Weinstein, M. and K. R. Castleman, (1971), Reconstructing 3-D specimens from 2-D section images, Proc. SPIE, 26: 131-138.