National Library of Canada
Acquisitions and Bibliographic Services Branch
395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada
Direction des acquisitions et des services bibliographiques
395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Canada

SEGMENTATION AND EDITING OF 3-DIMENSIONAL MEDICAL IMAGES

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF THE UNIVERSITY OF OTTAWA
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Jeffrey Stanier
August 1993

ISBN 0-612-00500-3

UNIVERSITÉ D'OTTAWA
UNIVERSITY OF OTTAWA

Abstract

Neuroradiologists rely on scanned images of the human brain to diagnose many pathologies. The images, even those collected in 3-dimensions, are typically displayed as a 2-dimensional collage of slices and much of the intrinsic 3-D structure of the data is lost.

Image *Atlases* are commonly used to delineate and label *Volumes Of Interest* (VOIs) in 3-dimensional, slice-type, medical data sets. They can serve many purposes: to highlight important regions, to quantify the size and shape of structures in the images, to define a surface for 3-D rendering and to help in navigation through a series of images. To performthese functions, an *individual atlas* is required for each data set.

The purpose of this thesis is to develop a link between the volume data and the individual atlas associated with each set of images. An automatic method of building an *individual atlas* from the volume data is proposed. The method uses a data-driven, bottom-up segmentation to produce a *primitive atlas* followed by a knowledge-driven, top-down merging and labelling stage to refine the *primitive atlas* into an *individual atlas.*

The system was implemented in software using an object-oriented approach which allowed for a high quality user interface and a flexible and efficient implementation of the concepts of an *atlas* and a *VOI.*

Tests were performed to judge the quality of the segmentations and of the *atlas* labellings. The results prove that the *individual atlases* created using the proposed method are sufficiently accurate to aid in visualizing 3-D structures in medical data sets and to quantify the sizes of these structures.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This work is driven by the needs of radiologists interpreting medical images, in particular, images produced using Computed Tomography (CT) and Magnetic Resonance Imaging (MRI). Radiologists make decisions daily about the diagnosis and treatment of patients based solely on their experience and the images in front of them. Any improvement to the presentation of these images should therefore be beneficial and welcome. To this end, an image editing and integration system is proposed in Chapter 4 to assist the radiologist in extracting information from the image data.

Neuroradiologists, radiologists who specialize in interpreting images of the nervous system, commonly use CT-scanning machines to visualize the anatomical structure of a patient's brain. Less commonly used is the relatively new technology of Magnetic Resonance Imaging. Both imaging technologies can collect data in all three spatial dimensions to a millimetric resolution throughout most of a patient's body. A brief description of the work of a clinical neuroradiologist and of the scanning-type medical imaging modalities are included as Chapter 2.

Data collected by a scanning-type medical imaging system is stored into a 3-D volume array or as a series of 2-D slices. Figure 1.1 shows a 3-D volume array and

some of the possible methods of displaying the data: a slice through the data in any plane can be displayed, a cutout can be removed from the data in order to display the voxels at the edges of the cutout, or successive cuts can also be displayed in an animation to simulate a movement through the image. As interesting and informative as these images are, the display possibilities of the raw data sets are limited. Indeed, not even the external surface of the head is explicitly described by the volume data.

Both CT and MRI scans are usually displayed in a slice format; a series of two-dimensional "cuts" are collected and sequentially displayed on a terminal or photographic transparency. Radiologists must, when interpreting these images, "construct" a three-dimensional model of the intra-cranial structures which caused the images in the scans. Constructing the model requires the visual interpretation of many (5 to 15) 2-D images simultaneously; usually presented in a collage of 2-D slices. Direct (3-D) display of the data, although an attractive idea, is not easily developed into a clinically useful system since display of a 3-D solid is difficult without segmenting the image into clinically significant regions. It is difficult to provide depth cues for these images since shading and rotation, two of the most effective methods of simulating depth, require explicit surfaces to be effective. The situation is further complicated since the neuroradiologist requires not just the surfaces of structures to be displayed but the internal texture as well. A system with which a radiologist could visualize each of the structures in isolation as well as view their relative positions in 3-D would be of great assistance.

Much work has been done on the three-dimensional display of medical images; particularly of CT-scan images. A review of the state-of-the-art in this area is included in Section 3.2. We have used two concepts from the literature for the present work: those of a Volume of Interest (VOI) and of an *Atlas*. A VOI is a 3-D region which delineates a specific, medically significant, structure in the data. An *Atlas*, in the

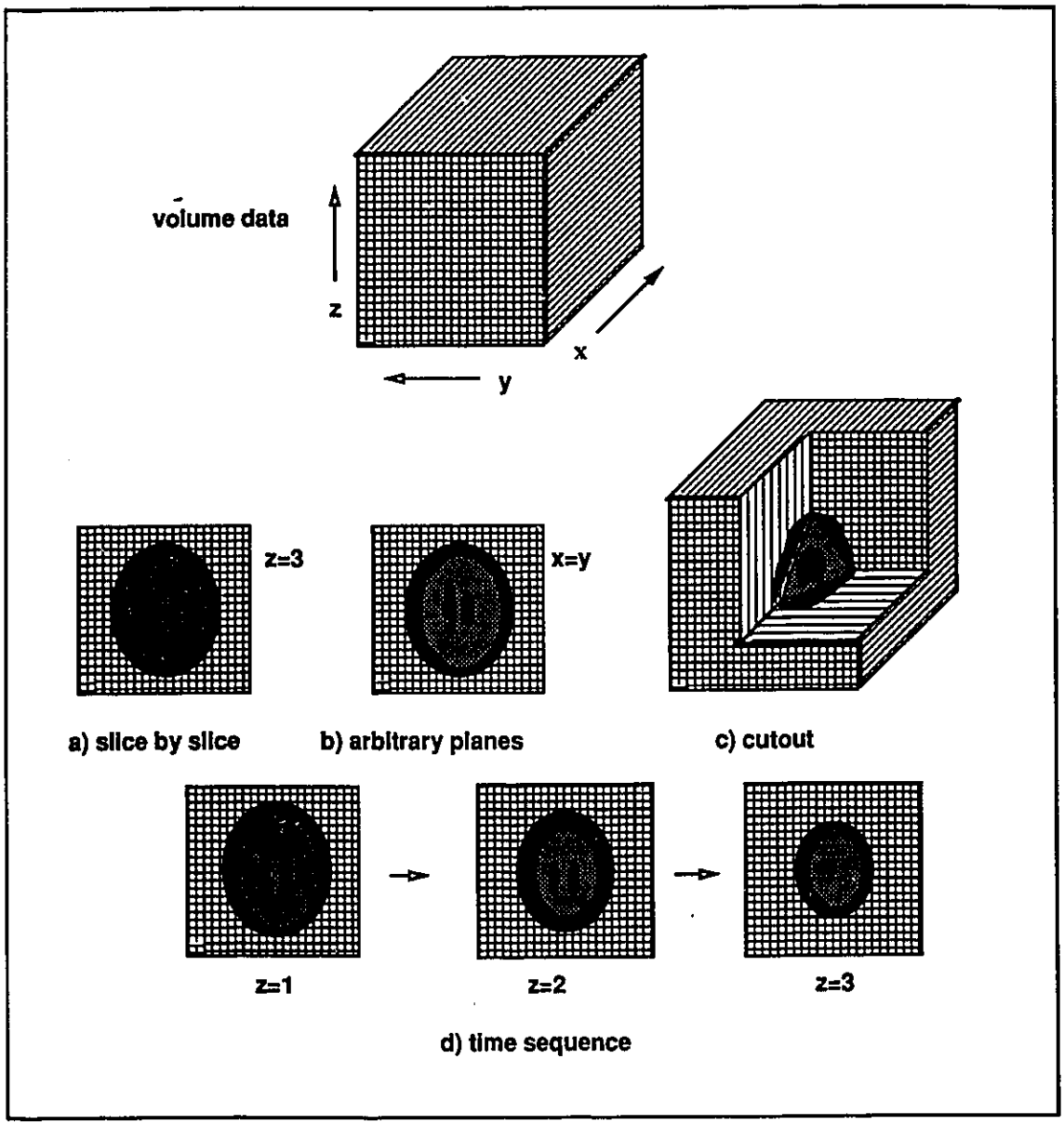

Figure 1.1: Methods of Displaying a 3-D Volume Array

realm of 3-D medical image processing, is a collection of VOIs which describes all of the structures in the data. We also use the concept of an atlas to describe a collection of primitive regions (VOIs) found as the output from a 3-D segmentation procedure or as a collection of 2-D Regions of Interest (ROIs) contained in a 2-D image.

We have investigated methods of improving the image presentation and access to the information in the images by exploiting the concept of an atlas. For example, knowledge of the surfaces of the VOIs contained in the image can be used for 3-D displays. The VOIs could also provide accurate surface and volume measurements, useful in analysing individual organs or pathologies.

An atlas allows for different types of access to the data. Renderings of the same structure taken in a succession of clinical visits could be used to track changes in a patient's condition. Images of the same structure in a patient taken using different imaging modalities could be directly compared on the same screen. It is also possible to overlay the atlas formed from one imaging modality onto the raw data from another imaging modality. All of these possibilities are discussed in Chapter 4.

Figure 1.2 shows one way in which the atlas structure could be used to access the structures in the image data. By selecting the name of the structure in the VOI list, the region is highlighted in the image and information such as average pixel value and the number of pixels are available. Conversely, by selecting a point on the image, the label assigned to that structure can be highlighted in the VOI list.

We must now be more specific about the type of information to be extracted from the 3-D volume data. Although many methods are available for image segmentation, the data suggests some which are more suited to intra-cranial structures in CT data. The images are usually collected under controlled and very similar conditions suggesting that structures in comparable studies using the same imaging modality will be similarly positioned. The imaging equipment has good dynamic range (usually 12

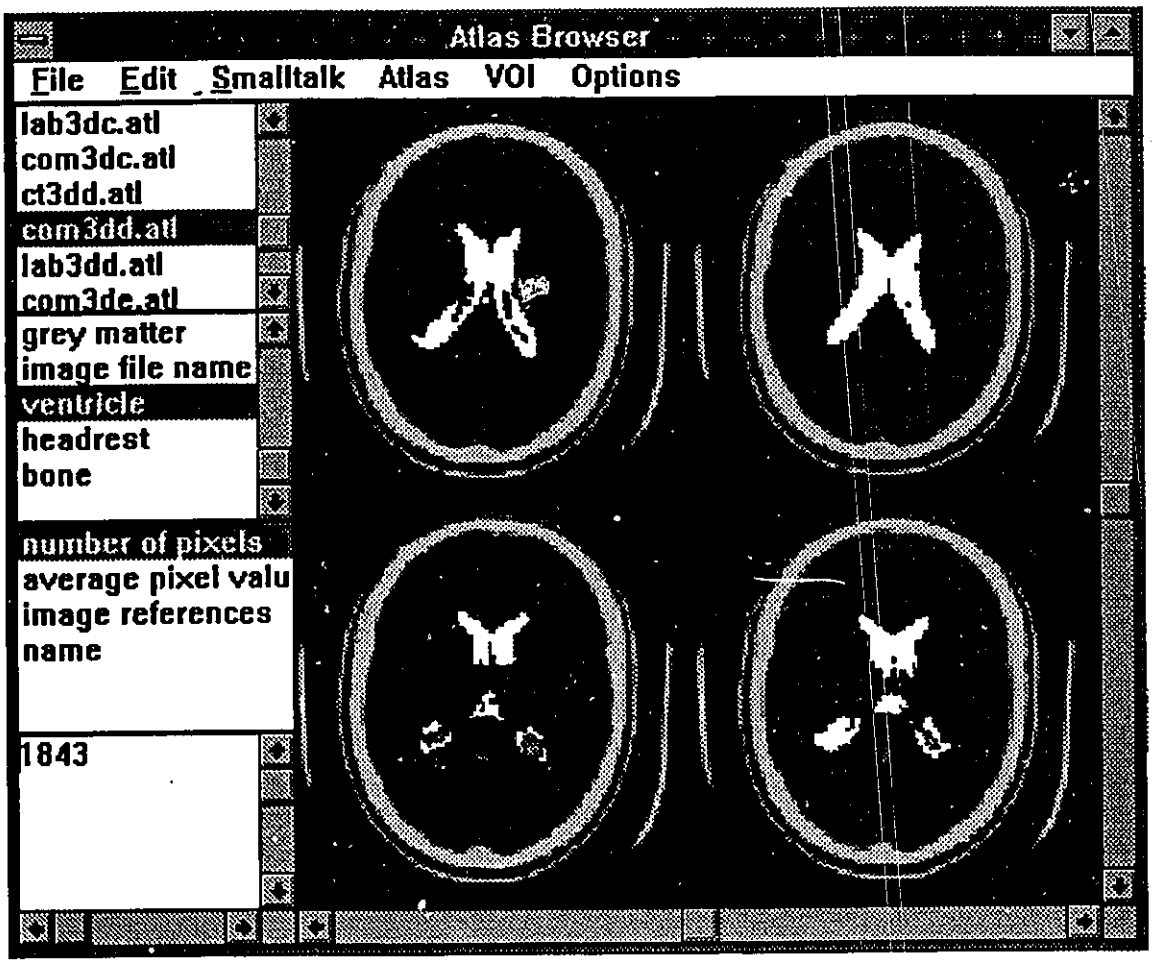

Figure 1.2: A Method of Display Using an Atlas

bits) and adequate resolution therefore the segmentation method must be sensitive to changes in individual pixels. Perhaps the most important factor is that the machines are routinely calibrated to ensure a representative pixel value according to the tissue type suggesting that *a priori* knowledge of the density of intra-cranial tissues could aid in the segmentation.

The method used to create the atlases from the raw images works in two stages: primitive VOIs are found by segmenting the image and then these VOIs are merged and labelled to produce a final atlas. In performing the segmentation, regions of relative homogenaeity are extracted from the data. Two different methods of segmentation are used: a gradient-based method and a region growing method. The two methods are somewhat complementary in their function; the region grower collects similar pixels together to form a region and the gradient method finds the boundaries between regions based on their dissimilitude. Both of these methods were modified so that knowledge of the pixel statistics of the tissue types contained in the images could be used to improve the segmentation process. A general discussion of image processing and particularly of segmentation techniques is included in Chapter 3. The segmentation methods used in this work are described in detail in Chapter 3.

Depending on the quality and type of the data, and on the expectations of the radiologist, the initial VOIs found in the image may or may not be sufficient for the needs of the radiologist. If the radiologist wants only a 3-D visualization of the external surface of the skull using CT data, a simple segmentation scheme may be sufficient to extract this information and render an image. If, however, the radiologist wishes to view a 3-D rendering of the *corpus callosum* with MRI data a more complicated segmentation procedure will likely be needed.

The output of the segmentation stage is converted to an atlas data structure and termed the *primitive atlas*. After the initial segmentation, a knowledge-based

searching and merging of the VOIs in the *primitive atlas* is performed to find and label medically important structures. The output of the merging and labelling stage is an *individual atlas.* A *search dictionary* is used to specify the list of VOIs to be found, the pixel statistics of each VOI and the percent confidence of its position. An *individual atlas* from a previously processed set of images is used as a guide to decide which VOIs should be merged to form each region listed in the search dictionary. The merging is performed based on the descriptors abstracted during the segmentation phase. There are several types of descriptors which can be abstracted from the data during or after the segmentation, such as: pixel statistics, shape, and topology. Within each of these broad categories several specific descriptors are possible; for example, information on average pixel value, standard deviation, and the number of pixels in each VOI.

The VOI data structure, generated by the segmentation stage, is important for the speed and accuracy of the search. A high degree of data abstraction is needed for easy comparison and merging of VOIs. Section 4.4 contains a complete description of the atlas matching process.

A software application was designed and built to test the process of automatically generating and using *atlases* as a means of manipulating and displaying medical image data. The software was written as a *MS Windows* application using *Smalltalk/V.* It can process 2-D or 3-D images from various formats including images extracted from the *GE Highlight Advantage* databases. The original images are compressed into an 8 bits/pixel format for processing and display. The application has three parts. First, an image processing application that pre-processes, filters and segments the images and which can produce an atlas data structure from the results of the segmentation. Second, an atlas matching application which merges and labels the results of the segmentation. Last, an atlas browser that allows interactive browsing of the atlas data structures produced by the other two applications. A complete description of

the software implementation is given in Chapter 5.

System trials consisting of two tests were performed at the Ottawa Civic Hospital: one test was of the image segmentation methods and a second of the atlas matching methods.

The segmentation test was performed with 2-D and 3-D images of the head from a GE *Highlight Advantage* CT scanner. The radiologists were asked to judge four aspects of the segmentation:

1. The differentiation of white and grey matter
2. The shape of the ventricles
3. The shape of the surface of the skull
4. The localization of sulci

The criteria used to judge the test results were designed in conjunction with the radiologists to best fit the practical uses of such a system. The boundary between white and grey matter is often difficult to visualize in CT images since their radiological densities are very similar and the enhancement of that boundary could be beneficial. The shape of the ventricles and sulci are are commonly used to evidence mass effect or intra-cranial pressure. The proper segmentation of the skull can be used for subsequent 3-D surface rendering and shading of the skull. The results from 2-D and 3-D images were compared to see if performing the direct, 3-D, segmentation has an advantage over segmenting each slice individually. Including *a priori* knowledge of the pixel statistics in the segmentation process greatly improved the segmentations of the CT images. Doing a direct 3-D segmentation of the data produced far fewer volumes than if the segmentation was done slice by slice. The trials are described more fully in Chapter 6.

The second test used dissimilar images to challenge the merging and labelling process. Previously labelled atlases were used to automatically merge and label the output of the segmentation stage. The quality of the labelling depends on how closely the atlas matches the image. For example, if the labelled atlas was derived from exactly the same set of images as the unlabelled atlas then the matching should find exactly the same VOIs and label them identically. However, if the images were of a severely abnormal brain and the atlas from a relatively normal brain then mismatches are expected. Mislabellings could take several forms; some regions may be mislabelled or may be too large. Other regions may not be found at all in the segmented images. Details of the trial procedure and the results are presented in Section 6.2.

The general conclusions of this work are included as Chapter 7. Overall, the system was able to segment and label CT images and to provide access to the structures in the image using an atlas. Including *a priori* knowledge in the image segmentation process greatly increased the performance of the system since it reduced the number of primitive VOIs searched during the merging stage. A direct 3-D segmentation of the volume data, as opposed to segmenting each 2-D slice separately, was beneficial for the same reason. Using a program based on this type of processing could extend the functionality of the 3-D medical workstations currently available. It would provide a more flexible access to the volume data and aid in the problem of 3-D display.

Chapter 2

Radiology and Medical Imaging

In order to provide the radiologist with more advanced image visualization tools we must first understand the environment in which these tools will be used and the nature of the data being collected. We need to know how the radiologist makes use of the images, what he or she is looking for in the images, and the strengths and weaknesses of the various imaging technologies. This chapter describes the work of the radiologists and the neuroradiologists, the medical imaging technologies that they use and a little about the images themselves.

2.1 Radiology

Typically, a patient is referred to the radiology department by an examining physician. In most cases there is little contact between the referring physician and the radiologist except the few words written on the requisition describing the radiographic procedure or querying a possible pathology. The radiological examinations are done by the department staff and, when the films are ready, they are sent to a radiologist, usually someone with training or experience in this particular speciality. If the patient has had previous radiological examinations, and they are available, they will be included

with the current study for comparison. The radiologist then reports on the current study. In most cases, a general examination of the image(s) is performed, looking for any deviations from normal. After the general survey the radiologist looks specifically for findings relating to the original purpose of the examination. The results of the study are put into a concise report describing exactly what was found or not found. In some cases, an interpretation or opinion of the findings is given by the radiologist in the report, especially when the case is a common one. This report is then sent to the referring physician and the films and a copy of the report are kept on the patient's file in the radiology department.

A radiologist reports on images from several modes of imaging including: conventional x-rays, subtraction angiograms, conventional and computer tomograms (CT), myelograms, magnetic resonance imaging (MRI), and ultrasound. In each case, a knowledge of the imaging process and human anatomy, as depicted by that imaging process, is necessary to correctly interpret what is shown in the images. The radiologist has an internal, three-dimensional, representation of human anatomy which he or she compares to what is shown in the images. In some cases, such as with normal x-rays, projections of a patient's anatomy are compared with the radiologist's internal representation. The 3-D position of any structure is found in two ways. For normal structures it is inferred from the internal representation of anatomy and, for abnormal structures, it is found by using back-projection of orthogonal views.

2.1.1 Neuroradiology

One of the specialities in radiology is neuroradiology, which is concerned mainly with the interpretation of images of the head and spine. The principle tools of the neuroradiologist were, in the past, conventional x-rays, tomograms, pneumoencephalograms,

and subtraction angiography [7]. Although subtraction angiography is still used to image the vascular system in the brain, the other imaging techniques have been largely supplanted by CT and MRI scans. These newer techniques offer better visualization of the intra-cranial tissue and collect data in all three spatial dimensions. Let us first look at the anatomy of the brain and then at the specific imaging modalities.

On a macro level, the anatomy of the brain has been extensively studied. There are many books on the subject, each with a slightly different viewpoint. The information presented here comes mainly from [62].

The *dura mater* is the most external membrane covering the brain. The outer side of the dura is next to the bones of the skull. The inner surface is covered by the *arachnoid.* The three important processes in the *dura mater* are: 1) the *falx cerebri* which separates the two hemispheres of the cerebrum, 2) the *tentorium cerebelli* which separates the cerebrum and the cerebellum and, 3) the *falx cerebelli* which divides the two hemispheres of the cerebellum.

The *pia mater* is a vascularized membrane which covers the entire brain surface and has processes into the interior of the brain. The *arachnoid* is another membrane which covers the pia mater.

The brain can be divided into four main parts: the *cerebrum*, which is the largest, the *cerebellum*, which lies inferior-posterior to the cerebrum, the *medulla oblongata*, which extends from the spinal cord to the *pons varolii*, and the *pons varolii*, which forms a process to connect the other three parts together.

The cerebral hemispheres include the major lobes of the brain. Referring to Fig. 2.1 [60], the lobes are the *frontal*, *parietal*, *temporal* and *occipital* and are separated by deep clefts called *fissures.* Within each lobe are ridges called *gyri* separated by smaller clefts called *sulci.*

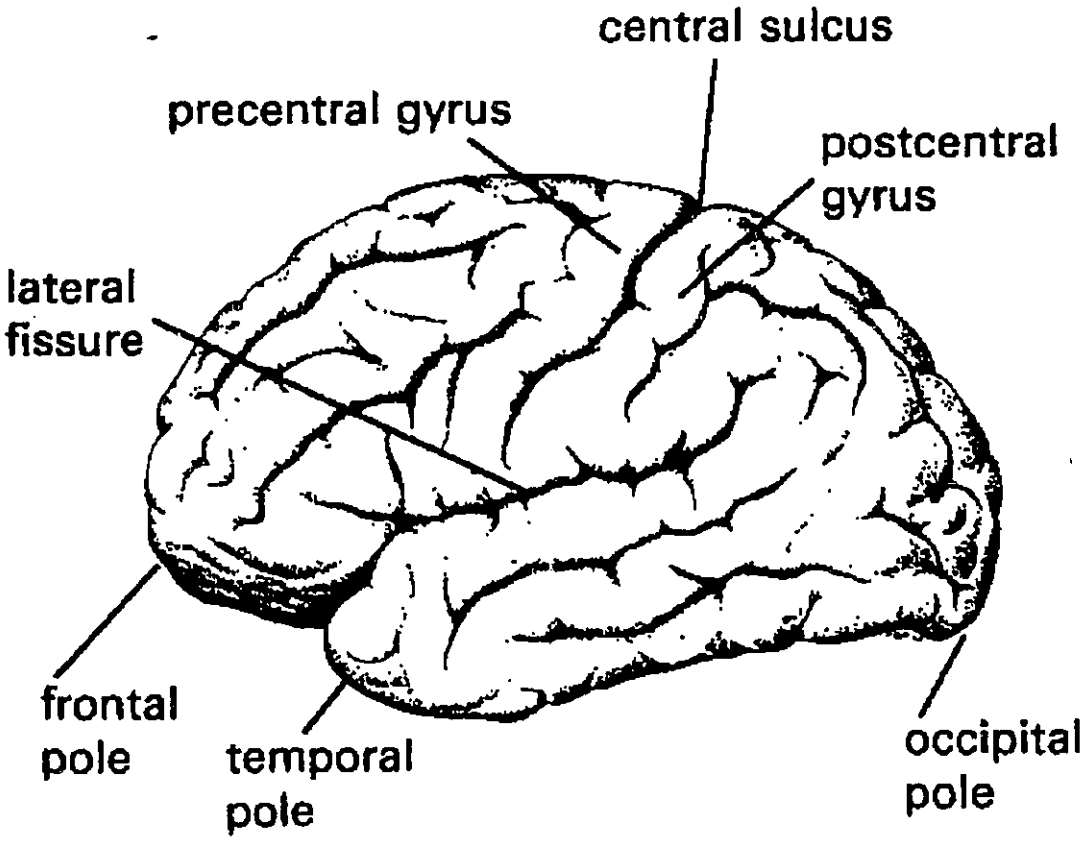

Figure 2.1: Lateral View of the Human Brain

The ventricular system includes the *lateral ventricles*, *third ventricle*, *fourth ventricle*, and several inter-ventricular connecting passages. The ventricular system carries the flow of *cerebrospinal fluid* (CSF) throughout the cranial cavity. The fluid acts mainly as a cushion for the brain, protecting it from impact against the skull, but may have other functions [44]. Changes in the shape of the ventricular system are very often used by neuroradiologists to indicate changes in the surrounding brain tissue. A drawing of a cast of the ventricular system is shown in Fig. 2.2 [60].

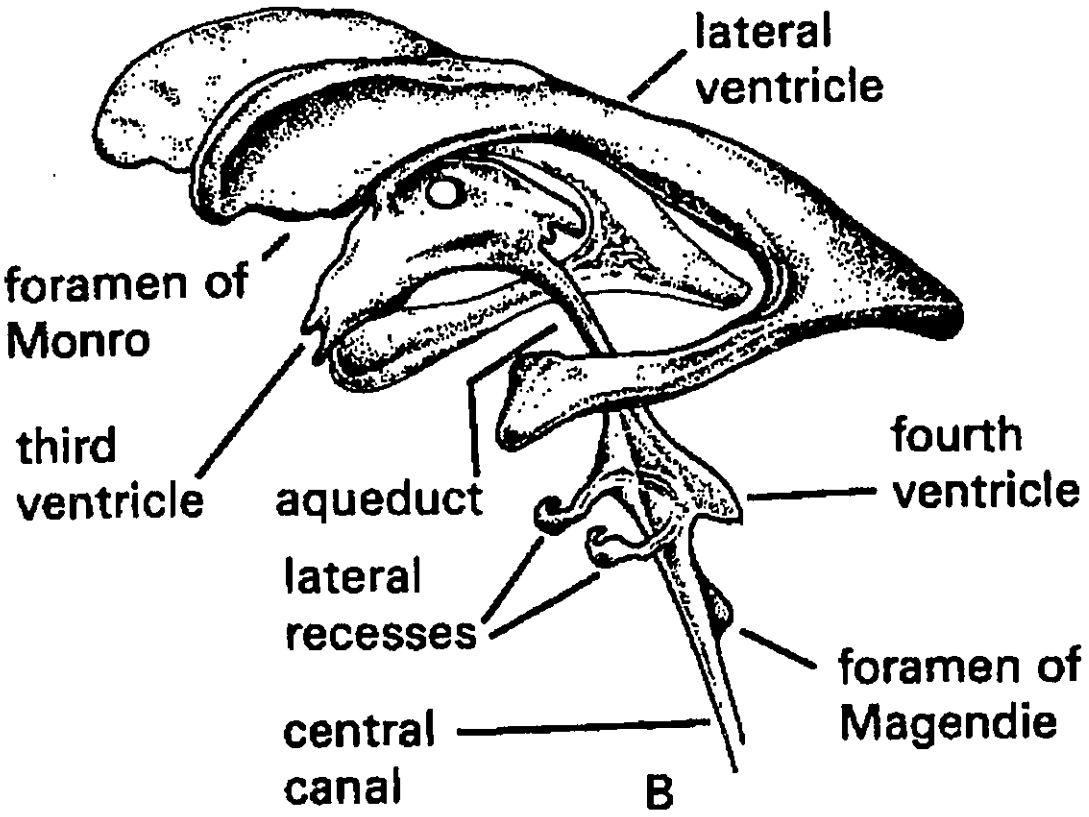

Figure 2.2: The Ventricular System

The structure of the brain matter surrounding the ventricular system is complicated. The *corpus callosum* forms the superior surface of the lateral ventricles. The

inferior boundary of the bodies of the lateral ventricles is formed by the *thalamus*, the *caudate nucleus* (laterally), and the *fornix* (medially).

The principle *basal ganglia* are the *caudate nuclei*, *lentiform nuclei*, and *thalamus*; all close to the third ventricle. The third ventricle is connected to the lateral ventricles through the *foramina of Monro* and to the fourth ventricle through the *aqueduct of Sylvius*. The *thalamus* and the *hypothalamus* form the superior and inferior lateral surfaces of the third ventricle respectively. The base of the third ventricle is formed posteriorly by the *pineal body* and anteriorly by the anterior columns of the fornix; the latter also form part of the anterior wall. Inferior to the third ventricle are a series of three *suprasellar cisterns*; the *interpunduncular cistern*, the *chiasmatic cistern*, and the *cistern of the lamina terminalis*.

The fourth ventricle is situated roughly between the *medulla oblongata* anteriorly and the *cerebellum* posteriorly.

An exhaustive treatment of the pathologies of the brain and skull is beyond the scope of this report (not to mention the author); however, definitions of several common terms used in reporting on scans of the head are given for completeness. This information comes from several books on the subject [68] [58] [36]. The focus is on conditions which are normally diagnosed using CT or MRI scans.

Soft tissue damage in the cranial vault can be due to trauma, congenital defects, vascular impairment, haemorrhage, and cancerous processes. Haemorrhage and carcinoma have a space occupying effect. Since the skull contains any expansion the increased pressure constricts the vessels of the CSF circulation system. In the case of large, space-occupying lesions the whole midline of the brain can be forced to distort. Smaller lesions may only constrict local parts of the ventricular system.

One possible organization for primary intracranial neoplasms (brain tumours) is [36]:

Skull osteoma, hemangioma, granuloma, xanthoma, osteitis deformans.

Meninges meningioma, sarcoma, gliomatosis.

Cranial Nerves glioma of optic nerve, schwannoma of 8th and 5th cranial nerves.

Supportive Structures gliomas.

Pituitary or Pineal Body pituitary adenoma, pinealoma.

Congenital Origin

Secondary *metastases* or lesions, which have spread from other sites, may involve any structure in the brain.

Some pathologies such as *porencephaly* are congenital. The radiological diagnosis for this disease is based on a limited development of brain tissue resulting in massive areas of lucency where the skull is full of CSF. Other causes of disease may be viral such as *meningitis* where there is inflammation of the meninges of the brain or spinal column. *Hydrocephalus* is a condition characterized by increased pressure in the ventricular system due to an obstruction to CSF flow. A dilated ventricular system can be an indicator of hydrocephalus. It can also indicate brain *atrophy* if accompanied by prominent solci.

Hemorrhages are also classified by where they occur. An *intracerebral hemorrhage* can arise anywhere within the brain tissue. A *subarachnoid hemorrhage* occurs when blood accumulates in the subarachnoid space. A *subdural hematoma* develops between the inner layer of the dura and the arachnoid membrane, and an *epidural hematoma* is an accumulation of blood in the space between the outer layer of dura and the inner surface of the skull.

The bony parts of the skull and spine can be imaged using conventional tomography and normal x-rays. Stereo views of the skull are sometimes used to assist the

radiologist in visualizing the bones in the sinus and ear region. Possible causes for abnormalities in these areas are trauma and congenital defects. Even if bony fractures are visualized using these techniques the extent of underlying tissue damage is not always apparent.

2.2 Medical Imaging Modalities

There are many different imaging systems in a modern radiology department, each used for a range of purposes. Just as a referring physician might use the radiology department to answer specific questions about a patient, the radiology department uses the different imaging modalities to answer specific questions about the inner structure of a patient's body. A brief explanation of some common imaging systems is given here.

Conventional X-ray images have a multitude of uses, among them: detecting bone fractures, screening for lung disease, and surgical planning. The image formed by a conventional X-ray procedure is a projection of the radiological density of the body onto a 2-D photographic plate. Although a trained radiologist can infer the relative positions of some of the imaged structures (assuming a normal anatomy), locating abnormal structures generally requires at least two orthogonal projections. A radiologist can then visualize quite accurately its position, except if the object is overshadowed in one of the views.

Ultrasound imaging uses the reflections of high frequency sound waves ($\approx 10\ MHz.$) from internal body structures to form an image. Modern ultrasonography uses digital sampling techniques and doppler processing to detect motion (*i.e.* blood flow) in the return. Since ultrasound cannot penetrate bone it is of limited use in imaging of the head and spine, however, it can be used for babies since the *fontanel* offers a

translucent "window" into the cranial cavity.

Most ultrasound imaging systems use a 2-D, fan-shaped, scan to obtain an image. Three-dimensional ultrasound imaging systems have been built using a transducer which scans in two orthogonal directions [76].

Myelograms can be used to detect blockage of the spinal cord. Radio-opaque dye is injected into the thecal sac and allowed to disperse through the spine. Normal x-ray views are then taken and constricted or ruptured regions can be visualized.

Imaging of the ventricular system was difficult and painful before the advent of the CT-scanner. In most cases, a pneumoencephelogram was done. The procedure involved injecting air into the ventricular system and taking normal x-rays. This highlighted the ventricles and other vessels which normally contain CSF. Tomographic techniques could also be used to image only a specific region of the brain. This technique could detect abnormalities in the shape of the ventricles, but could not detect what was causing the distortion. A side effect was severe headaches suffered by the patient for several days following the procedure.

Computed tomography (CT) and magnetic resonance imaging (MRI) have rendered pneumoencephalograms obsolete. A discussion of these two, more modern, methods of imaging follows.

2.2.1 Computed Tomography

Computed Axial Tomography (CAT) and Computed Tomography (CT) are both names given to a process which collects data by mechanically or electronically scanning the patient and computing an image from the collected data. Unlike a conventional x-ray image, the images formed by the CT scanner are not 2-D projections of the patient but a cross-sectional slice.

The scanning system is of primary importance to the CT technique. In all cases, an x-ray source and a detector system are located on opposite sides of the patient. The source has a narrow beam; typically 1 *cm* or less in thickness and wide enough to illuminate the width of the detectors. The detector system can be either a single detector scanned across the patient or a row of detectors distributed along an arc. In both cases, the detector should have a narrow beamwidth so only rays passing straight through the patient (with attenuation) are detected, and scattered rays from other paths are excluded. After a view is completed the source and detector system are moved to a different angle and another view taken. This is repeated until data is collected at enough angles to provide the required resolution in the final image. This data comprises one slice through the patient; if more slices are required the patient is moved through the scanner to a new position and the process repeated.

The speed of the scanning can be increased by using the more modern technique of helical scanning. In this case, the patient is steadily (but slowly) moved through the scanning plane and the data is collected continuously.

There are two basic algorithms used for reconstructing the tomographic image; iterative algebraic reconstruction and Fourier transform. Both of these techniques will be discussed in the section on the mathematics of reconstruction.

At photon energies typical of CT scanning systems the photons are attenuated either by photoelectric absorption or by scattering. In photoelectric absorption the photon is absorbed and its energy transferred to an electron. With scattering the photon may change direction but maintain its energy level (Raleigh) or it may give up some of its energy to an electron and continue in a different direction with lower energy (Compton). All of these effects contribute to the exponential attenuation of a monochromatic beam of photons as they pass through a homogeneous material. This

absorption can be quantified by the Lambert-Beer law as [64];

$$I = I_0 e^{(-\mu x)} \tag{2.1}$$

where I_0 is the input intensity, μ is the attenuation of the material, and I is the intensity after a distance x.

If we consider a human body with heterogeneous attenuation $\mu = \mu(x, y)$ then the output intensity can be written as (see Fig. 2.3);

$$I(x) = I_0(x) exp\Big(-\int_L \mu(x, y) dy\Big) \tag{2.2}$$

where L is the path of the x-ray through the body. The projection is written as;

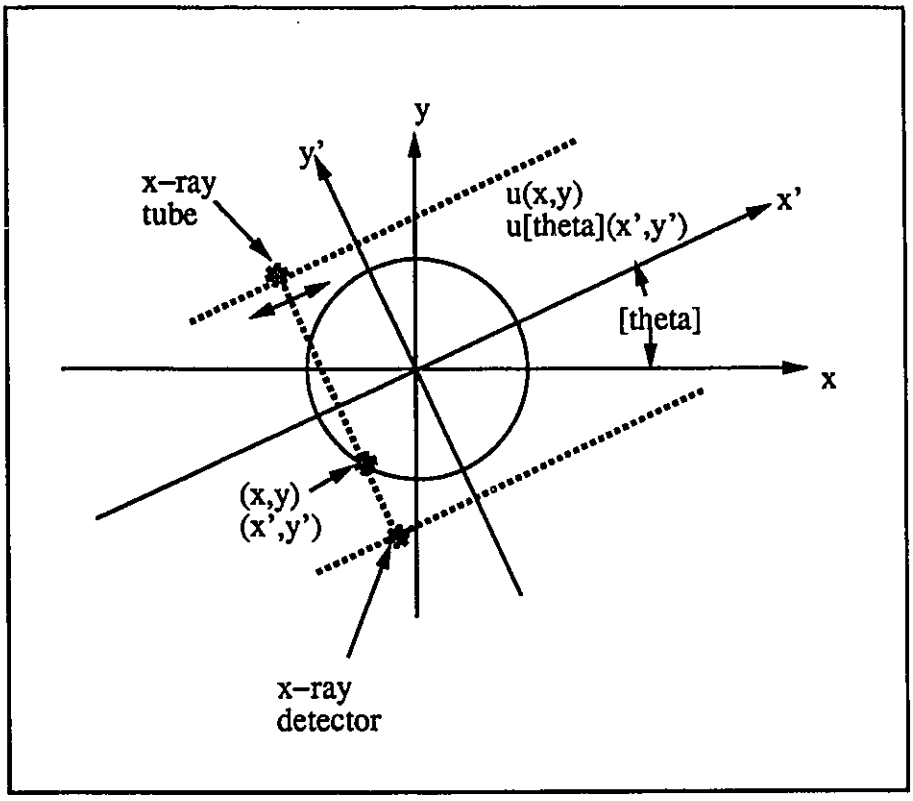

Figure 2.3: Projection Geometry of a CT-Scanner

$$P(x) = -ln\left(\frac{I(x)}{I_0(x)}\right) = \int_L \mu(x, y) dy \tag{2.3}$$

and is what is recorded by an ordinary x-ray film.

The CT-scan needs approximately 11 *bits* of data precision in order to reconstruct an image adequately. For this reason, any fluctuations in the input x-ray intensity or in the output intensity not directly due to variations in absorption must be accounted for. A detailed calibration is necessary to ensure accurate results. By placing a reference x-ray detector in the unperturbed path of the source, and sampling the detector during each scan, the input intensity can be constantly measured. Fluctuations in the beam pattern and detector responses can be measured daily to provide a calibration. A third source of data fluctuation can be the high gain amplifiers used to amplify the detector signals. Once again, a known input can be applied and the variations in the output measured as often as is necessary to maintain accurate results.

In order to understand exactly how CT images are formed it is necessary to look at the mathematics of tomographic reconstruction. Referring to Fig. 2.3, and following the notation of [64], a point on the body can be represented in either a fixed (x, y) coordinate system or a rotating $(\hat{x}, \hat{y})$ system. They are related by;

$$\begin{aligned} \hat{x} &= x\ \cos\theta + y\ \sin\theta \\ \hat{y} &= -x\ \sin\theta + y\ \cos\theta. \end{aligned} \tag{2.4}$$

Since $\hat{x}$ is the detector position and $\hat{y}$ is the path of a ray, the projection at any view angle is;

$$p_\theta(\hat{x}) = \int_L \hat{\mu}_\theta(\hat{x}, \hat{y})\ d\hat{y}. \tag{2.5}$$

The Fourier transform of Eq. 2.5 is;

$$P_\theta(f) = \int p_\theta(\hat{x})\ exp(-j2\pi\hat{x}f)\ d\hat{x}. \tag{2.6}$$

Substituting Eq. 2.5 into Eq. 2.6 yields;

$$P_\theta(f) = \int\int \hat{\mu}_\theta(\hat{x}, \hat{y})\ exp(-j2\pi\hat{x}f)\ d\hat{x}\ d\hat{y}$$

$$= \int\int \hat{\mu}_\theta(\hat{x},\hat{y})\ exp(-j2\pi(f\hat{x}+0\hat{y}))\ d\hat{x}\ d\hat{y}. \tag{2.7}$$

$$P_\theta(f) = \hat{U}_\theta(f,0) = U(f\cos\theta, f\sin\theta) \tag{2.8}$$

This is known as the projection theorem and means that the Fourier transform of a projection is the same as a cross-sectional cut of the Fourier transform of the image. The reconstructions can be done in the frequency domain or the equations can be transformed into the spatial domain. To transform Eq. 2.8 into the spatial domain we must take the inverse transform;

$$\begin{aligned}\mu(x,y) &= \int_0^\pi \int_{-\infty}^{\infty} P_\theta(R) exp(j2\pi(r\cos\phi R\cos\theta + r\sin\phi R\sin\theta))|R|dRd\theta \\ &= \int_0^\pi \int_{-\infty}^{\infty} P_\theta(R)|R| exp(j2\pi(rR\cos(\theta-\phi))dRd\theta,\end{aligned} \tag{2.9}$$

where; $x = r\cos\phi$, $X = R\cos\theta$, $R = sgnY\sqrt{X^2+Y^2}$, $y = r\sin\phi$, $Y = R\sin\theta$, $\theta = arctan(\frac{Y}{X}) mod\pi$, $0 < r < \infty$, $0 < \phi < 2\pi$, $-\infty < R < \infty$ and $0 < \theta < \pi$.

If we allow that $F^{-1}(R) = k(x)$ and call it the kernel then by using the convolution theorem we can write the inner integral of Eq. 2.9 as;

$$g_\theta(s) = \int_{-\infty}^{\infty} p_\theta(s')k(s-s')ds'. \tag{2.10}$$

The outer integral of Eq. 2.9 is a backprojection in that it sums the contributions from each view at each image point. Equation 2.9 can now be written as;

$$\mu(x,y) = \int_0^\pi g_\theta(r\cos(\theta-\phi))d\theta. \tag{2.11}$$

Although the kernel, $k(x) = |R|$, is a simple function in the frequency domain it has no exact analytical transform in the spatial domain. Several approximations to this function have been used but if we assume that $\mu(x,y)$ is bandlimited to frequencies less than $B\ cm^{-1}$ and that the sampling spacing is $a = \frac{1}{2B}\ cm$ then the kernel can be written as;

$$k_L(x) = 2B^2\ sinc2Bx\ -\ B^2\ sinc^2Bx. \tag{2.12}$$

This is called the Lakshminarayanan kernel. Although the algorithm is different if a fan-beam geometry is used the basic principles of the reconstruction are similar and this case will not be treated here. The reader is referred to [64] for more details.

The CT-scan collects information in the axial plane since the opening in the scanning mechanism cannot accommodate a patient in any other orientation. The patient is incrementally moved through the scanner and data is collected at each increment. The width of each "slice" is typically about 2mm and the increment spacing can vary to anything upwards of 1mm. If the increment spacing is small enough, (*i.e.* 2 mm), a reasonable image in the coronal and sagittal planes can be constructed by reformatting the data collected on successive slices. This can be done on data collected at greater increment spacing but the resolution in the coronal and sagittal planes is proportionately reduced. Only the area of interest (*i.e. head, lumbar spine*) is scanned and the typical increment spacing is about 4-8 mm for a head scan. Scans of the spine tend to be done in planes parallel to the interface between the closest disk-vertebra junction so the data is not collected in a regular pattern as in the head scans.

CT-scans can be used in place of the pneumoencephelogram and the myelogram. The ventricles are very well visualized and a space occupying lesion which distorts the ventricular system can be detected and is often directly visualized on the CT-scan. Vascular deformities and haemorrhages can be detected by injecting contrast dye into the arteries. CT-scans of the spine can show the size of the spinal canal as well as the shape and position of the thecal sac.

Recently, with the advent of helical-scan CT imagers, CT-angiography studies have been possible. Accurate contrast studies are possible since the scanner collects data very quickly. A clinical trial of CT-angiography is described in [20].

A representative image from a CT-Scanning machine is shown in Fig. 2.4.

Figure 2.4: An Image from a CT-Scanning Machine

2.2.2 Magnetic Resonance Imaging

This section is only a brief introduction to MRI; the interested reader is referred to several introductory papers on the technical background to MRI including [32] [54]. Discussions of the radiological aspects of MRI can be found in [12].

Although MRI and CT scan images can appear similar they utilize very different physical phenomena. One major difference is that, unlike the X-ray CT-scan, MRI does not use ionizing radiation. It also does not use rotating mechanical sensors to collect the data. MRI images can be collected and displayed as slices in any plane through the body. These slices can either be selected from data collected in all three dimensions or can be generated directly by using special hardware.

The MRI scan can be used to investigate a wide range of pathologies including: brain tumours, congenital defects, edema, haemorrhage, and trauma. Bone does not produce a strong return in the MR Image. Contrast materials are available for MRI studies to enhance the blood or CSF filled vessels. Another technique uses the flowing blood or CSF itself to produce contrast enhanced images. Since the blood is flowing from an area with a different magnetic field strength it will have different excitation characteristics from the surrounding tissues. This phenomenon can be used to enhance the appearance of blood vessels in an MRI study.

Exposure to the strong magnetic field is considered safe, exceptions being people with pacemakers or ferromagnetic material within their bodies. Other equipment in the MRI clinic must be shielded from the strong fields. MRI machines are expensive when compared with CT-scan systems, costing three to four times as much. Figure 2.5 shows an axial slice from an MRI study.

Figure 2.5: An Image from an MR-Imaging Machine

Chapter 3

Image Processing and Computer Vision

Vision has been described as the process of inferring the structure of the world from images [83]. Building machines which can simulate some of the functions of the human visual system is a goal of much of the research done in the field of computer vision. The central issue in computer vision is to implement the inference part of vision; how can the machine "reason" to extract the structures from an image and ignore the irrelevant parts.

A systematic approach to computer vision has been developed which details the steps in going from raw image data to a functional description of a scene. The process is similar for all applications of computer vision [6]. First, the image or images must be collected. Preprocessing and image registration is done at this stage resulting in a generalized image. Second, the image is segmented and regions of interest are highlighted. Third, the regions of interest are classified and individual structures are described. Fourth, the relationships between the structures are inferred.

This same approach can be applied to 3-D medical images to assist radiologists in

visualizing the structure of a patient's body. A combination of classical image processing and knowledge-based image interpretation is needed to perform the complete process.

This chapter is a discussion of the relevant material from the fields of image processing, medical image analysis and display, and computer vision. The first section deals with methods of image segmentation. Three approaches are discussed, each in its own subsection: thresholding and clustering, edge detection, and region extraction. These discussions are not meant as an exhaustive survey but merely as an overview of the different approaches to solving the segmentation problem. There is a wealth of information available on classic image processing techniques [57] [28] and only the basic principles will be discussed here.

Section 3.2 is a review of some voxel-based medical imaging systems. Most of the important work in the display of 3-D medical images has been in this area. Section 3.3 is a review of knowledge based approaches to medical imaging. Section 3.4 presents some of the work done on multi-modality image integration and data set registration.

A comprehensive review of 3-D medical imaging systems and techniques was given in [67]. Numerous systems were described from fast surface and volume rendering and display technologies to image segmentation schemes. Of particular interest was a description of the unique aspects of 3-D medical imaging which distinguish it from the general field of 3-D image processing. First, the large medical data sets imply high computational cost even for simple processing. Second, mathematical models cannot be used to simplify the data, the integrity and accuracy of the data is paramount. Third, a 3-D display of the data must be flexible, able to display the data set from any angle quickly and interactively. It was suggested in the paper that, while none of these aspects is individually unique to medical imaging, the confluence of all these aspects is what makes the field unique.

A discussion of the trends in 3-D medical imaging was given in [15]. Four tasks related to 3-D imaging systems were highlighted: image formation, image segmentation, fusion of information and simulation. Image formation is rarely treated in articles on 3-D image processing, most assume previously formed images. In this case, the problem of forming a 3-D image from a severely limited number of projections was discussed. Image segmentation was described as an open problem which needed further work. Direct segmentation in 3-D as opposed to 2-D segmentations of individual slices was suggested as possibly offering better performance. Data fusion leading to multimodality imaging was presented as an important area of research including data set registration, feature specification, elastic image matching and simultaneous multimodal display. Simulation, in terms of surgical and radiation therapy planning, was presented as a high-level research activity which fuses all the features of 3-D medical imaging systems. This research activity was cited as being important for education as well as for increasing the accuracy of treatment and surgery.

3.1 Image Segmentation

Image segmentation utilizes the differences in texture, colour, gray level, orientation, position and motion between objects in an image to create divisions between them. Following Fu and Mui [25], approaches to segmentation can be divided into three classes; feature thresholding or clustering, edge detection, and region extraction. These general classes will guide this discussion and are included as the three subsections.

A definition for segmentation has been formulated by several authors as, for any grid X, a predicate P partitions X into disjoint, non-empty subsets $X_1, X_2, \ldots, X_N$

under the following conditions [25],[82],[34];

$$\begin{aligned} \cup_{i=1}^{N} X_i &= X \ \ (all\ points\ processed) \\ X_i\ ,\ i &= 1,2,\ldots,N\ is\ connected\ \ (regions\ are\ contiguous) \\ P(X_i) &= true\ for\ i = 1,2,\ldots,N\ \ (determines\ segmentation\ criterion) \\ P(X_i \cup X_j) &= false\ for\ i \neq j\ \ (maximal\ area). \end{aligned} \tag{3.1}$$

Image segmentation techniques can use parametric input to include heuristic knowledge and tune the segmentation to specific types of images. Knowledge based approaches to image segmentation use explicit information about the scene being imaged to either decide if a given segmentation is correct, choose between alternative segmentations, or search for specific objects. This class of image understanding systems generally relies on a low-level initial segmentation and uses the knowledge-based approach for higher level segmentation. Examples of these types of systems are given in Section 3.3.

3.1.1 Thresholding and Clustering

Thresholding can be described mathematically as [25];

$$\begin{aligned} S(x,y) &= k\ if\ T_{k-1} \leq f(x,y) < T_k \\ k &= 0,1,2,\ldots,m \end{aligned} \tag{3.2}$$

where (x,y) are the spatial (2-D) coordinates of the pixel, $S(x,y)$ is the segmented image function, $f(x,y)$ is the feature image function, $(T_0,T_1,\ldots,T_m)$ are the threshold values and m is the number of levels of segmentation. The threshold values, T, can be written in general form as;

$$T(x,y,N(x,y),f(x,y)) \tag{3.3}$$

where $N(x,y)$ is a computed property local to the point (x,y). A classification of thresholding schemes can be made from this functional relation. If T is a function only of $f(x,y)$ it is a global threshold. If T is a function of both $f(x,y)$ and $N(x,y)$ it is a local threshold. However, if T depends on all four parameters, it is a dynamic threshold.

The gray level histogram has been used as a method of setting the threshold levels. Thresholds can be chosen at valleys in the histogram thus compressing the image into fewer grey levels. Or, if the percentage of pixels taken up by a single object in high contrast with the background is known, a threshold level which includes that percentage of pixels can be set. The major drawback of these thresholding schemes as applied to medical images is that the pixels in the segmented regions are not necessarily contiguous.

An extension of the standard thresholding scheme using a mathematical morphological algorithm was applied to MRI scan images by Schmitt and Preteux [63] [9]. This algorithm was useful for finding specific features in an image. The gray level height and average slope of the feature were given beforehand and the image was scanned to find only those features with the appropriate height and slope. In this way, features such as brain tumours and mammary nodules were found in MRI scans and tumours in the spinal canal were found in CT scans.

Clustering is an extension of the 1-D thresholding technique which uses a multi-dimensional feature space to segment an image. Features such as texture or colour as well as gray level are commonly used for clustering algorithms. The advantage of using a multi-dimensional feature space over a straight thresholding scheme is that, if no single feature has sufficient discrimination to segment the image, a combination of two or more may work.

An approach to clustering using fuzzy set theory was discussed by Ortendahl

and Carlson [50]. Fuzzy clustering was applied to the problem of segmenting an MRI data set using all the measured parameters including; relaxation times T_1 and T_2, spin density, and motional states of the nuclei. It was postulated that a more accurate segmentation of the data could be achieved using the entire 3-D data set. For the latter stages of segmentation, a model for the partial volume filling phenomena was presented.

Fuzzy C-mean clustering was described as a method of automatically segmenting a feature space into a preset number of regions [50]. Initial guesses at the center points for each region were designated randomly but the accuracy of the initial guesses did affect the run time to a solution. The procedure to find the true centers of the regions started by computing a membership function using a distance metric measurement between the feature vectors and the initial centers. New cluster centers were then calculated based on these membership functions. If the cluster centers had converged, and the new positions were not significantly different from the old ones, then the procedure finished. If not, the process iterated again. It was not necessary to use all of the points in the image to calculate the centers. A random selection of points can be used or just points from a specific area of the image. Although the method was described as having been quite slow, it was found to be effective in performing the correct segmentation.

Statistical approaches to segmentation have also been used. In [39], an investigation which uses a stochastic model of the CT imaging process to decide on segmentation was described. The image was divided into a preset (but variable) number of regions and regions were constructed in the image based on the statistical properties of the pixel values. As with most thresholding methods, the program did not require the regions to be homogeneous or connected. For example, all of the pixels classified as bone in the image would be considered as one region whether or not they were all

connected.

3.1.2 Edge Detection

Discontinuities of image features identify edges between separate regions in an image. Inherent in these edges is most of the information contained in the image. Therefore, edge detection has been an important method for image segmentation. There is also some evidence that edge detection is employed by the human visual system for image understanding [61].

Both the objects themselves and the boundaries between them are important. There are two complementary approaches to the segmentation problem; either find the boundaries between the regions and thus the regions themselves or find the regions directly and thus the boundaries. This Subsection will detail the boundary approach and the next the region approach.

Edge detection schemes can be classified as follows [25];

A Edge element extraction

1. High-emphasis spatial frequency filtering
2. Gradient operators
3. Functional approximations

B Edge element combination

1. Heuristic search and dynamic programming
2. Relaxation
3. Line and curve fitting

With respect to edge element extraction, there are two general methods of performing segmentation by edge detection; a parallel approach and a sequential approach. With a parallel approach, the decisions are based only on the local feature data and are not dependent on decisions made elsewhere in the image. Therefore, the processing in different regions of the image may be done in parallel. The sequential approach uses decisions made about other regions of the image to make decisions about the local region.

Of the edge extraction techniques listed, the high frequency filtering and the gradient operator approaches are commonly used and require no further explanation. The functional approximation approach was first used by Hueckel [35]. His approach searched in a circular region for an edge at a point which could be characterized by a step function.

With respect to edge element combination; heuristic search and dynamic programming use graph searching techniques to find the optimal combination of edge elements to form a complete edge. Various schemes have been proposed; all relying on some form of cost function to evaluate the relative merits of path choices and all sequential in nature.

Relaxation techniques have also been employed to combine edge elements. The position of the edge is updated after each iteration; guided by previous results. The processing can be done in parallel thus saving time, however, convergence may be slow.

A relatively recent method of segmentation by edge detection used deformable surfaces to find specific regions in the image. This technique required a roughly sketched initial estimation of the surface which is then iteratively deformed to match the boundary of the object. The deformation is driven by opposing internal and external forces until the system is in a minimum energy state. The method has been

used on 2-D images [37] and has recently been extended to 3-D data sets [16].

The method of watersheds (WS) or *la ligne de partage des eaux* (LPE) is suitable for segmentation when applied to a gradient image. This method has been investigated using mathematical morphology by [78] and [59].

The following analogy helps explain the concept of the watershed segmentation method. If the pixel values in a 2-D gradient image are thought of as elevation values on a map, then the valleys in the elevation map form the drainage basins for each local minimum in the image. Finding the watersheds is like flooding the image from the bottom through the minimum of each drainage basin. To carry the analogy further, if a small hole is placed at the local minimum of each basin and the whole image slowly immersed one pixel level at a time, the water will start to flood each basin from the bottom as soon as the water level reaches the minimum level for that basin. Wherever the water from two separate basins touches, a continuous line is formed which corresponds to the outline of the watershed.

The process can be formally described in the language of mathematical morphology following [78]. The threshold of an image, I, at a level h, can be written as:

$$T_h(I) = p \in D_I, I(p) \leq h \tag{3.4}$$

where p is the pixel and D_I is the domain of definition of the image. A drainage basin $C(M)$ is the basin associated with a minimum M. The subset of $C(M)$ which have a level less-than or equal to level h is then;

$$C_h(M) = (p \in C(M), I(p) \leq h) = C(M) \cap T_h(I) \tag{3.5}$$

The set of pixels belonging to a minimum at level h is written as $Min_h(I)$. To define the WS by the immersion method, the lines can be described as the complement of the set of drainage basins contained in the image and obtained from the following

loop;

$$X_{h_{min}} = T_{h_{min}}(I), \tag{3.6}$$

$$\forall h \in [h_{min}, h_{max} - 1], X_{h+1} = Min_{h+1}(I) \cup IZ_{T_{h+1}(I)}(X_h). \tag{3.7}$$

where h_{min} and h_{max} are the smallest and largest pixel values in the image domain and $IZ_{T_{h+1}(I)}(X_h)$ is the union of the geodesic zones of influence of the drainage basins associated with the threshold at $h+1$ within the region X_h. X_h is the assembly of drainage basins in the image I at the level h and $X_{h_{max}}$ is the assembly at the highest pixel value denoting the final state of the immersion.

3.1.3 Region Growing

A classic reference on region growing was written by Zucker [82]. In it, the following five approaches to region growing were discussed:

1. Regional neighbour search
2. Multiregional heuristics
3. Functional approximations and merging
4. Split and merge
5. Regional interpretations and semantics

The regional neighbour search, originally conceived by Muerle and Allen [47], started with small regions of pixels (2×2−8×8) with similar gray levels. The statistics of the gray levels within the regions were calculated and, starting in the upper left of the image, neighbouring regions were combined if they had similar statistics. After testing, the regions were appropriately labelled. Successive neighbouring regions were

tested until all possibilities were exhausted. At that point, the next unlabelled region was used as the initial seed for the region grower.

The second approach, multiregional heuristics, was first developed by Brice and Fennema [14]. Initial boundaries between atomic regions, each having a constant gray level, were marked on a set of lattice points interlaced with the picture points. The regions separated by the initial boundaries were then merged using two heuristic algorithms which tested how "weak" the intervening boundary was. The first heuristic joined two areas if the strength of the boundary was less than a preset threshold and the resultant boundary was shorter than the original two. The second heuristic merged two regions if the length of the "weak" part of their common boundary was a certain percentage of their total boundaries. Finally, the boundaries were smoothed.

Functional approximation, although quite similar to the first two region growing techniques, offers a more solid mathematical basis for this type of region growing [51]. Each atomic region was described by a functional approximation within some measure of error. For example, the gray level over a region might be approximated by a constant value. Merging decisions were then made based on a comparison of the functional coefficients of neighbouring regions. With constant valued functional approximations this technique was very similar to the multiregional heuristics and regional neighbourhood search techniques, however, more complicated functions could be used to solve problems which had regions with non-constant gray level.

All of the previously discussed region growing techniques have been based on the concept of combining sub-regions together to form the region. The opposite approach is to start with one region consisting of the entire image and break it down into regions. The next two approaches operate on this principle.

The split and merge approach to region finding, developed by Horowitz and Pavlidis [34], extended the concept of functional approximation to a pyramidal data

structure. The data structure was organized into a series of views of the same image at successively greater resolutions. At one extreme, the image was represented as a single average value; at the other extreme each pixel in the image was described. Merging consisted of combining regions at a higher resolution into one at a lower resolution; splitting performed the opposite function.

The pyramidal structure also served as an analogy for the segmentation tree. Nodes of the tree were the regions and edges were the split and merge operations. Segmentation was a process to find a cutset of the tree in which all of the regions satisfied the functional approximation predicate. Using only the hierarchical split and merge procedures formed only square regions so an additional procedure which combined regions into irregular shapes was used to do the final merging.

The last approach discussed in Zucker's paper [82] was regional interpretations and semantics; based on work done by Feldman and Yakimovsky [23]. In this approach the goal was to maximize the probability of correctly segmenting the image. The probability of correct segmentation was calculated as the product of the probabilities that each region was correctly segmented and that inter-regional relationships were correct. The correctness of each region was estimated by using *a priori* knowledge about the scene being imaged. In order to reduce the data, the initial segmentation was done using a simple thresholding scheme. Then a tentative segmentation was proposed using shape criteria for each of the regions. From this point a guided search for the most likely segmentation was done.

3.2 Voxel-Based Medical Image Processing

This section is a review of several medical imaging systems which use voxels as a basis for creating and displaying 3-D images on a video screen. The data sets used by the

systems were collected on a CT-scanner or a MR imager.

An early contribution in this field discussed methods of region filling between boundaries in a 3-D image [71] . The 3-D image was a series of 2-D cuts in the (x, y) plane. In order that the voxels be cubes and not just rectangular parallelepipeds the slices were interpolated in the z-direction. Object edge contours were found in each 2-D slice based on an 8-connectedness of the voxels at the edges of the object and, by using specific information about the edge voxels, the interior of the object was identified. The voxels in the interior of the entire 3-D object were then the intersection of all the sets of voxels interior to the object in each slice. The border voxels of the object in 3-D were found directly from the total set of voxels by identifying voxels which lay on the edges. A further application of these ideas was to use an operator-drawn contour to perform electronic surgery by excluding certain voxels from the interior of the object.

Several medical applications using 3-D images were given in [31]. Identification methods were discussed including an interactive segmentation procedure used when automatic classification will not work. Methods of manipulating 3-D images based on a cuberille environment and on a directed contour representation were given. The display of 3-D medical images was also discussed. Four depth cues were used: hidden surface removal, shading, transparency effect, and motion parallax by rotation. A brief description was given of six projects where these ideas were currently being applied. These include a study of joint disease, an NMR imaging application and the use of 3-D images for surgical planning.

A comparatively simple system for 3-D display of intracranial tissues was described by Vannier *et al* [75]. It was implemented on an unmodified CT-scanner with no extra hardware. High resolution scans of the head with 2mm spacing were used to construct 3-D images of patients with craniofacial and intra-cranial abnormalities and

neopla ..c diseases of the head and neck. Ten, 3-D, views were computed from each session; frontal, rear, 45° oblique (anterior and posterior), lateral and top and bottom. Separate sets of images were created for both bony and soft-tissue surface displays.

The thresholds used by the contour extractor were set by the operator; all other processing was automatic and ran unattended. The discontinuities between contours in adjacent slices were smoothed using a non-linear, anisotropic, digital filter and were displayed without perspective depth. In choosing the contour extractor thresholds, the operator was selecting the surfaces to be displayed. No performance data was given in the paper but these images were formed overnight for display the following day.

The application of 3-D image presentation systems to radiation therapy and surgical planning was discussed in [10]. In radiation therapy planning it is paramount that a uniform radiation dose is delivered to the target area while minimizing the dosage to surrounding critical structures. Dosage calculations should account for absorption changes in the radiation field and the best angles for the radiation to be delivered should be used. Three-dimensional imaging from CT-scan data was very useful in this application since the geometry and strength of the radiation therapy beams could be superimposed on the anatomical detail provided by the CT-scan.

The surgical planning application had two main goals. First, an interactive capability to perform electronic surgery on representations of organs generated from CT-scan data was required. Second, a method of translating the manipulations done by computer into surgical procedures would be necessary if this tool was to be useful clinically. Although the paper does not contain many details, some results of electronic surgery were presented. Their first goal of being able to perform the electronic surgery was at least partially realized. However, as described in the article, the problem of translating the electronic procedures into the operating room, "needs

considerable further work".

A more complete explanation of the surgical planning problem was described in [13] . A nomenclature was developed where a 3-D collection of data (CT-slices) was termed a *scene* and the meaningful objects in the data called *organs*. The manipulation was carried out using *models* of arbitrary shape which can be removed from or added to any *organ*. The process was iterative since the complex models must be built-up step-by-step.

As in previous work [72] , a distinction was drawn between working with slices and working directly with the 3-D data. The 2-D ISP (Interactive Surgical Planning) techniques described in this paper used 2-D models on each slice. Geometric models were generated from algorithms and may be modified by changing the parametric inputs to the algorithms. Organs were displayed either by colouring the associated pixels or by outlining the organ with a contrasting boundary. As discussed previously, the model and organ limits were defined by *directed contours* in 2-D but they can be displayed using one of the 3-D display techniques by connecting the 2-D contours in the axial direction. In the 3-D ISP scheme, models were created and modified in 3-D and, once again, geometric models could be modified by changing the parametric input.

The greatest problem with these ISP schemes, as described by the authors, was the speed of the process. Not only was the display slow to update but the model generation and modification process took a considerable (but unspecified) amount of time. Better methods would have to be found if this system was to be implemented clinically.

Another system for colour 3-D imaging of intra-cranial structures was discussed in [22] . Structures were not separated by finding the surfaces as in other studies but were interpreted directly from the display by the operator. Although several

depth cues were implemented the main technique was to use different transparent colours to display voxels at different ranges. Four advantages of this method were given: different structures were displayed as different colours (but only if they were at different ranges), during image generation the slices were successively built-up on the screen, it was fast and the data need not be continuous. Other options for display were; point of view, angle of oblique projection, cutout, transparency, contours, and stereoscopy. The transparency was achieved by displaying only every second image point in a checkerboard fashion. Stereoscopy was implemented in this system by generating two projections $15° - 20°$ apart and viewing through either a reflecting stereoscope or by cross-eyed viewing. A further comment was that these images were formed using a slice thickness of 5 *or* 10 $mm.$ and could be significantly improved by using slices $1.5 - 2.0$ $mm.$ in width.

A hardware driven approach to a 3-D medical imaging workstation using a "Voxel Processor Architecture" was given in [26] . This article stressed that interaction in three dimensions and real-time performance are two capabilities which should be part of any clinically useful display system. Real-time was considered to be near the refresh rate of the video screen (≈ 30 $Hz.$). The authors claimed that the difference between this and a few frames a second was analogous to the difference between a movie and a slide show and that depth cues from motion parallax were significantly better at the faster rates.

Also included in this paper was a list of desirable features of a 3-D medical image display system. They were listed as; selection of a spatial window, selection of a density window, rotation of a volume of interest, translation and scaling of a displayed object, dynamic object display, direct editing of the scene, and special interactive facilities. These features are self-explanatory except for the last one; it simulated surgical procedures and radiation therapy planning sessions. To implement these

features it was stated that four fundamental capabilities were necessary: segmentation of the volume of interest, projection from 3-D to 2-D, hidden surface removal, and realistic shading.

The four main components of the software environment were also discussed in this paper; preprocessing utilities (interpolation, filtering), volume of interest extraction (sub-sampling), voxel display processor control (real-time interaction) and off-line image generation (necessary only because the prototype was only a $64 - cube$).

A full implementation ($256^3, 16\ bit$) of the proposed hardware system had not been done but a prototype ($64^3, 4\ bit$) had been tested and this was found to be adequate for small objects. Images in the region of the eye and of the temporo-mandibular joint were presented; the latter application being particularly interesting since the movement of the joint was visualized. A more complete discussion of the *Voxel Processor Architecture* was given in [27] . More detail was given about the hardware and the relative merits of 1-D (contours), 2-D (patches), and 3-D (cuberille) primitives were discussed.

A semi-automatic three-dimensional image segmentation algorithm for CT and MRI scan images has been presented by Acharya and Benchimol [1]. The four stages of the algorithm were:

1. Preprocessing
2. Specification of search space
3. Determination of surface candidate elements
4. Surface element estimation

The preprocessing stage employed noise cleaning filters to improve the overall quality of the image as well as histogramming techniques designed to normalize the average

intensity and range of gray levels. This aided in making comparisons between images.

The specification of search space was used to confine the 3-D volume search for the region borders. This decreased the amount of time spent searching and made the search more reliable. The search space was specified using a number of methods, namely: manual input, input from adjacent groups of slices, orthogonal back projections, mathematical morphology techniques and, anatomical knowledge. It was not mentioned exactly how these techniques were used. The determination of the surface candidates was done using a 3-D Sobel edge operator and the bicubic patch approximation used to represent the surface.

The surface approximation program used a hierarchical filtering scheme which incorporated anatomical knowledge into the process. The hierarchical filter was apparently implemented so the overall shape was found while maintaining the local details of the surface. The proposed applications for this program were tumour surface extraction in CT scans of the head as well as organ surfaces in CT body scans.

A novel hardware architecture for processing and displaying 3-D medical images called *Cube Architecture* (or *MediCube*) was presented in [38] and [5]. Of particular interest to the present discussion was the two methods used for image segmentation in *MediCube*; spatial and density segmentation. Both could be used to edit the image. Spatial segmentation was implemented by defining box-like regions within the image. These regions could be combined and manipulated in several ways and were used to isolate a particular spatial region. Density segmentation was implemented using a straightforward thresholding scheme.

A system for developing a 3-D database of human anatomy was briefly described in [65] . This system used photographs of thin ($250-300\ \mu$) slices of a frozen cadaver as input to a 3-D database. The pin-registered photographs were digitized with a colour camera and loaded into a PIXAR imaging system which can take sections in

any plane for display. The article mentioned that future work may include segmenting the database into areas of anatomical functionality.

Software developed by NASA for multispectral image processing of satellite images has been applied to MRI scans by Vannier *et al* [74]. Since MRI scanning equipment can collect data on several different parameters, the multispectral approach was ideally suited to the task. It was found that a maximum likelihood classifier performed best on the data and that the accuracy of tissue classification varied between 71-94% over 8 patients. It was also found that an engineering workstation would be a necessary addition to the computer facilities in a radiology clinic in order to run this software.

3.3 Knowledge-Based Medical Image Processing

Extracting abstract information from images requires an organizational representation or knowledge base of the possible situations which can exist in that image. It is also necessary to identify or match the structures in the image with *models* in the knowledge base. These models can be *analogical* or *propositional*; a distinction which becomes blurred in many applications but which is, nonetheless, a useful concept.

Analogical models are applicable mostly for low-level processing. As the name suggests, the model is an analogy of the situation being imaged. For example, a 3-D geometrical model of a human kidney using a voxel representation would be considered an analogical model. A *propositional* model would be a higher-level, more abstract representation of the object. A description of the human kidney using a set of predicate calculus clauses is an example of a propositional model. In any particular vision system both of these models can be useful and a combination of the two is sometimes more efficient [6].

An example of a knowledge representation which contains aspects of both analogical and propositional models is a semantic net. Objects in an image are represented as *nodes* and the relationships between objects as *arcs*. As an analogical model, a semantic net would have a structure similar to the relations in the scene being imaged. If the scene was a plan view of a town, all important buildings would be represented as nodes and the spatial relationships between the buildings as arcs. A logical extension of this model would be a complete map of the town stored in a 2-D array. All the necessary positional information would be contained in the structure and procedures could be written to extract distances, angles or more complex information.

An important aspect of image understanding is *matching*; forming the connections between the knowledge base and the structures in the data. These connections might consist of a one-to-one correspondence between the unknown representation and the knowledge base representation when the two models have the same structure. For example, if the input image has been segmented and the size and shape of the regions have been computed these could be directly matched to the size and shape parameters described in a knowledge base.

Another type of matching could be between two different types of representations; for example, matching an image with a relational structure. Since there is no one-to-one correspondence in this case, the term *matching* really means interpreting one or the other of these representations and then making a comparison.

A third type of matching might be testing a structure against a set of rules in the knowledge base. As the unknown structure was tested with inquiries from the knowledge base a coherent idea of what the unknown structure was would emerge; provided the right questions were asked.

Parameter Optimization techniques are a useful means of matching if an appropriate merit function can be found to measure the differences between the two representations. This technique is usually applied to matching problems involving geometric shapes or mathematical models. There are several measures of similarity possible and in all cases a global search for a minimum of the error measure is done by systematically varying the parametric inputs to the model.

Matching graph representations can be a complicated process since it may be impossible to find a matching metric and the search may be unguided. If a perfect graph isomorphism is not required; a cost based on the number of templates found, amount of displacement from expected position and the number and importance of missing templates can be calculated. The costs of misshapen or imperfectly located (according to the knowledge base) templates can be non-linear. Exhaustive graph searches can also be done using one of several search schemes; for example a straight backtracking search. In this case, when a proposed matching (even partial) is obviously wrong the search can stop, discount that possible route, reverse the last step taken in the search and start again on the next possible matching. Although this system is exhaustive it can be lengthy and several more sophisticated searches have been developed [49]. Several specific applications of these types of matching can be found in [6].

A *frame* implementation of a knowledge base is a particular example of a semantic net [45]. The term *frame* refers to a partitioned piece of knowledge which contains information on various aspects of a node or arc. This information is contained in a highly structured system of *slots* which can be instantiated upon creation to default values, updated, or left unfilled until that knowledge is obtained. One reason this highly compartmentalized structure was developed was that it provided a method of recalling all information relevant to a particular situation by matching with only some

of the slots. So, a frame can be completely recalled with only partial information.

Another important issue in the organization of a system which uses knowledge to describe relationships in an image is whether the control is driven by the data (bottom-up) or by the knowledge base (top-down). To explain this distinction further, each structure found in the data during image segmentation could be matched against the entire set of possibilities described in the knowledge base; this is *bottom-up* control. *Top-down* control starts with a concept from the knowledge base of what could be found and then, dividing the problem into smaller problems, searches through the data to find if any structures match the prediction. It is, of course, possible to mix the two control strategies in an application according to the stage of processing for increased efficiency. A good general reference on this subject is Nandhakumar and Aggarwal [48].

Knowledge-based models have also been successfully applied to the problem of labelling blood vessels in carotid angiograms [79]. The processing was divided into three stages. First; short, straight, lengths of vessel were found. Next, these bars were strung together to form ribbon sections without any crossings or branches. Finally, the ribbon sections were combined to form whole vessels which could be identified and labelled. The system was tested successfully on several binary images and may be extended in the future to non-binary carotid angiograms and eventually to cerebral angiograms.

Two types of knowledge were used in this system; *facts* knowledge pertaining to the anatomy and the imaging process and *strategy* knowledge to determine when and how to apply the facts. The backtracking system used confidence parameters for both types of knowledge to guide the search and to solve conflicts. Another system, able to segment non-binary angiograms, was proposed which would use forward chaining for the initial segmentation.

An automated image analysis program has been developed for 2-D, CT-Scan images by Winter [80]. It used contour lines to separate regions with unit differences in density. This contour map then provided the structural organization for an image recognition program. The contour lines were non-intersecting, closed curves which either completely enclosed (or were enclosed by) other contour lines or were adjacent to each other. These contour lines divided the image into non-intersecting sets of pixels; the resultant image being equivalent to a chain coded image. In this case, the contour line approach was preferable to straight thresholding since the regions were contiguous and the random density fluctuations of the soft tissue were high enough to cause the thresholding approach to fail.

A semantic net was constructed using arcs from the inside to the outside of the contour lines connecting the regions. The parameters of the inter-contour regions such as area and perimeter were stored explicitly to make further segmentation easier. Preliminary segmentation of sub-regions was done by bottom-up construction of the semantic net; up to the level of defining "lucency" and "density". At that stage, a top-down search of the entire image was initiated. A sophisticated control strategy was used for searching the tree which chose the ten most promising leaves and used heuristics to find the important areas in the scan. For example; a CT-scan of the head, at the level of the Pineal gland, was used as a test case. The program found and correctly identified the skull, the atria of the left and right ventricles, the frontal horns, the third ventricle and a large pineal tumor. There was no indication given in the reference of how long it took to do the processing.

A system called IBIS (Interpretation of Biomedical Images of the Slice type) for interpretation of 2-D NMR scans has been described by Vernazza *et al* [77]. A propositional representation of shape, spatial and densitometric features of important intracranial structures was contained in the database in the form of a semantic

network. These features had wide enough tolerances to permit matching with a reasonable range of scanned images. Although the system was designed to process single slices, some of the anatomical models were represented as 3-D structures to avoid the redundancy of describing each structure for each possible slice. A choice of four slice planes were represented in the system. The control for the search was hybrid; both top-down and bottom-up and relied on *a priori* knowledge to trim the search tree. The basic strategy was:

1. Look for distinctive organs.
2. From results of step 1, hypothesize a tomographic plane.
3. Search for other organs using intrinsic and relational information.
4. Assign the unrecognized areas to the most likely organs.
5. If the general match is quite close, concentrate on finding missing organs.
6. Otherwise, restart and consider a new hypothesis.

Steps 1 and 4 were data driven; step 3 was driven by the model.

Only an overview of the software organization will be given here since a detailed description would be too lengthy. There were four main parts to the system; the low-level subsystem, a middle-level subsystem, a high-level subsystem, and a global database.

The low-level subsystem provided calibration, filtering, and smoothing and did a preliminary segmentation of the image into elementary regions using thresholding and region growing. The middle-level subsystem contains modules for feature extraction. Various statistics of all the generated regions were computed such as area, average grey level, position, direction, and approximate shape. The high-level subsystem

handled the data at a symbolic level and had four main parts: domain knowledge, high-level procedures, control structure, and man/machine interface. This included rules for everything from segmentation to object recognition to overall control. The global database contains the results of the recognition process at the various stages of processing.

A more recent description of the IBIS system is given in [18]. New additions to the system are discussed including fuzzy matching of the segmented data to a propositional model. Only 2-D images are treated in the paper. The first stage of the process is to perform a low-level segmentation of the images into regions by thresholding. Next, a feature tree is used to classify the regions found in the slice into specific categories according to extracted features. Fuzzy criteria are used to make the classification decisions and the easily classified regions regions are decided first. Manually segmented training images are used for the feature classification.

A 3-D medical image display program was recently described which focussed on quick visualization in an interactive viewing environment [81]. The software was built around a high-performance, parallel-processing, computer system and much of the paper described the research done on high speed rendering algorithms. Of particular interest, however, were the approaches to segmentation and classification. Useful definitions of segmentation and classification were given in the paper: segmentation was described as, "the division of a data set into often non-overlapping subvolumes that have some cohesive meaning unto themselves" and classification was described as, "the process of labelling various portions of an image by type".

Two types of classification were distinguished; *syntactic* and *semantic*. Syntactic classification uses only the information in the data set such as voxel intensity, local gradient and connectivity. The semantic grouping of small segments of 3-D data was used as a basis for a shape driven classification. The user was called upon to

group these primitive volumes together to form larger, more meaningful, regions. The authors of the paper contended that this kind of control best rests with the knowledgeable user of the system and should not be done automatically. Two techniques for syntactic segmentation in a pre-processing stage were described; a reverse-gravity watershed technique and a ridge-flow technique. Although the details of the implementation were not given, it was clear that a tree structure was found with the connections in the tree set by the user in an interactive interface.

Another knowledge-based system used for 3-D image analysis is described in [19]. Two separate examples are given of segmenting medical images; one study of the heart and one of the head although the main use of the system was for CT and MR studies of the chest. The first level of the processing was a segmentation using K-mean clustering, region extraction and rule-based merging. Masks were used as an overlay to limit the segmentation to medically significant areas on the images. Once the initial, low-level, segmentations had been performed, the segments were matched against a model using features such as average grey level, shape and area. The shape of the segments was described using the centroid and the central second moment. To "fine-tune" the initial matching, adjacent segments were added and subtracted from the aggregate regions to improve the match.

A different procedure was employed for segmenting the ventricles in an MRI study of the head. Using three seed values from each of the two peaks in the grey level histogram, a K-means clustering was performed on each slice. A binary image was then created from the three highest clusters corresponding to the grey levels for the sulci and the ventricles. A slice dependant mask was used to separate the ventricles from the sulci. Finally, the ventricle segmentation was adjusted manually for the final display.

Understanding of scan-type medical images was discussed by Stiehl [66]. The

acronym SISU (Spatial Image Sequence Understanding) was used to describe the process of automatically or semi-automatically interpreting the data sets to aid in 3-D display, surgical planning and radiation therapy planning. No specific systems were detailed but a general overview of the research area was presented. Image segmentation was identified as a severe bottleneck in SISU systems since it often required interactive contour designation in each slice of the data set; a time consuming process.

3.4 Multimodality Image Integration

Combinations of images from different medical imaging modalities including CT-scan, MRI, Positron Emission Tomography (PET) and angiography can provide not only better visualization for the radiologist of the internal structure of the brain but can directly aid in surgical and radiation therapy planning. Since it is necessary to match the images in three spatial dimensions, a secondary benefit and goal of this research is the 3-D interactive display of these composite images.

A prime motivation for developing systems to assist in neurosurgery planning is the amount of data needed to do the planning; up to 400 separate images can be used for just one patient. Although the systems may not decrease the number of images which must be processed, they can make interpretations of the images easier.

An interactive, stereotactic, display workstation for neurosurgical planning was developed by Peters *et al.* [55] [29]. The system was designed to merge volume rendered CT and MRI data with stereo digital subtraction angiograms (DSA) into a single image which could be viewed on a stereo display apparatus.

The term *stereotactic* comes from the neurosurgical techniques used to precisely locate a point in the brain. A stereotactic frame was rigidly fitted to a patient's head

which provided a reference coordinate system common to both the surgical procedures and the images. Panels with specially coded markers embedded in them were inserted into the faces of the stereotactic frame for each different imaging procedure. These fiducial markers were then automatically located by the image processing system in each image and provided the reference.

The images were collected on commercially available equipment. The CT images were 512^2 pixels by 12 bits and the field of view was 345 *mm*. The MR images were 256^2 by 12 bits with a field of view of 325 *mm*. The stereo DSA images were 512^2 pixels by 10 bits with a 190 *mm* or 270 *mm* field of view. All of the image processing and volume rendering of the tomographic data was done using a PIXAR image computer. Preprocessing included thresholding to minimize noise and enhance tissue contrast, correction of skew caused by gantry tilt, and interpolation and resampling to create cubic voxels. Before stereo projections of the tomographic data were taken, the geometry of the DSA images was calculated and the same parameters applied to the tomographic data. The data was volume rendered producing a pair of stereo projections corresponding to the DSA images. Final scaling was done after the rendering. Of special interest in the display was the use of a colour monitor and a *hot metal* colour scale for the MR images. It was claimed that this scale provided greater perceived dynamic range in the image.

The stereoscopic workstation allowed 3-D interaction with the image via a 3-D mouse controlled cursor. Vectors could be drawn by specifying the end points thus simulating probes or needles in a surgical procedure. In this way, a path for a biopsy or electrode implantation could be found which avoided major blood vessels and important neural connections. The stereo DSA images were considered accurate to between 3 and 8 *mm*, depending on the method used to collect them, but greater accuracy was attained ($\approx$ 1 *mm*) by using both AP and lateral views. The system

was being used in hospitals on a daily basis by physicians. The authors were planning to upgrade the DSA unit to a 1024^2 pixel system with a stereo shift of 6.5 *cm*.

The same general stereotactic system has been used for beam calculations for radiosurgical treatment of brain lesions [56]. This treatment uses the concurrent rotation of a photon beam linear accelerator and the patient couch around a common point within the lesion. This concentrates the energy at the site of the lesion while minimizing the energy outside the lesion. By using the stereo imaging system with the stereotactic frame the exact location of the lesion could be found as well as the dosage delivered to other sites. The dose distributions were plotted as isodose contours in three orthogonal planes: transverse, sagittal and coronal centered on the lesion.

Positron Emission Tomographic (PET) images show neural function within the brain but do not specifically image anatomic details. Work has been done using regions of interest (ROI's) to describe 2-D anatomical regions and use these regions to relate anatomical and functional images [21]. A 2-D template was overlaid onto MRI slices and positioned to most exactly fit the structures. Individual region boundaries within the template could then be adjusted. By using a stereotactic frame for both the MRI and the PET scans those two sets of images were registered and the edited template could be overlaid on the PET scan. Assuming there were no significant distortions in either imaging system, the functional activity from the PET could be related to the anatomy from the MR images. It worked best if the template was in the same plane in both the MRI and PET slices.

The extension of the regions of interest (ROI) template into a three dimensional volume of interest (VOI) was the next step. A system for anatomic interpretation of PET images has been developed which relies on integration of images from MRI and PET [42]. It was a significant advancement since a template could then be generated in any plane through the volume corresponding to the PET imaging plane. As well,

neural activity in various structures could be easily calculated over the whole volume of the structure and not just in one plane.

The VOI atlas was matched to the MRI data in a two stage process. First, a subset of the set of anatomical features in the VOI were matched to those features in the MR study. From these points, an affine transformation was calculated for a crude match. This match was then refined by the user by selecting translation, rotation and scaling parameters. Second, the match was fine tuned by moving individual surface points on the VOI using the MRI data as a guide. Thus, the VOI was as anatomically correct as the user wanted to make it. The VOI atlas could be displayed either as individual slices in an arbitrary plane or be volumetrically rendered as a wire frame or shaded surface model with polygonal surfaces.

Another approach to matching biomedical images by Apicella *et al.* used a fast correlation algorithm to register images from different modalities [2]. The authors proposed using digital image processing techniques as an alternative to the rigid stereotactic frame technique. Three approaches were discussed: the method of moments, relaxation techniques and correlation analysis. Although the correlation analysis was found to be the most suitable, the authors claimed that previous attempts had not sufficiently decoupled the translation, rotation and scaling shifts which led to computationally intensive algorithms.

The fast correlation matching algorithm decoupled the matching variables to achieve an order of magnitude decrease in the number of FFTs computed over the standard correlation techniques. The fast correlation technique was tested in two ways: 1) using two identical images with Gaussian noise added to one and 2) using two images from different imaging modalities. Rotations of up to $\pm 6°$ were introduced between the images. In all cases, the algorithm identified the direction of rotation correctly but the magnitude was off by $0 - 1.5°$. It was suggested by the authors that

this was well within the estimated 2.8° effective resolution of the system due to the coarse (64^2) image matrix used. Future work included an extension to 3-D correlation analysis.

An overall approach to combining anatomical (CT,MRI) and functional (PET) imagery was proposed by Valentino *et al.* [73]. The authors used criteria proposed by Mazziota [43] for the optimal solution to image requisition and analysis. The overall plan for a system to fit the criteria was:

1. Image data collected from various imaging modalities
2. Distortion corrections on image data
3. Three dimensional correlation of image volumes
4. Morphometric measures in an individual
5. Morphometric measures in a population

The distortion corrections involved mainly proposals for frequent calibrations of imaging equipment. The proposed correlation procedure for the image volumes used a surface fitting technique proposed by Pelizzari [53]. This technique used surfaces of large anatomical structures such as the skull and the ventricles to match the data sets with each other. It was not explained how these structures would be discriminated in the PET data. A method of volume visualization of the images was also discussed. Although many details about the rendering were not provided in the paper, voxels were classified and their display opacity adjusted to highlight them.

An important part of the study was to perform morphometric analysis on the structural and functional organization of the brain. This was done both at the individual patient level and among all the patients. The goal of this part of the research

was to establish a database which included a large number of entries for statistical analysis of the general population.

A three dimensional brain atlas for matching functional and structural images has been developed by Dann *et al.* [17]. This atlas was developed using 135 myelin-stained sections of a young normal brain at 700 micron spacing. Two dimensional contours from these sections were stacked and the resultant 3-D atlas could be rotated, scaled and resliced in three different planes. The contours in the general atlas could be elastically deformed to match a patient's CT scan images. The automatic matching process was done in two stages. First, a global translation, rotation, and scaling registration was done. This was followed by an elastic deformation of the atlas to conform to the structures outlined on the CT scan images. The elastic matching was done at three successively higher resolutions.

The system was tested against experts for proper placement of regions. In one third of the cases there was no difference and in 18% of the regions the computer derived regions were better than at least one on the experts. These individualized templates could then be modified, edited and used to interpret functional images. The display of the atlas or of segmented images was not discussed in this paper.

An early method of combining CT and PET scans using a flexible brain atlas was proposed by Bohn *et al.* [11]. The atlas was developed by slicing cadavers' brains and photographing them at 0.25 *mm* intervals. Using an interactive graphics display system, the important structures of the brain were segmented and labelled by an expert. The transformations available with this atlas were translations, linear stretchings, and planar rotations. The two steps in fitting the atlas to a stack of CT images were to first find a rough global fit on all the stack and then make adjustments on individual slices. All the fitting was done by an operator. It was mentioned that an elastic stretching transformation would be advantageous for matching to a wider

range of structure sizes and shapes.

A slightly different approach to structural mapping of brain images was taken by Herholz *et al.* [30]. An interactive program was used to assist in the contouring of 2-D regions on PET images with high inter-user reproducibility. It was also possible to calculate the metabolic rates within the contours using the PET data. The first step in the procedure was to define the frontal, occipital, and both temporal poles of the brain with a cursor. An outer contour was then fitted to the brain image by thresholding. A standard parameter file was used to provide a rough template for the regions of the brain. Although these regions were very rough; being either circles, rectangles or sectors of circles, they only guided the automated thresholding algorithm used to draw the final contour. Before the automatic searching was done, the raw regions could be adjusted by the user. The contours were drawn using both geometric information from the rough outlines and expected metabolic activity levels from previous studies. This set the threshold for the algorithm but caused problems when an inhomogeneous activity region was encountered. A study of seven PET slices which found 140 regions took about 1 hour to complete.

A comparison of data set registration using an anatomical atlas approach or a mathematical technique using image features was given in [41]. Images collected using CT or MRI were registered to PET or SPECT (Single Photon Emission Computed Tomography) images to correlate the functional images (PET and SPECT) to anatomical features and to improve the accuracy of the SPECT reconstruction algorithm. Five approaches to image registration were discussed: analytical with respect to structure, analytical with respect to surfaces, procedural, anatomic atlas, and external markers. The first two approaches use mathematical techniques such as Fourier analysis, least-squares searches, and moment-matching techniques to register data sets. The procedural approach uses a stereotactic frame fitted to the patient's

head to perform the registration. The anatomic atlas method was described earlier.

The external markers approach was considered the most useful in this investigation. The radiologist identified 7-12 landmarks in both of the data sets and these landmarks were used to create a set of simultaneous equations which related the reference data set (PET or SPECT) to the CT or MRI data set. A linear regression analysis of the system was performed to find the best match based on the chosen landmarks. Two drawbacks of using atlases to perform a similar function were mentioned; they require previously collected data, and the ROI placement over the reference image must be done interactively.

Another brain atlas was implemented by Bajcsy *et al* [4] which used both 2-D and 3-D brain atlases. This system was designed to quantitatively measure structure size and density as well as acting as an individualized brain atlas to overlay onto other images. For display, the contours of the atlas were filled in with values similar to typical values for that region. This gave the displayed atlas slices the appearance of a standard CT image.

The most interesting part of this study was the generation of the personalized atlas from the generalized atlas. The original image was divided into squares of equal size; possibly overlapping. A set of features was derived for each of these regions; six features were used for the 2-D case and four features for the 3-D case. The 2-D features were: average density, horizontal edges, vertical edges, circles within circles, and two types of intersection edges. The 3-D features were average density and edges in the three principle planes. These values in feature space were used to match the elastic atlas to the CT image by solving a set of simultaneous equations which computed the "least energy" state of the atlas.

The algorithm was reported as working fairly well once the atlas had been positioned by the user; provided the two were similar. Errors, when compared to a

human expert were 10% in the 2-D case and 20% in the 3-D case. The 3-D atlas was, however, deemed preferable since the image planes of the atlas and the scan did not have to be coplanar. As well, since PET scans are not typically in the same plane as CT scans, the 2-D atlas was not directly usable on PET images whereas the 3-D atlas could be used by reslicing in the same plane.

This work was expanded by Bajcsy and Kovačič to include 3-D matching using a multi-resolution approach [3]. The matching process was formally described as consisting of a global registration and a global or local matching. The registration procedure used translation, rotation and scaling to match the *model* with the *pattern*. The matching procedure assumed a deformable model acted upon by external forces. The forces were calculated by deriving a local similarity measure which determined the direction and magnitude of the differences between the model and the pattern. The matching was performed interactively; firstly at a course resolution and successively at finer and finer resolutions. In this way, the matching was limited to small deformations at each stage. The processing was performed on 3-D images with cubic voxels generated by resampling and interpolation (in the axial direction). The results shown in the paper matched a brain model to a CT data set. Another study of elastic matching is described in [46].

An interesting approach to combining functional and anatomical data in a 3-D display was given in [52]. The concept of surface mapping was used to display quantitative PET data at specific surfaces of the brain such as the cortex or an artificial surface a fixed distance "below" the cortex. Surfaces could also be defined by intersecting planes or cutouts. It was found that displaying only the functional data at the surface did not accurately depict the 3-D nature of the data so shading was added to the display. Since the colour of the "light" reflected from the surface was different than the colours used to represent activity in the PET data, the shading

enhanced the 3-D effect without degrading the utility of the image. The application included one other feature; semi-transparent surfaces could be shaded to provide a 3-D reference to the data. For example, the cortical surface could be shaded semi-transparently and a second shaded surface, 2 cm below the cortex, could be used to display the PET data.

A comparison of an object-oriented and a prolog based implementation of a multi-modality medical image interpretation system (MMIIS) is described in [24]. Both implementations are currently being pursued but in this, an interim analysis, preference was given to the Smalltalk implementation due to its programming environment and extensive language features.

3.5 Conclusions

At the heart of almost all of the 3-D medical imaging workstations described above is a method of identifying surfaces or volumes in the data set which correspond to medical structures in the patient. Without this information, accurate surface renderings cannot be displayed or surfaces used to register data sets.

As we have seen however, most medical imaging workstations concentrate on fast image display and use relatively simple methods to select surfaces or sub-volumes from the data. Thresholding is commonly used but, although it works well for for segmenting obvious structures, it cannot adequately distinguish many of the more subtle structures. Many of the most advanced systems use an interactive segmentation and labelling scheme involving a radiologist or a skilled operator. While the interactive approach is accurate and relatively robust, it can be labourious and a more automatic method is desirable.

Some systems have improved on the basic thresholding method by using geometric

overlays to identify sub-volumes (or sub-regions) of the data set on which to perform the thresholding. This method is best suited when searching for a specific structure, but is more difficult to implement for a complete segmentation of a data set.

How can these structures be found? As described in the next chapter, computer vision techniques can be used: preprocess the data set, segment it into primitive volumes and merge and label the primitive volumes into medically significant objects. Although these techniques are normally used to understand 2-D images, they can be used on a 3-D data set to great advantage. In some ways, the 3-D data sets are easier to process than most 2-D images since the 2-D images are projections of a 3-D scene thus confounding the process of image understanding.

The problem is thus reduced to several constituent parts: preprocessing, segmentation into primitive volumes, and merging and labelling of primitive volumes. From the numerous segmentation methods described in Section 3.1, we must choose one which best suits the data. We have seen that the initial segmentation of the data is difficult and critical to the success of the entire process. Finding a single segmentation method which works even for a single class of images is difficult due to the variable nature of the images. The success of the segmentation is critical since all of the subsequent processing is dependent on a good initial segmentation. To reason about the segmented image. the merging and labelling stage must compare the segmented image with a model contained in the knowledge base. As we have seen, both analogical and propositional models are useful for performing the comparison and we must decide how to best exploit these types of knowledge representation.

As we will see in Chapter 4, by exploiting some of the features of medical images, by performing the segmentation directly in three dimensions, and by later merging the primitive regions using both analogical and propositional models, a solution can be found for segmenting these data sets. The solution depends, however, on the

development of a strong model of the anatomy using the concept of an anatomical atlas. Previously, using atlases has implied interactive matching by the user; the present approach avoids this by using previously labelled atlases to guide the merging and labelling of the primitive regions created by the initial segmentation.

Chapter 4

A Medical Image Editor

The utility of a multi-modality integration and display system for medical images has been shown in Chapter 1. Three dimensional display of intra-cranial structures can assist the radiologist in visualizing data sets collected from CT, MRI and PET scanning equipment. Although 3-D display tools are well developed, the problem of finding and representing the important structures in the data sets remains.

The solution presented in this thesis is to use an *individual atlas* to describe, quantify, and display the medically important structures in the data. The individual atlas pertains to a single data set. It holds representations of the shape, size, and location of the structures found in that data set as well as quantifiable parameters such as first order pixel statistics. Each medically important structure in the data set is described by a *Volume of Interest* (VOI) and is referenced by name.

An atlas can be used for comparison of images from studies of the same patient, of different patients, or of the same patient from different imaging modalities. Section 4.1 describes the concepts of an atlas and a VOI and how they can be used to abstract the information in the images.

How do we form the atlas? The present approach differs from previous approaches

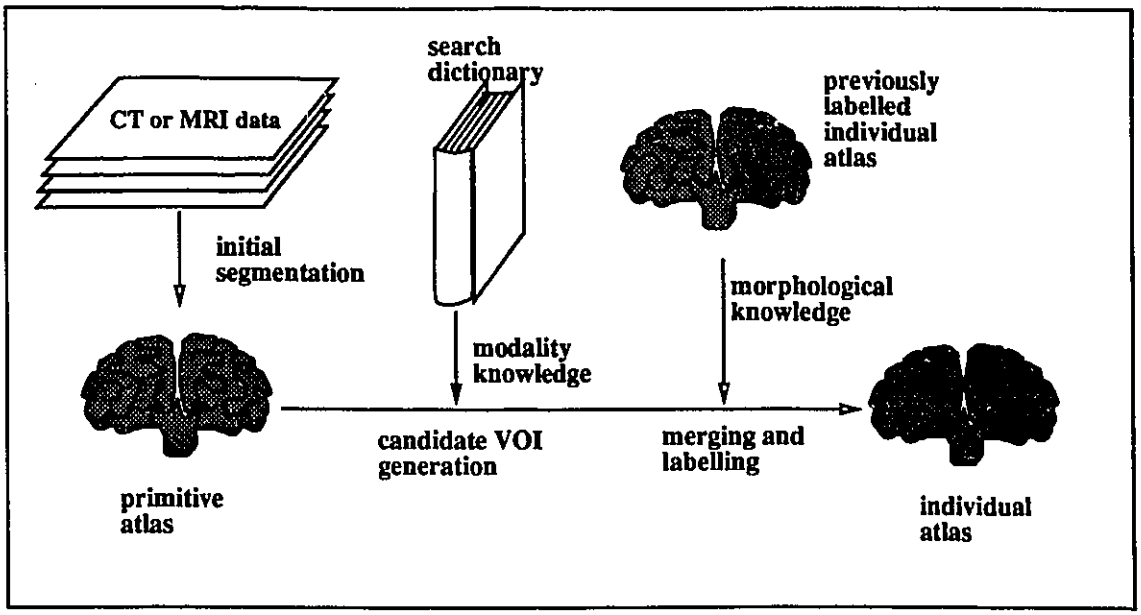

Figure 4.1: The Overall Plan of the System

since it uses the data to generate the boundaries between the regions and it incorporates a considerable amount of knowledge into the atlas creation process. It uses a two stage process: an initial, data driven, over-segmentation of the data into primitive VOIs followed by a knowledge-driven merging and labelling to form the final segmentation and the individual atlas. A schematic plan for such a system is shown in Fig. 4.1.

Before any processing starts the data must be manipulated into a suitable form using preprocessing (sub-sampling, compression, windowing) and filtering. A description of the techniques used to perform these functions is included as Section 4.2.

The segmentation of the data set into primitive VOIs is the first stage in the atlas creation process. Ideally, if each primitive VOI represented a single structure in the brain, the subsequent matching and labelling steps would be simple. Alternatively, the primitive VOIs could be a gross over-segmentation of the data and dozens of VOIs

in the primitive atlas would form each VOI in the individual atlas. A more serious problem would arise if the primitive atlas was an under-segmentation of the data which, since the primitive VOIs cannot be split, would never result in an acceptable atlas. The best we can reasonably expect is a slight over-segmentation of the data; one which does not over-estimate the size of the primitive VOIs but which allows for a minimum of choices in the merging stage. Two methods which are particularly pertinent to segmentation of 2-D and 3-D medical images are described in Section 4.3.

The merging of primitive VOIs into medically significant regions is an important step in the creation of individual atlases. How can knowledge of human anatomy be included in the merging procedure? How can 3-D shapes be compared? What features of the images can be exploited to aid in the merging? A discussion of these questions is given in Section 4.4

Finally, how can an atlas be used to aid in the display and interpretation of scan type images of the head? Once the images have been segmented, several image editing, integration and display features can be imagined. These include: extracting and displaying intra-cranial structures, direct comparison of a structure from one modality with the same structure from another, 3-D visualization using depth cues, and orthographic projections. The subject of 3-D editing and display is treated in Section 4.5.

4.1 The Atlas

There are many published books and digital atlases of the human brain [70] [33]. These show the idealized structure of the brain in a series of slices or are available in digital form for 3-D display. These idealized atlases are used mainly as references when studying or identifying structures and regions in the brain and each important

region is outlined and labelled. The labellings are taken to be correct for the "normal" brain used to illustrate the atlas.

Through experience, study, and practice, neuroradiologists have the same type of information available to them when they view images. That is to say that the idealized anatomy of the brain resides in the knowledge they use when reporting on images. In the same way that one develops a mental "map" of a city through studying printed maps and travelling through the city a radiologist develops a model or "atlas" of the anatomy of the brain.

While reporting on a set of images, a neuroradiologist matches this internal "atlas" to the images. This step of matching the internal "atlas" to the image is complex and learning to do this is a large part of a neuroradiologist's training.

A computer-based system which hopes to segment and label a set of images must perform a similar task. The individual atlas should overlay the real data so that the labelling of the atlas is as close as possible to the correct labelling for the raw data, the labelling a radiologist would assign while reporting on an image.

A distinction can be made between an *anatomical atlas* and a *tissue atlas.* An *anatomical atlas* identifies the regions in the brain based on their anatomical names. The borders between the regions in this type of atlas may not be directly visible to the radiologist but may rely on landmarks or conventions for their definition. A *tissue atlas*, however, demarcates regions based on differences in radiological density. The difference between the two atlases is analogous to the difference between a map showing political boundaries and one showing topographical features. The classification of regions on the topographical map is based solely on visible features whereas boundaries on the political map may or may not follow geographic features.

As discussed in Section 3.4, a digital atlas can be manually deformed to label a set of images. For a perfect match the data would have to be identical to the data used to

draw the atlas or a correspondence would have to be made at every point; the latter is difficult to do unless the image has been previously labelled. An acceptable match may, however, be possible without making a correspondence at every individual voxel.

As shown in Fig. 4.1, this thesis introduces a new methodology for creating an individual atlas. Rather than matching the idealized atlas directly to the raw data, an intermediate data structure is used to perform the matching. The raw data is segmented and stored in a data structure, a data structure which describes each 3-dimensional region found during the segmentation. Each 3-dimensional region is called a *Volume of Interest* (VOI). The data structure is called a *primitive atlas* but at this stage it is unlabelled and the VOIs represent an over-segmentation of the raw data; several regions in the primitive atlas may represent a single medical structure. It is the primitive atlas which is searched by the idealized atlas to find the intra-cranial structures.

The product of the merging process is an *individual atlas* which represents a key to the raw data set. This is a tissue atlas since it is created from the image data. With this atlas, individual structures can be rendered and displayed, and calculations of the size, texture characteristics and voxel statistics can be made. The individual atlas also provides for alternative methods of displaying the data such as highlighting or rendering medically important regions in the raw data.

The atlas data structure is a collection of VOIs. Each VOI represents a region in a volume description of a 3-D image. Table 4.1 is a list of the information which could be contained in a VOI. Each data field is an abstraction of the information contained in the image within the domain of the VOI.

A VOI is the lowest level of abstraction in the atlas. It refers to the data set directly and connects the higher level abstractions contained in the atlas with the data set.

Information	Field
voxel statistics	average voxel value standard deviation number of pixels
morphology	center of mass position shape
topology	number of holes

Table 4.1: List of Data Fields in a VOI

The most difficult feature of a 3-D region to abstract is the morphology. The shape of a 3-D object can be complex and may not lend itself to mathematical descriptions. How then can the shape, size, and position of the VOI be represented? The solution we chose is to use a volume representation of the 3-D region. With this representation, the exact size and shape of the region can be abstracted and recorded.

There are disadvantages to using a volume representation. For example, the surface of the VOI is not specifically modelled and thus 3-D surface renderings require the generation of the VOI surface. The representation does, however, provide for easy access into the original data and for exact shape representation.

The exact method of storing the volume occupancy representation of the shape and position of the VOI is discussed further in Section 4.4.2.

4.2 Pre-processing

As a first step in the procedure, the data must be manipulated into a common format. For images which have been collected with the gantry tilted, it is necessary to correct for the skew introduced by the gantry tilt. The skew can be corrected using one of one of two methods: resampling the data set onto a rectangular grid, or, shifting each 2-D image in the data set a small amount.

For the second method, the amount of shift in the i^{th} slice is given by,

$$\delta y = ib \sin \alpha, \tag{4.1}$$

where α is the gantry tilt angle and b is the amount the gantry table is moved between slices, expressed as the number of pixels. If there are 20 pixels between the top (or bottom) edge of the slice and the closest extremity of the patient's head image, then for a series of 10 images with 4 pixel interslice spacing, the tilt angle can be 30 degrees before the important image information encroaches on the edge of the slice.

Sub-sampling can be used to reduce noise in the data and decrease the subsequent computation time by decreasing the resolution of the image. Averaging each (2×2×2) block in the data set into one voxel in the subsampled image decreases the resolution of the data set by two in each dimension. Once again, this is usually not desirable in the axial direction with CT images since there are normally fewer slices in that direction.

For display of the image, gamma correction can be used to alter the appearance of the data and to provide non-linear scaling. The correction is performed using the following formula:

$$y(i) = 255 \left(\frac{x(i)}{255} \right)^{\frac{1}{\gamma}}, \tag{4.2}$$

where γ is a positive, real, number and $x(i)$ and $y(i)$ are the input levels and output levels respectively.

Other processing is necessary to convert the 12 bit image data to 8 bit data. The 12 bit data is compressed using a non-linear and monotonic transfer function. Similar to the gamma correction, it uses an S-curve to expand certain ranges of densities and compress other ranges. A typical, soft tissue, windowing transfer function is shown in Fig. 4.2.

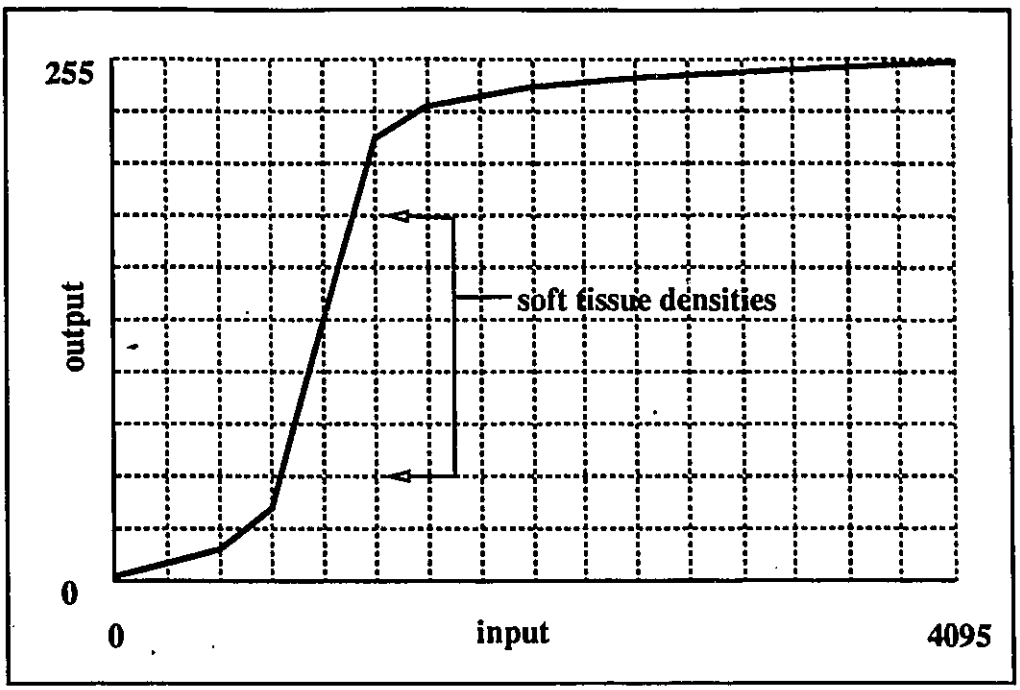

Figure 4.2: A Soft Tissue Window Transfer Function

After the geometric manipulations and corrections, but while the data is still in a raw form, it forms the baseline image. In later stages, what is extracted from the image is a "key" or atlas to this baseline data.

Filtering can be used to reduce noise in the image and improve the performance of the subsequent segmentation. The data sets can also be subsampled to smooth the data and decrease the computation time.

Two common filters which are useful for noise cleaning are the average filter and the median filter. The filters work equally well in 3-D as in 2-D but it is important to remember that the resolution in the 3-D images may not be the same in all three dimensions and a filter should have a variable window. If the data set is isotropic then the support for the filter can be equal in all three directions. If, however, the resolution is poorer in the axial direction, a filter with more support in the other

two directions may be more beneficial. The filter would still allow for noise cleaning but would not further degrade the already poor resolution in the axial direction. An example of the difference between the two filters is shown in Figures 4.3 and 4.4. The loss of resolution in the 3-D filter can be seen by examining the ventricles in the two images.

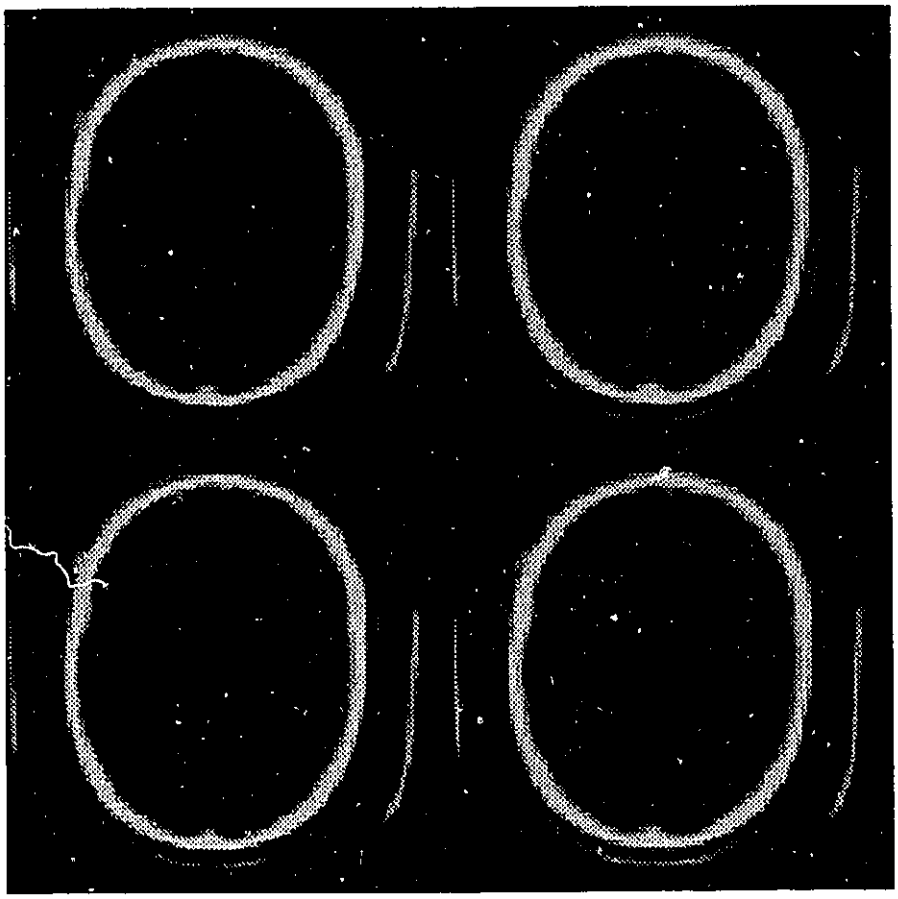

Figure 4.3: Result of Average Filtering a 3-D Image with Equal Support in all Three Spatial Dimensions

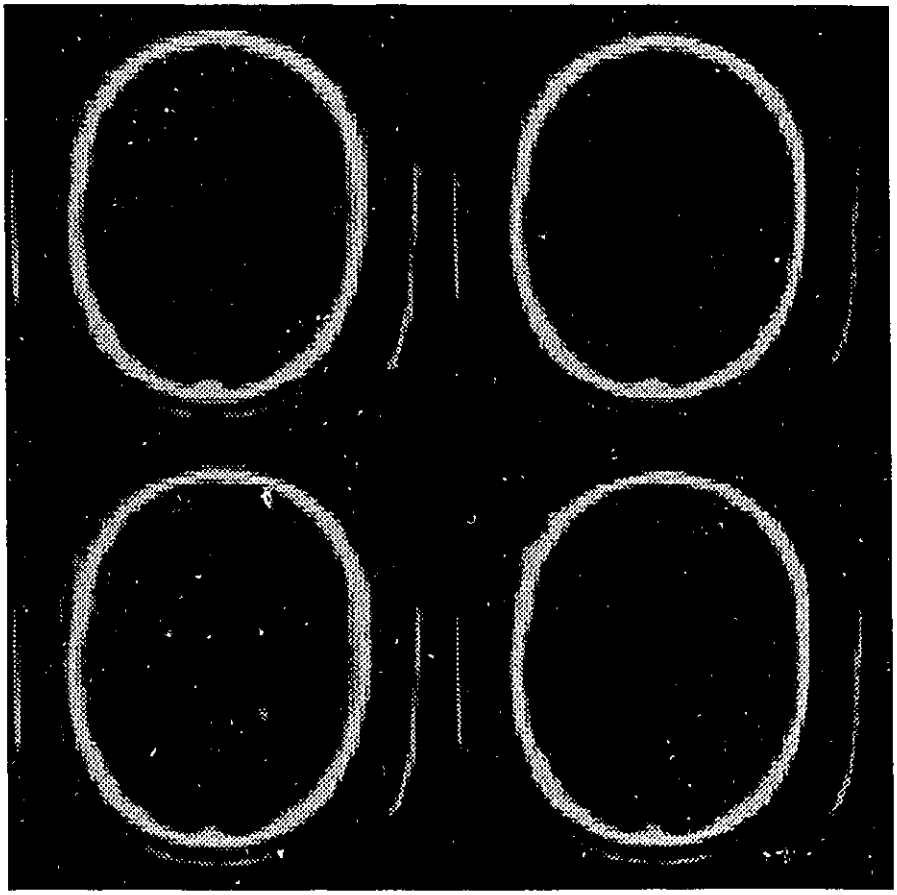

Figure 4.4: Result of Average Filtering a 3-D image with Reduced Support in the Axial Dimension

4.3 Initial Segmentation

The success of the later merging stage depends on the quality of the initial segmentation. Ideally, each medically important structure in the raw image data would be contained in its own VOI in the primitive atlas which results from the segmentation. In this situation the merging stage would be relatively easy. However, this is not the case when real data is used; the segmentation stage is difficult for medical images since the regions are rarely homogeneous and the borders between some tissue types are weak.

The following stage of the process merges VOIs and labels the result as a single VOI. It is not, however, intended to split the original VOIs into smaller sub-regions; they are indivisible. For this reason the original segmentation should be an over-segmentation of the data.

Although many methods are available for 3-D image segmentation, the data suggests some which are more suited to intra-cranial structures in CT and MRI data. Medical images are usually collected under controlled and very similar conditions suggesting that structures in comparable studies using the same imaging modality will be similarly positioned. The imaging equipment has good dynamic range (usually 12 bits) and adequate resolution, therefore the segmentation method must be sensitive to changes in individual pixels. The CT imaging equipment is also routinely calibrated to ensure a representative pixel value according to the tissue type, suggesting that *a priori* knowledge of the density of intra-cranial tissues could aid in the segmentation. There is no equivalent absolute calibration used with MR images so using grey scale levels to help in their segmentation is not as useful. MR images do, however, tend to have a higher contrast between tissue types than CT images, suggesting that a segmentation method which relies on differences would be more appropriate.

Two different methods of segmentation are discussed in this section: a gradient-based method (Watersheds) and a region growing method. The two methods are somewhat complimentary in their function; the gradient method finds the boundaries between regions and the region grower collects pixels together to form a region.

The other limiting factor in the choice of a segmentation scheme is its applicability to 3-D data. Most 2-D segmentation methods can be adapted directly to 3-D but slice-type medical images are often not isotropic and the distance between slices can be several times the pixel width. If a gradient method is employed to segment the image in 3-D then it must be able to account for the different scalings in each direction.

At the end of this final stage of the segmentation there is a set of 3-dimensional VOIs. Each VOI has a 3-D description of its shape as well as descriptions for the pixel statistics of that region such as average pixel value, number of pixels, and standard deviation. All of these VOIs are organized into an atlas data structure but this atlas represents an over-segmentation of the data and is called a *primitive atlas.*

4.3.1 Watersheds

The algorithm for performing watershed segmentations given by Vincent [78] and described in Section 3.1 can be modified to better suit 3-D medical data sets. The concept of watersheds can be extended to 3-D images by extending the definition for the neighbourhood of a voxel to 3-D. Although the formal definition does not change, the watershed analogy does break down. The results are similar except that the watersheds are now surfaces enveloping a volume. The simplest 3-D neighbourhood is 6-connected where only the 6 voxels adjacent to the central voxel are included. At a higher level of complexity and computational cost is the 26-connected neighbourhood containing all the voxels within a $(3 \times 3 \times 3)$ block enclosing the central voxel. An

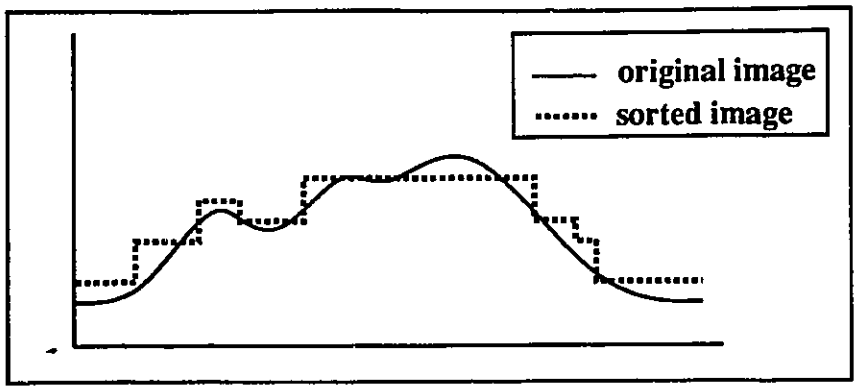

Figure 4.5: Pixel Binning of Images

intermediate level is the 18-connected neighbourhood which eliminates the 8 corner voxels from the 26-connected neighbourhood.

The size and shape of the neighbourhood necessary to correctly segment the image is dependent on the image features to be extracted. If thin diagonal lines or surfaces (1-2 voxel widths) are to be segmented then a suitable neighbourhood which can track these features should be used. The 6-connected neighbourhood is well suited to 3-D medical images since there are few narrow regions. There is an added benefit since it is also the fastest to compute.

The WS algorithm is implemented by sorting the gradient image into bins; each bin containing the indices of all the pixels in the image which have that particular grey level. Normally, a 256 level image would be sorted into 256 bins and the first bin would contain the indices of all the pixels which have the level "0". The sensitivity of the WS algorithm can be adjusted by sorting the gradient image into bins larger than one pixel level so that a bin contains the indices to pixels within a range of values. For example, the first bin could contain the indices of all the pixels with values between "0" and "10". An example of a 1-D signal before and after binning is shown in Fig. 4.5. Since regions with nearly constant pixel values in the image will have values

in the gradient image close to zero, the size of the first bin normally determines the sensitivity of the algorithm to the gradient values. It is difficult, however, to adjust the bin size to include weak borders without destroying some important image features. Referring to Fig. 4.6, setting the first sorting bin to level **1** eliminates the signal variation in region **a** of the original image but also eliminates the edge at **b**. If

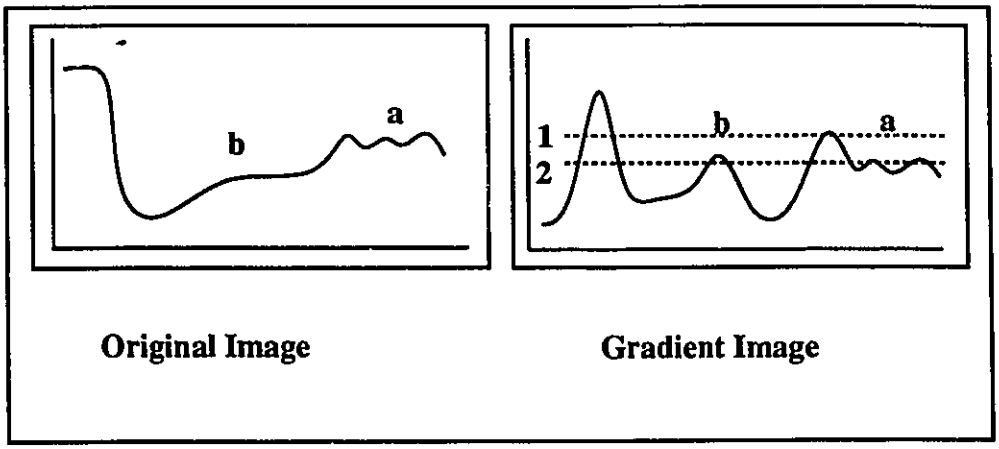

Figure 4.6: Adjusting the sensitivity of the WS algorithm

the first bin is reduced to level **2** to include the edge at **b** then the noise in the image in region **a** results in a gross oversegmentation of the image in region **a**. Filtering can help the problem but excessive filtering may destroy some image features.

To alleviate the problems with sorting the gradient image, an alternative method was used. The original image was sorted into bins prior to the gradient operation to make use of the *a priori* knowledge of the image. The resulting image has a compressed dynamic range as is shown in Fig. 4.7. The WS was then performed with a bin size of one pixel level. The incorporation of the grey level information into the gradient image allowed for much better control of the segmentation method. Fig. 4.8, shows a gross over-segmentation of the image using the WS method. The white lines in the image are the watershed lines. The gradient image is sorted in an

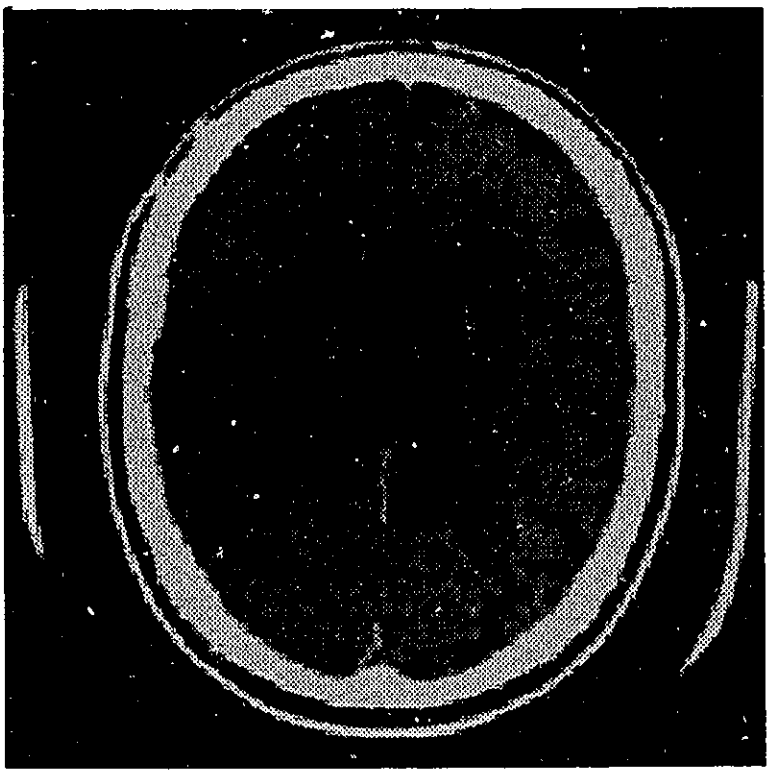

Figure 4.7: Example of an Image after the Binning Process

attempt to reduce the sensitivity of the segmentation to small changes in grey level while retaining the grey matter/white matter differentiation. The first sorting bin has values that range from 0 to 5.

Figure 4.9 shows the segmentation of the same image using the WS method but with the image sorted before the gradient is performed. This results in improved performance and the easily segmented areas (ventricles and bone) are well defined.

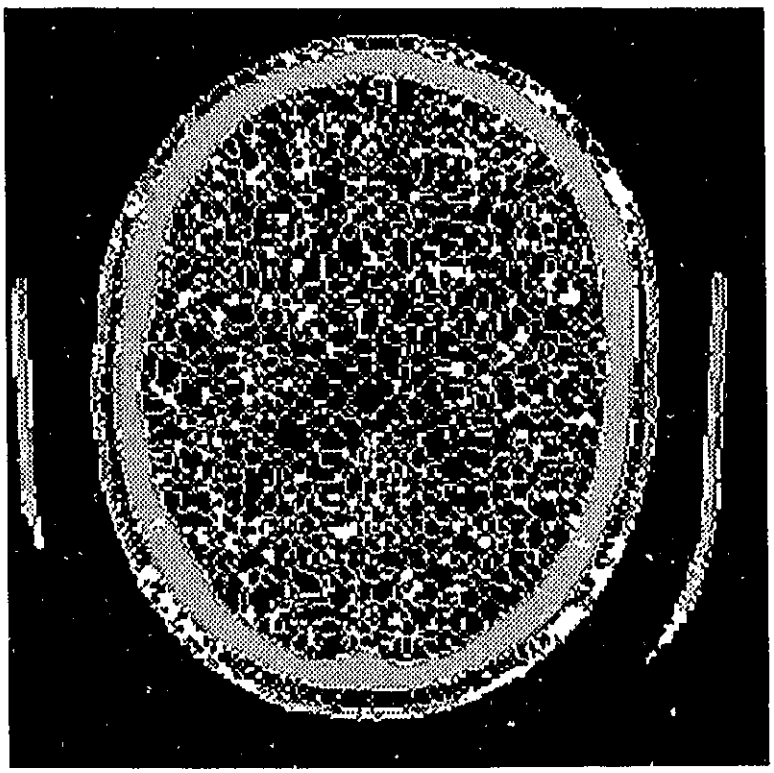

Figure 4.8: Oversegmentation of Image Using WS With a First Bin Size of 5

4.3.2 Region Growing

The region-growing method employed in the study is an adaptation of a watershed algorithm described in [78] and in Section 4.3.1.

The basis for the region grower is sorting the image pixels into bins which have

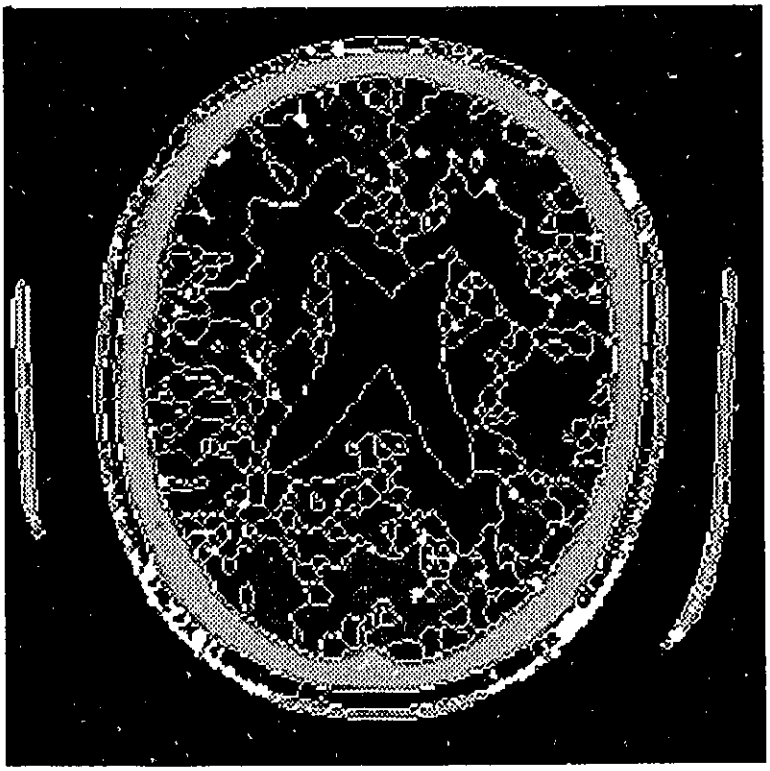

Figure 4.9: WS Segmentation With Presorting of the Image

the same value for the purposes of region growing. As with the WS method, a sorting bin can contain indices to pixels within a range of values.

The sorting process groups pixels according to their values. The next stage of the process groups pixels which are members of the same bin together based on connectivity. The algorithm is adapted from the WS algorithm in [78] and is shown below. The *fifo* is a first-in first-out queue.

```
MASK := 'belongs to current bin'
for all levels do: {
   for all pixels (p|I(p)=h) {
      J(p) := MASK}
   for all pixels (p|I(p)=h) {
      if (J(p) = MASK) {
         increment current label
         fifo_add(p)
         J(p) := current_label
         while fifo_not_empty {
            pp := fifo_first
            for all neighbours of pp {
               if J(ppp = MASK) {
                  fifo_add(ppp)
                  J(ppp) := current_label}}}}}.
```

The region grower described in the above algorithm has been implemented to work on 3-D images as well as 2-D images and the extension to 3-D is an extension of the definition of neighbourhood to three dimensions. In this case, the neighbourhood was six connected (in 3-D) but other neighbourhoods based on 18 or 26 connectivity could

also be used. The idea was adapted from the WS algorithm since it led to an elegant and efficient algorithm for region growing.

4.3.3 The Primitive Atlas

The primitive atlas contains the results of the initial segmentation. It contains the name of the associated data set and a collection of the VOIs found during segmentation. The VOIs are labelled according to the order in which they were found. Each VOI contains some elementary pixel statistics such as average pixel value and number of pixels as well as a reference to an associated array where a volume occupancy representation of the position and shape of the VOI are kept.

4.4 Merging

The final merging process builds an individual atlas using the regions found in the initial segmentation and contained in a primitive atlas. It uses a top-down approach to the merging in which specific structures are searched for in the primitive atlas.

The *search dictionary* guides the search. It determines the order in which the structures are searched and the parameters used to generate candidate VOIs from the primitive atlas. The order can be important to the ease with which the structures are found since the locations of some can help in locating others.

Two forms of knowledge are used to do the merging; general *(a priori)* knowledge of the imaging procedure and patient specific anatomical information. *A priori* knowledge about each type of intra-cranial tissue is contained in the search dictionary. Patient specific knowledge is incorporated by using a previously created individual atlas to make decisions on the correct segmentation. These methods of representing knowledge are fundamental to the merging procedure. The idealized atlas is split into

two parts, the pixel density characteristics and the shape and position characteristics.

4.4.1 The Search Dictionary

The search dictionary is a special type of atlas. It does not contain information from an individual data set but rather the averaged results from several data sets. It contains only relative position information and does not include a description of the shape of the VOI. It does, however, contain information about the pixel statistics for each region and for each imaging modality.

The search dictionary contains information about all of the cranial and intracranial structures to be searched for in the primitive atlas. Not all of the regions listed in the search dictionary have to be found; indeed they will not all occur in every set of images.

A graphical depiction of a search dictionary is shown in Fig. 4.10. A search dictionary has an analogous structure to an atlas as outlined in Section 4.1; it is arranged as a collection of search VOIs, each referring to a specific structure in the head.

Each search VOI describes attributes associated with each imaging process, the average and range of the grey scale values for a specific tissue type and imaging modality. Only one example of the MRI values is given here but several different sets are used to account for different stimulation and collection procedures. As well, a measure of the positional variation, expressed as a percentage overlap, is included for each VOI.

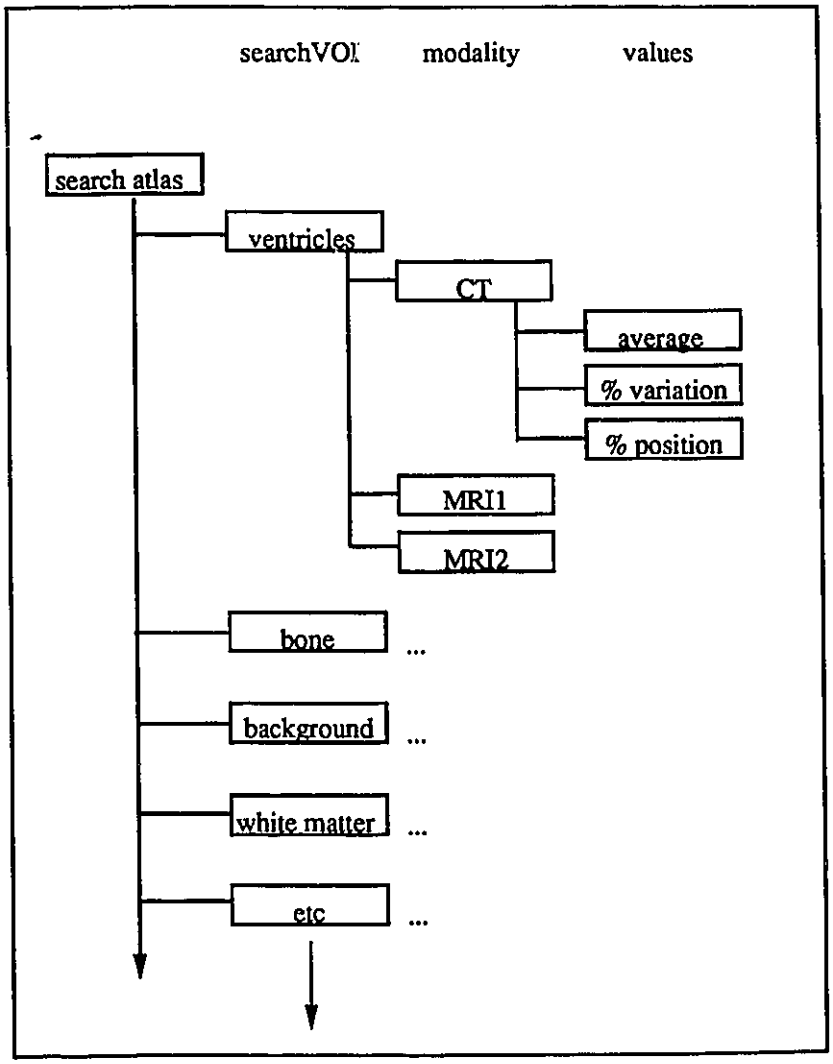

Figure 4.10: The Search Dictionary

4.4.2 The Individual Atlas

The individual atlas, described in Section 4.1, is specific to a given data set. It contains a description of the regions found, their attributes and their shapes.

An individual atlas is the final product of the merging and labelling stage and also the "gold standard" used to decide on a merging of primitive VOIs. It is the final product since it contains representations of all the structures found in the data set. In its role as a "gold standard", an individual atlas is used to represent the positional information contained in an idealized atlas. The positions and shapes of the VOIs contained in a previously created individual atlas are used as a template to decide which of the candidate VOIs from the primitive atlas should be included in the present VOI.

The individual atlas is a data structure that holds the name of the associated data set and a collection of VOIs; each VOI has a name and is associated with a particular medical structure depicted in the data set. Each VOI also contains the pixel statistics of the region and a volume occupancy representation of the shape and position of the structure. Figure 4.11 shows the organization of an individual atlas.

4.4.3 Building the Atlas

The individual atlas is built up by merging VOIs in the primitive atlas into medically significant structures. *A priori* knowledge, contained in the search dictionary, guides the search by listing the structures to be searched for and providing an average pixel intensity and measure of positional variation for each structure. Since a specific structure can appear with different intensities in each imaging modality, the approximate value for each structure and for each modality is available. The process is shown graphically in Fig. 4.12.

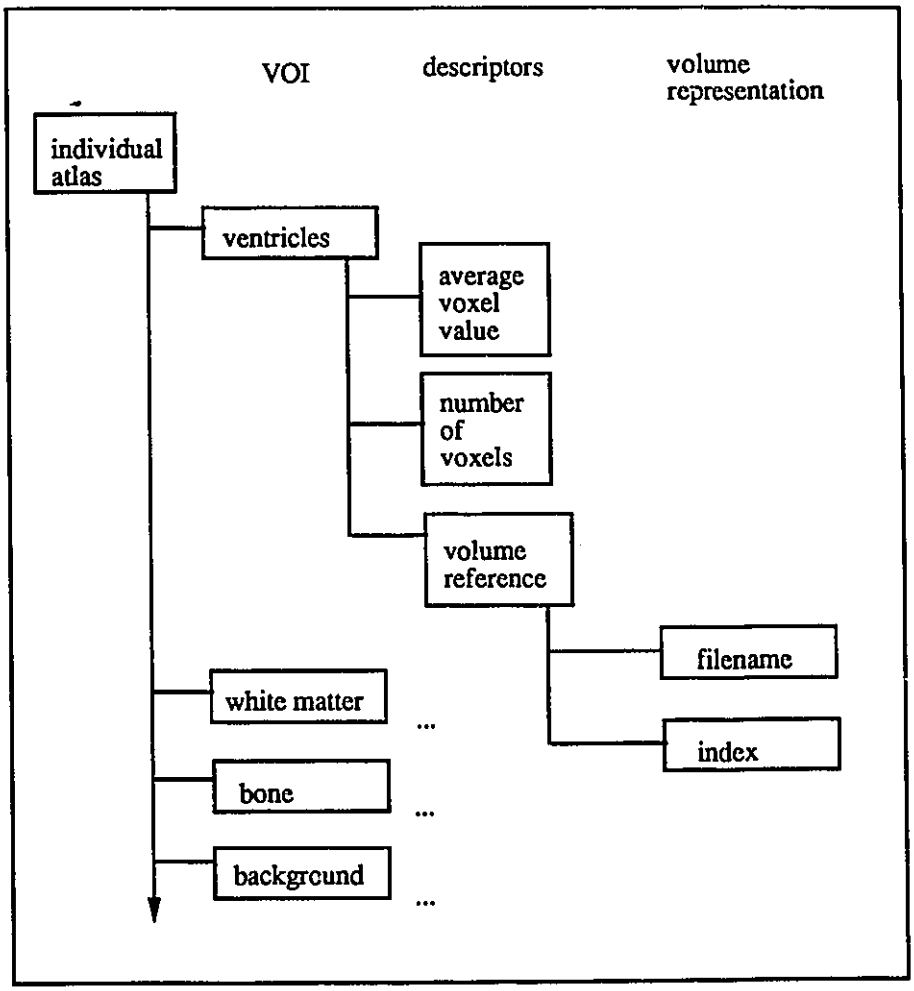

Figure 4.11: The Individual Atlas

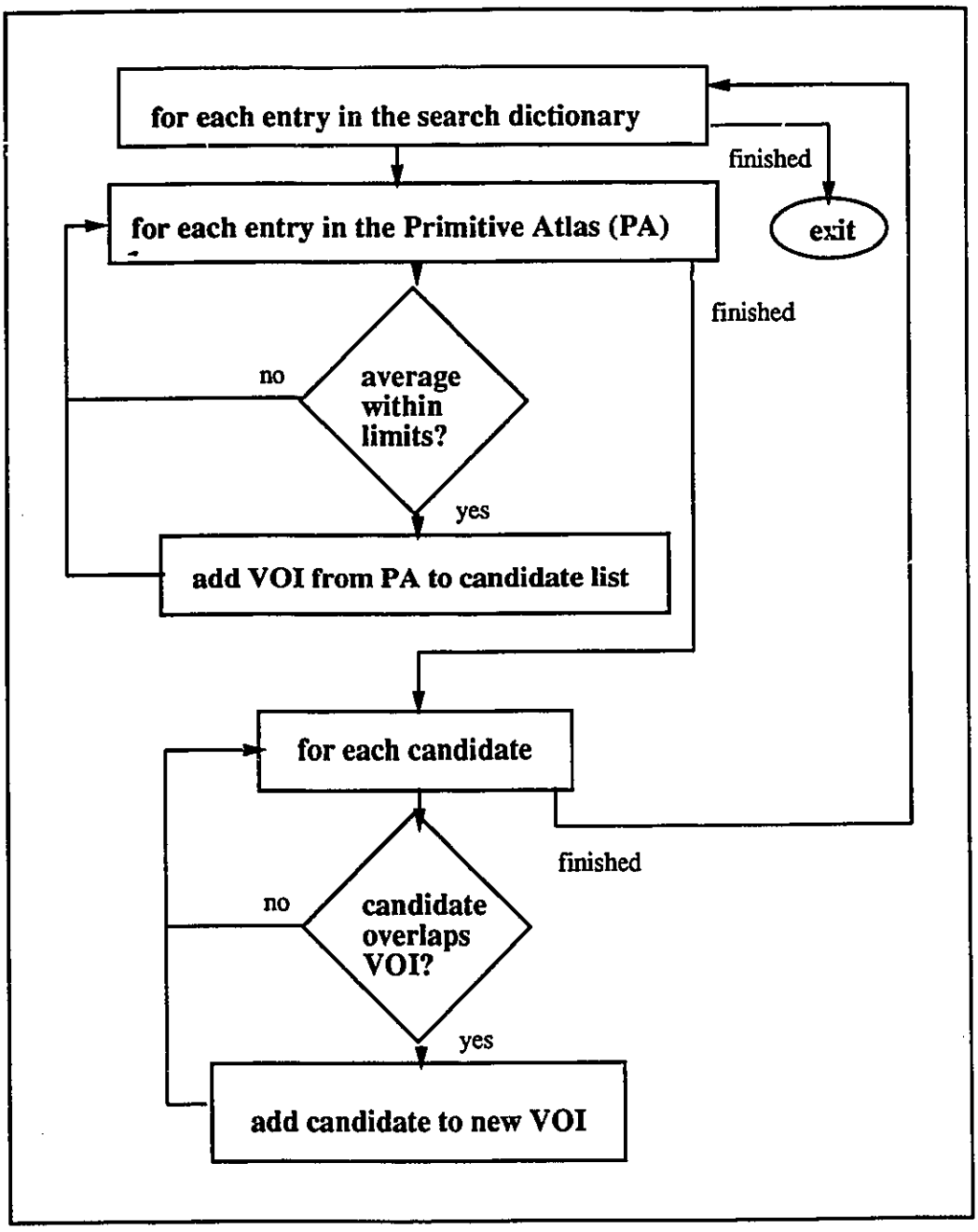

Figure 4.12: A Flow Chart Depicting the Merging Procedure

The structures are searched for individually in the order determined by the search dictionary. For each structure, a set of candidate VOIs is generated using the search dictionary estimates of pixel density for that structure and imaging modality. The order of the search is important since this candidate list is only chosen from the remaining, unmatched, VOIs in the primitive atlas. It is from this set of candidate VOIs that the final set of VOIs which form the structure is chosen.

Just how can the correct subset of VOIs be chosen from the candidates? Since the candidate VOIs have been selected based on voxel intensity, the next decision need only be based on position, shape and size. Two solutions to this problem were found. The first leaves the decision to the user. By browsing through the primitive atlas the user can select the subset of candidate VOIs which compose the structure. The second solution uses a single, previously created, individual atlas.

An individual atlas contains all the information necessary to decide which subset of the candidate VOIs should be included provided there is sufficient similarity between the two data sets. If an individual atlas to a data set of the same patient is available, then it could justifiably be used as a "gold standard". We must remember that it is only the shape of the structure which must be tested at this point.

What if a previously created individual atlas of the same patient is not available or if the anatomy of the patient has changed since the last study? In order to build in some margin for mismatch or differences between the two data sets, a field in the search dictionary is used to specify the necessary overlap between the candidate VOI and the VOI in the individual atlas used for comparison. The measure is expressed as the number of overlapping voxels as a percentage of the total number of voxels in the candidate VOI. This method has the added advantage of being able to match candidates to a structure in the absence of any overlap by setting the overlap criterion to "0" for that structure in the search dictionary. This is useful for matching

anomalous structures, which can appear anywhere in the data set.

Differences in scaling, translation and rotation between the two volumes could cause errors in the matching. Because of this, the choice of the previously segmented atlas is quite important to the success of the merging.

4.5 Image Editing, Integration and Display

How does the creation of an individual atlas help the radiologist visualize the data? The main benefits are a result of the segmention and labelling of the medically significant structures in the data. Once the structures have been identified in the data, a new set of tools become available to the radiologist.

The most obvious benefit of having an individual atlas to a data set is the direct display of the structures' shape in three dimensions. This is possible since the surface of the structure can be generated from the volume representation held in the VOI data structure.

Another benefit is the ability to visually compare the corresponding structures from two different data sets by simultaneously displaying them on the screen. This feature is useful when comparing data sets taken of the same patient at different times to spot trends in the shape, size or density of the structure. The comparison can also be made using data sets from different patients or different imaging modalities.

The use of an individual atlas can be extended to include image registration. If the same structure is well-defined in two different data sets, this structure can be used to define the transformation between the data sets and to register the data sets. Once the data sets are in registration, direct, quantitative comparisons of the structures are possible.

To take the image registration idea further, integration of multimodal images can

be performed. If the images are taken of the same patient, at the same time but with different imaging modalities, and they can be put in registration, then simple editing tools can be used to integrate the data.

4.5.1 Image Editing

Since the individual atlas references the raw image data, it can be used to perform image editing functions on data sets. For example, functions such as *cut* and *paste* can be used to move structures from one data set to another. Fig. 4.13 illustrates this idea. The VOI in the destination atlas is used to access the data set associated

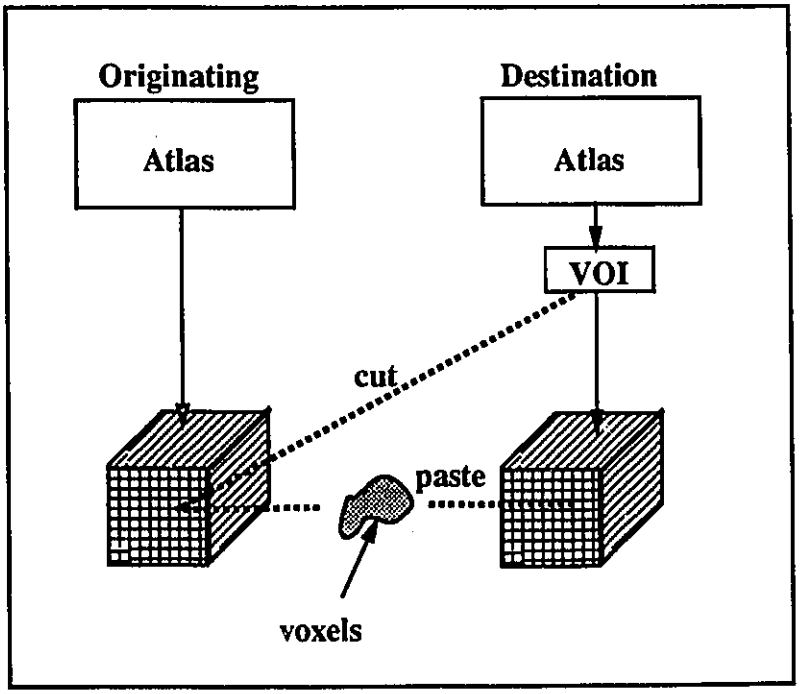

Figure 4.13: Image Editing: Cutting and Pasting a VOI

with the originating atlas and the voxels are copied over to the destination data set. New voxel statistics are calculated for the VOI and the destination atlas is updated. The two data sets need to be well registered for the process to work properly.

4.5.2 Image Integration

Several types of image integration methods can use an atlas. Some methods require that the data sets are in good registration.

One method of image integration is to superimpose the outline of VOI boundaries onto another data set. If the two data sets are of the same patient, and are in registration, the atlas will outline the structure in the new data set even if the boundaries between the regions are not well visualized in the second data set.

Another method of multimodality image integration is to replace a structure in one data set with the data in the same structure from another data set. To do this, the two data sets must be in excellent registration so the voxels can be transferred directly from one data set to another. Image editing functions can be used to do the transfer. This feature is useful for inserting structures from CT or MRI data sets such as the skull or the ventricles into PET data to provide landmarks for the radiologist.

4.5.3 Image Display

The most compelling reason for creating individual atlases to 3-D data sets is for 3-D display. An individual atlas allows the data to be displayed using a wide range of 3-D display technologies and methods by providing the surfaces of the medical structures in the data set. Without an individual atlas, 3-D renderings are generally limited to surfaces generated by a simple thresholding. Depending on the threshold window used, it is usually not possible to select just the desired structure and to exclude noise or structures with a similar grey level. The surfaces can be generated by the user using interactive selection. This process can be quite time consuming unless the user has extensive computer assistance.

One approach to 3-D display is to use a collage of 2-D slices as used in most

radiology departments and shown in Fig. 4.14. Successive slices are displayed in a 2-D array thus allowing for complete viewing of all the data on a single screen. The individual atlas can be used to outline the VOIs in the data. The example in Fig. 4.15 shows the ventricular regions outlined in white in each slice. In some cases, such as small highly convoluted regions, a complete depiction of the volume is easier to interpret and the pixels contained in the VOI are highlighted as shown in Fig. 4.16.

One of the advantages of having an individual atlas as a "key" to the data set is that information about the data can be easily accessed. For example, to highlight a certain VOI in the data set one can select its name from a list of all the VOI names in the data set. Likewise, using the cursor to select a voxel in the image can select the VOI which contains that voxel.

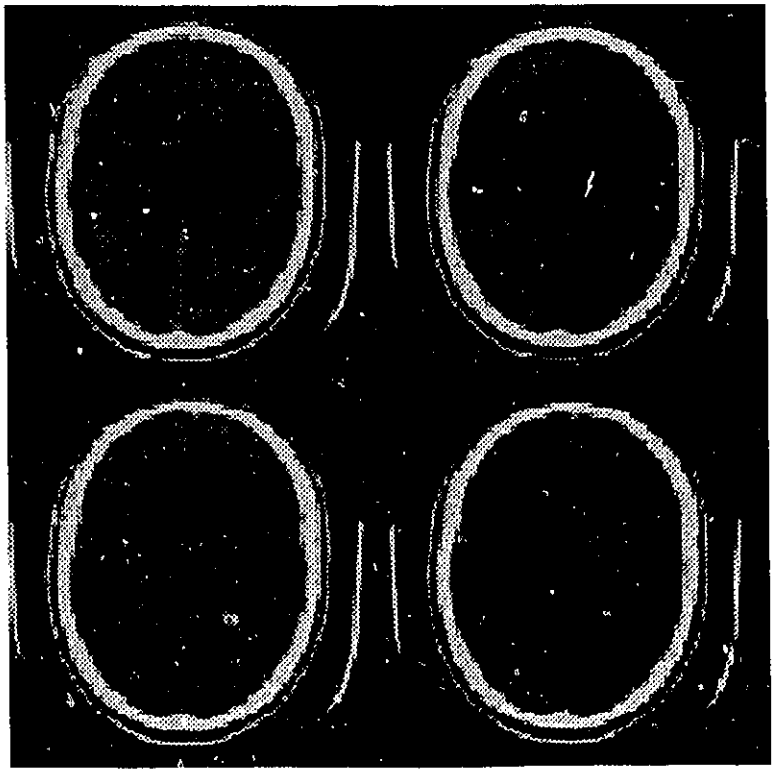

Figure 4.14: Display of a 3-D Data Set in a Collage

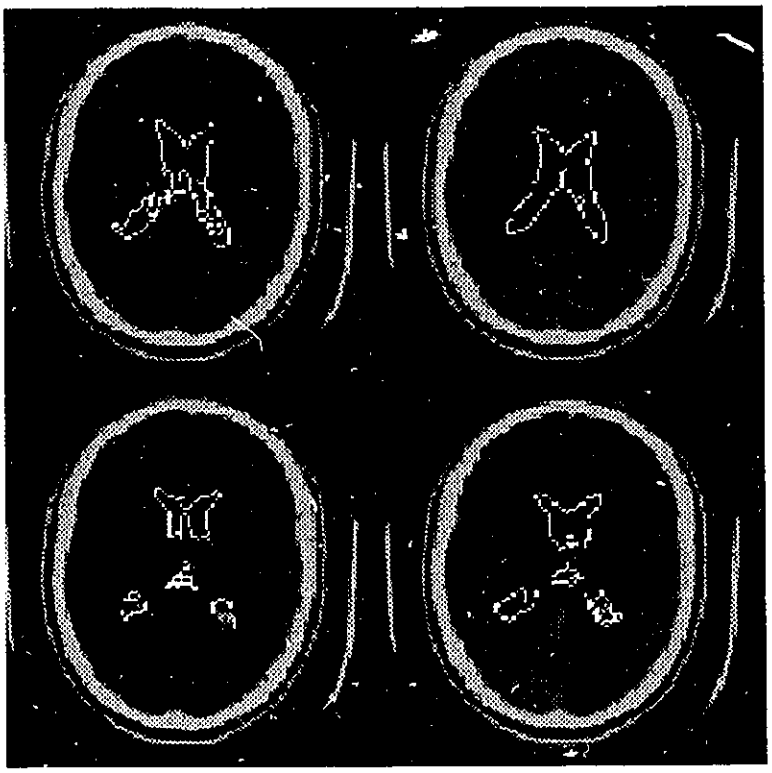

Figure 4.15: Outline Display of a VOI

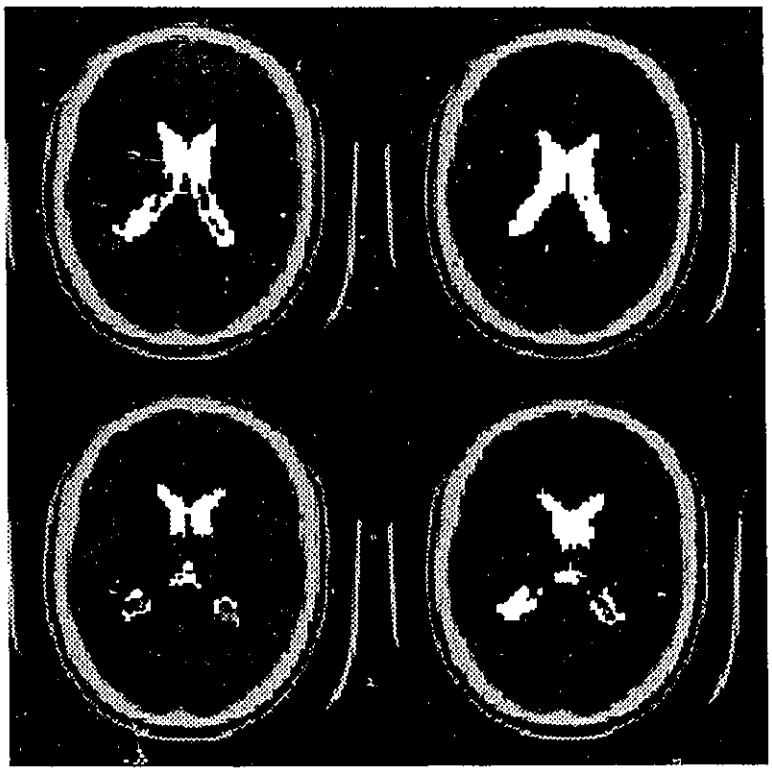

Figure 4.16: Direct Display of the Volume Representation

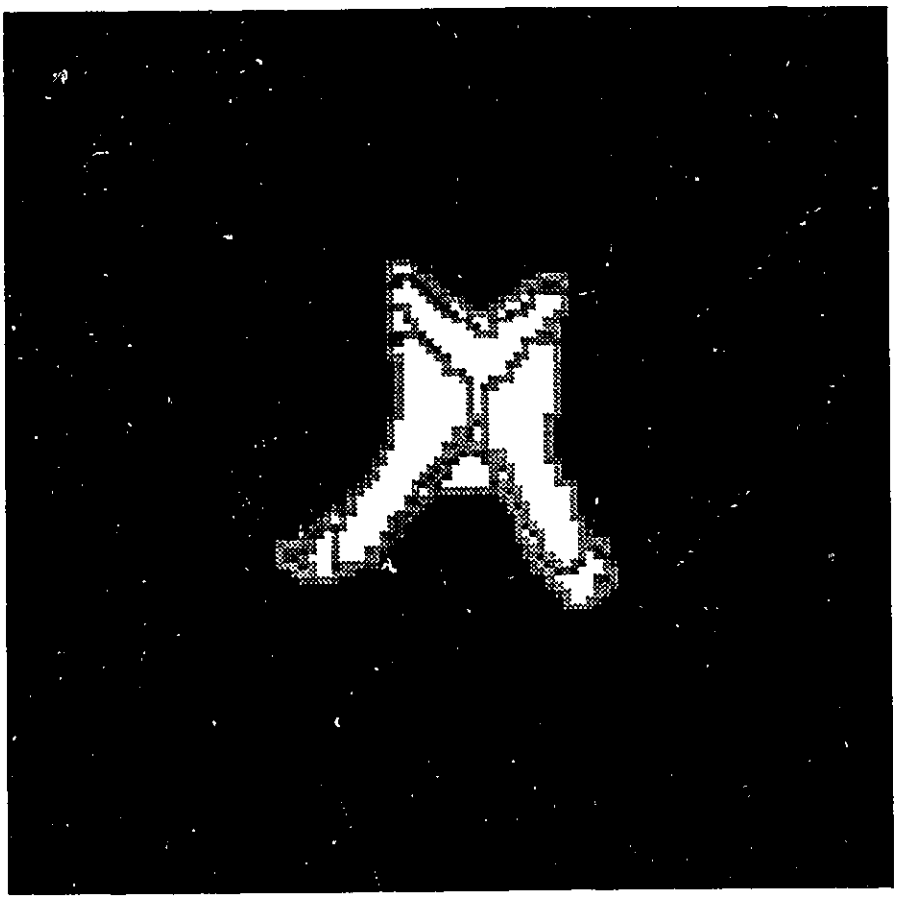

Figure 4.17: A Rendering of the Ventricles

Another method of 3-D display is surface rendering. This method yields dramatic results when a series of shadings are done at small angular increments and are displayed sequentially in a cine-loop. Figure 4.17 shows an unsmoothed rendering of the ventricles in the data set.

As mentioned earlier, the surface of the structure is implicitly contained in the volume representation of the VOI. Thus, an individual atlas contains the information necessary to render any of the VOIs contained in the atlas. Further processing to threshold the image or to interactively segment the image is not necessary. Additionally, each VOI is named explicitly so the data can be accessed by name.

Most other 3-D representations such as stereo viewing and animated projections rely on a surface representation of the structure. In these cases, an individual atlas can provide the surface representations and can provide access to the data by name.

Chapter 5

An Object Oriented Implementation

This chapter is a description of the software implementation of the ideas presented in Chapter 4. The implementation was written in *Smalltalk/V Windows*[1], a software programming system produced by *Digitalk* and which runs as a *Microsoft Windows*[2] application.

To assist in describing the implementation, it is helpful to describe the Smalltalk programming environment. Section 5.1 is an overview of the main ideas in Smalltalk and the naming conventions. To implement the concepts in Chapter 4 such as atlases, 2-D and 3-D images, and search dictionaries, extensions were made to the Smalltalk system. The important additions are described in Section 5.2. The application consists of 3 user interfaces, *MedView*, *AtlasBrowser* and *AtlasMatcher* which are described in Sections 5.3, 5.4, and 5.5 respectively.

[1]Smalltalk/V is a registered trademark of Digitalk, Incorporated
[2]Microsoft Windows is a registered trademark of Microsoft Corporation

5.1 An Object Oriented Approach

Although this section is not intended as a tutorial on object-oriented programming, it is necessary to explain some of the concepts used in Smalltalk/V before describing the application. Three concepts that will aid in the discussion are: a *class*, an *object* (an *instance* of a class), and a *method* (which operates on an object).

The entire Smalltalk system is organized into classes. Each class defines a specific type (or class) of object and the class defines the behavior of the objects when they are used. For example, the class *Integer* defines the behavior of integers used in a program. Each object in the program is an instance of a class. In object-oriented programming, objects are the building blocks in creating an application. Objects are created, used, and destroyed as the program executes.

An object holds the data values assigned to it and the behavior of an object is defined by the *methods* contained in the object's class definition. The only way to access or change the data is by using the *methods* contained in the definition of the class. The methods are said to *encapsulate* the data.

So how do the objects and the methods fit together and how are programs written in Smalltalk? Objects interact in a Smalltalk program by sending *messages* to each other. If a *sender* object is executing one of its methods and it requires information from another object, it sends a message to the other object. The other object is called the *receiver*. The message that is sent is a call to one of the methods understood by the receiver object. The messaging cycle is shown in Fig. 5.1. The result of the method, an object, is returned to the sending object.

There is a strict hierarchy amongst the classes in Smalltalk. Each class inherits all of the attributes and methods of all the classes above it in the hierarchy. *Inheritance* is one of the features of object-oriented programming languages that allows for re-use

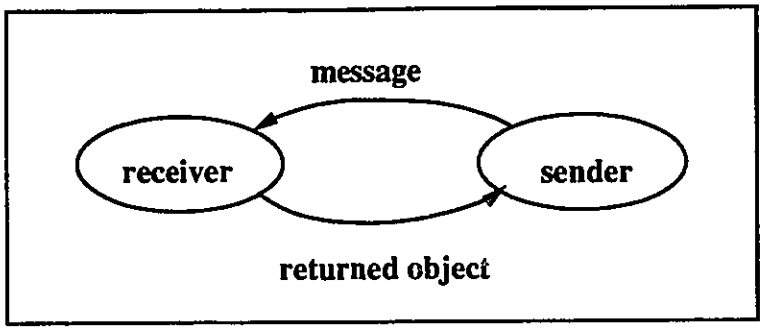

Figure 5.1: Messaging between objects

of computer code. At the top of the hierarchy is the class *Object*; therefore, all of the classes in the system inherit the methods in the Object class. The methods defined in the Object class are the most general and are redefined in subclasses as necessary. In other words, the subclass inherits all the methods and variables of its *super-classes*, all the classes above it in the hierarchy.

Smalltalk provides a convenient method of creating Graphical User Interfaces (GUIs) under the *Windows* operating system. The basic functionality of a *Windows* GUI is included in the Smalltalk class, *ViewManager*. The programmer creates a new subclass of *ViewManager* for each GUI and includes in that class the functionality specific to the application and the rest of the behavior is inherited from the super-classes.

Smalltalk is a general purpose programming language and, like other general purpose languages, is not specifically designed for image processing applications. There are three features of *Smalltalk/V*, however, which make it well suited for this application. First, the ease of developing interactive user interfaces including the display of bitmaps and the use of dialog boxes for user input allows for quick prototyping. Second, the *Dictionary* class in Smalltalk, which forms a list of key/value associations, fits well with the ideas of an atlas and of a VOI. Third, code inheritance allows for the

extension of the algorithms from 2-D to 3-D by simply redefining the neighbourhood.

A brief explanation of the naming and labelling practises followed by *Smalltalk/V* is necessary at this point. Class names start with a capital letter and, if composed of multiple sub-words, each sub-word also starts with a capital. As an example, *ByteArray* is the class name for an array of byte-sized numbers. Objects are usually designated by the class name with the appropriate indefinite article as a prefix. For example, *anAtlas* is an instance of the class *Atlas*. Methods are the functions associated with each class and are designated starting with a small letter but each subsequent sub-word starts with a capital. For example, *drawToScreen* might be a method in the Bitmap class.

5.2 Classes

To develop the M^3I^2E application using Smalltalk/V, numerous new classes were created, these are shown in Fig. 5.2.

A sub-class of *Object*, called *Image*, is used to perform the image processing functions including filtering, resampling, and low-level segmentation. While the Image class provides the 2-D image processing functions, a sub-class of Image called *Bitmap3D* extends the functionality to include 3-D volume images. Instance methods in the Bitmap3D class perform the same type of image processing functions for 3-D images.

The atlas data structures were implemented as subclasses of the *Dictionary* class as this class already provides most of the functionality required for holding and referencing the data.

The three subclasses of *ViewManager: MedView*, *MedRegion*, and *AtlasMatcher*, were created to control the GUIs for the applications and are covered later in this

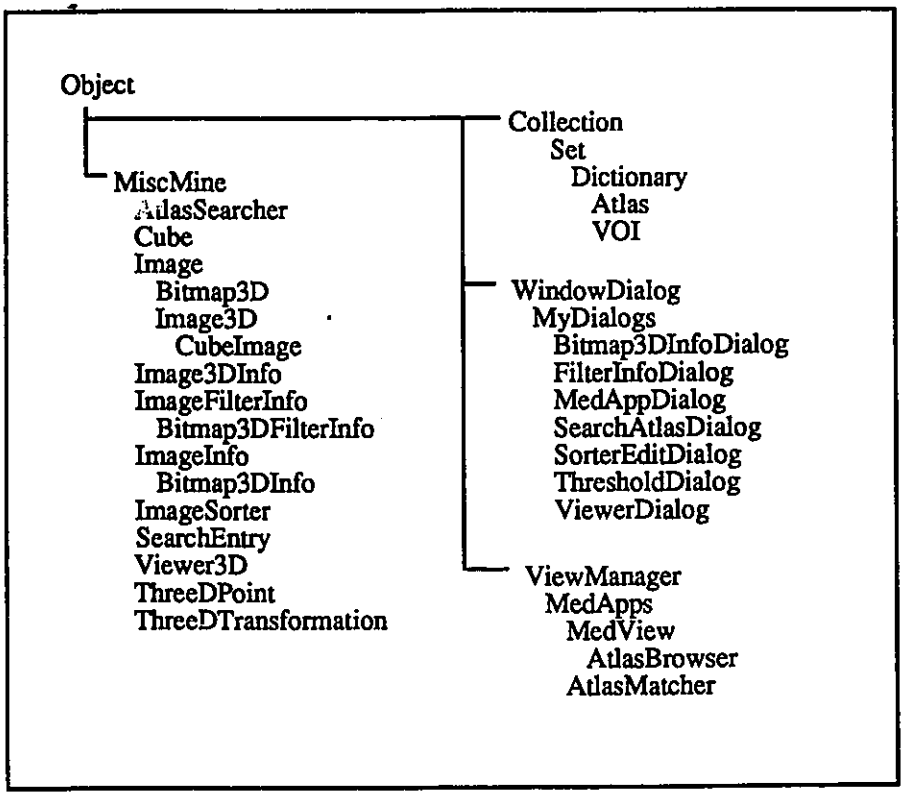

Figure 5.2: A Partial Smalltalk Class Hierarchy Showing the Important Classes Added to the System

chapter.

5.2.1 Image Class

This class is a subclass of *Object* and handles all of the image processing and manipulation functions. Each instance of the *Image* class contains the data for a specific image in *aByteArray* and the dimensions of the image in *anImageInfo*. An instance of an Image can be initialized in two ways: from a file or from *anImageInfo*. An image initialized from a file transfers the data from the file into aByteArray and anImageInfo. However, if the image is created from anImageInfo, each element in the byte array is initialized to zero before being updated with processing results.

The Image class provides numerous instance methods for performing image processing functions such as: filtering, subsampling, and segmentation. The methods operate on the data contained in the byte array.

The two segmentation algorithms, the methods of watersheds and region growing, are implemented using a mathematical morphology approach. The neighbourhood of each pixel is determined based on the structuring element defined.

Two objects are used to hold the interim segmentation results; *aLabelTable* and *anOutputArray*. The output array holds the results of the labelling. It has the same number of elements as the byte array which holds the pixel values but can hold numbers larger than a single byte. The values for each pixel position in the output array refer to the index of the associated label table. As it is being formed the label table uses three arrays to hold the pixel statistics of each region. One array holds the average pixel level for the region, another holds the number of pixels in the region, and the third keeps track of region mergings. All three arrays are accessed by the number assigned to a region in the output array. For example, if region A in the output

array is labelled with "25", the pixel statistics of that region would be contained in the label table at the index "25" and the merging array would have a value "25" at that index if the region had not yet been merged with another region. If region **B**, with a label of "32", was to be merged with region **A**, then rather than changing the labellings in the output array and changing the label table only the index in the merging array would be changed. In every case when two regions are merged, the value of the larger label is made to point to the smaller label. Therefore, the entry in the merging array at index "32" would be set to "25". After the entire image has been processed, the label table is rationalized, the pixel statistics are updated, and the arrays are compressed to eliminate surplus labels. Lastly, the output array is updated.

Images can be read in from two file formats, the internal file format and the format output from a *GE Highlight Advantage* database. For information on the *Highlight Advantage* database the reader is referred to the GE documentation [69]. The internal file format is shown in Fig. 5.3.

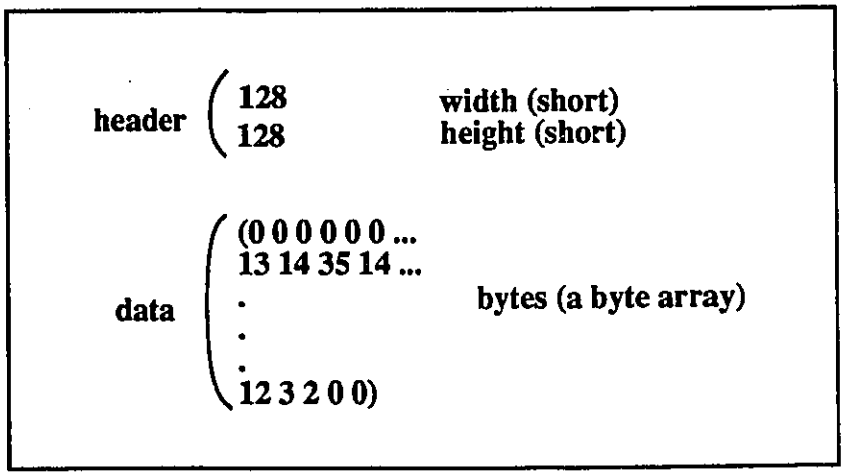

Figure 5.3: The Internal Image File Format

5.2.2 Bitmap3D Class

This class is a subclass of *Image* and adds 3-dimensional functionality to the *Image* class. The internal format of the Bitmap3D file is shown in Fig. 5.4. As with the image class, aBitmap3D can be initialized either from a file or from anImageInfo.

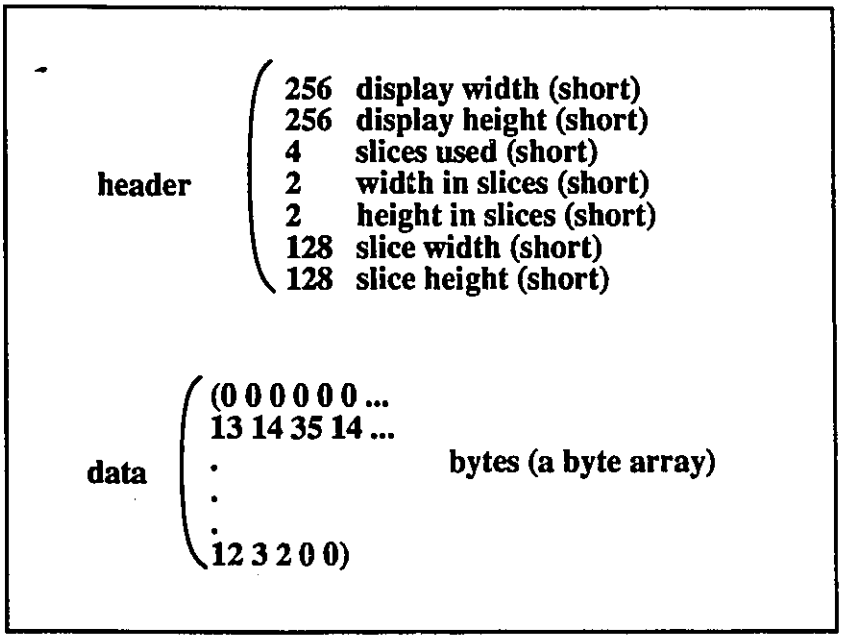

Figure 5.4: The Internal Bitmap3D File Format

The 3-D image segmentation methods are inherited from the image class since the processing is identical. The *(getNeighbourhood: aPixel)* method is re-implemented by the Bitmap3D class since a 3-D neighbourhood is needed.

The methods used for filtering aBitmap3D were re-implemented in the Bitmap3D class rather than inherited from the Image class. This was done as an optimization to improve the run-time performance.

5.2.3 Image3D Class

This class is also a subclass of Image and performs 3-D rendering functions such as building and storing a surface description of a 3-D volume. Along with another class, *Viewer3D*, it renders and displays shaded surfaces generated from the volume descriptions of the VOIs. The *marching cubes* algorithm is used to generate the surfaces [40].

5.2.4 Atlas Class

This class holds both the primitive and individual atlases described in Chapter 4. It is implemented as a subclass of *Dictionary* to provide keyed access to the VOIs in the atlas. Methods are provided in the Dictionary class to add and remove key/value pairs, to step through the keys and to step through the values. Each key or value can be any Smalltalk object; in the case of an atlas, each key is the name of a VOI and each value is a VOI. As new VOIs are found they are named and the key/value pair added to an instance of *Atlas*.

The functionality of the Dictionary class was extended by adding several important methods in the class *Atlas*. Methods were written to save anAtlas to disk and to instantiate a new atlas from the object stored on the disk. A method to create a new Atlas from aBitmap3D or anImage output from the segmentation processes in MedView was also added.

5.2.5 VOI Class

This class is a subclass of Dictionary and contains the data structure used to describe the volumes of interest in the atlases. It contains key-value pairs for pixel statistics and for accessing the volume occupancy array. Although the dictionary structure is

not necessary for implementing the VOI class since there is only a small and fixed number of elements in each instance, it is very useful for accessing that data and for implementing the user interfaces.

The data structure for a VOI is simple, the fields are mainly integers containing the pixel statistics. The field which contains the references to the volume representation of the VOI is more complicated.

A parallel image is used to store the volume occupancy data for each VOI and the data is referenced with *anImageReference*. For example, anImageReference might be **(d:ima23.des, 64)**. In this case, the image contained in the file named in the reference has the same dimensions as the original image and all of the pixel positions containing the value "64" belong to the volume occupancy array of the VOI containing the reference. Since anImage contains only single byte values, it is necessary to use more than one instance of Image to hold the volume occupancy data if the number of VOIs in the atlas is greater than 256.

5.2.6 SearchAtlas Class

The *SearchAtlas* is also a sub-class of Dictionary and has a similar structure to the *Atlas* class. Each key/value pair is a name and searchVOI respectively. The searchVOI does not contain any volume occupancy information but does contain representative, first order, pixel statistics for each imaging modality as well as variational position information. Figure 5.11 shows the interface used to edit a search dictionary.

5.3 MedView Class

This class, a subclass of ViewManager, provides the capability to view and manipulate images on the screen. It provides access to general purpose image processing tools

in the *Image* class and controls the image display on the screen. To compare images, several MedViews can be opened at once, each displaying a different image.

MedView also provides an interface to the segmentation methods used to generate a primitive atlas from the baseline image.

5.3.1 The Graphical User Interface

The user interface for MedView is shown in Fig. 5.5. It consists of a single graph pane with 5 custom menus at the top. The graph pane is capable of displaying instances of the Image and Bitmap3D classes.

The menus are entitled: **File**, **Tools**, **Image**, **Processing** and **Atlas**. The File menu is generated by *Windows* while the other four are controlled by the application.

The **Tools** menu provides both imaging tools and display palette control. The imaging tools include utilities to subsample, flip, and expand the dynamic range of the displayed image. Other tools are a histogram generator and a thresholding tool that highlights the image pixels within a range of values. Several display palettes are available for image display including a 256 level grey scale and a 256 level "hot metal" palette, a palette which approximates the colour of a metal body as it heated from room temperature (black) through orange, yellow and finally white.

The **Image** menu allows the user to retrieve an image from the disk, save an image to the disk and to create a new bitmap3D from a series of 2-D images. Three types of images are supported, 2-D and 3-D images in the internal format and images in the GE Highlight Advantage Database format. Alternatively, just the file header can be retrieved so the user can check the file information before reading the entire file into the MedView.

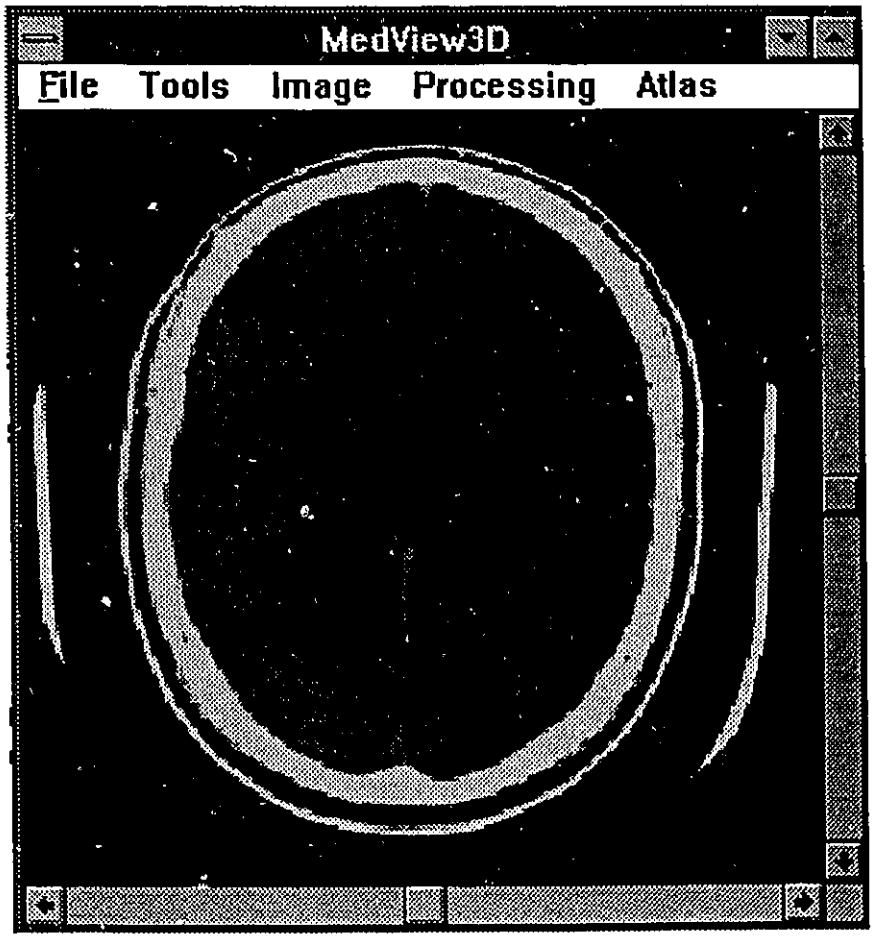

Figure 5.5: The MedView Graphical User Interface

The **Processing** menu provides image processing functions such as filtering, sorting and segmentation. Dialog boxes are used to input parameters into several of the image processing routines. To set the filter size a *FilterDialog* was created as shown in Fig. 5.6. The size of the filter in all spatial dimensions can be set interactively using the dialog box.

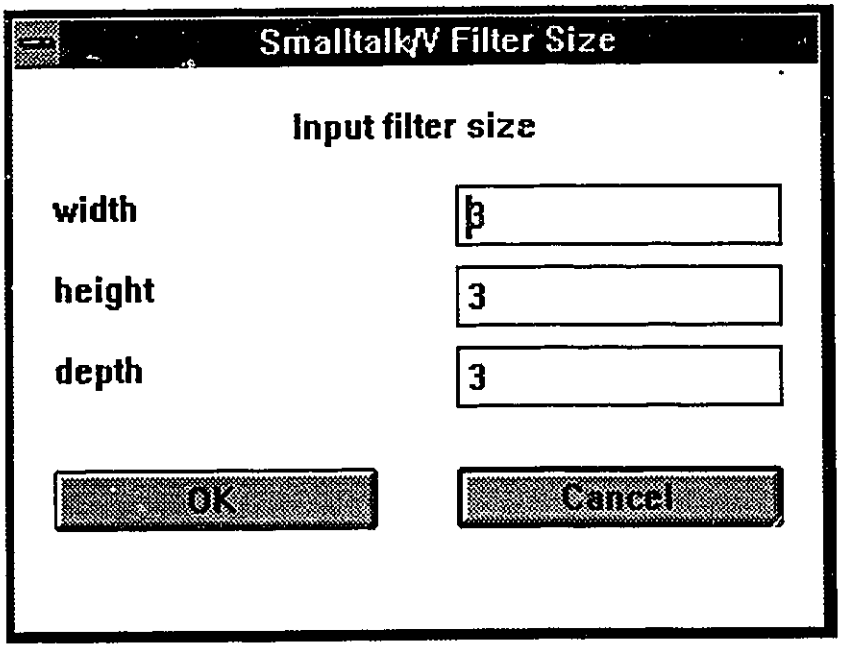

Figure 5.6: The Filter Size Dialog Box

The image sorting algorithm also requires that the correct sorting levels be set for image type. A dialog box is used to save and recall the sorting parameters to disk as well as to modify the sorter. Figure 5.7 shows the Image Sorter dialog box containing the list of levels used by the sorter as well as buttons to add or delete levels, save and open sorter files and cancel or accept the changes.

The processing menu also contains items to call the region growing and watershed segmentation algorithms. Both of these algorithms use the sorter described above therefore, if the sorter has not been defined when the segmentation method is selected,

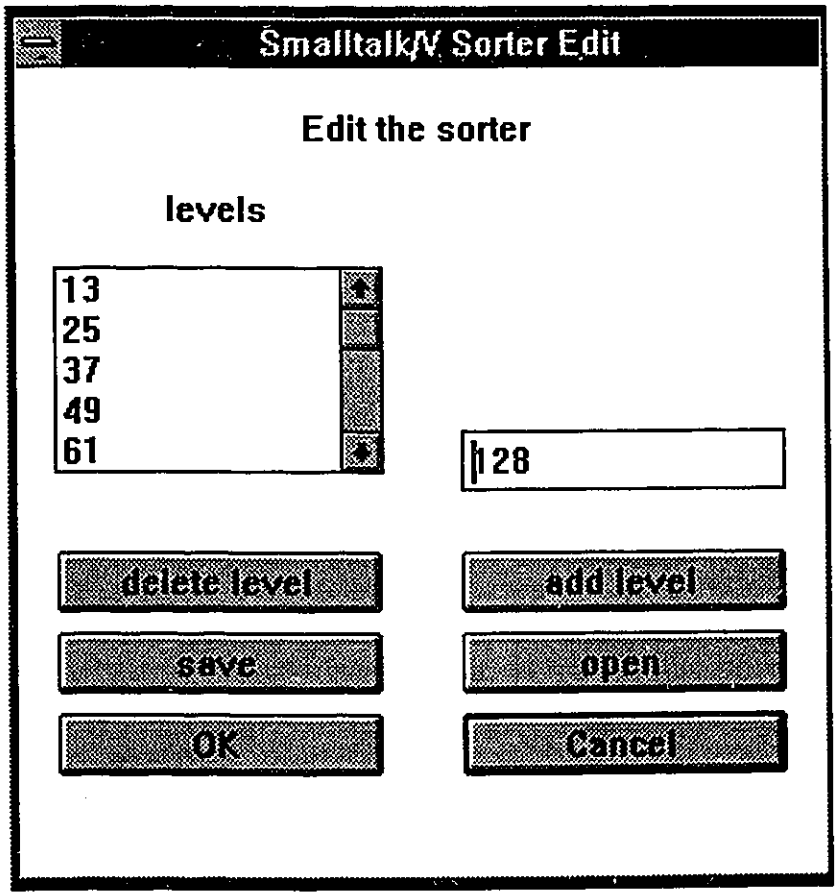

Figure 5.7: The Image Sorter Dialog Box

the sorter dialog box is displayed automatically so it can initialized.

The **Atlas** menu contains two items: one used to reset the atlas creation process and the other used to create a new atlas. The process of creating a new atlas starts with the results of the segmentation as input, it then creates a new primitive atlas, and writes the atlas to disk. The atlas can then be inspected in the AtlasBrowser or processed further in the AtlasMatcher.

5.4 Atlas Browser Class

The *AtlasBrowser*, as its name suggests, is used to browse an Atlas. It provides a flexible interface for inspecting the atlas data structures by displaying all the information contained in each VOI, that is the pixel statistics and the volume occupancy information. A secondary method of accessing the atlas data is also provided by finding the VOI membership of a pixel position on the screen.

The atlas browser can only be used to browse anAtlas and cannot display aSearchAtlas.

5.4.1 The Graphical User Interface

The GUI for the AtlasBrowser application is more complicated than for MedView. There are four boxes (panes) down the left side of the screen, and a single graph pane takes up the right of the application pane, as shown in Fig. 5.8.

The top three panes on the left of the application pane are list panes. The top pane lists the names of all the files in the current directory containing atlases. Upon start-up, only this pane is filled. Once an atlas file has been selected in the top pane, the atlas is read-in from the file, a list of the keys contained in that atlas is written to the second list pane, and the baseline image is displayed in the graph pane.

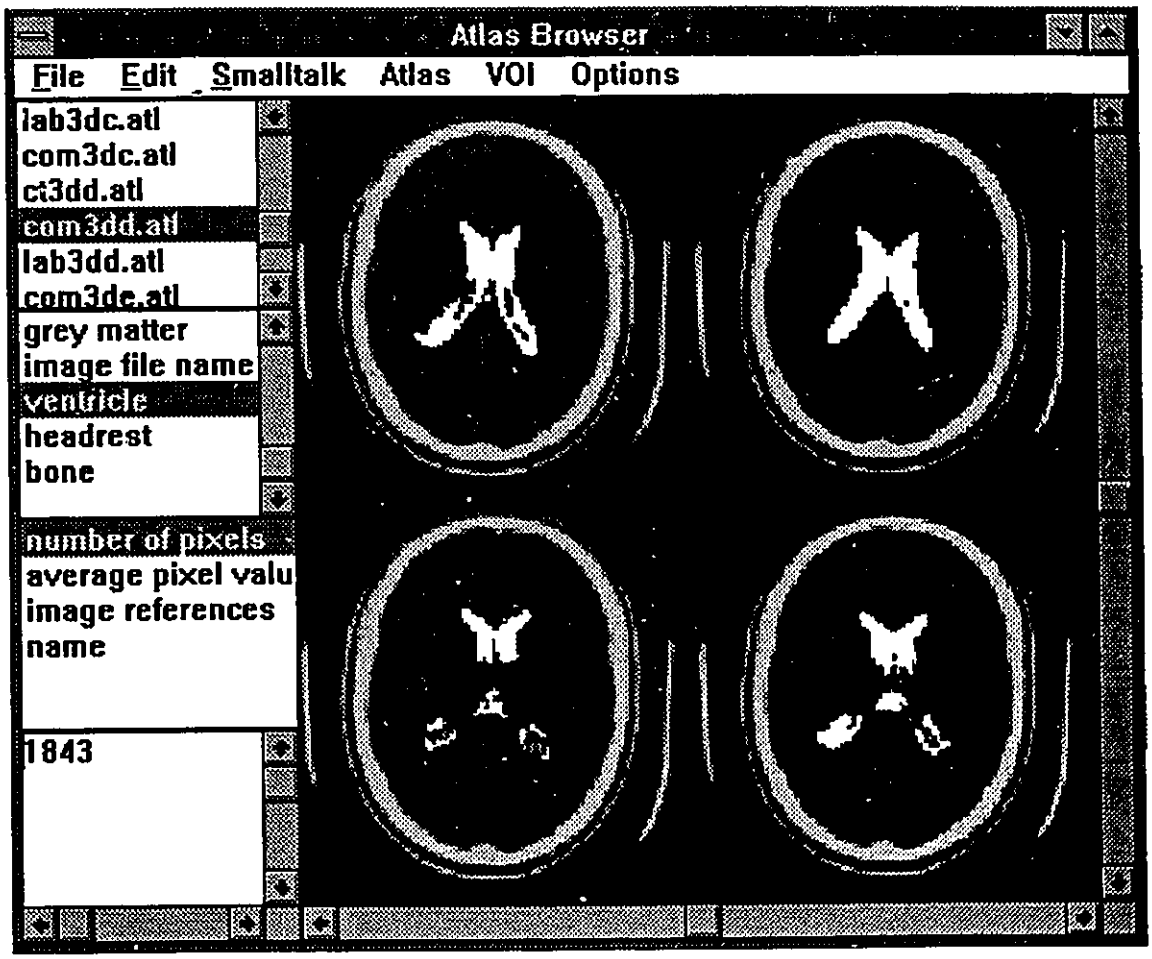

Figure 5.8: The AtlasBrowser Graphical User Interface

Selecting a key in the second list pane causes the list of keys in the selected VOI to be written to the third list pane and highlights the pixels that are in the graph pane image and are contained in the volume occupancy array of the VOI. In Fig. 5.8, the key, **ventricle** has been selected in the second list pane. This caused the keys in the VOI to be displayed in the third list pane and the areas in the data set labelled as ventricles to be highlighted in white in the graph pane.

Selecting one of the VOI keys in the third list pane displays the value associated with that key in the text pane, the bottom pane of the four. Selecting a pixel in the graph pane by clicking the left mouse button on the image causes the VOI containing that pixel to be selected in the atlas keys pane (second from top).

There are six menus in the atlas browser interface. The first three: **File**, **Edit** and **Smalltalk** are provided by the system and are not important to our discussion. The next three menus: **Atlas**, **VOI** and **Options** are used to control the user interface and perform processing funtions on the atlas.

The **Atlas** menu is used to delete atlases. This is an important function since there are several files associated with each atlas, possibly in different directories, and keeping track of which files belong to which atlas can be difficult. With this function, all of the files associated with an Atlas except for the baseline image can be deleted.

Using the **VOI** menu, the user can choose the method of displaying the volume data associated with a VOI. There are three choices, 2-D outline, 2-D volume, and 3-D surface rendering. The default is the 2-D volume display, as shown in Fig. 5.8, as it is the fastest, however, once the VOI has been selected the display mode can be changed using the VOI menu. Figure 5.9 is an unsmoothed 3-D rendering of the grey matter from the example data set.

The **Options** menu allows the user to change the current directory and to save the image displayed in the graph pane to disk for reference. Once the image has been

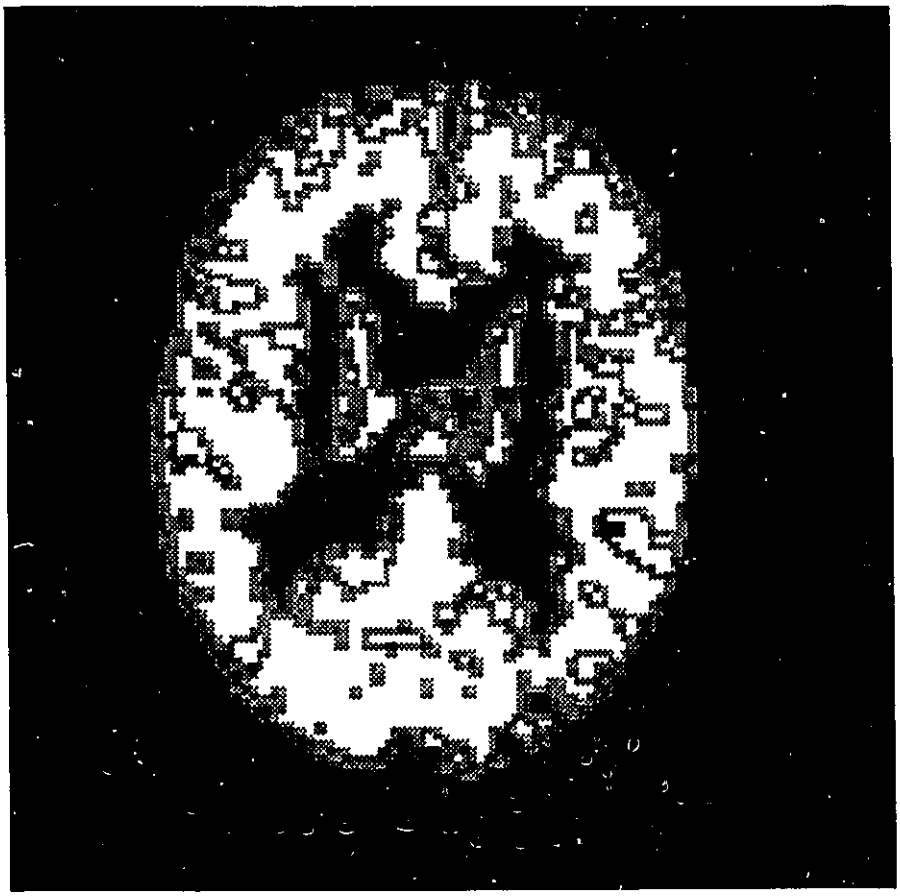

Figure 5.9: A 3-D Rendering of the Grey Matter from the Example Data Set

saved to disk it can be later displayed using a MedView.

5.5 AtlasMatcher Class

The Atlas Matcher is used to create an individual atlas by merging and labelling the primitive atlas. Four atlases are used to perform the merging: the primitive atlas that is yet to be merged, the search dictionary that controls the search and is used to generate candidate VOIs for merging based on pixel statistics, the previously created individual atlas that can be used to select a subset of the candidate VOIs for merging based on position and shape, and the individual atlas to be created.

Since the methods used for performing the matching have been described in Section 4.4.3, the following section discusses only the user interface.

5.5.1 The Graphical User Interface

In Fig. 5.10, the leftmost column shows a list of the VOIs in the search dictionary. This list contains the names of all of the structures which are to be searched for in the primitive atlas. Each item in the list references a searchVOI in the searchAtlas and can therefore access the information necessary to generate a candidate list of VOIs from the primitive atlas.

The second column initially contains the list of keys to the previously created individual atlas, but it is later used to list the keys in the individual atlas under creation. As soon as the merging begins, this column is cleared and entries are added as the VOIs are found.

The third column has two modes depending on whether the key is selected in the first or second column. If a key in the first column is selected, a list of the VOIs already assigned to that key in the individual atlas under creation is shown. However,

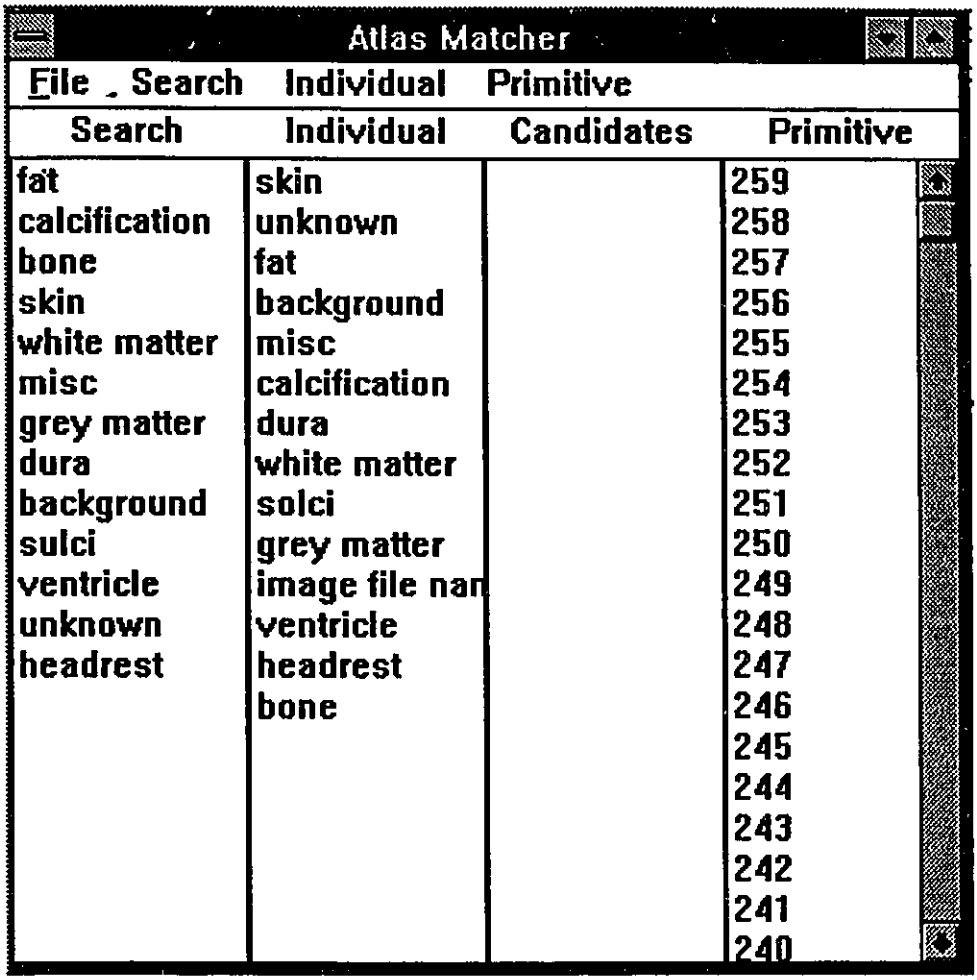

Figure 5.10: The Atlas Matcher Graphical User Interface

if a key in the second column is selected, and that key is not yet contained in the individual atlas under creation, then the third column is filled with a list of candidates from the primitive atlas.

The fourth column contains the list of the unassigned VOIs remaining in the primitive atlas.

There are four menus in the AtlasBrowser user interface; the standard **File** menu, and the **Search, Individual** and **Primitive** menus.

The search dictionary is initialized by selecting **new search dictionary** from the **Search** menu. This action initiates a dialog box used to recall search dictionaries from the disk, modify them, and then save the modified version back to disk. As shown in Fig. 5.11, the values for each VOI and for different imaging modalities can be set individually.

The **Individual** menu allows the user to manipulate the individual atlas in the second column. The user can retrieve a previously created individual atlas from disk and, after a new individual atlas has been created, save it to disk. When an individual atlas is first created by the AtlasMatcher, each VOI comprises several VOIs from the primitive atlas and the original image references are kept. The **compact atlas** function is provided to create a new volume occupancy image for the atlas and a new image reference for each VOI; replacing the multiple original image references with a single image reference to the new volume occupancy image.

The **Primitive** menu has items enabling the user to retrieve a primitive atlas from disk and to start the automatic labelling process. It is not necessary to use the automatic labelling process, labelling can be done by the user with assistance from the AtlasMatcher in generating the list of candidates from the primitive atlas.

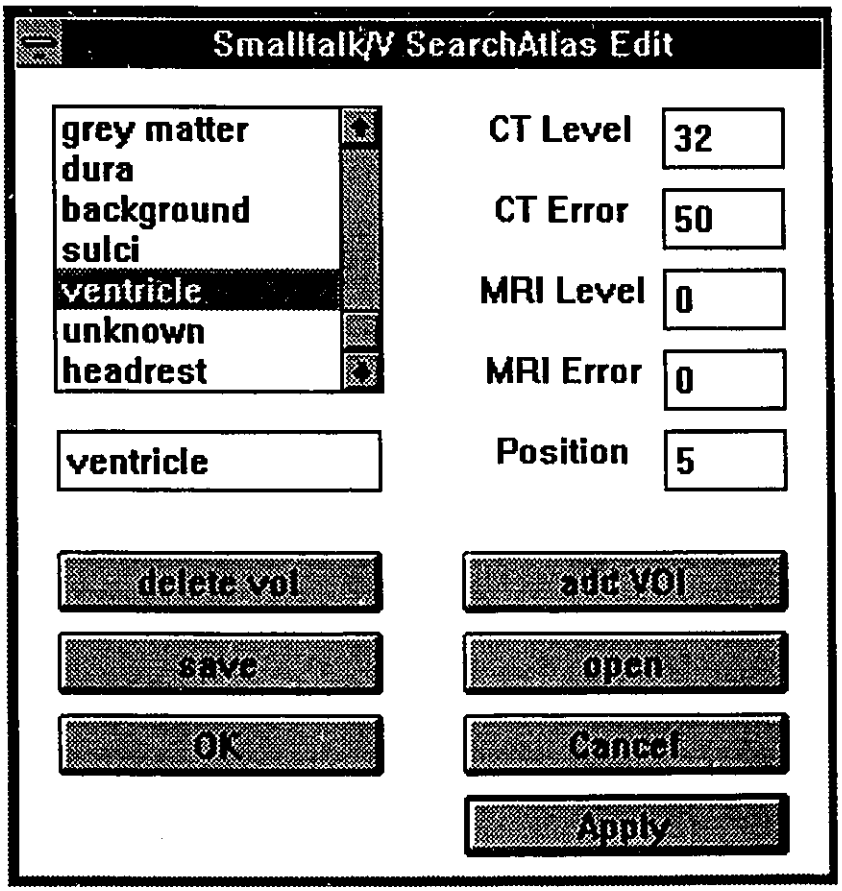

Figure 5.11: The Search Dictionary Dialog Box

5.6 Conclusions

How can this software be used by a radiologist? All three applications can be run simultaneously to create an environment where images can be read in from the scanner database, displayed, processed, and displayed again using the atlas as a key to the data. MedView is used to perform preprocessing, initial 3-D or 2-D segmentation, and then to store the resultant segmented image in a primitive atlas. AtlasBrowser is used to view the primitive atlas or AtlasMatcher is used to merge and label the primitive atlas directly into an individual atlas. The results of the merging and labelling can then be viewed using the AtlasBrowser in any of the available display formats.

The classes and methods described in this chapter were created to implement the ideas presented in Chapter 4. However, only the classes and methods relevant to this discussion have been described, others have not been included.

The next chapter attempts to quantify the performance of the software using CT images.

Chapter 6

Experimental Results

Trials were performed at the Ottawa Civic Hospital to test the accuracy of the initial segmentation and the robustness of the atlas matching. It was necessary to use radiologists to judge the results since no quantitative measurements were possible. CT images were used for all of the tests since they were the only type available at the Civic Hospital.

6.1 Initial Segmentation Test

This test compared the two segmentation methods, watersheds and region growing, to identify the method that produced results closest to the correct segmentation, as judged by the radiologist. Tests were performed with 2-D and 3-D images from a GE *Highlight Advantage* CT scanner. The primitive VOIs were merged into medically meaningful regions and false coloured for presentation to the radiologists since it was felt that the false coloured images would be easier to judge than the raw segmentation results. The merging of the primitive VOIs was done by hand to ensure that the segmentation was being tested and not the accuracy of the merging.

The test methodology is described in more detail in Section 6.1.1. The images

used for testing and the false coloured images depicting the results are provided in Appendix A.

6.1.1 Test Methodology

The radiologists were asked to judge four specific aspects of the segmentation:

1. The differentiation of white and grey matter
2. The shape of the ventricles
3. The shape of the surface of the skull
4. The localization of sulci

The criteria used to judge the results were selected in conjunction with the radiologists because they are important and to best fit the practical uses of such a system. The boundary between white and grey matter is often difficult to visualize in CT images since their radiological densities are very similar. The shapes of the ventricles and sulci are commonly used to evidence mass effect or intra-cranial pressure. The proper segmentation of the skull is used for subsequent 3-D surface rendering and shading of the skull.

Radiologists are accustomed to visually gauging the extent of structures in images. They routinely compare images or image sets and express changes in the size and position of pathologies in approximate terms. In comparing the segmentation methods, their approximations were sufficiently accurate to quantify the results.

2-D images were used for most of the segmentation trials since they were quicker to process, easier to analyse, and there was no advantage to testing with 3-D images.

Image No.	Method	Skull	White/Grey	Sulci	Ventricles
1	WS	excellent	moderate	poor	excellent
1	RG	excellent	moderate	good	excellent
2	WS	good	moderate	poor	excellent
2	RG	excellent	good	good	excellent
3	WS	excellent	moderate	moderate	excellent
3	RG	excellent	moderate	moderate	excellent
4	WS	excellent	moderate	poor	excellent
4	RG	excellent	good	poor	excellent
5	RG	excellent	moderate	poor	excellent

Table 6.1: Results of the Segmentation Trial

However, as a final test, the result from a 3-D image was compared with the 2-D results to see if performing the direct, 3-D, segmentation has an advantage over segmenting each slice individually.

6.1.2 Test Results

The results of the segmentation test are summarized in Table 6.1. A grading scheme for judging the performance of the methods was used by the radiologists to quantify their findings. The rating system was as follows: excellent = 90-100%, good = 75-90%, moderate = 50-75%, and poor = 0-50%. The percentage refers to the percentage of correctly segmented voxels in the labelled image for the specific image feature under test. The performance of the region growing algorithm was judged to be superior to that of the watersheds. From a practical point of view the region grower is simpler, faster and does not have watershed lines in the final output.

The watershed algorithm resulted initially in an oversegmentation of the image. Although an oversegmentation is desirable for the merging stage, the resulting regions were too small and numerous to be effectively searched. To improve the control of the WS method a presorting of the image was performed as described in Section 4.3.1.

In both cases, the watersheds and the region grower, the performance was improved by incorporating *a priori* knowledge of the grey level distribution in the images into a presorting of the images. Although this method worked well for these types of images, it would be difficult to implement in a general purpose segmentation method since *a priori* knowledge of the grey levels of important regions might not be available.

In comparing the performance of the region grower on 2-D and 3-D images, it was found that the 3-D segmentations (4 slices) contained only $\approx$ 25% more VOIs than a single slice at a comparable resolution. This is advantageous for the merging stage of the process since there would be far fewer VOIs to search through using a direct 3-D segmentation.

Figure A.1 is one of the images used for the 2-D segmentation study. Medically, the image is normal except for an area of hypo-density on the right of the image near the ventricle. The result of processing this image and merging the regions found in the watershed (WS) segmentation is shown in Fig. A.2. Although the bone and ventricles are well visualized, the white/grey matter differentiation is moderate and the sulci are less than 50% visualized. As well, the WS algorithm leaves lines around each region and some narrow or small regions were not well depicted due to these lines. The results of performing a region growing on this image and then merging the regions is shown in Fig. A.3. The white/grey matter differentiation was not complete and some small areas of white matter, particularly those in narrow fossa, were labelled as grey matter. The depiction of the sulcal anatomy was good with about 85% being visualized. The segmentations of both the ventricles and the skull were considered accurate.

The results of processing Image No. 2 (Fig. A.4) using the WS and the region growing algorithms are shown in Figs. A.5 and A.6 respectively. The results for the

region grower were substantially better than for the watersheds for the skull, sulci, and the grey/white matter differentiation. In this image, the narrow features such as the sulci and the skin layer surrounding the skull were particularily poorly segmented by the watershed method.

The segmentation results of Image No. 3 (Fig. A.7) show minimal differences between the two segmentation methods. This is attributable to the simple structure of the image which is largely devoid of narrow details which might have highlighted the differences. The results are shown in Figs. A.8 and A.9.

The last 2-D image used in the test was Image No. 4 shown in Fig. A.10. This figure contained a considerable amount of fine structure which challenged both of the segmentation methods. In both cases, the ventricles and skull were well segmented but, as evidenced in Fig. A.11, the WS algorithm did not segment the sulci properly and the and the grey/white matter differentiation appears imprecise. The region growing algorithm (see Fig.A.12) was better able to differentiate the grey/white matter boundary but there are many miscellaneous spots inside the white matter and the sulci are incompletely segmented. These problems in segmentation were likely caused by the noise in the original image.

Lastly, Fig. A.14 (Image No. 5 from Table 6.1) displays the results of the 3-D region growing and merging process applied to a 3-D set of images from the same study. It was found that the segmentations of the skull and ventricles were, once again, accurate and that the white matter regions were under-estimated. The performance on the sulci was not as good as in the 2-D case. This was attributed to the decreased image resolution in the 3-D image ($128 \times 128 \times 4$).

Overall, the ventricles, bone, and background were the easiest to segment for both methods since they are large and homogeneous regions. In those three cases it was usual for the primitive atlas to contain a single entry for each of them and

no merging was necessary in creating the individual atlas. For small regions such as sulci the region grower generally performed better than the watersheds method. This difference in performance was attributed to the watershed lines themselves having to be included in the matrix containing the segmented results which, for narrow regions such as sulci, interfered with the correct segmentation.

The averaging process tended to blur the strong edges in the image causing "transition" regions to be created in these blurred edges. This caused regions to be included in the primitive atlases which were not significant and which were not classifiable. This effect can be observed in the results of the region grower around the ventricles and the outside of the skull in Fig. A.6. The above-mentioned problem of the watershed algorithm with narrow regions caused many of these transitions to be missed and consequently included the transition in one or the other of the neighbouring regions.

The problem of incorrectly specifying the boundary between grey and white matter was due mostly to the poor differentiation between these two regions in the image data. In some cases the sorter could be set to differentiate accurately, but in general the differences between patients were too great to do this consistently. This area would benefit from the integration of MRI data, since this data could more exactly specify the grey/white boundary.

6.2 Atlas Matching Tests

The atlas matching tests were used to test the robustness of the matching process. For example, if the individual atlas was derived from exactly the same data set as the primitive atlas the matching should find exactly the same VOIs and label them identically. Whereas, if the primitive atlas was of a severely abnormal brain and the individual atlas was from a relatively normal brain, then mismatches could be

expected. Mislabellings could take several forms: the VOIs in the final individual atlas could include extra VOIs from the primitive atlas, VOIs could be missing from some of the structures, or some structures may be not found at all in the primitive atlas. A measure of the robustness was taken as the percentage of mislabelled regions.

There were two testing phases. The first used data sets from five different patients. Primitive atlases were created from each data set. One dataset was hand merged and labelled to act as the individual atlas to merge and label each of the five primitive atlases. The test images and the results of the first phase are included as Appendix B.1. The second stage used 5 sets of images extracted from several levels in a single data set. The same procedure was used as in the first part of the test, one of the primitive atlases created from the data sets was hand merged and labelled, and used to automatically merge and label each of the primitive atlases. The original images and the test results in image form are provided in Appendix B.2.

6.2.1 Test Data

For the first phase, 5 data sets were extracted from the hospital database. The data sets were roughly registered since they were all collected with similar programs. For the second phase, five 3-D images with varying degrees of dissimilitude were created from a 40 slice study. Each image comprised 4 consecutive slices of a closely spaced data set. The images are listed in Table 6.2.

For both trials, an *individual atlas* was constructed by hand to act as the morphological knowledge during the merging and labelling procedure. Unfortunately, these atlases suffered from some of the segmentation problems identified during the segmentation tests and were therefore less than perfect. For the first phase of the trial, Image No. 1 (see Table 6.2) was used to construct the individual atlas. As shown

Image No.	Slice 1	Slice 2	Slice 3	Slice 4	Note
1	5	6	7	8	patient 1 (baseline)
2	5	6	7	8	patient 2
3	5	6	7	8	patient 3
4	5	6	7	8	patient 4
5	5	6	7	8	patient 5
6	21	22	23	24	baseline
7	22	23	24	25	1 up
8	20	21	22	23	1 down
9	24	25	26	27	3 up
10	18	19	20	21	3 down

Table 6.2: Test Images for Both Testing Phases

Trial No.	Primitive Atlas	Individual Atlas
1	1	1
2	2	1
3	3	1
4	4	1
5	5	1
6	6	6
7	7	6
8	8	6
9	9	6
10	10	6

Table 6.3: A Listing of the Trials

in Fig. B.1, it is a four slice data set of a "normal" brain. Image No. 6 (Table 6.2) was used in this capacity for the second phase of the trial and is shown in Fig. B.14. The baseline individual atlases used to merge and label the other primitive atlases are shown in Figs. B.2 and B.13 respectively.

Table 6.3 is a listing of the merging trials. The second column shows the image used to create the primitive atlas for each trial.

6.2.2 Test Methodology

The radiologists were asked to identify the mislabelled regions in the final individual atlas. The identification of the mislabelled regions was done by a direct comparison of the individual atlas and the original image. An estimate of the percentage of mislabelled voxels was made by using the primitive atlas to the same data set. Once again, radiologists were considered to be the best arbiters of the results.

To perform the comparison, the volume representations of the individual atlases were false coloured using representative grey levels and printed for presentation to the radiologists along with a printed image of the original. The radiologists identified regions on the images where the labelling differed from that which they would have assigned.

6.2.3 Test Results

False coloured images were used to display the results of the matchings. Representative grey levels were used for each tissue type and miscellaneous volumes, ones which were too small to be categorized, were given a distinct level (128). Unknown regions were given a (0) level. This format was chosen to display the results since the images are then directly comparable to the originals and can be printed on hardcopy. Outlining the boundaries was also considered but a method of designating the label for each outlined region could not be designed.

The trials are listed in Table 6.4 and the false coloured images are included as Appendices B.1 and B.2. The results show that the merging and labelling procedure was quite robust for the skull, background and ventricles. The process was also able to correctly label brain tissue as either white or grey matter but in several cases the differentiation between the two was not accurate. The process had some difficulty,

Trial No.	Accuracy	Problem Areas
1	excellent	ventricles and skull overestimated
2	moderate	ventricles overestimated, grey matter and sulci underestimated
3	moderate	ventricles overestimated, grey matter underestimated
4	good	ventricles overestimated
5	good	grey matter underestimated
6	excellent	sulci and white matter underestimated
7	excellent	sulci and white matter underestimated
8	excellent	sulci and white matter underestimated
9	excellent	sulci underestimated
10	good	sulci poor, white matter underestimated

Table 6.4: Results of the Merging and Labelling Trial

however, in correctly labelling the sulci and, in phase 1 of the trial, segmentation problems caused the extent of the ventricles to be overestimated.

In the case of the ventricles, they were accurately segmented and labelled in the data set containing thin slices. In the first phase of the trial, however, the partial volume filling effect due to the thick slices ($\approx 1cm$) caused some areas near the ventricles to have indeterminate grey levels. This led to an over-estimation of the size of the ventricles. There was no "correct" labelling for these indeterminate areas since the volume was neither wholly ventricle nor surrounding tissue. This problem was not observed in the second phase of the trial indicating that it can be avoided by using thin slices.

The problem of correctly identifying the grey/white matter interface is mainly one of segmentation. In a few of the trials (1,4,9) the grey/white boundary was correctly identified, however, in the majority of cases the boundary was incorrect. To solve this problem, the segmentation of these two regions must be improved. One solution which was tried was to increase the fidelity of the sorter over an interval which bracketed the expected transition level between white and grey matter. In this way, the fidelity

of the segmentation was improved but the number of VOIs in the primitive atlas increased so dramatically as to make hand labelling of the atlas impractical.

The last problem identified by these trials was the incorrect labelling of the sulcal anatomy; many sulci which were visible in the image were not labelled correctly in the resulting atlas. The problem arose since each individual sulcus was highly convoluted and their positions were highly variable between patients. The problem was further aggravated by the small size of the sulci since the subsampling and averaging made them indistinct and difficult to segment. Both of these factors rendered the positional matching portion of the labelling process ineffective for narrow sulci. The best solution for improving the performance on the sulci would probably be to process the data at full resolution (512×512) but this would only address the segmentation issue and not the positioning issue. A solution to this problem has not been found.

The results from phase 2 of the trial were consistently better than those from phase 1. This was attributed to both the better image registration (since the images in the 2nd phase were from the same data set) and the thin slices in the second phase which reduced the partial volume filling effect. As was expected, the more similar the individual atlas and the primitive atlas, the better the results.

Chapter 7

Conclusions

This chapter has two parts: a discussion of the functional approach taken to solve the problem of 3-D image editing and display and, since a large component of the project was the software written to test these ideas, a discussion of the implementation issues.

As discussed in Chapter 1, the problem we are trying to solve is how can we access medically significant structures in volume data for display and editing. Solving this problem would benefit radiologists by providing more flexible image displays and renderings, and by aiding in image registration and multi-modality image integration. Currently, most 3-D medical imaging workstations rely on simple thresholding to extract volumes from the data set and most methods of image registration rely on user input for finding the transformation between data sets. In Chapter 4 a plan to use an *atlas* data structure to solve this problem was proposed.

A unique aspect of this work was to create *individual atlases* for each data set. Thus, rather than deforming a generalized atlas to fit each data set, the process started with a segmentation of the data set. Specific structures were then searched for in the segmented data to create the individual data set. Knowledge of the expected grey levels for each imaging modality and morphological knowledge of the structures

were included to aid in the searching process.

This approach of using an atlas to abstract the information contained in the individual data sets was found to be both practical and useful. It allowed for intuitive browsing of the structures and for novel methods of accessing the data. For example, an atlas can be used to display a rendered image of a single structure or to highlight a structure in a collage display. It can also be used in the reverse direction; if a point is selected in the data set the atlas can be used to find which structure that point belongs to. An atlas can aid in image registration since, if the same structure is well defined and labelled in both data sets, that structure can be used to calculate the transformation between the data sets. Matching the two structures in binary form provides an easy starting point for the calculation.

As shown in Chapter 6, the segmentation problem is central to this work. In the most general case of segmenting an arbitrary scene, the problem is extremely difficult to solve since no prior knowledge can be used to assist in the segmentation. However, in the case of 3-D slice-type images, the problem is more simple. Not only do most of the images have a common structure, but specific tissues are displayed with representative grey levels. By exploiting these two traits, a method of reliably segmenting and labelling the data has been developed. As may be expected, the segmentation works well for large homogeneous structures but is less able to segment small convoluted structures, especially after subsampling and and filtering has been applied to the image.

To further improve the performance of the system, it would be necessary to improve the segmentation process. The two problem areas identified with the segmentation, differentiation between white and grey matter and finding the sulci, require further study. A more sophisticated sorter which used a stochastic model of the expected grey levels instead of the simple thresholds which were used could improve

the performance of the initial segmentation. It is hoped that this new sorter could offer increased fidelity without increasing the number of segmented regions. Since the subsampling and averaging create significant problems for the current sorter due to edge blurring, perhaps a sorter which uses a stochastic model of the expected grey level distribution could be used on a non-averaged image. If successful, this would have the added benefit of reducing the number of regions found in the image and thus reduce the number of VOIs in the primitive atlas.

Overall, the results proved that the process could be used to segment and label volume data with sufficient accuracy to create an individual atlas to a data set which could assist a radiologist clinically. The individual atlases could be beneficial in the 3-D display of the data set, in registering data sets, and in extracting quantitative measures of specific structures from the image data.

The merging and labelling stage forms an individual atlas by combining the VOIs in the primitive atlas into VOIs which correspond to meaningful medical structures. Morphological information from a previously created individual atlas is used as a guide for performing the matching. The success of this matching is strongly dependent on the similarity of the structures in the previously created individual atlas and the primitive atlas. Using images of the same patient resulted in improved performance of the matching stage over using an individual atlas from a different patient.

7.1 Implementation Issues

Smalltalk/V for *Windows* was found to work well for prototyping and testing the idea of an atlas and to develop user interfaces under *Windows*. Many of the system classes were useful for the implementation and the programming environment provided by *Smalltalk/V* allowed for quick and easy evolution of the code.

The design of the user interface was not investigated or tested specifically but the *Windows* environment offered several useful features such as pull-down menus and support for bitmap display and palette manipulation. The *dialog box* was found to be the best method of entering parametric data and prompting the user for information.

There are several features which could be added to the image processing application which would aid in the creation of image sorters. To display thresholding results to the screen more quickly, palette animation could replace the current method of thresholding and displaying the result. Therefore, instead of having to display an entire image after each thresholding operation, only the palette entries associated with a range of grey levels would have to be changed and the effect would be visible almost instantaneously.

Better access to the quantitative grey levels in the images would aid the user. Currently, the user can display the value of a single pixel but a display of the values in a neighbourhood would be more useful.

More work should be done on managing the image data and the atlas files. Currently, the user must keep track of the image and atlas files and no provision has been made for recording this information in the file headers. Thus there is no method of automatically recording in the atlas file which image, search dictionary, or sorter was used to create an atlas. A method of automatically managing the data by patient or by study would be necessary if the system were to be used clinically. The clinical usefulness of the software could also be enhanced by improving the 3-D surface rendering software. The current implementation does not perform any smoothing of the surface once it is found and does not create a convincing 3-D effect.

Bibliography

[1] R.S. Acharya and C. Benchimol, "A 3-D segmentation algorithm for extracting object surfaces from computed tomography scans", In *Proceedings of the International Electronic Imaging week Second Image Symposium*, pages 452–460, April 1986.

[2] A. Apicella, J.S. Kippenham, and J.H Nagel, "Fast multi-modality image matching", In *SPIE vol. 1092, Medical Imagery III: Image Processing*, pages 252–260, 1989.

[3] R. Bajcsy and S. Kovačič, "Multi-resolution elastic matching", *Computer Graphics, Vision and Image Processing*, 46:1–21, 1989.

[4] R. Bajcsy, R. Lieberson, and M. Reivich, "A computerized system for the elastic matching of deformed radiographic images to idealized atlas images", *Journal of Computer Assisted Tomography*, 7(4):618–625, 1983.

[5] R. Bakalash and A. Kaufman, "Medicube: A 3-D medical imaging architecture", *Comput. and Graphics*, 13(2):151–157, 1989.

[6] D.H Ballard and C.M. Brown, *Computer Vision*, Prentice-Hall Inc, Englewood Cliffs New Jersey 07632, 1982.

[7] G. Bélanger, private communication, 1989, Neuroradiologist Ottawa Civic Hospital.

[8] F. Bloch, "Nuclear induction", *Physics Review*, 71(3):460–474, 1946.

[9] I. Bloch and F. Preteux, "Démare morpho-mathématique pour la segmentation application aux lésions cérébrales en irm", In *Onzieme Colloque*, pages 575–578, Gretsi-Nice, juin 1987.

[10] P. Bloch and J.K. Udupa, "Application of computerized tomography to radiation therapy and surgical planning", *Proceedings of the IEEE*, 70:351–355, March 1983.

[11] C. Bohm, T. Greitz, D. Kingsley, B.M. Berggren, and L. Olsson, "Adjustable computerized stereotaxic brain atlas for transmission and emission tomography", *A.J.N.R.*, 4:731–733, 1983.

[12] M. Brant-Zawadzki and D. Norman, editors, *Magnetic Resonance Imaging of the Central Nervous System*, Raven Press, New York, 1987.

[13] L.J. Brewster, S.S. Trivedi, H.K. Tuy, and J.K. Udupa, "Interactive surgical planning", *IEEE Computer Graphics and Applications*, 4(3):31–40, March 1984.

[14] C. Brice and C. Fennema, "Scene analysis using regions", *Artificial Intelligence*, 1:205–226, 1970.

[15] J.L. Coatrieux, C. Toumoulin, C. Hamon, and L. Luo, "Future trends in 3-D medical imaging", *IEEE Engineering in Medicine and Biology*, 9(4):33–39, December 1990.

[16] I. Cohen, L.D Cohen, and N. Ayache, "Using deformable surfaces to segment 3-D images and infer differential structures", *Computer Graphics, Vision and Image Processing: Image Understanding*, 56(2):242–263, September 1992.

[17] R. Dann, J. Hoford, S. Kovacic, M. Reivich, and R. Bajcsy, "Three dimensional computerized brain atlas for elastic matching: creation and initial evaluation", In *SPIE vol. 914, Medical Imagery II*, pages 600–608, 1988.

[18] S. Dellepianie, G. Venturi, and G. Vernazza, "Model generation and model matching of real images by a fuzzy approach", *Pattern Recognition*, 25(2):115–137, 1992.

[19] A.P. Dhawan and L. Arata, "Knowledge-based 3-D analysis from 2-d medical images", *IEEE Engineering in Medicine and Biology*, pages 30–37, December 1991.

[20] S. Napel *et al*, "CT angiography with spiral CT and maximum intensity projection", *Radiology*, 185(2):607–610, November 1992.

[21] A.C. Evans, C. Beil, S. Marrett, C.J. Thompson, and A. Hakim, "Anatomical functional correlation using an adjustable MRI based atlas with PET", *Journal Cerebral Blood Flow and Metabolism*, 8(4):813–830, 1988.

[22] E.J. Farrell, R.A. Zappulla, and W.C. Yang, "Colour 3-D imaging of normal and pathological intracranial structures", *IEEE Computer Graphics and Applications*, 4(9):5–17, September 1984.

[23] J.A. Feldman and Y. Yakimovsky, "Decision theory and artificial intelligence, a semantics based region analyzer", *Artificial Intelligence*, 5:349–371, 1974.

[24] A. Forte, M. Bernadet, F. Lavaine, and Y. Bizais, "Object oriented versus logical conventional implementation of a MMIIS", In *SPIE Medical Imaging VI: Image Processing*, pages 23–28, Newport Beach, CA, 1992.

[25] K.S. Fu and J.K. Mui, "A survey on image segmentation", *Pattern Recognition*, 13:3–16, 1981.

[26] S.M. Goldwasser et al., "Physicians workstation with real time performance", *IEEE Computer Graphics and Applications*, 5(12):44–57, December 1985.

[27] S.M. Goldwasser and R.A. Reynolds, "Real-time display and manipulation of 3-D medical objects", *Computer Graphics, Vision and Image Processing*, 39(1):1–27, July 1987.

[28] R.C. Gonzalez and P. Wintz, *Digital Image Processing*, Addison-Wesley, Reading, MA, 1977.

[29] C. Henri, L. Collins, T.M. Peters, A.C. Evans, and S. Marrett, "Three-dimensional interactive display of medical images for stereotactic neurosurgery planning", *SPIE vol. 1092, Medical Imagery III: Image Processing*, pages 67–74, 1989.

[30] K. Herholz, G. Pawlik, K. Wienhard, and W.D. Heiss, "Computer assisted mapping in quantitative analysis of cerebral positron emission tomograms", *Journal of Computer Assisted Tomography*, 9(1):154–161, 1985.

[31] G.T. Herman and J.K. Udupa, "Display of 3-D digital images: computational foundations and medical applications", *IEEE Computer Graphics and Applications*, 3(5):39–46, August 1983.

[32] M.S. Hinshaw and A.H. Lent, "An introduction to nmr imaging", *Proceedings of the IEEE*, 71(3):338–350, March 1983.

[33] K.H. Hohne, M. Bomans, M. Riemer, R. Schubert, U. Tiede, and W. Lierse, "A volume-based anatomical atlas", *IEEE Computer Graphics and Applications*, pages 72–78, July 1992.

[34] S.L. Horowitz and T. Pavlidis, "Picture segmentation by a directed split and merge procedure", *Proc. 2nd International Joint Conf. on Pattern Recognition*, pages 424–433, 1974.

[35] M. Hueckel, "An operator which locates edges in digital pictures", *Journal Ass. Comput. Mach.*, 18:113–125, 1971.

[36] R. Berkow (Editor in Chief), editor, *The Merck Manual of Diagnosis and Therapy*, Merck, Sharp and Dohme Research Lab, Rahway N.J., 15th edition, 1987.

[37] M. Kass, A. Witkin, and D. Terzopoulas, "Snakes: Active contour models", In *Proceedings, First International Conference on Computer Vision*, pages 259–268, London, June 1987.

[38] A. Kaufman and R. Bakalash, "Memory and processing architecture for 3-D voxel based imagery", *IEEE Computer Graphics and Applications*, pages 10–23, November 1988.

[39] T. Lei and W. Shewchand, "Statistical approach to X-ray CT imaging and its applications in image analysis- Part II: A new stochastic model-based image segmentation technique for X-ray CT images", *IEEE Transactions on Medical Imaging*, 11(1):62–69, March 1992.

[40] W.E. Lorensen and H.E. Cline, "Marching cubes: A high resolution 3-D surface construction algorithm", *Computer Graphics*, 21(4):163–169, July 1987.

[41] G.A. Maquire, M.E. Noz, H. Rusinek, J. Jaegar, E.L. Kramer, J.J. Sauger, and G. Smith, "Graphics applied to medical image processing", *IEEE Computer Graphics and Applications*, pages 20–28, March 1991.

[42] S. Marrett, A.C. Evans, L. Collins, and T.M. Peters, "A volume of interest (voi) atlas for the analysis of neurophysiological image data", In *SPIE vol. 1092, Medical Imagery III: Image Processing*, pages 467–477, 1989.

[43] J.M. Mazziotta and S.H. Koslow, "Assessment of goals and obstacles in data aquisition and analysis from emission tomography: Report of a series of international workshops", *J. Cereb. Blood Flow Metab*, 7:s1–s31, 1987.

[44] T.H. Milhorat, *Cerebrospinal Fluid and the Brain Edemas*, Neuroscience Society of New York, New York, 1987.

[45] M. Minsky, "A framework for representing knowledge", *PCV*, 1975.

[46] M. Moshfeghi, "Elastic matching of multimodality medical images", *cvgipgm*, 53(3):271–282, 1991.

[47] J.L. Muerle and D.C. Allen, "Experimental evaluation of techniques for automatic segmentation of objects in a complex scene", In *Pictorial Pattern Recognition*, pages 3–13, Washington, 1968. Thompson.

[48] N. Nandhakumar and J.K. Aggarwal, "The artificial intelligence approach to pattern recognition - a perspective and overview", *Pattern Recognition*, 18(6):383–389, 1985.

[49] N.J. Nilsson, *Principles of Artificial Intelligence*, Morgan Kaufman Publishers, 1980.

[50] D.A. Ortendahl and J.W. Carlson, "Segmentation of magnetic resonance images using fuzzy clustering", Technical report, Radiologic Imaging Laboratory, University of California, San Francisco, 1987.

[51] T. Pavlidis, "Segmentation of pictures and maps through functional approximation", *Computer Graphics and Image Processing*, 1:360–372, 1972.

[52] B.A. Payne and A.W. Toga, "Surface mapping of brain function on 3-D models", *IEEE Computer Graphics and Applications*, 10(5):33–41, 1990 1990.

[53] C.A. Pelizzari, G.T.Y. Chen, D.R. Spelbring, and R.R. Weichselbaum, "Accurate three dimensional registration of CT, PET and mr images of the brain", *Science*, 1987.

[54] T.M. Peters, "Principles and applications of magnetic resonance imaging (MRI) in neurology and neurosurgery", *Journal of Mind and Behavior*, 9(3):241–262, 1988.

[55] T.M. Peters et al., "Stereotactic neurosurgery planning on a pc-based work station", *Journal of Digital Imaging*, 2(2):75–81, May 1989.

[56] B. Pike, T.M. Peters, E. Podgorsak, C. Pla, A. Olivier, and A. de Lotbiniere, "Stereotactic external beam calculations for radiosurgical treatment of brain lesions", *Applied Neurophysiology*, 50:269–273, 1987.

[57] W.K. Pratt, *Digital Image Processing*, Wiley-Interscience, New York, 1978.

[58] R.G. Ramsey, *Neuroradiology with Computed Tomography*, W.B. Saunders Co., Toronto, 1981.

[59] J-F Rivest, *Analyse automatique d'images geologiques: Application de la morphologie mathematique aux images diagraphiques*, PhD thesis, l'Ecole Nationale Supérieure des Mines de Paris, 1992.

[60] C. Romero-Sierra, *Neuroanatomy; A Conceptual Approach*, Churchill Livingstone Inc., New York, 1986.

[61] A. Rosenfeld and A.C. Kak, *Digital Picture Processing*, Academic Press, New York, 1976.

[62] G. Salamon and Y.P. Huang, *Radiologic Anatomy of the Brain*, Springer-Verlag, New York, 1976.

[63] M. Schmitt and F. Preteux, "Un nouvel algorithme en morphologie mathematique: les r, h maxima et r,h minima applications en scanner irm, angiograpie", In *Semaine Internationale de L'image Electronique, Deuxieme Colloque image*, pages 469–475, Nice, April 1986.

[64] H.J. Scudder, "Introduction to computer aided tomography", *procIEEE*, 66(6):628–637, June 1978.

[65] V.M. Spitzer and D.G. Whitlock, "A 3-D database of human anatomy", *Advanced Imaging*, pages 48–49, March 1989.

[66] H.S. Stiehl, "3-D image understanding in radiology", *IEEE Engineering in Medicine and Biology*, 9(4):24–28, December 1990.

[67] M.R. Stytz, G. Frieder, and O. Frieder, "Three dimensional medical imaging: algorithms and computer systems", *ACM Computing Surveys*, 23(4):421–499, December 1991.

[68] D. Sutton, *Radiology and Imaging for Medical Students*, Churchill Livingstone, Edinburgh, 4th edition, 1982.

[69] General Electric Medical Systems, *Image Extract Tool: Release CT1.0, Operators Manual*, General Electric, January 1992.

[70] J. Talairach and P. Tournoux, *Coplanar Stereotactic Atlas of the Human Brain*, Thieme, Springer Verlag Ed., Paris, 1988.

[71] J.K. Udupa, "Interactive segmentation and boundary surface formation for a 3-D digital image", *Computer Graphics Image Processing*, 18(3):213–235, March 1982.

[72] J.K. Udupa, "Display of 3-D information in discrete 3-D scenes produced by computerized tomography", *procIEEE*, 71(3):420–431, March 1983.

[73] D.J. Valentino, J.C. Mazziota, and H.K.Huang, "Mapping brain function to brain anatomy", In *SPIE vol. 914, Medical Imagery II*, pages 445–451, 1988.

[74] M.W. Vannier et al., "Validation of magnetic resonance imaging (MRI) multispectral tissue classification", In *9th International conference on Pattern Recognition*, volume 2, pages 1182–1186, 1988.

[75] M.W. Vannier, J.L. Marsh, and M.H. Gado, "Three-dimensional display of intracranial soft-tissue abnormalities", *American Journal of Neuroradiology*, 4:520–521, 1983.

[76] M.W. Vannier, G.L. Melson, and J. Barbier, "Efficient three-dimensional surface reconstruction from real-time b-scan model sonography", *Journal of Ultrasound in Medicine*, 2(10):72, October 1983.

[77] G.L. Vernazza, S.B. Serpico, and G. Dellepiane, "A knowledge based system for biomedical image processing and recognition", *IEEE Trans. Circuits and Systems*, 34(11):1399–1416, November 1987.

[78] L. Vincent, *Algorithmes morphologiques a base de files d'attente et de lacets. Extension aux graphes*, PhD thesis, l'Ecole Nationale Supérieure des Mines de Paris, 1990.

[79] H.Q. Wang, R.T. Ritchings, and A.C.F. Colchester, "Image understanding system for carotid angiograms", *Image and Vision Computing*, 5(2):79–83, May 1987.

[80] J. Winter, "Automated computer tomography image analysis using contour map topology", *IEEE Trans. Medical Imaging*, 3(4):163–169, December 1984.

[81] T.S. Yoo, U. Neumann, H. Fuchs, S.M. Pizer, T. Cullip, J. Rhoades, and R. Whitaker, "Direct visualization of volume data", *IEEE Computer Graphics and Applications*, pages 63–71, July 1992.

[82] S.W. Zucker, "Region growing, childhood and adolescence", *Computer Graphics and Image Processing*, 5:382–399, 1976.

[83] S.W. Zucker, J.F. Traub, B.J. Grosz, B.W. Lampson, and N.J. Nilsson, "The emerging paradigm of computational vision", *Annual Review of Computer Science*, 2:69–89, 1987.

Appendix A

Segmentation Trial

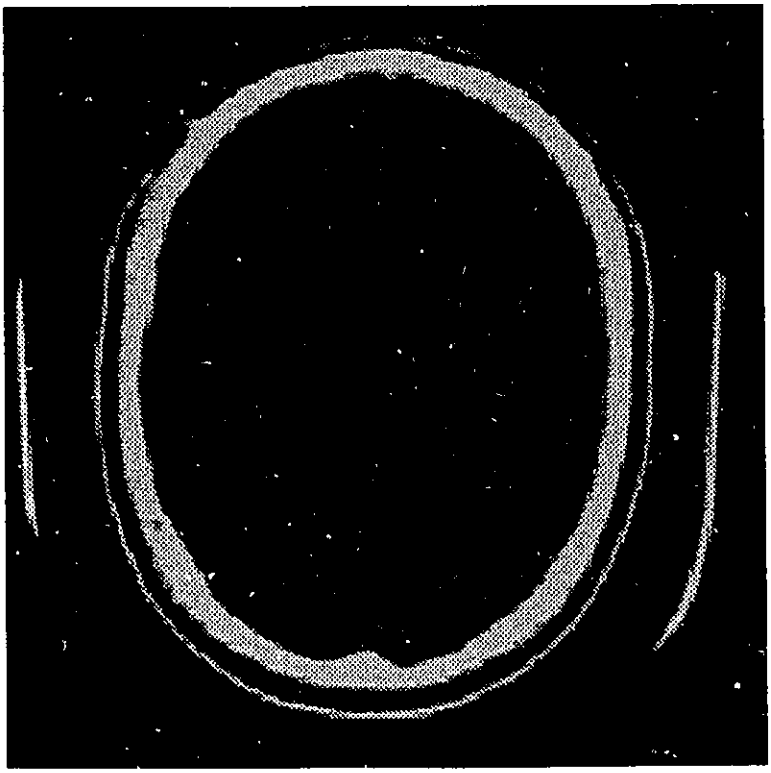

Figure A.1: Image No. 1 in the Segmentation Trial (256 × 256)

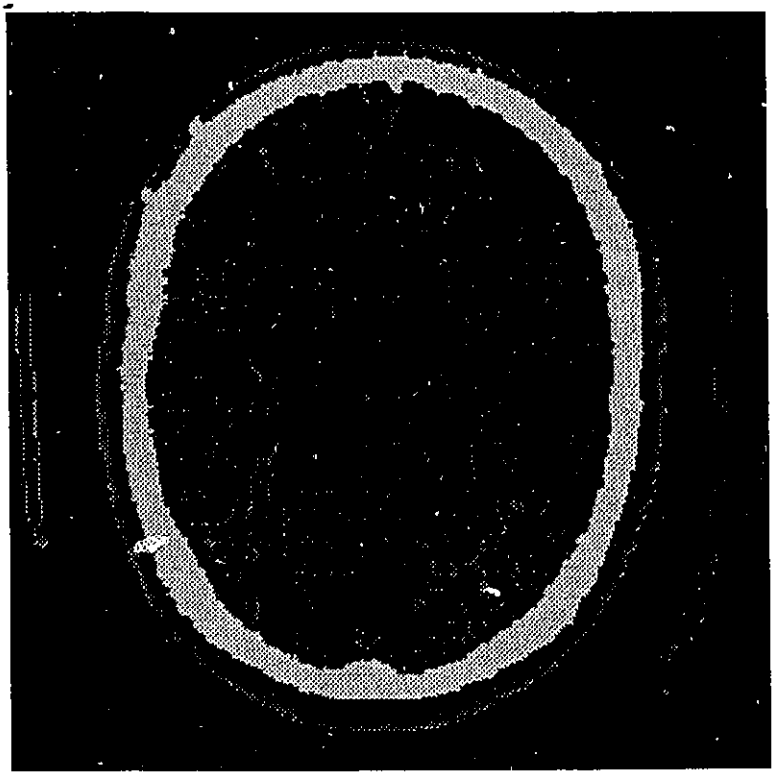

Figure A.2: Segmentation of Image No. 1 using WS and merging the regions

Figure A.3: Segmentation of Image No. 1 using the region grower and merging the regions

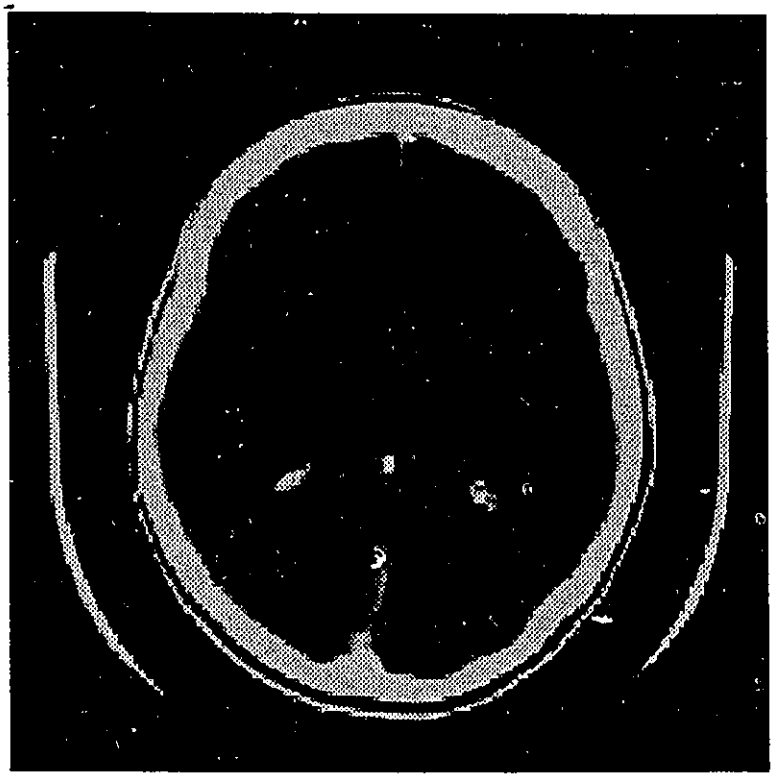

Figure A.4: Image No. 2 in the Segmentation Trial (256 × 256)

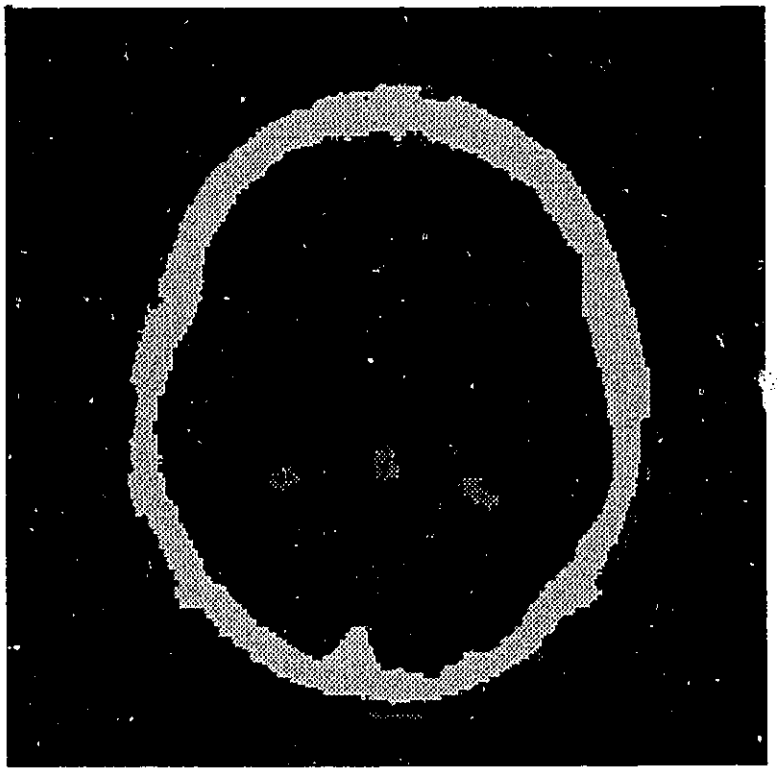

Figure A.5: Segmentation of Image No. 2 using WS and merging the regions

Figure A.6: Segmentation of Image No. 2 using the region grower and merging the regions

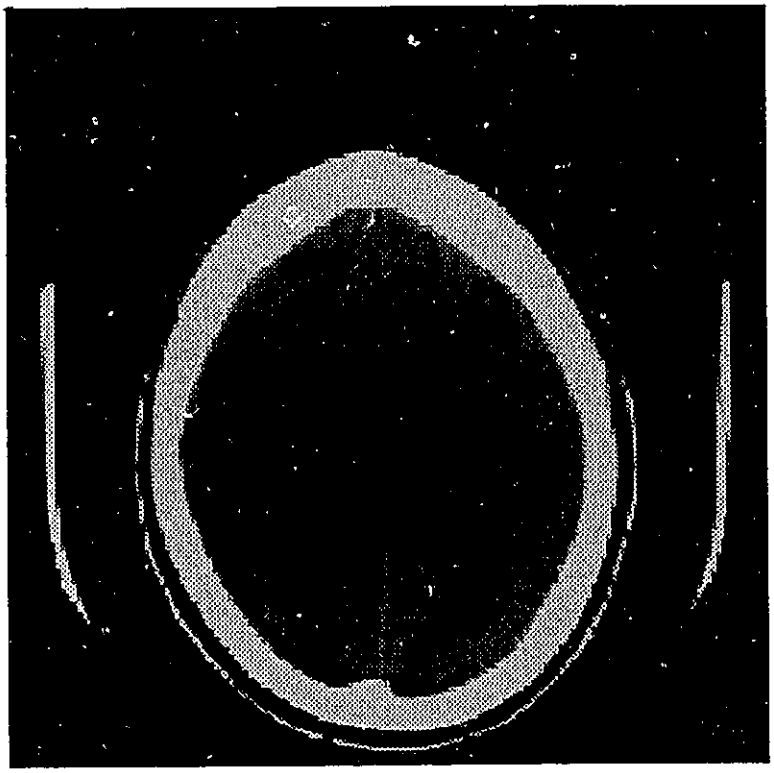

Figure A.7: Image No. 3 in the Segmentation Trial (256 × 256)

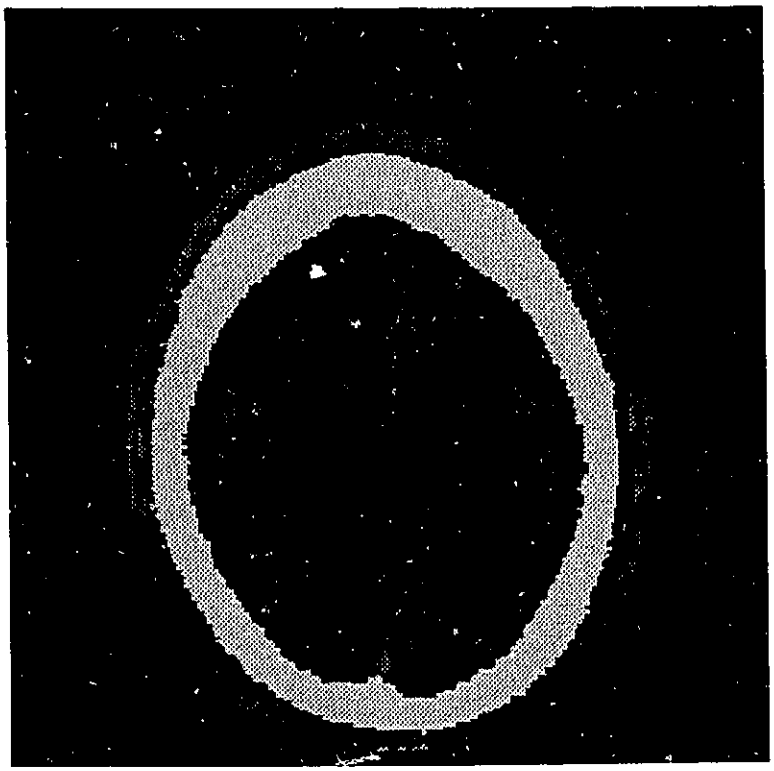

Figure A.8: Segmentation of Image No. 3 using WS and merging the regions

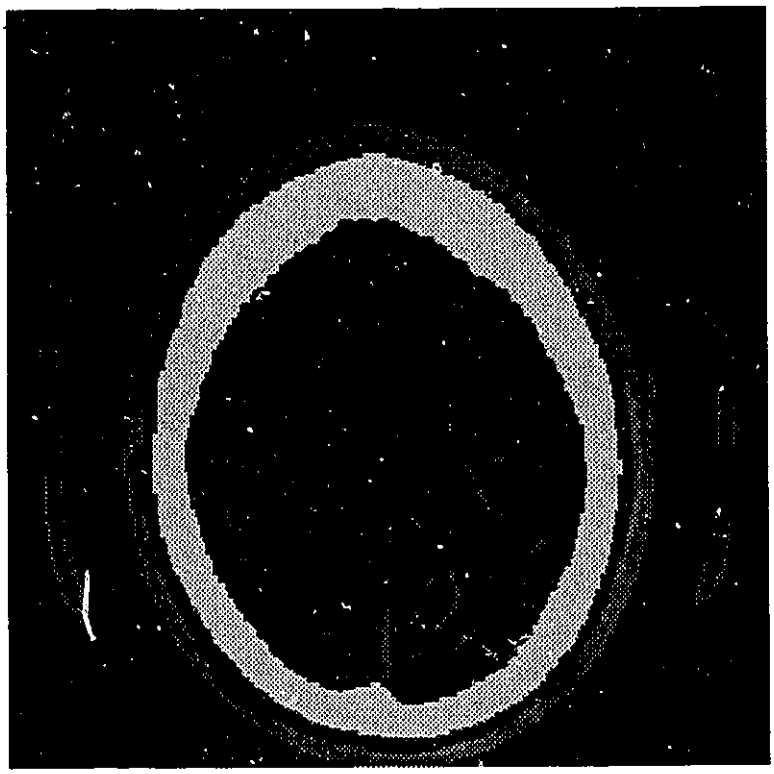

Figure A.9: Segmentation of Image No. 3 using the region grower and merging the regions

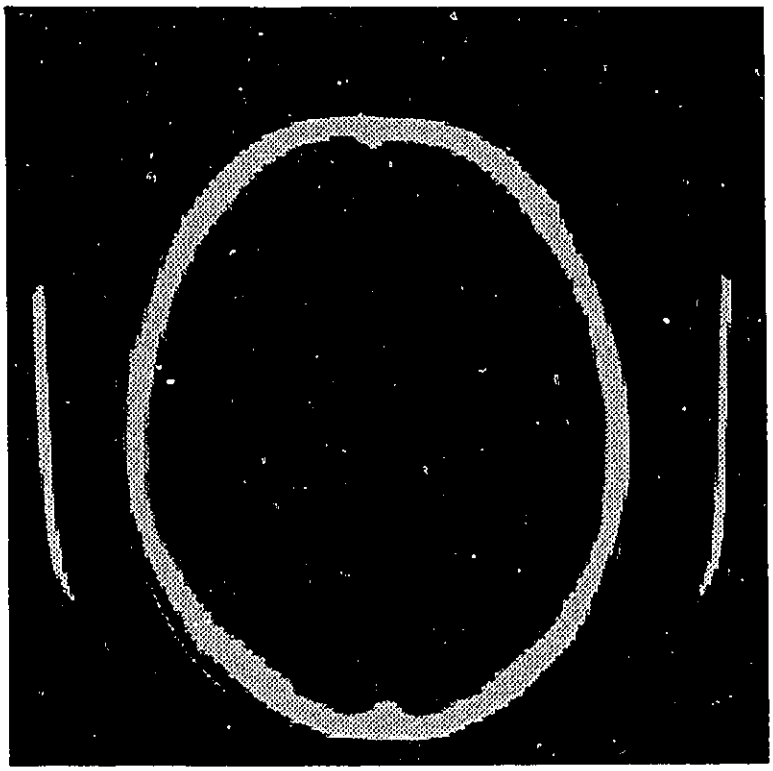

Figure A.10: Image No. 4 in the Segmentation Trial (256 × 256)

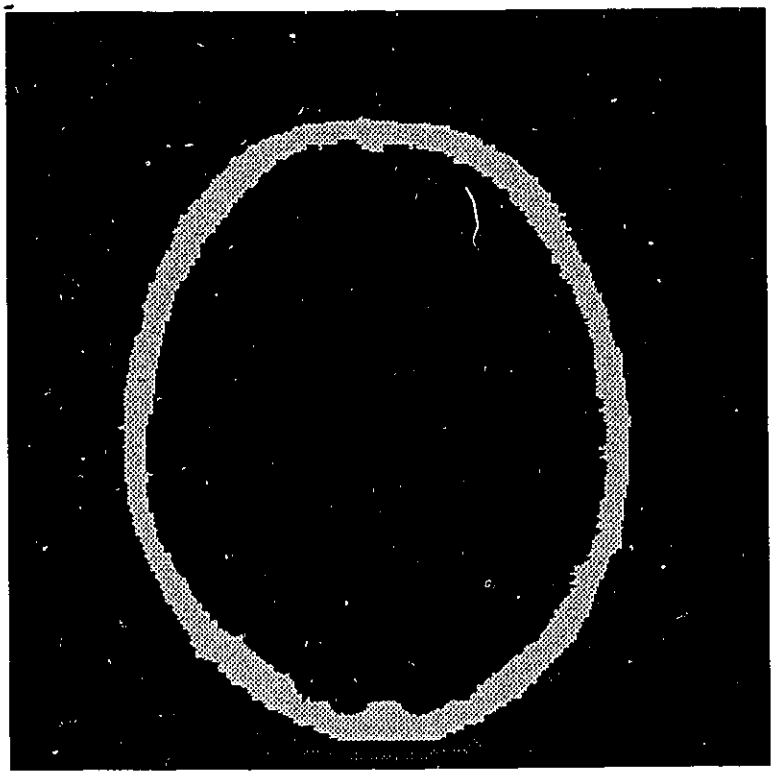

Figure A.11: Segmentation of Image No. 4 using WS and merging the regions

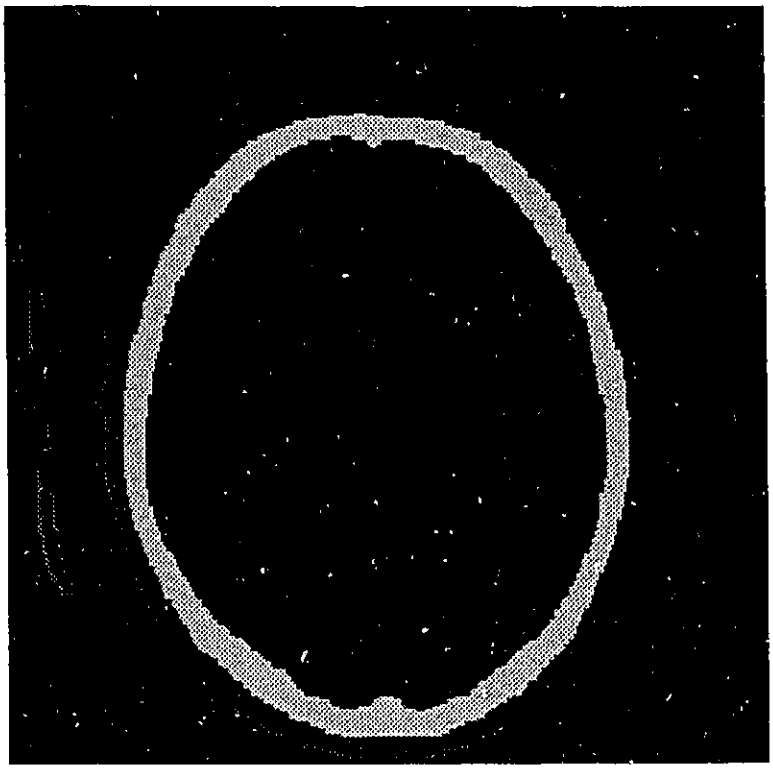

Figure A.12: Segmentation of Image No. 4 using the region grower and merging the regions

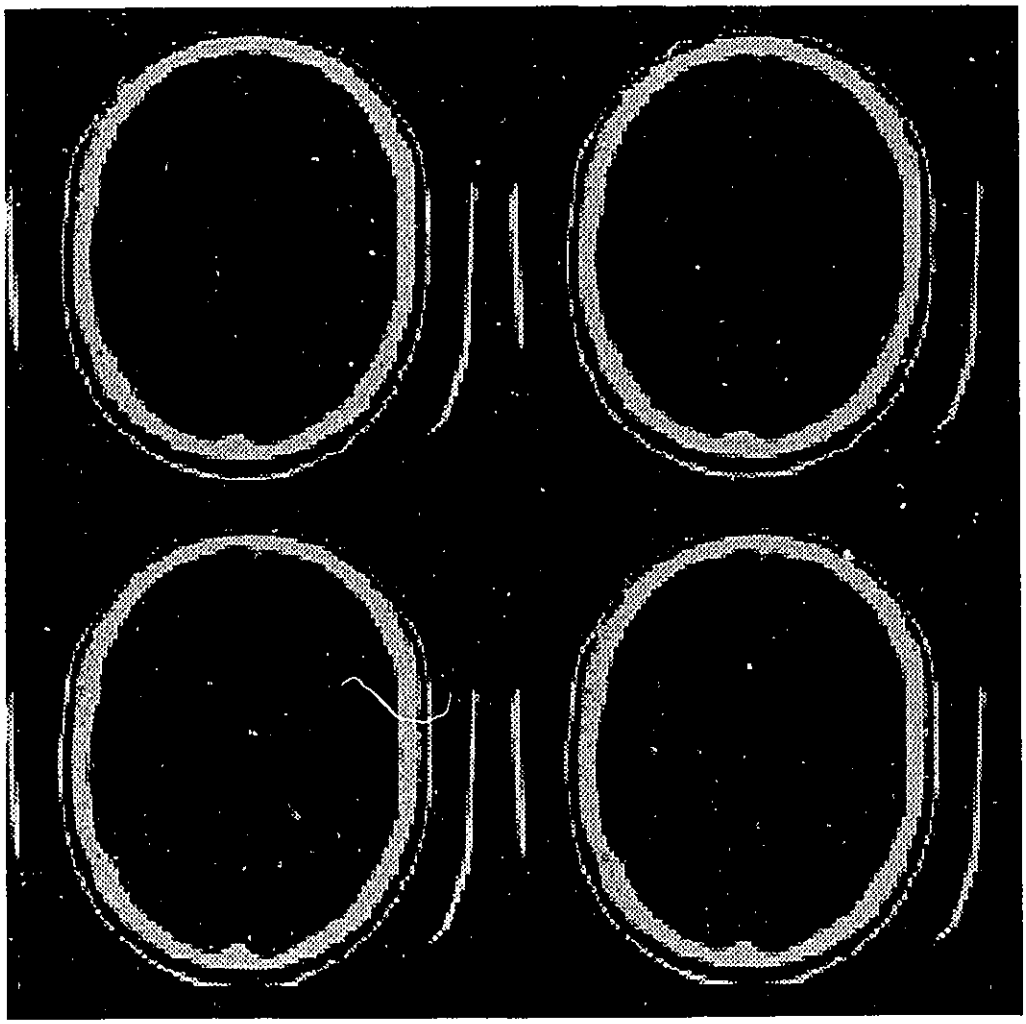

Figure A.13: Image No. 5 in the Segmentation Trial

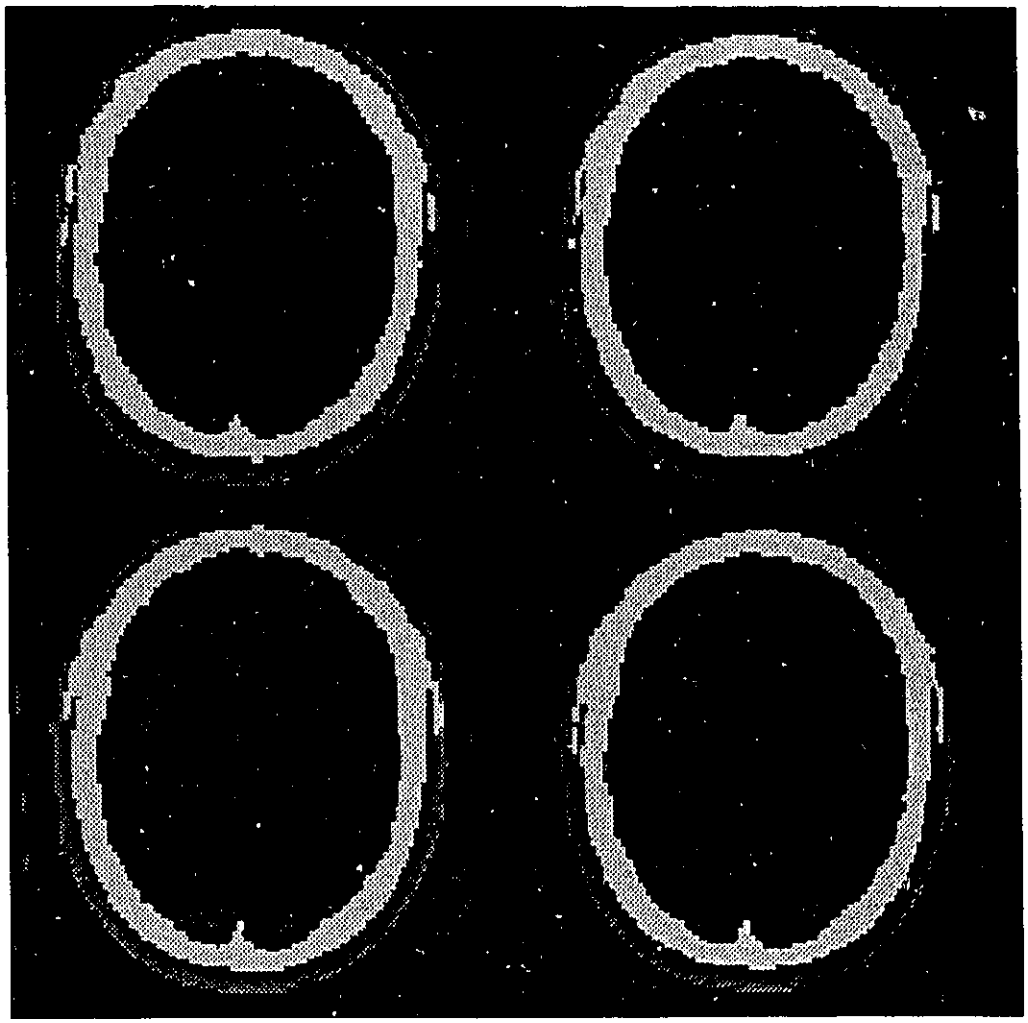

Figure A.14: Results of the region grower on a 3-D image (Image No. 5) ($128 \times 128 \times 4$)

Appendix B

Merging Trials

B.1 Phase 1 of the Merging Trials

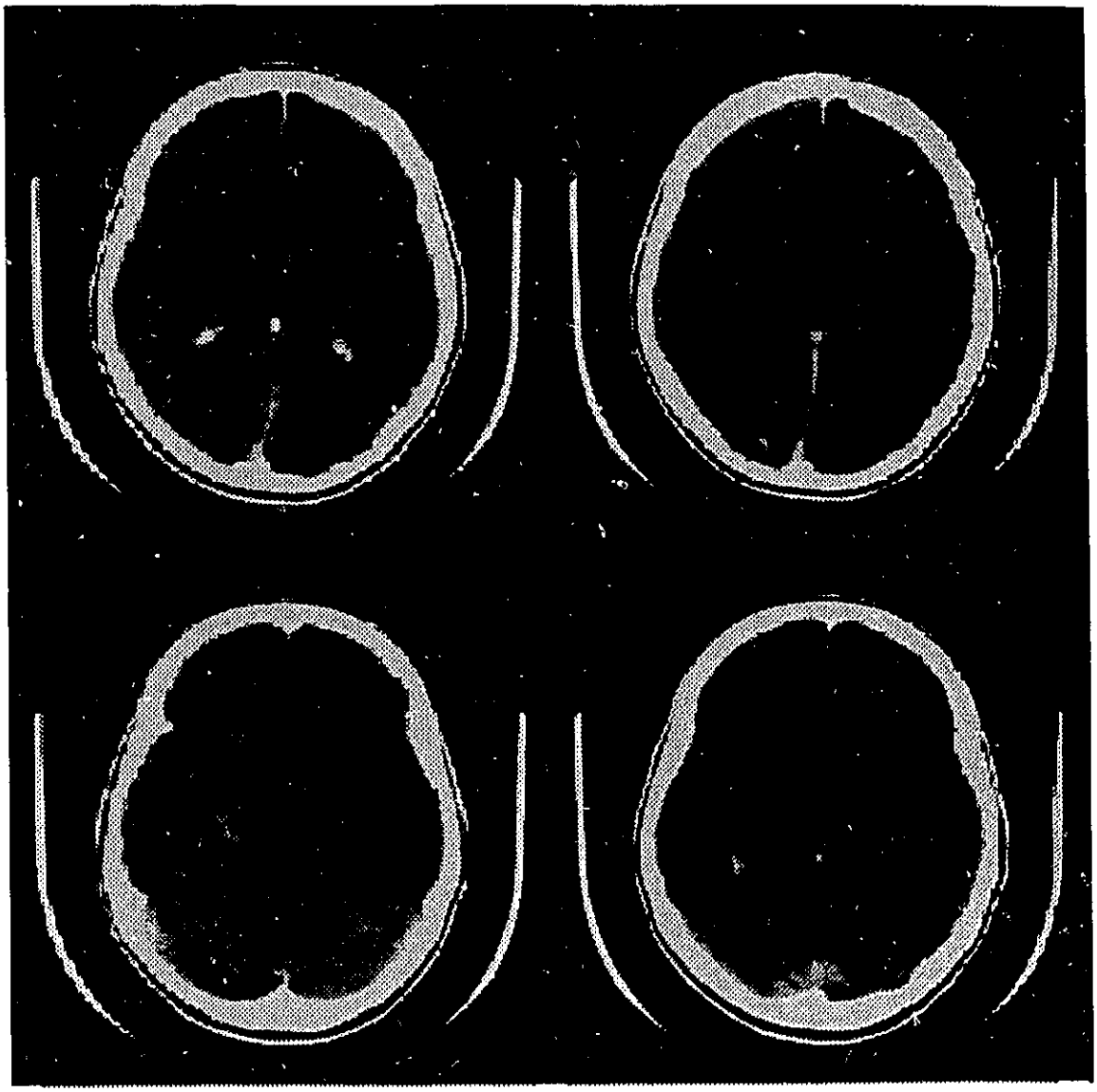

Figure B.1: Test Image for Trial No. 1

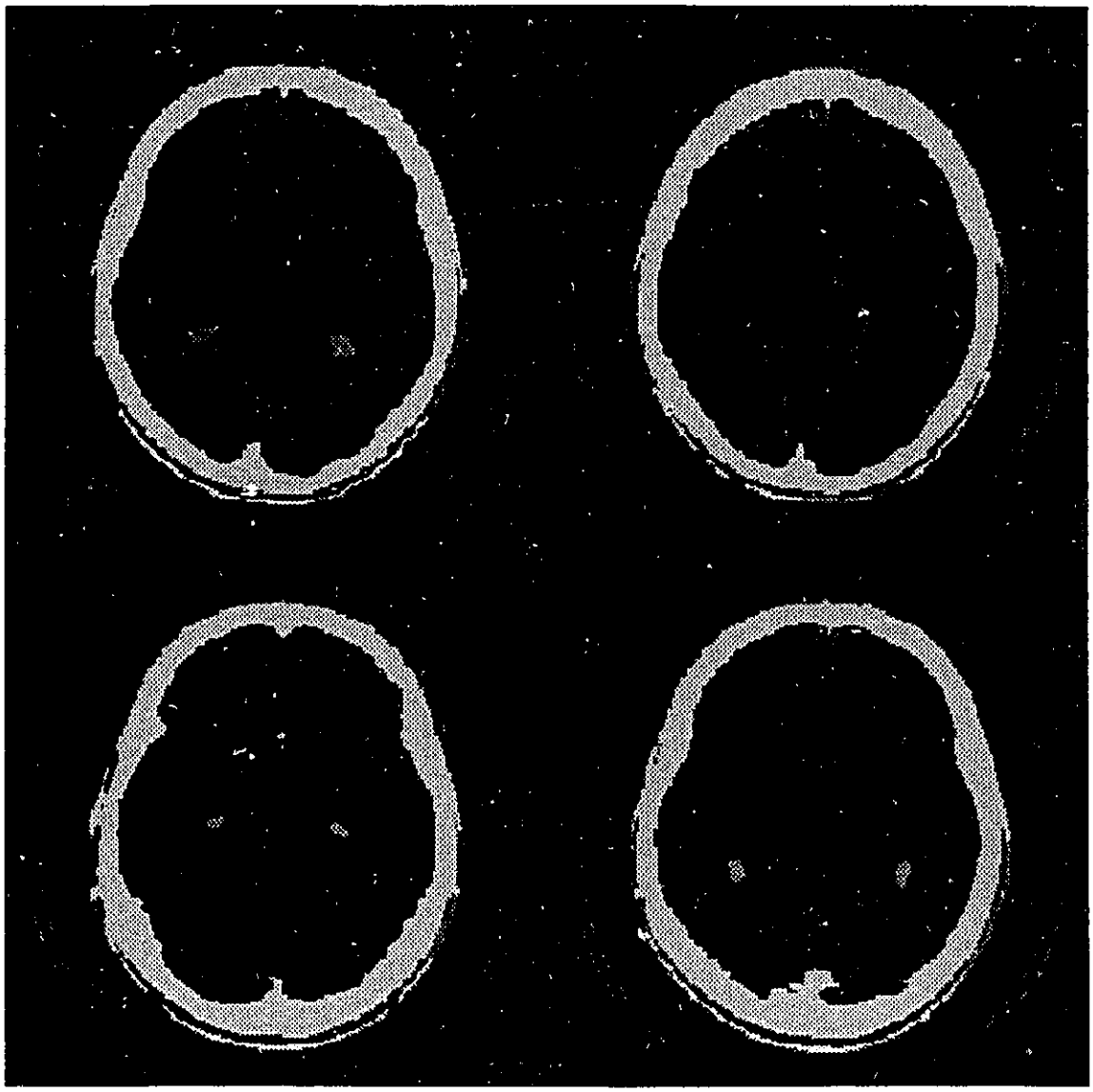

Figure B.2: The Hand-made Individual Atlas for the First Phase (from Image No. 1)

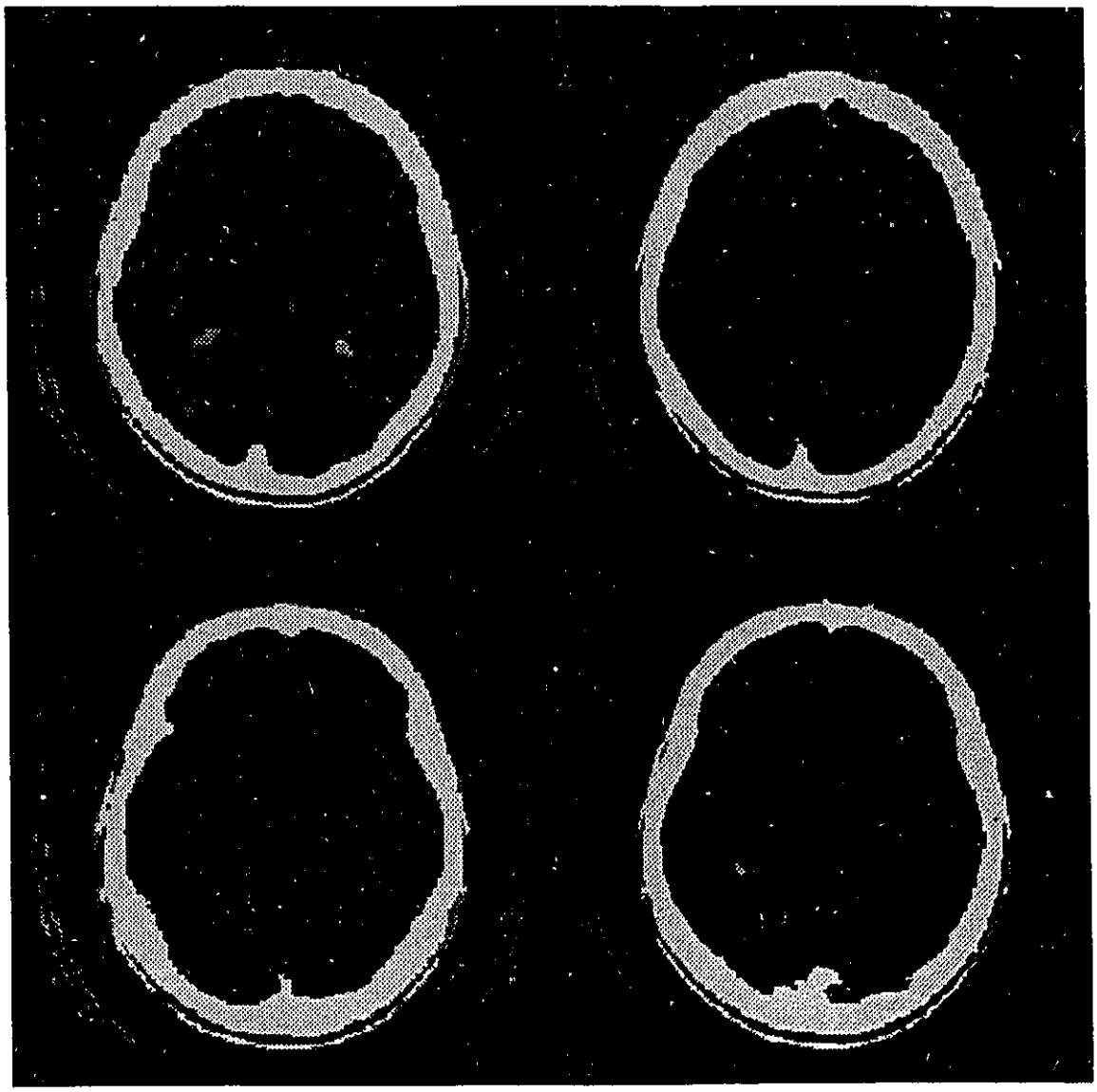

Figure B.3: Results of Trial No. 1

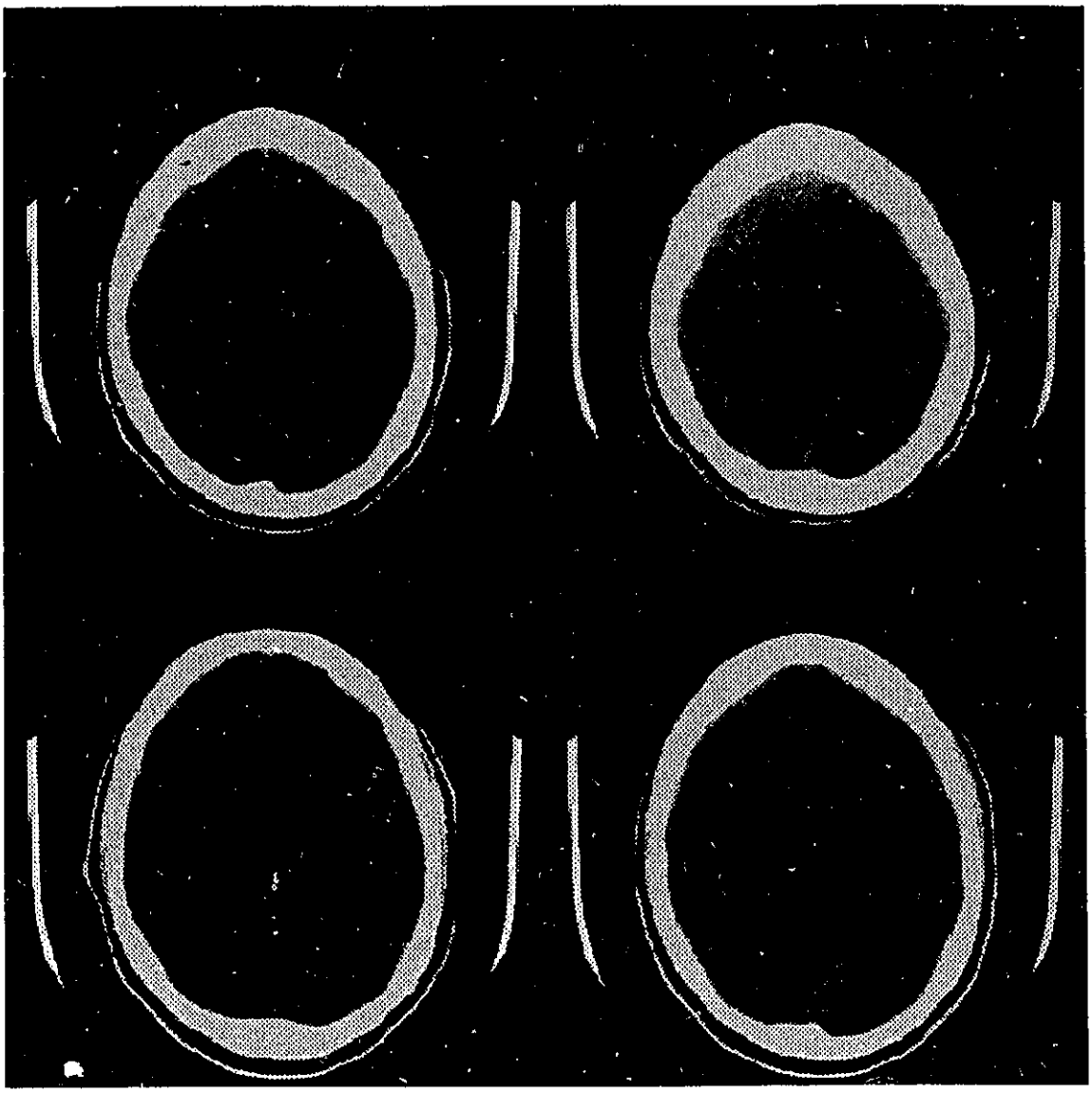

Figure B.4: Test Image for Trial No. 2

Figure B.5: Results of Trial No. 2

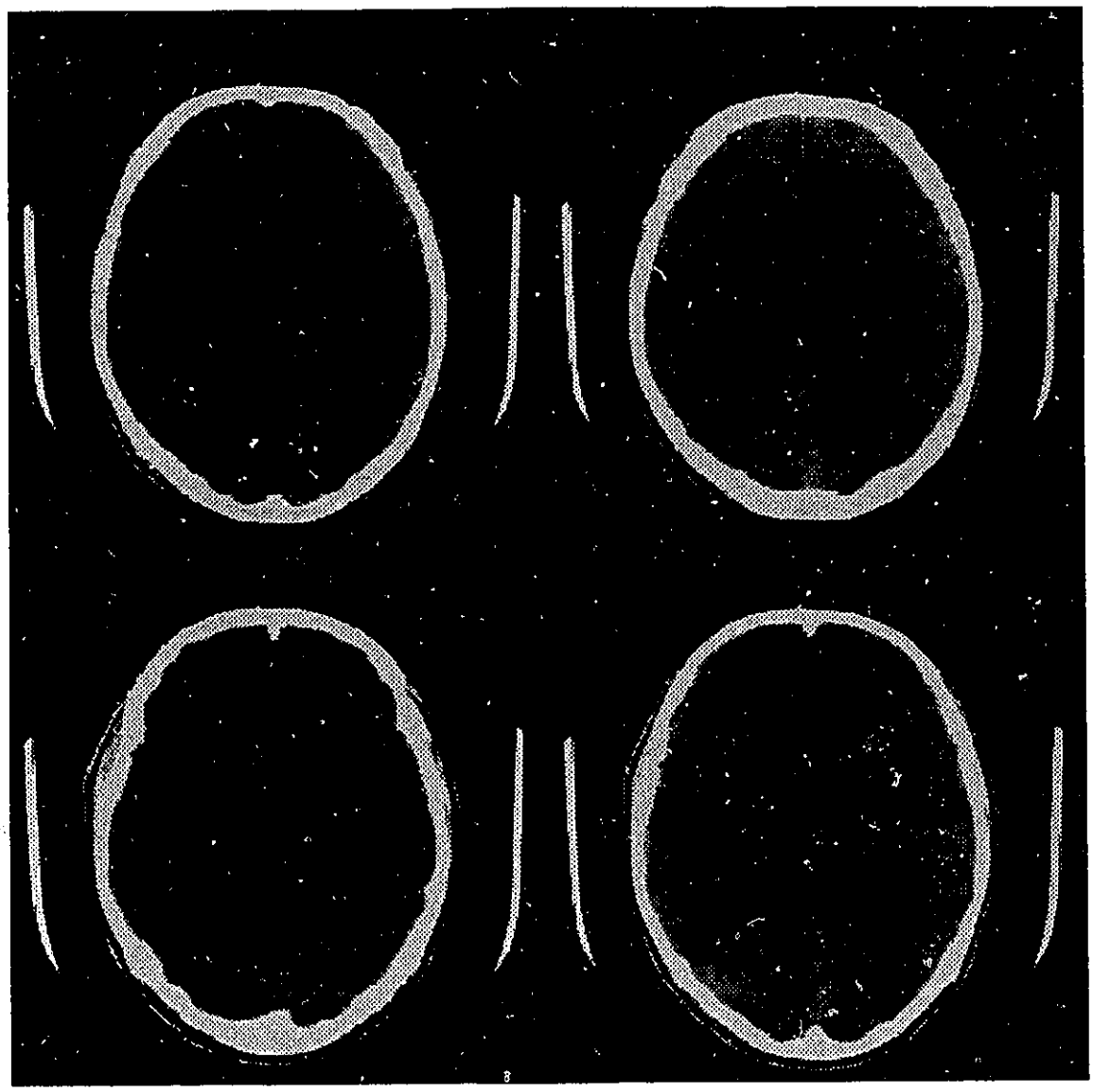

Figure B.6: Test Image for Trial No. 3

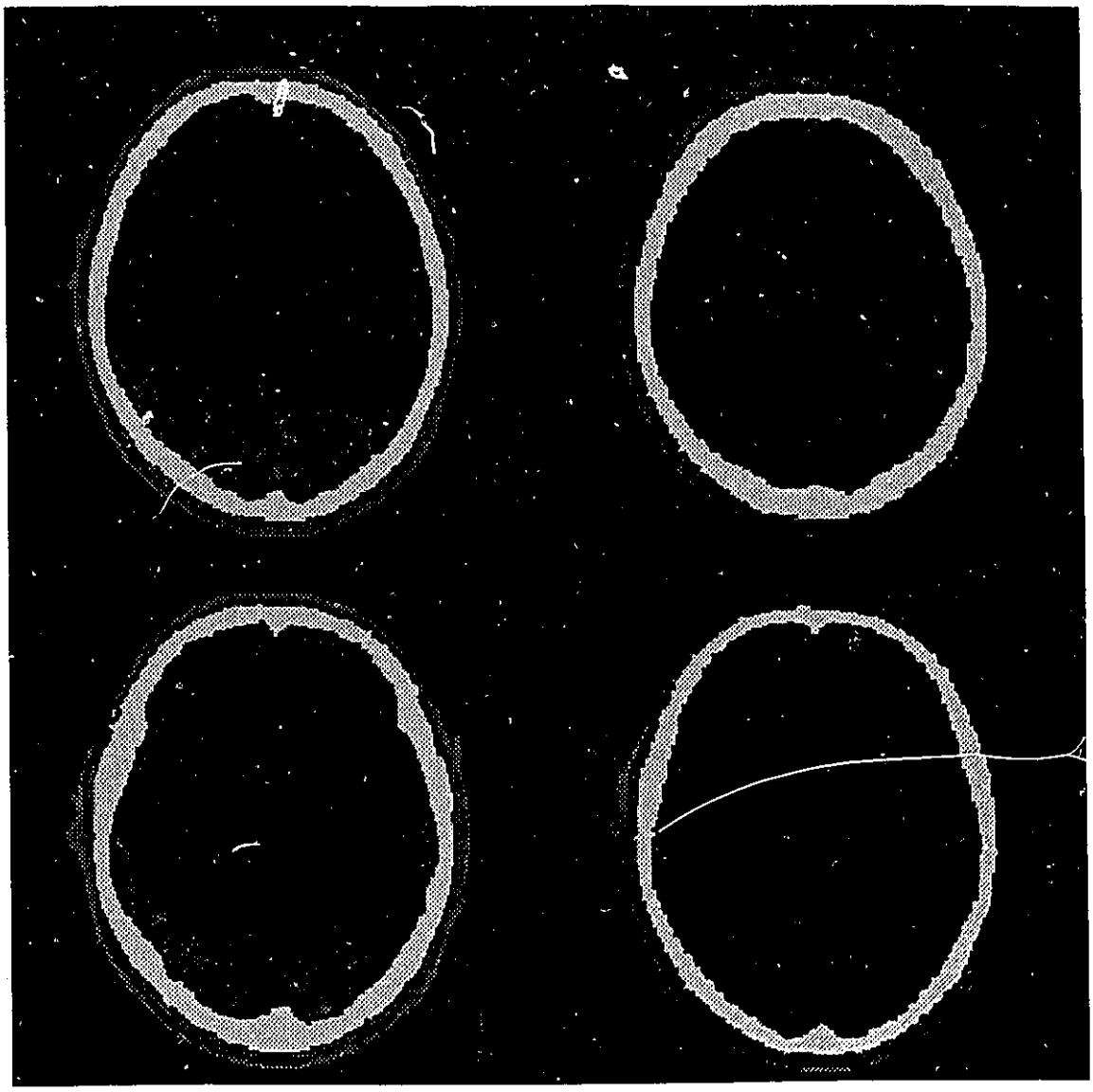

Figure B.7: Results of Trial No. 3

Figure B.8: Test Image for Trial No. 4

Figure B.9: Results of Trial No. 4

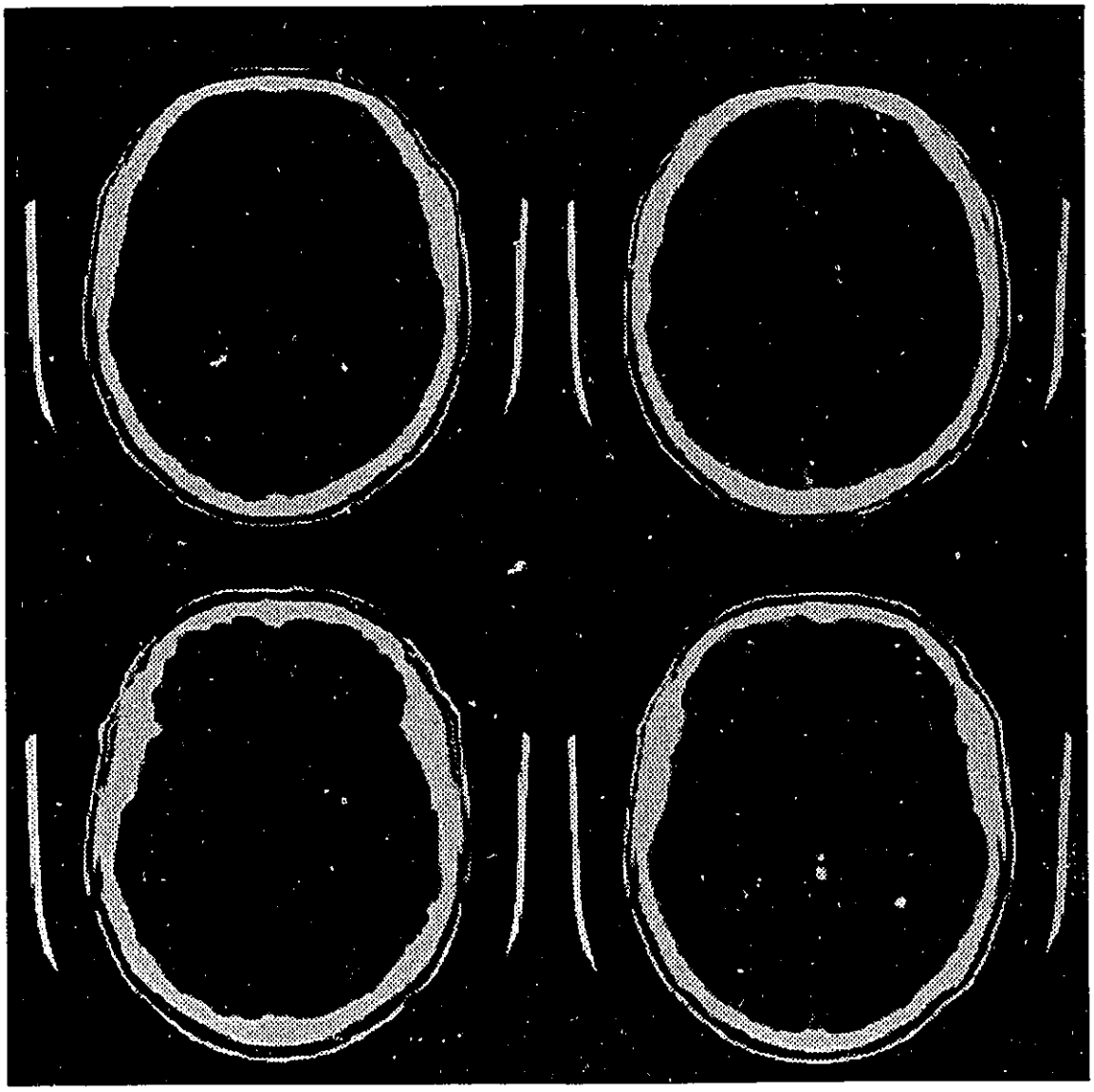

Figure B.10: Test Image for Trial No. 5

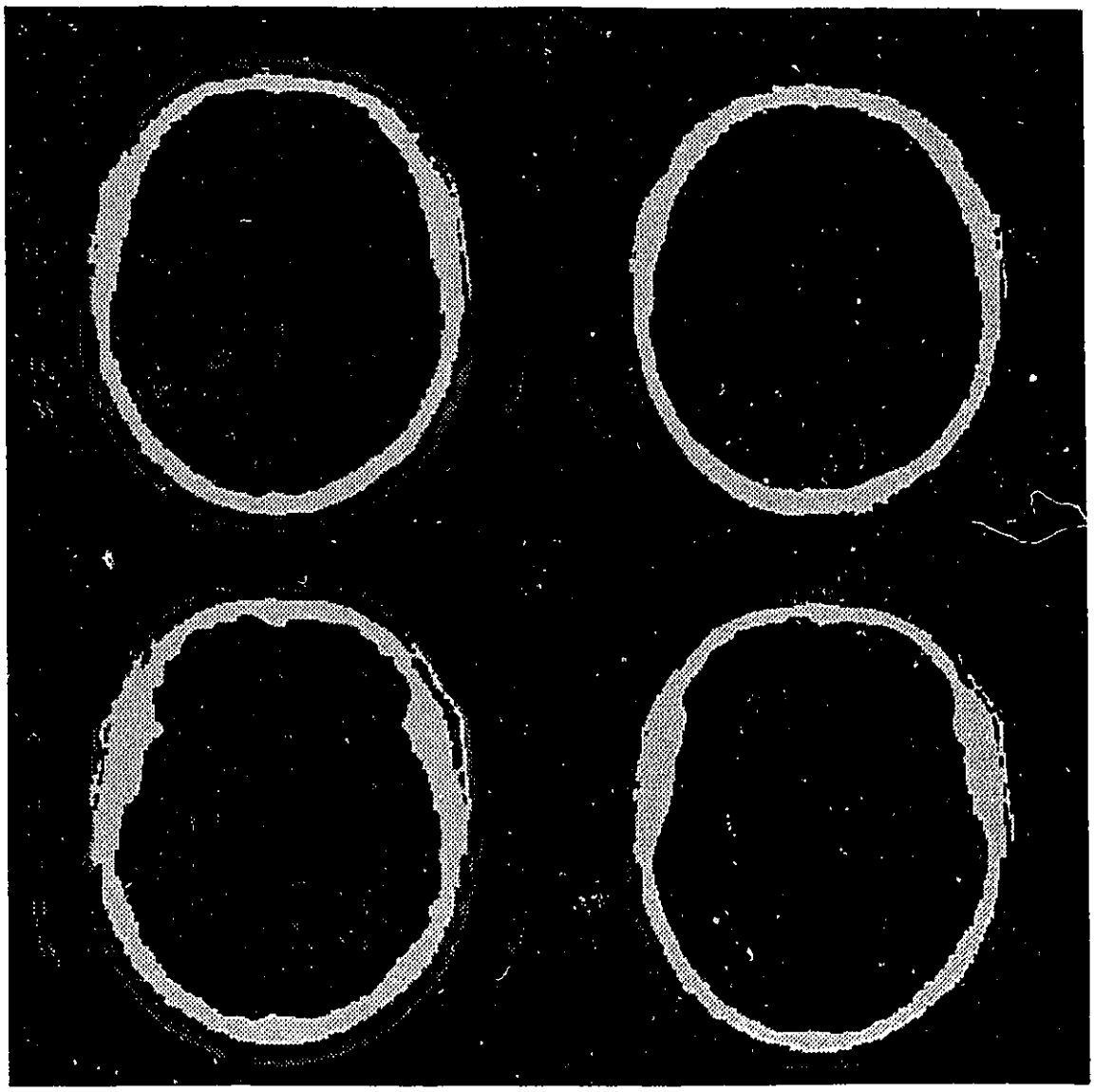

Figure B.11: Results of Trial No. 5

B.2 Phase 2 of the Merging Trials

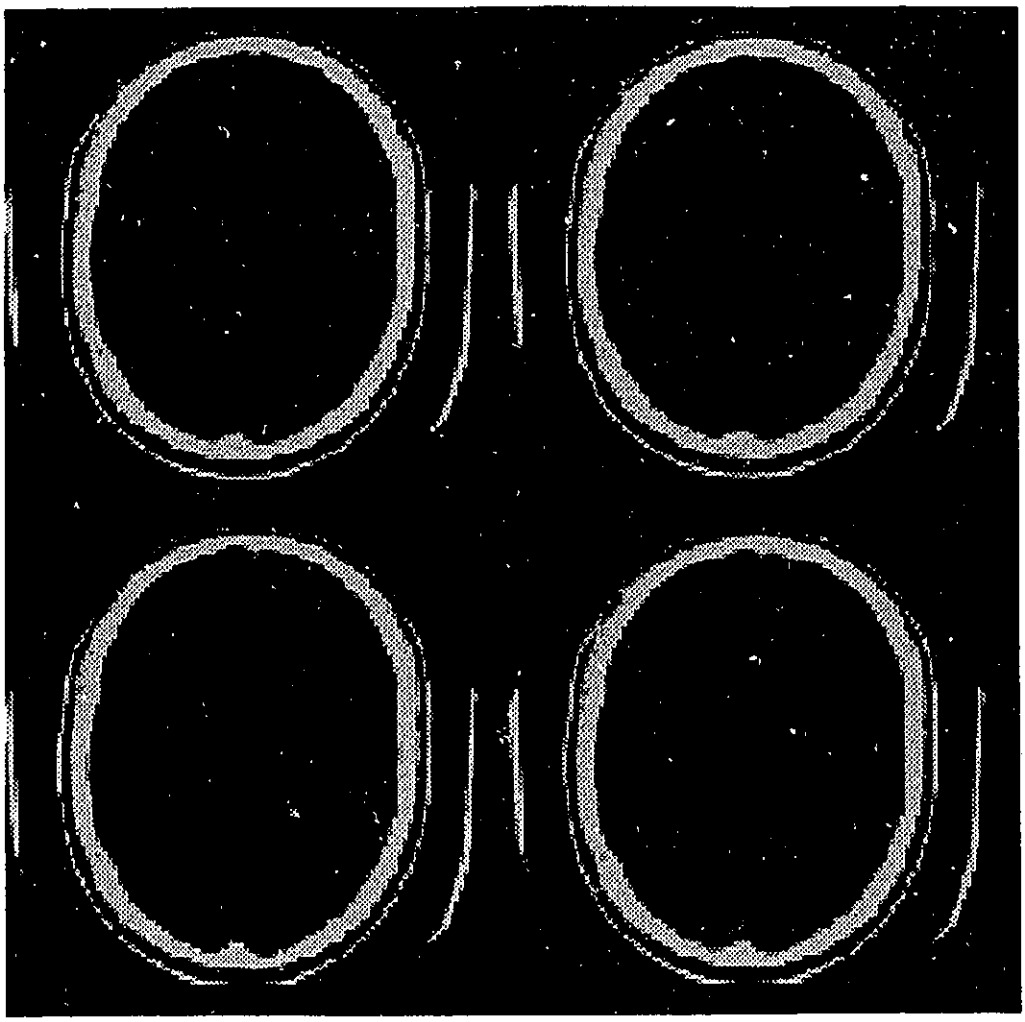

Figure B.12: Test Image for Trial No. 6

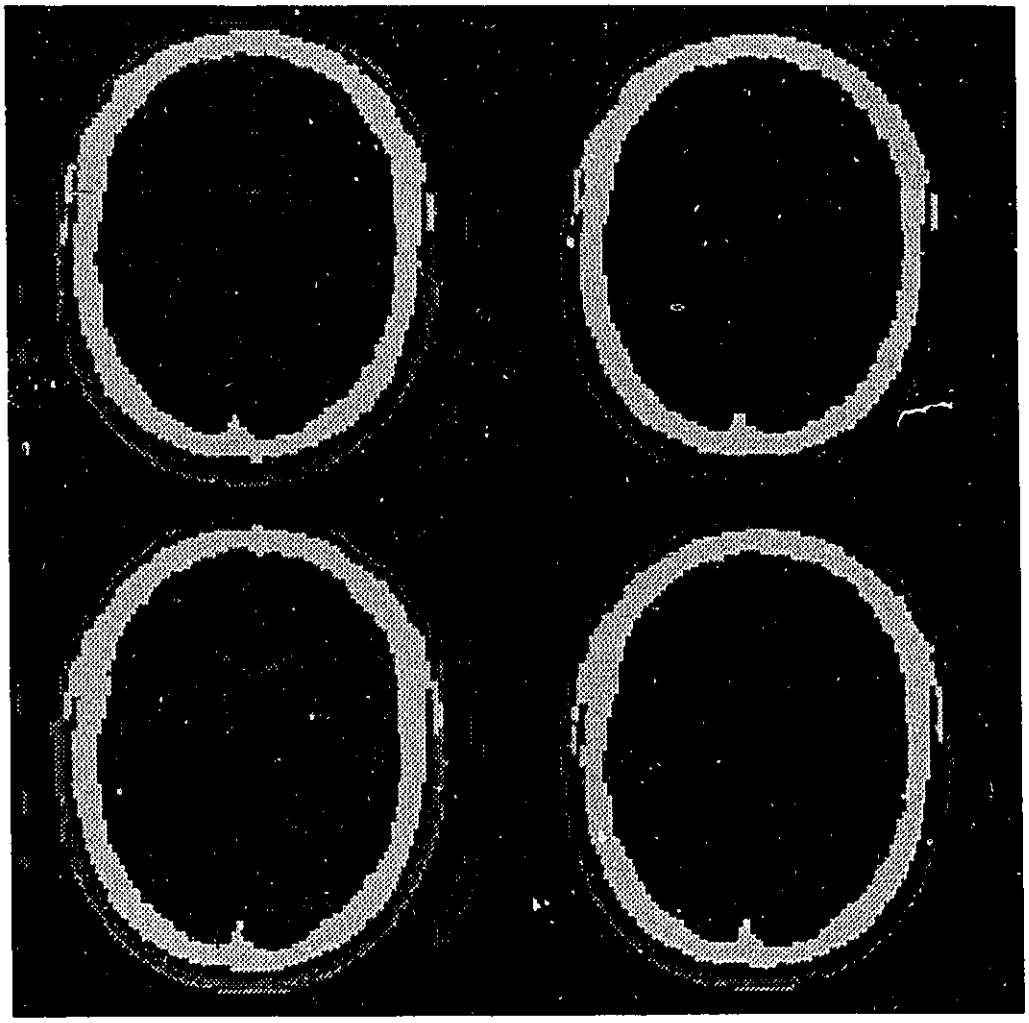

Figure B.13: The Hand-made Individual Atlas for the Second Phase (from Image No. 6)

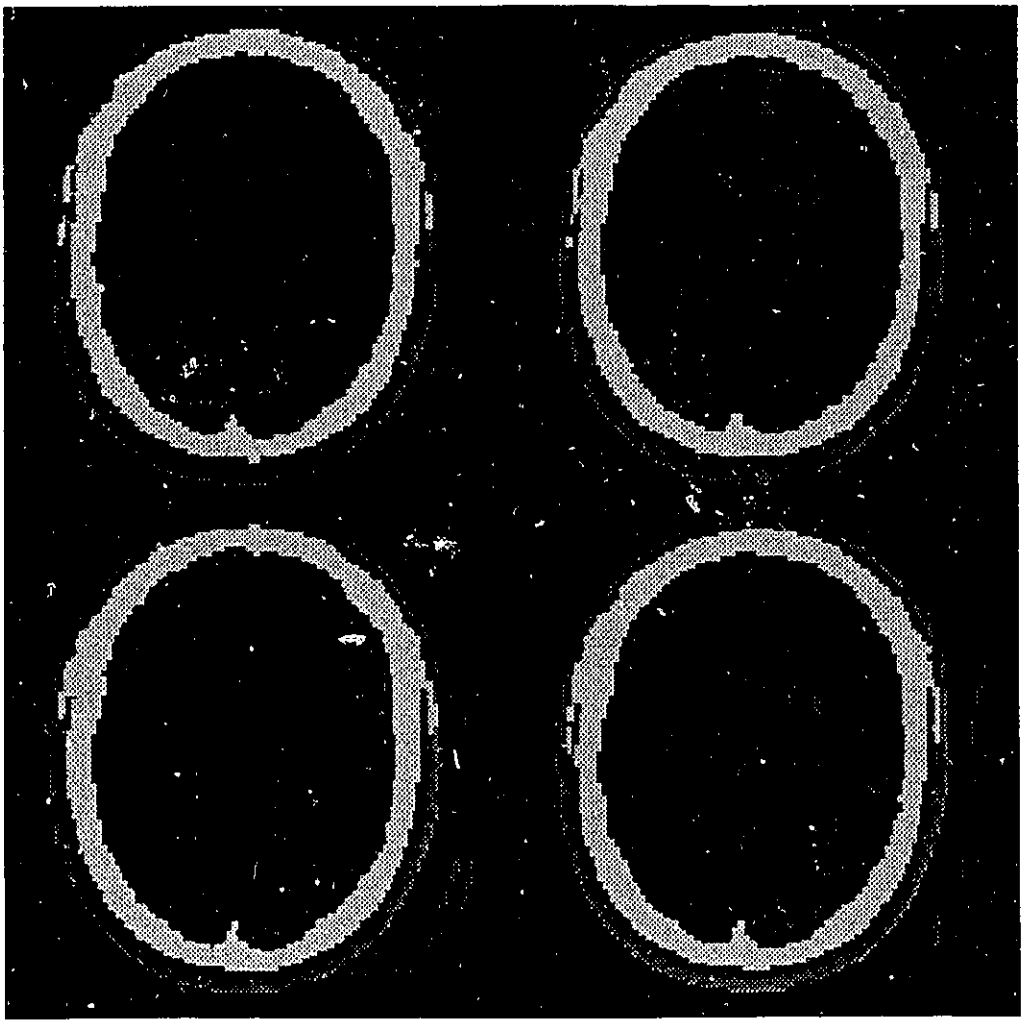

Figure B.14: Results of Trial No. 6

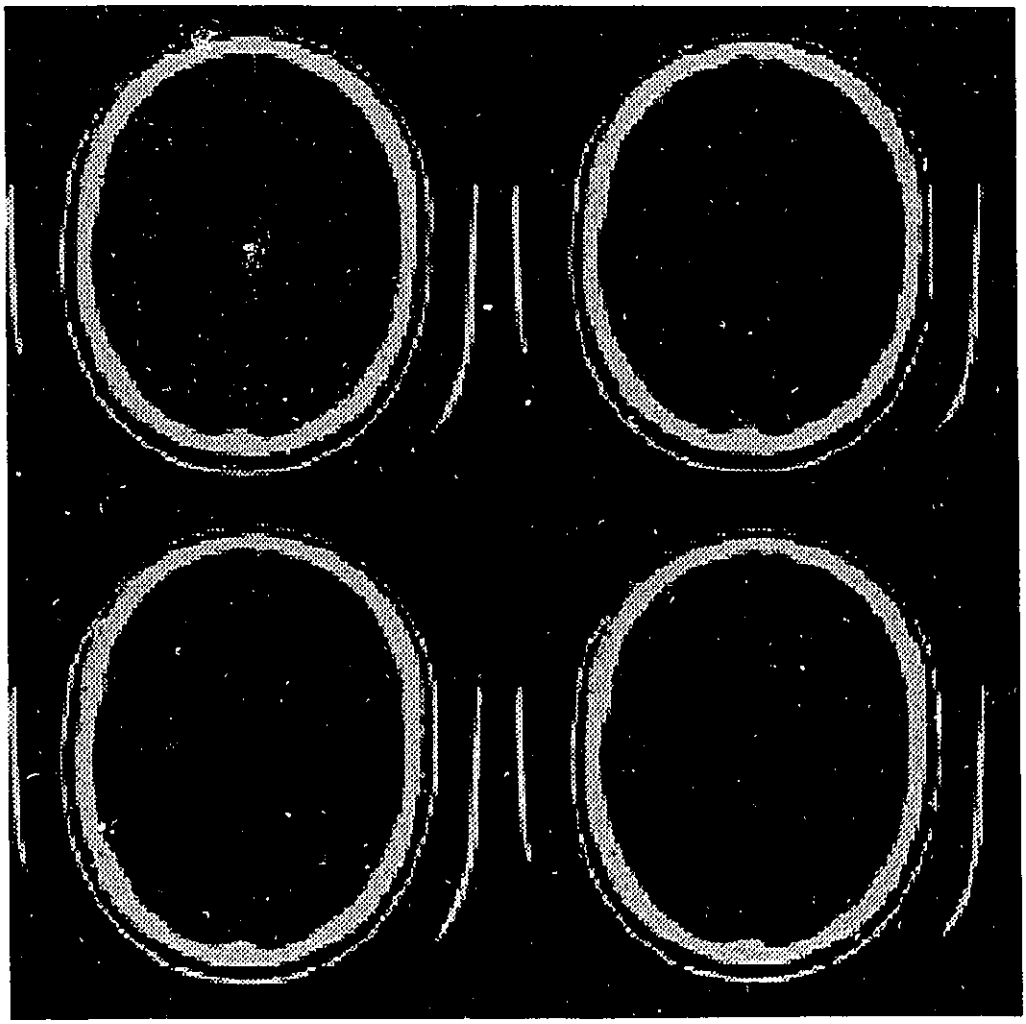

Figure B.15: Test Image for Trial No. 7

Figure B.16: Results of Trial No. 7

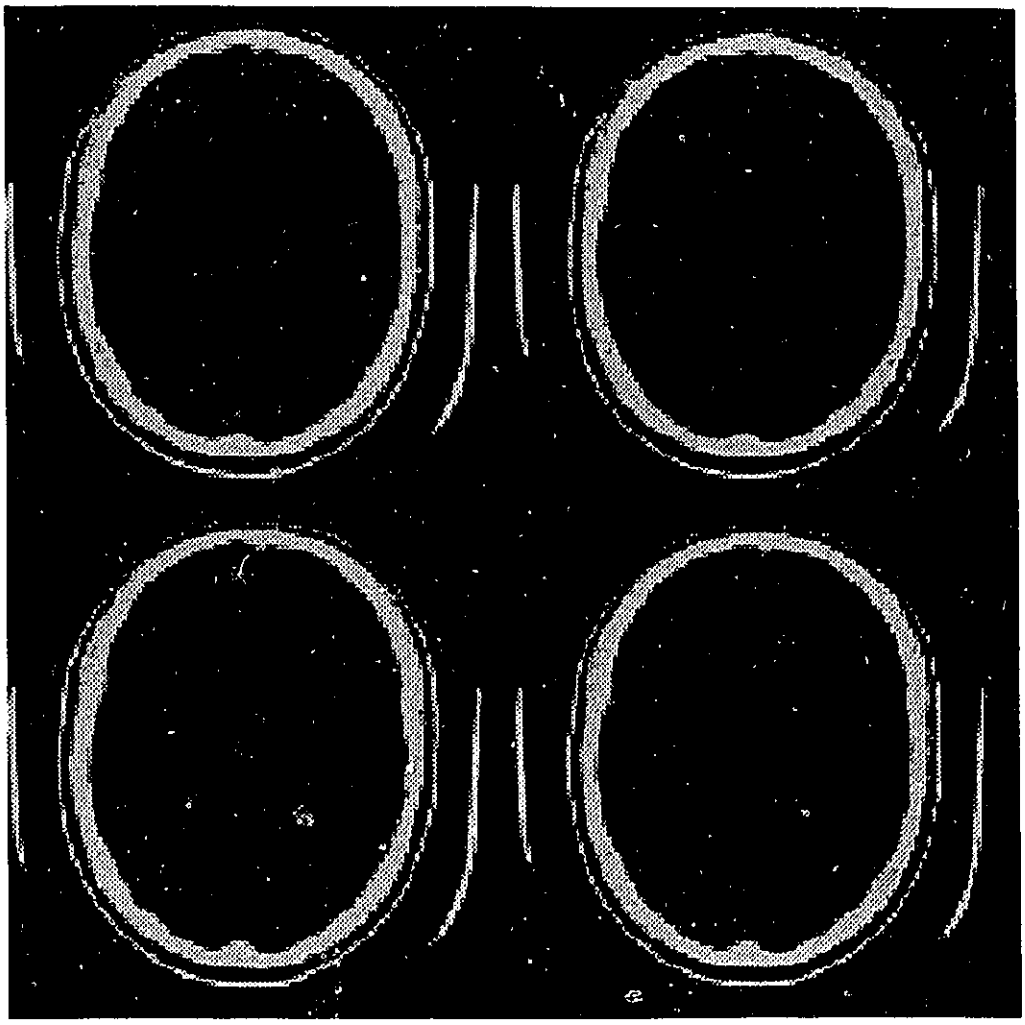

Figure B.17: Test Image for Trial No. 8

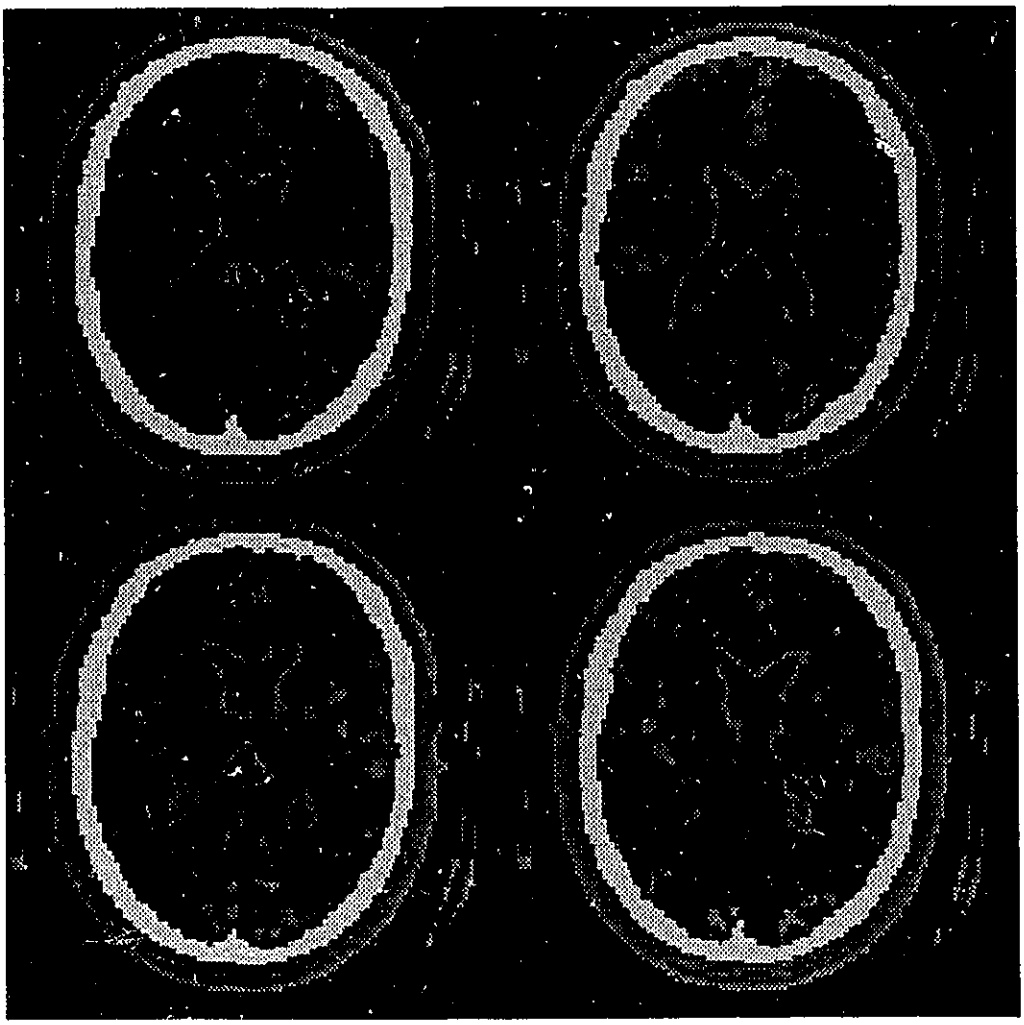

Figure B.18: Results of Trial No. 8

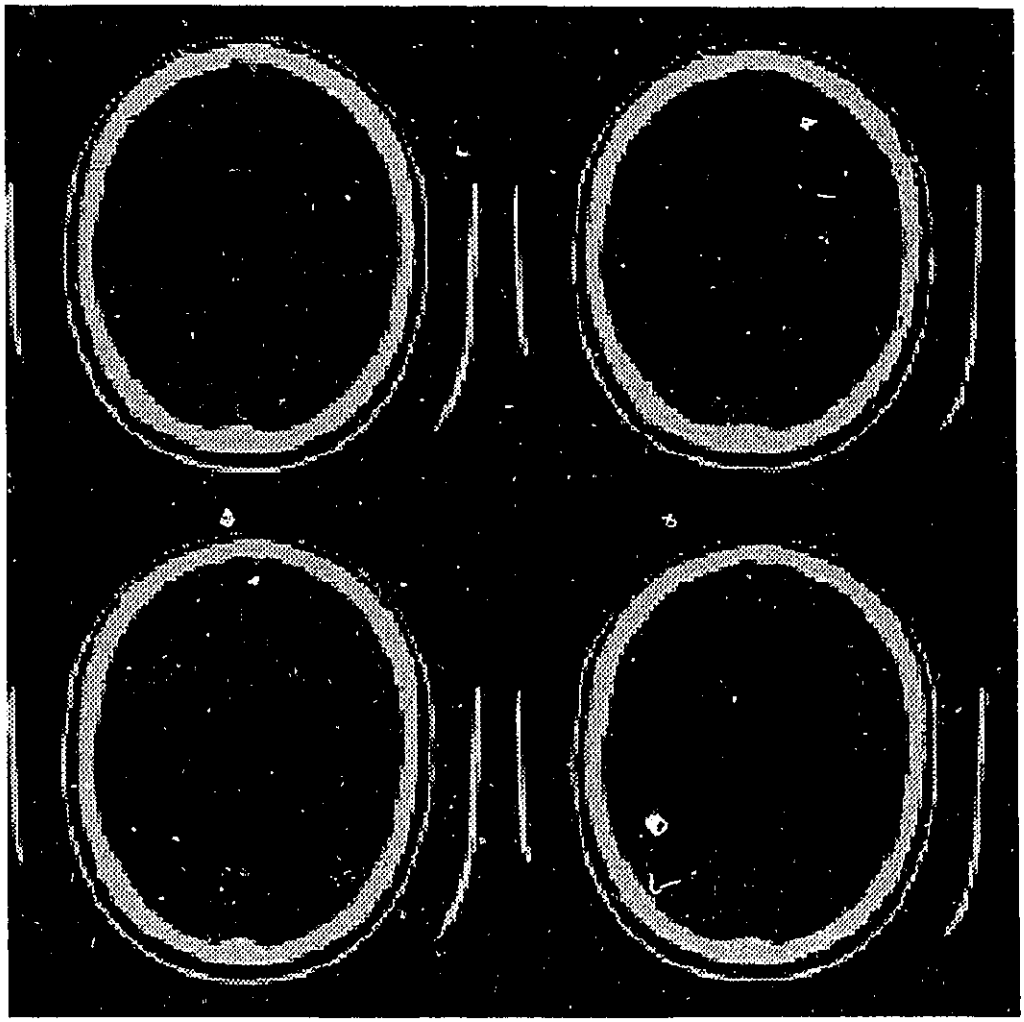

Figure B.19: Test Image for Trial No. 9

Figure B.20: Results of Trial No. 9

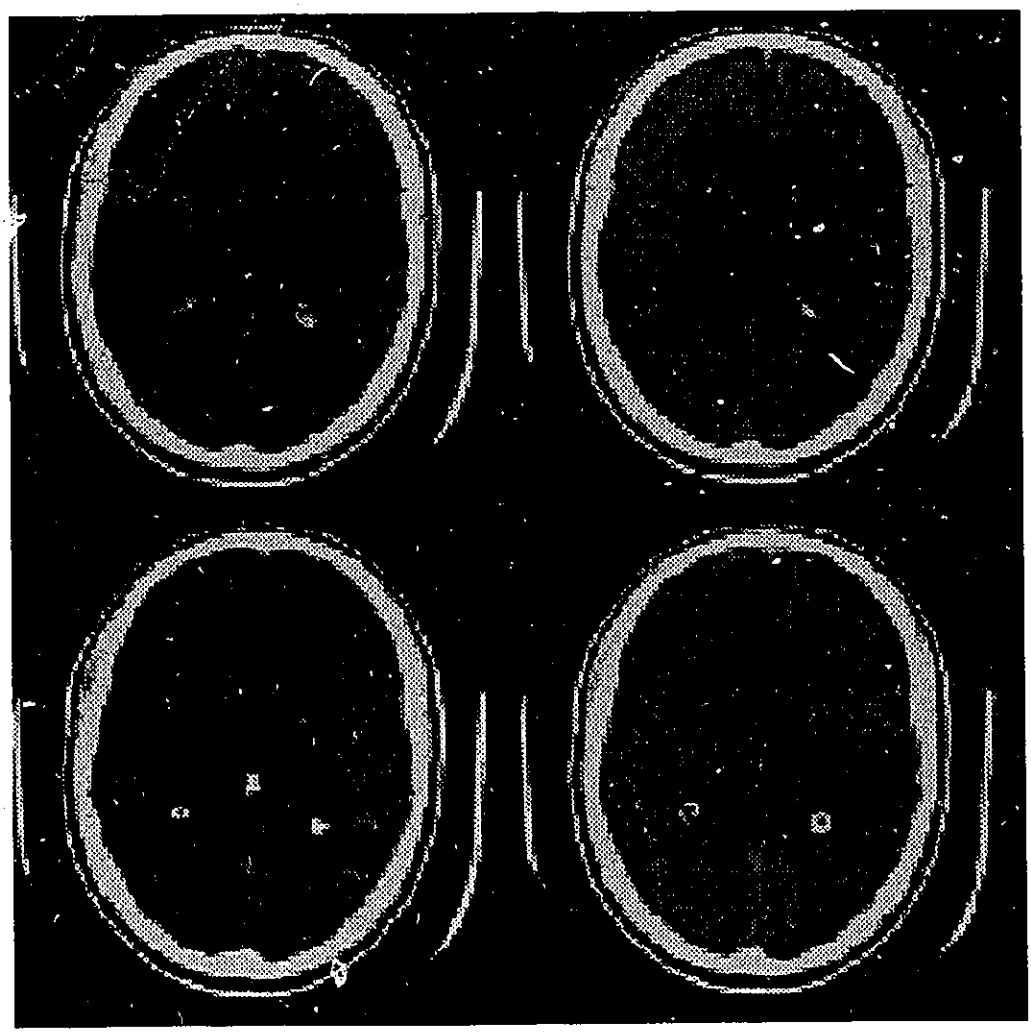

Figure B.21: Test Image for Trial No. 10

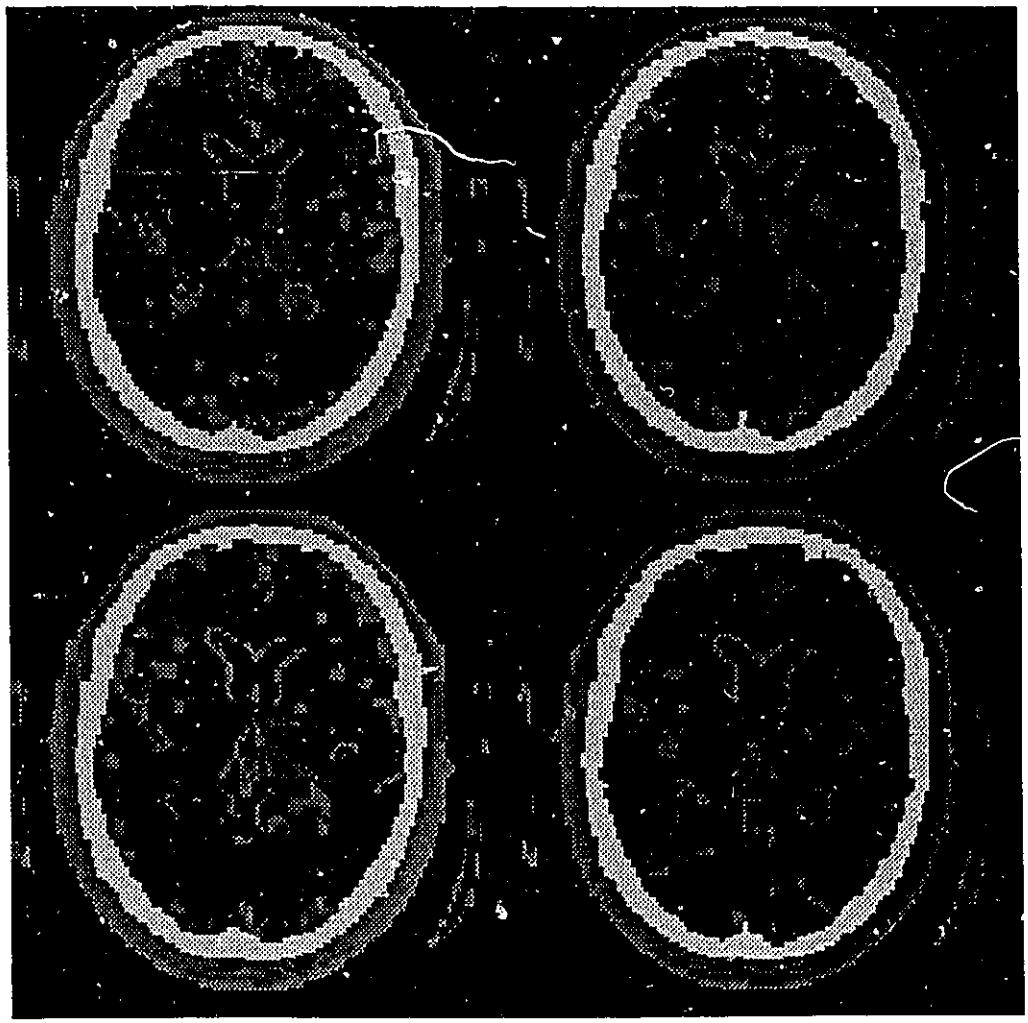

Figure B.22: Results of Trial No. 10