A computer vision system for tracking proliferating cells

Kim M. Wheeler

Department of Electrical Engineering

McGill University

Montréal

August, 1992

A thesis submitted to the Faculty of Graduate Studies and Research

in partial fulfillment of the requirements for the degree of

Master of Engineering

Abstract

A computer vision system which tracks and analyses living cells and their offspring from sequential two-dimensional images is described. The cells are segmented using a method which combines global thresholding, component labeling and noise filtering. The *shapes* of the cells are then analyzed using a fast skeleton algorithm, and the *states* of the cells are determined as the cells change shape and move. This vision system recognizes the occurrence of *cell mitosis* and then tracks the offspring by evaluating the overall cell shape and position to determine the cell state. Both the cell *shape* and cell *state* are represented as fuzzy sets. The cell *state* takes into account previous shape and state knowledge as the mitotic process is described by a specific series of shape changes. This *Mitosis Recognition System* (MRS) has been developed for cell physiology research in the study of cell locomotory characteristics evolving over several generations.

Résumé

Un système de vision par ordinateur pour l'analyse de cellules vivantes est décrit. Ce système permet, à partir d'une séries d'images en deux dimensions, d'analyser et de suivre le développement de cellules et de leur progéniture. Une méthode sophistiquée, combinant seuillage global, étiquetage de composantes et filtrage de bruit, est utilisée pour la segmentation des cellules. Les *formes* de cellules sont analysées en utilisant un algorithme de squelettisation rapide et leurs *états* sont déterminés à mesure que les cellules se modifient et se déplacent. Ce système de vision reconnait l'avènement d'une *division cellulaire* et ensuite suit la progéniture en évaluant la forme générale et la position d'une cellule afin de déterminer son état. La *forme* et l'*état* d'une cellule sont tous deux représentés à l'aide d'ensembles flous. L'état d'une cellule est déterminé en tenant compte de sa forme précédente ainsi que la connaissance du fait qu'une division cellulaire procède par une séries de changements de forme bien spécifique. Ce système: *Mitosis Recognition System* (MRS) est développé pour la recherche en physiologie cellulaire dans l'étude des caractéristiques de locomotion cellulaire, évoluant sur plusieurs générations.

Acknowledgements

I would like to thank several people. Firstly, I thank my thesis supervisor, Professor Martin Levine for his support and guidance. I extend thanks to David Gauthier for his tremendous technical guidance, support and his encouragement throughout my research. Thanks to Peter Noble for not only providing the cell cultures, but also for the overall motivation of the research from the biomedical standpoint: the advancement of the understanding of cell motion, shape, and proliferation for medical research.

Thanks to Frédéric Leymarie whose previous work in cell shape and skeletonization and his constant advice provided valuable assistance in my work. Thanks to Kenong Wu for his continual interest in the cell tracking work and for his assistance with the cell segmentation and preprocessing sections of the research.

Thanks to Prof. Levine, David Gauthier, and Frédéric Leymarie for reviewing selected chapters of the thesis, and thanks to Gerard Blais for helping in the translation of the abstract into French.

I am grateful to all of my friends, colleagues, and professors at McRCIM for their guidance, and for making my research term at McGill enjoyable. Finally, thanks to Dennis Whyte for his assistance and personal encouragement he provided during my years at McGill.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Cell biology is a field of research that can benefit considerably by advances in computer vision and artificial intelligence (AI) techniques. Primitive methods which require tremendous human effort are still being used by microbiologists for data analysis. Great improvements in data collection, statistical analysis, and experimental control can be achieved by combining computer vision and AI principles with microscopic imaging experiments. An increased number of data and statistics can be gathered with a greater amount of consistency if computer vision and AI principles are combined with image acquisition.

Likewise, the field of computer vision can be advanced by exploring the problems of cell segmentation, representation, and tracking. How is it possible that trained microbiologists can so easily distinguish cells from their background environment? By carefully watching moving cells, they can quickly determine which cells are undergoing division and which cells are about to collide. Tracking individual cells and their offspring is quite simple for experienced microbiologists, however, it is a tedious and sometimes difficult task for them to track many cells *and* monitor their behaviour simultaneously.

Cell research would benefit considerably from automated techniques such as the Mitosis Recognition System (MRS) discussed in this thesis. The MRS is a computer vision tool developed as part of the more general

automated cell tracking system called TRACES (TRacking CEll System) [18]. TRACES is a real-time system for the automatic tracking and imaging of cells as they move across a microscope's field-of-view. The stage of the microscope is adjusted automatically in real-time by feedback from image processing results of the digitized microscopic scenes. After manual initialization, cell tracking proceeds without human input or manual adjustment.

TRACES provides a means to simultaneously study shape and cell locomotion. Hence, this tracking system is very suitable for studying shape changes of cells during cell division (or mitosis) in conjunction with studying the tracks of their offspring. The MRS tracks generations of cells (within TRACES) by automatically monitoring their states of mitosis using computer vision techniques.

Cell tracking refers to the monitoring of a living cell's position over time. Techniques range from using particular mediums in which cells leave visible tracks [37] to real-time tracking and position recording by computer. The former method has the disadvantage that a cell's physiological behaviour may be unpredictably affected by the unnatural medium required to record its movements. The latter method involves the use of videotape for capturing the traces of the cells within a single field-of-view. Because tracking is not automated, there is no adjustment of the microscope field when the cells move out of view. Tracking of individual cells is done later, at the convenience of the researcher by replaying the videotape and manually noting the cell paths. Automatic Computer vision systems are also sometimes used to track individual cells from pre-recorded images (see for eg. [14] and [63]).

Pre-recorded methods of cell tracking have the disadvantage of being limited to the field of view of the stationary camera used during the ex-

periments. In these methods researchers must decide between recording a large number of cells at a very low resolution (or magnification) from recording fewer cells at a high magnification. High magnification is fine for shape analysis of cells which do not move very much, but for purposes of studying cell locomotion, high magnification (and hence a small field of view) often results in cells moving *out* of the field of view, prematurely ending the experiment. Low magnification, however, is appropriate for cell tracking, but cell *shapes* cannot be analyzed because of low resolution. Cell tracks are typically much longer than the diameters of the cells under study.

The measure of cell diameter is generally shorter than the lengths of typical cell tracks. Correlation of cell shape and cell locomotion is an interesting biological problem that requires magnifications high enough to perform proper shape analysis *and* a field of view *wide* enough so that cells cannot locomote out of view. Automatic tracking and real-time control of the microscope stage allow the effective field of view to be the extent of the cell chamber or slide, while affording a high magnification window required for high-resolution shape studies.

The study of changes in cell behaviour, such as membrane shape activity and cell locomotion from generation to generation provides valuable insight to many important medical questions, such as the mechanisms of cancer [23], [48], aging [5], [1], and differentiation [10]. Cell division is the central activity of all of these important issues. An automatic means then, of determining and recording when cell division occurs, and what happens to subsequent generations will greatly aid medical researchers in their quest to solve these medical puzzles.

Present techniques to study cell heterogenity require injecting inert tracer dyes which are passed on to offspring [10]. Disadvantages of this

method include eventual dispersion of the dye after many generations, and the unknown physiological effects the dye may have on the cells under study.

TRACES and the MRS are developed for tracking cells in an undyed environment developed to simulate *in vivo* conditions [18]. The *in vivo* environment is three dimensional, unlike the common two-dimensional environment created by microscope slides. TRACES has the ability to track cells in the direction perpendicular to the focal plane as well as in the usual two-dimensional field. The MRS, in its present form, however, is based on two-dimensional images, and assumes that the cells being tracked do not move out of the depth of field of the microscope. However, the MRS has built-in mechanisms to monitor other objects possibly appearing or disappearing in the image plane. Tracking is disabled for cells which move completely out of focus.

Advanced image processing techniques must be used to segment unstained cells from their textured backgrounds. The segmented cells must then be processed by an appropriate shape representation that is independent of cell size and position. Time-dependent state knowledge must be used in conjunction with the shape changes to determine the cell process state. Living cells are of particular interest to computer vision researchers who study shape morphology, curvature representations, and dynamic shape processes [46] [40]. The dynamics of a "cell's world" [1] are interesting as well from a *knowledge engineering* perspective. The "cell's world" can be defined by a finite number of rules describing the shapes of cells, their ancestry and their state. This world is particularly

[1]The use of the term "cell's world" is meant to emphasize a confined, limited environment, such as that described by the well-known term "blocks world" used for example by [19].

interesting in that it is time-dependent: cell states depend not only on the present cell shape, but on the previous progression of states leading up to the present image.

This thesis applies computer vision and AI techniques to the study of proliferating cells. The movements of living cells are automatically tracked and the shapes of the cells are analyzed to determine their states of mitosis. Once cell division has been detected, the new daughter cells are then tracked to continue the study of subsequent generations.

Cells are segmented from individual images using a region-based thresholding technique [60] enhanced to reduce non-cell artifacts (for the mitosis application) and to allow segmentation of more than one cell. The cell shape and center are computed using a fast multiple-resolution skeleton method based on boundary erosion [13]. The skeleton shape information, previous knowledge, and segmentation information are mapped into a fuzzy evidence representation which is then analyzed to produce a final fuzzy state set. Cell division is observed when the post-mitotic state has a maximum membership value relative to the other possible states. A new cell is then initialized for tracking, and the generation numbers of the two daughter cells are updated.

In summary, this thesis has several contributions.

An adaptive segmentation method has been applied to accurately segment multiple unstained cells. The method is based on the work of Wu *et al* [60], and expanded to suit the cell division problem.

A fast skeletonization algorithm and boundary representation has been developed particularly for the MRS application. It is modified from Dill *et al's* [13] multiple resolution skeleton approach to emphasize both the shape's characteristic of mitosis and the computational efficiency re-

quired for real-time processing. The erosion nature of the skeleton algorithm leads to an alternative definition of *cell center* for tracking the significant body of a cell. Unlike standard center-of-mass definitions, the alternative cell center is based on the erosion process of skeletonization.

This thesis develops a method of converting image data and miscellaneous measurements into a representation of overall cell shape and state. The expert system uses matrices of rules, fuzzy weights, and time-dependent processes to quickly convert the input data into a fuzzy shape and state.

Finally, this thesis integrates the segmentation, skeletonization, and fuzzy processing into a single computer vision system which segments, analyzes, and tracks cells and their offspring by recognizing the major states of mitosis.

1.2 TRACES: System Functions and Hardware

The entire set of computations and tracking is implemented as a single package called the *Mitosis Recognition System* (MRS) developed for a specialized imaging application called TRACES (an acronym for **TR**acking **CE**lls **S**ystem) [18]. TRACES operates in a SUN Station environment to process either previously stored image sequences or real-time images during actual tracking experiments.

TRACES consists of six main functional components for real-time tracking experiments: the environmental system, the positioning system, system control, image acquisition, image processing, and image understanding. These functional components are shown in Figure 1.1. The MRS adds the additional functional sub-component of *image understanding* or *image description* to TRACES, apart from the other main

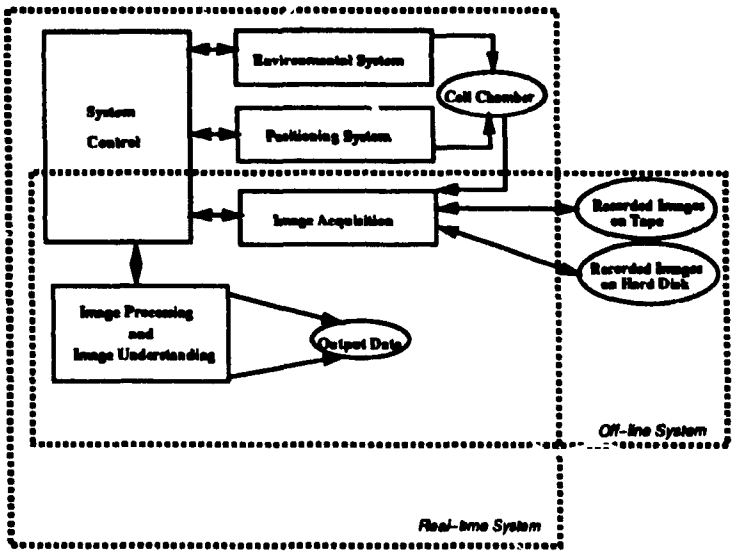

Figure 1.1: TRACES Functional Components. The dotted boxes surround the functions necessary for the off-line system and the real-time system as indicated. Arrows indicate flow of data and/or control information. The cell chamber and recorded data modules (enclosed by ellipses) are inputs to the real-time tracking functions and off-line analysis functions respectively.

subsystems outlined in [18].

The *environmental control system* is functionally independent from the imaging and positioning components. Its role is to maintain the cell chamber at a constant temperature in a humidified and gas-controlled atmosphere [18].

The *positioning system* has the function of moving the cell chamber so that any point along its three axes can be centered and focused within the field-of-view of the microscope and imaging system. More than one cell can be studied during an experiment by constantly repositioning the stage for each cell before image acquisition. The hardware responsible for the positioning function is a computer controlled, linear, three-axis positioning system and a locating fixture located beneath the mounted cell chamber [18].

System control is responsible for the overall high-level control of an experiment, with a functional interface to the image processing, image acquisition, positioning, and environmental systems. The system controller updates the positioning controller with new information from the image processing component, controls the image acquisition frame rate, and controls the general sequence of events during an experiment. The system controlling tasks are performed on a SUN workstation with data communication to and from the servo positioning controller, the peripheral controller, and the frame grabber (Figure 1.1).

The *image acquisition* component is responsible for the delivery of digital grey-level image data to the image processing function through the system controller. Hardware includes an inverted light microscope, an illumination and shutter controller, a monochrome video camera, a digital frame grabber, and an analog time-lapse video recorder [18] (Figure 1.2). Real-time experiments may store image data on videotape, on hard disk, or store only the processed output data produced by the image processing and image understanding systems. Off-line experiments are performed with previously recorded images from videotape or hard disk without the need of the positioning and environmental systems.

The *image processing* function consists of tasks dedicated to the preprocessing: segmentation, boundary extraction, and center of mass computations, as well as the coordination and interface with image understanding activities. The cell center positions are fed back to the system controller so that the updated information can be used for the positioning control system. This information, along with the boundary coordinates of the cells, is stored on hard disk for processing and analysis outside of TRACES. All image processing tasks are performed on a SUN workstation.

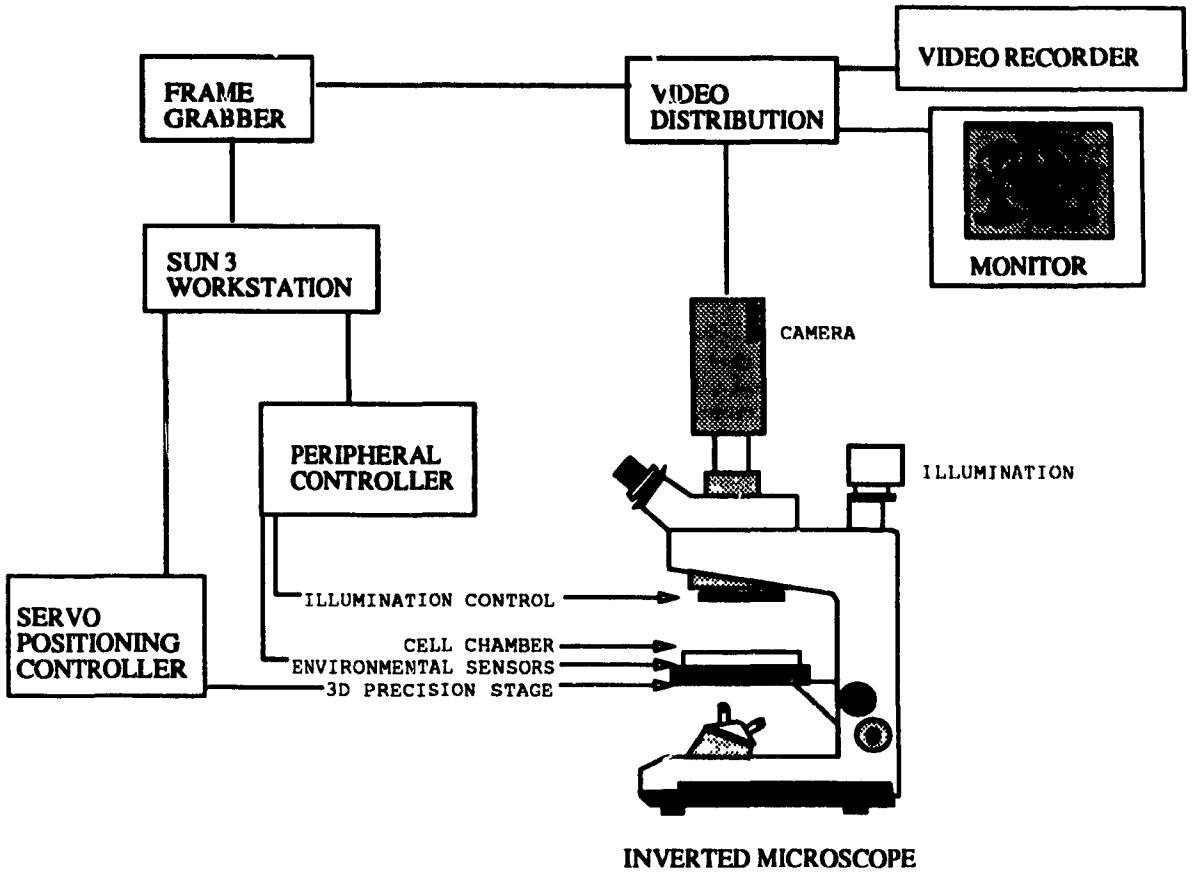

Figure 1.2: TRACES (TRAcking CEll System) Hardware Components. (adapted from [18])

The understanding functions include reasoning about the shape processes of the cells being tracked. Both the shape and mitotic state of cells are determined using a fuzzy-logic expert system which integrates time-dependent data in the analysis of mitosis. If the system determines that cell division has occurred, the information about the the daughter cells is used to create a new entry for the positioning control system. The image understanding functions of TRACES consists only of the Mitosis Recognition System (MRS) at present, but future plans include the integration of additional high-level tasks. These would include, for example, *three-dimensional* analysis and classification of living cells.

1.3 Related Work

It is difficult to summarize the related work within the scope of this thesis because of the many areas of research contained within its parts. The MRS system explores image segmentation and tracking of cells, biological shape analysis, skeletonization, and fuzzy expert systems.

Other systems have been developed to image, segment and track cells for a variety of applications ([57] [34] [45] [14] [40]). Cell image processing and shape analysis activities (generally done with previously stored data) are becoming more and more prominent both in the fields of computer vision and in cell biology ([46] [63] [45] [40] [13]). The field of cell imaging and automatic cell processing is becoming increasingly popular with the introduction of computer workstations and image processing packages. More specifically, *skeletal* methods of shape processing and analysis applied to living cells are explored [40] [13] [46]. Tracking generations of cells by means of recognizing mitotic states has not been developed elsewhere, to our knowledge, however, it has been explored to a certain extent by Ferrie [14] and Wheeler [59] (unpublished). We believe we are the first to develop a Mitosis Recognition System which is able to track and analyze dividing cells automatically in a real-time environment.

Expert system approaches have also been developed for various problems related to cell tracking or analysis of microscopic images ([15] [36] [58]). Fuzzy logic systems applied to image processing problems are addressed by [31], however, there are few existing systems applied to microscopic cell shapes. Lee [30] applies the concepts of fuzzy shape classification to three different configurations of chromosome images. Our fuzzy shape concepts of *ROUND*, *OBLONG*, and *COMPLEX* applied to

cells are similar to the *median*, *submedian*, and *acrocentric* chromosome shape classifications of Lee. Lee's application, however, does not involve the problem of dynamic shape processes; they are concerned mostly with pattern classification problems. Fuzzy set theory applied to the shapes and states of dividing cells is also believed to be an original application of this thesis.

1.4 Thesis Outline

General biological aspects of the cell cycle and cell locomotion are explored as a background to the main thesis (Chapter 2). These cell biology problems are discussed in terms of *computer vision* research. Quantitative and data measurement issues for biological analysis are examined with a brief discussion of other cell tracking applications and methods in relation to the MRS system.

The main thesis is divided into two important sections: a section on the *image processing* principles needed to first extract cells and basic shape information from raw images (Chapter 3), and a section on the *image understanding* methods used to intelligently analyze the preprocessed data for deciding cell shape and life cycle state (Chapter 4).

The principles developed in Chapters 2 and 3 are then demonstrated in Chapter 5 by means of actual cell experiments. The results, methodology, and limitations are discussed before final conclusions are drawn (Chapter 6).

Chapter 2

Background

2.1 Cell Biology: The Cell Cycle, Cell Locomotion, and Cell Shape

All living cells reproduce by a process called *cell mitosis*. Although cells may differ considerably in size, shape, and behaviour from one type to another, the basic stages that occur during the cell cycle are common to all living cells, both plant and animal.

The DNA contained in the chromosomes of the parent cell is replicated during mitosis so that the daughter cells contain genetic information identical to that of the original parent. The extent to which *locomotory* behaviour is inherited is not known. One of the main purposes of the development of the mitosis recognition and tracking system is to study these locomotory traits over several generations of cells.

Standard cell biology texts usually take a *functional* approach to describe the physiological changes which take place during the cell cycle. Figure 2.1 illustrates the processes which occur during a cell's life cycle [2]. The stages of mitosis are: prophase, metaphase, anaphase, and telophase. Interphase follows cell division and precedes the next mitosis sequence. These stages are based mainly on the internal changes in the cell nucleus and the behaviour of the chromosomes.

Plant cells in general undergo the same internal processes as animal cells during mitosis, however, certain types of plant cells do not exhibit the same type of shape changes as individual free-roaming cells, such as

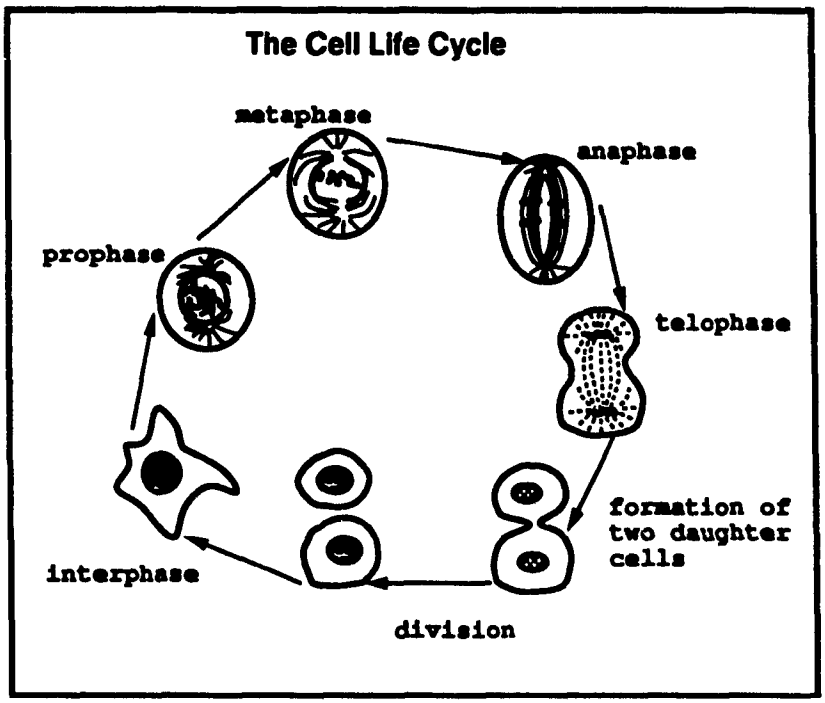

Figure 2.1: The life cycle of a cell. The life cycle and the stages of mitosis are characterized by internal changes in the cell such as changes in the cell nucleus and the replication and movement of the chromosomes. The telophase stage, however, is also characterized by the *division furrow* of the membrane. (Adapted from [2] and [28].)

blood cells. This thesis considers only unclustered "free-roaming" animal and plant cells.

The cell tracking system does not use the *internal* behaviour and structure of the cells because the cells are tracked at a relatively low magnification and they are *not* stained, making internal organelles difficult to segment. The overall shape changes of the cell boundary are used to determine the occurrence of cell division. A cell cycle with a notation based on *shape* changes during mitosis is used to define the possible cell states in the Mitosis Recognition System (MRS), as shown in Figure 2.2.

Unlike the life cycle diagram of Figure 2.1, the shape cycle (Figure 2.2) contains paths which allow backtracking from certain states to pre-

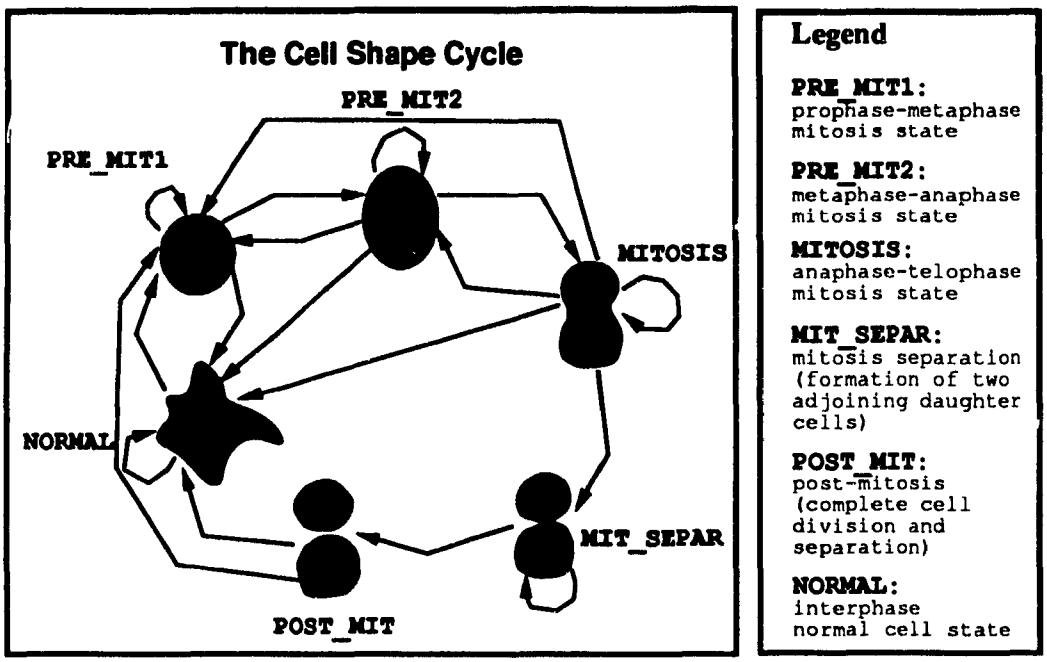

Figure 2.2: The shape cycle of a cell. The life cycle of a cell can be described by its sequential changes in overall *shape*. The MRS determines the cell shape and state and uses the changes in state to predict the occurrence of cell division. Note that not only is the *shape* important, but also the *order* in which the shape changes. Certain types of cells often undergo drastic changes in shape during the "normal" state; shapes which may be characteristic of late stages of mitosis. Certain state transitions are declared to be forbidden, such as from POST_MIT back to MIT_SEPAR.

vious ones. Changes in cell shape may, under some situations, lead to an ambiguous cell state condition. For example, a cell in the **NORMAL** or *interphase* state may exhibit roundish and figure-8 shape changes similar to the pre-mitotic states, but then revert back to a more complex shape. It is not possible to distinguish the two conditions since the behaviour of the internal structures (eg. the nucleus) is not processed. However, since backtracking is allowed, when the cell once again exhibits complex shapes, the system will recognize that the cell is still in the normal non-mitotic state.

Once the cell is determined to be beyond the mitosis state and exhibits features of the mitosis separation in the **MIT_SEPAR** state, cell division and separation is then expected to occur in the next series of images. A cell in the **MIT_SEPAR** state has an important shape feature consisting of two touching knobs.

Figure 2.2 is only a simplified description of the shape cycle used in the Mitosis Recognition System. The system actually uses a fuzzy state representation or state vector rather than a single state. The complete representation is described in Chapter 4 which include additional states needed to represent cell collision conditions.

Cell locomotion takes place only during the non-mitotic, **NORMAL** state. Free-living amoebae such as cellular slime-moulds and other free-roaming *amoeba-like* cells such as fibroblasts and leukocytes change shape and move about during this stage. An otherwise stationary cell may produce many appendages which move about wildly while examining its local environment. A cell will also locomote by extending pseudopodia to *push* and by attaching anchors to *pull* itself along a solid substratum. A study of these extrusions are crucial in understanding the mechanisms of cell motion. The biological aspects of cell movement and behaviour

are covered very well in [28] and [12].

Cells exhibit characteristic locomotory patterns during the interphase period ([45] [12]). Cells of the same *type*, such as leukocytes, can be subdivided into classes whose members exhibit similar track-patterns. The track-pattern features such as stationary waiting times, turn angles, and directional memory times are being investigated. Little is known about the mechanisms responsible for these cell classes, but active research is underway. Some studies have suggested that locomotory characteristics may be inherited from parent to daughter [37]. This is one aspect which can now be studied using the mitosis recognition and cell tracking system. Thus, this research is expected to provide very valuable information for cancer and immunological studies [23], [48].

2.2 Computer Vision: Image Segmentation and Image Understanding

Living cells pose interesting problems for computer vision researchers. A cell can grow, shrink, translate, rotate, and change shape in any manner. How can a machine adequately represent such a creature? Even a moving person does not undergo the type of radical and unpredictable changes a cell may experience. Most robotic vision applications deal with very controlled environments consisting of rigid, well-defined objects. However, a cell's world is reasonably confined and a cell's life cycle changes are characterized by a limited number of functional states. Specialized computer vision procedures can be developed specifically for experimental conditions encountered in particular cell applications.

There are generally two aspects to any computer vision problem:

- the *image processing* part needed to enhance or preprocess an image

and the extraction of data about the objects contained in the image.

- the *image understanding* part which uses the data obtained by image processing, along with expert knowledge or other information to produce a higher-level understanding of the image in terms of what the objects are, how they are arranged, or how they are behaving.

The *image processing* problems encountered in the application of cell tracking are very challenging. Unstained cells blend very well with their background culture, making the membrane of the cell very difficult to segment. The segmentation must also be performed in real-time so that the cell can be tracked with an adequate frame-rate.

Apart from these preliminary image processing and single object tracking problems, there are three main aspects of cell behavioural research of interest to the computer vision community:

- In order to automatically track cells from generation to generation, a method must be devised to recognize cell division. The method must be able to distinguish division from other possible situations such as cell collision and subsequent separation [14].

- A meaningful representation of cell *shape* must be constructed to characterize the shape changes of a cell over time. The representation would help cell biologists study the activity of appendages during both the stationary period and the locomotory activity. The relationship between the pseudopods and the direction of cell motion could then be studied ([33] [63] [13]).

- The cell-path patterns in space and time are signals which can be analyzed using computer processing methods of pattern analysis,

digital signal processing, and stochastic theory ([45] [9]).

The present thesis emphasizes the first issue, but provides a foundation for exploring the other two. The computer vision system in this thesis recognizes mitosis and tracks cells and their offspring by using skeletons (Chapter 3) to analyze the shapes in real-time. The skeleton approach can also be used for pseudopod study, as done elsewhere ([13] [40]). The MRS records the skeleton data for further analysis.

Cell positions are also stored for pattern analysis at a later time. The location of the cell is marked by an estimate of the cell center; two definitions of the center are used in this thesis and compared. Analysis of the cell paths is not done in this thesis. We emphasize that one of the main goals of this thesis is to provide a *means* for the automatic collection of these data.

2.3 Shape Measures and Related Work

Whatever the *method* of shape description, the ultimate purpose in object classification applications is to use the computed description to produce measurable data for the characterization of the important features of an object. The type of data and the chosen method of shape description depend upon the goal of the application, since an emphasis may be placed on certain features under study.

Most shape applications require that the measurements be invariant under image size or magnification, object rotation, and object translation. Living cells, in particular, move about in all directions and are studied under different magnifications. Measurements that are invariant for these criteria are crucial for consistent study during different experimental conditions. There are various methods of shape representation,

for example, polygonal, boundary chain-code, and skeletal [35]. Once an object's shape is clearly represented, important data such as feature points and size or length measurements should be extracted easily.

A shape's skeleton is a compact way of representing an object's form, combining both region and boundary information into a graph-like structure [40]. Biological shapes are represented more appropriately by a skeleton-based description than methods based on strict geometrical constructs ([40] [6]). Given a skeleton and the erosion time of each point on the skeleton, the original object can be reconstructed. TRACES does not presently use the skeleton for object reconstruction purposes, but records the data for an efficient record of the important cell shape changes during experiments.

The trace of a skeleton is used in many applications for the recognition of objects with well-defined structure. The number of branches, lengths of the branches, location of nodes, and angles of the branches with respect to one another at a node are used as a graph representation for object recognition and classification problems. Blum and Nagel [7] used this type of connected node information from the SAT (Symmetric Axis Transform) to describe a child's face in profile.

However, in this thesis, the principle motivation for using a skeleton representation is that meaningful data measures can be extracted very easily to ultimately classify objects into shape categories. For example, the simple measure of *end branch count* is the number of shape protrusions or extruding parts of an object. For a cell, protrusions indicate pseudopods, and extruding parts indicate either major pseudopod activity or the process of mitosis. Thus, a cell which has very many protrusions or pseudopods would be categorized as a complex shape, indicating a normal cell behavioural state.

Only *three* major shape categories, combined with other knowledge, are used to help determine the cell state. These are *ROUND*, *OBLONG*, and *COMPLEX*. The complete shape description is actually a vector or fuzzy shape consisting of membership values in each of the three basic shape categories.

Cells with *COMPLEX* shapes are *never* undergoing mitosis, so it suffices for this application to have such a broad shape category. Other applications, such as motion or pseudopod analysis, would require a further breakdown of the complex shape category.

The onset of mitosis is characterized by a very round shape which lasts for a significant period of time. The *ROUND* shape category is used to help determine this important state. The next shape category describes the *elongation* that must follow. The skeleton of this *OBLONG* shape category remains very stable as the cell becomes pinched-in before cell division actually occurs.

In determining an adequate shape representation for an application, it is important that one is chosen carefully in order to maximize the amount of both biologically interesting data and data needed for further system analysis. The shape representation must also be quick to compute for *real-time* processing during on-line cell tracking experiments.

Many AI rule-based applications and computer vision processing algorithms are just too time consuming to be practical for real-time tracking. The rate of imaging for real-time cell analysis depends upon both the *features* of the cell to be studied and on the *type* of cell being used. The data requiring the most samples per unit time determines the imaging rate. The measure of cell position is needed frequently in order to monitor the movements of the cell and to re-adjust the position of the tracking window so that the cell never moves out of view. Slime moulds,

for example, need to be imaged at a rate from 10 to 30 seconds per frame because they move about quite quickly. Blood cells, however, tend to move more slowly, and can be imaged less frequently.

Relative to cell translations, cell shape changes during mitosis occur more slowly, and would not require processing as often. Pseudopod analysis (when a cell is not undergoing division), would require a fast frame rate because a cell's local extrusions may occur very quickly even if the cell's overall position is stable. Thus, as with any dynamic shape problem, scale across *time* must be considered in conjunction with scale across the spatial dimensions of an image ([26], [27]).

Cell shape changes as related to pseudopod activity and locomotion have been studied extensively by Levine, Noble, and Yousseff ([33] [63] [45]). Cell shapes are described by their static features which define different *properties* of the cell under the subsets of *shape, location, structure, and motion*. The set of properties describing *shape* include area, perimeter, circularity, average bending energy, circularity and properties based on a polygonal approximation of the cell, such as angle regularity and side regularity. This polygonal representation is also used to decompose a cell into its body and pseudopod parts. The quantitative shape properties of a cell are then related to qualitative shape *qualifiers*. The shape property *average bending energy* for example, has the following shape qualifiers associated with it: *very jaggy, jaggy, almost smooth , smooth, and very smooth*. The average bending energy is defined as the average rate of change of the tangent along the boundary [33]. Levine and Noble use the polygonal decomposition to create a labeled graph to represent the cell body and its parts [45]. The graph description, (like a skeleton graph) is both translation and rotation invariant.

Ferrie *et al* combine relevant shape data into *feature vectors* for each

cell region [15]. A vector contains the following important features: 1) region centroid, 2) average intensity, 3) coordinates of the minimum bounding rectangle for the region, 4) a shape measure based on the ratio of length over width of the minimum bounding rectangle, 5) orientation of the major axis of the region, 6) area, 7) perimeter, and 8) an overall boundary curvature measurement ([15] [14]). A reference vector based on the features of a cell about to divide is created to approximate the expected characteristics of a hypothesized daughter cell in order to perform the best matching of cells after division occurs. For example, the intensity of the daughter cells should approximate the parent cell before division, and the area should be one-half that of the parent. Problems of scale invariance are avoided for state classification since the features are matched using *relative* measures. However, raw feature data (such as area and perimeter) are *not* scale invariant so would not be meaningful under a comparison of different data sets from different experiments.

Our method emphasizes skeleton data to describe the shape features of a cell. Other data, such as area and perimeter, are normalized in order to be mapped consistently into fuzzy representations which are comparable from one experiment to the next. This additional step of converting the raw (normalized) data and skeleton data into fuzzy membership sets not only provides us with comparable data sets across experiments, but it also allows us to integrate different types of data into descriptive measures of overall shape and state.

Chapter 3
Image Processing of Living Cells

3.1 Segmentation

3.1.1 Introduction

Cell shape processing cannot proceed until the boundaries of the cells are correctly identified given the input digitized grey-level image. The process of extracting objects or regions of interest from a raw image is called *segmentation*. The segmentation process for living, unstained cells at a magnification suitable for tracking is a particularly difficult task since the cell boundary is difficult to distinguish from the non-uniform background texture. It is also desirable to segment all possible cell candidates within an image window, and to distinguish cells from other artifacts.

A cell segmentation method specially designed for unstained, dividing cells is being used for the Mitosis Recognition System (MRS). The basis of the method is the segmentation procedure proposed by Wu *et al* [60], which was developed to accurately find the bounding contour of a living cell within a three-dimensional collagen gel. The image features that characterize cells under these imaging conditions are the grey level intensity and the local variation of this intensity. They have overcome most of the problems normally encountered in this type of segmentation problem, and the algorithm is practical for real-time processing.

Their method is applied with the assumption that there is a *single* cell within the image window to be segmented. The cell is determined to always be the *largest* component found after thresholding. This segmen-

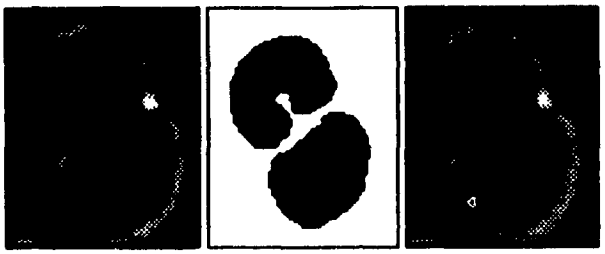

Figure 3.1: Mitosis segmentation problem. (a) raw grey-level image of cell near complete division; (b) preliminary segmentation of two cell parts; (c) final segmentation of single cell.

tation method was easily modified for the more general case required by the mitosis tracking experiments. This application requires the segmentation of *all* cells within a window in order to determine the positions of the daughter cells after cell division has occurred. To be robust under real experimental conditions, the segmentation method must be able to differentiate cells of varying sizes from smaller non-cell artifacts. There may also be a problem during mitosis when a cell is in a figure-8 shape (Figure 3.1). The image of the cell along the cleavage furrow tends to be lighter than the rest of the cell due to the absence of internal organelles. Unscrupulous segmentation using Wu *et al*'s standard algorithm results in the cell segmented into two halves *before* the cell membrane has actually divided. The object is still actually one cell, even though the area around the cleavage furrow is very light in contrast. A subsequent image may have a slightly darker cleavage furrow, resulting in the segmentation of a single, figure-8 shape. This characteristic of division results in very unstable segmentation during mitosis - at the stage most crucial to our application. The enhanced segmentation technique used by the MRS provides additional stability in the segmentation of cells undergoing division.

3.1.2 Segmentation Overview

Wu *et al's* algorithm is a *multi-stage* strategy [60]. The image is first partially segmented into approximate regions by the application of a global threshold to the local *variation* of intensity. The variance at each pixel is computed using a square mask centered at each pixel by summing the difference between the pixel intensity and the mean intensity within the mask. A minimum error thresholding method proposed by Kittler and Illingworth [24] is used to segment the regions from the original image. This first variance thresholding step is considered to be a *global* method.

These regions are then filled to obtain single, connected regions without holes using a component labeling procedure [60]. The resulting high-variance regions may contain one or more objects whereas the non-regions consist only of the background texture. The region of high variance provides a more ideal area in which to apply a final global grey-level threshold to extract objects, parts of objects, and other dark artifacts. This thresholding is now based on Otsu's optimal thresholding procedure [47]. In this application, cells are assumed to be darker than their backgrounds. The segmentation procedure can easily be reversed to extract *light* objects rather that *dark* objects.

These thresholded components are expanded, contracted and merged together locally using a component labeling approach [18] so that object boundaries and parts become connected and any remaining holes are filled. Wu *et al*'s application then considers the largest cohesive object to be the imaged cell.

3.1.3 Adaptive Size Thresholds

The application of Wu *et al* included the assumption that a single cell is being segmented from a given image window, so the concern was with extracting the largest high-variance region and then the largest component within this region. In the present implementation of this method for the mitosis system, preprocessing does *not* assume that the largest component is the desired cell. The Mitosis Recognition System attempts to recover *all* possible cells from a given image. It does not assume *a priori* that there is a single cell within the image window, however, other criteria are used to help filter out extraneous non-cell particles.

The image segmentation consists of two main steps:

1. the segmentation of the high variance regions (Figure 3.2(a)(b)), and

2. the segmentation of the dark objects or components within the regions (Figure 3.2(c)).

Filtering is performed during each of these steps.

Region Filtering

The image is first segmented into high variance regions, as described above. The *non-regions* are discarded, since they should not contain cells. The *regions* are examined, however, and sorted according to their total area in pixels. Any regions which are smaller than the *minimum region size* do not contain cells, and are discarded. The size of a region depends upon the mask size used in the variance computation, and the computed variance threshold, but nonetheless must always be larger than a typical

cell if the region actually contains a cell. Figure 3.2(b) illustrates the high-variance regions before size filtering.

The *minimum region size* can either be set as a static parameter determined by typical cell sizes for the application, or set as a dynamic parameter for applications in which cell sizes can change drastically (*ie.* orders of magnitude) for different tracking experiments. The mitosis application performs sufficiently with a pre-determined value.

Component Filtering

Each of the remaining regions is now thresholded and their components labeled and filled. The sizes of the components are compared to a *minimum component size* threshold, and the small, extraneous objects are discarded.

The mitosis application computes the *minimum component size* from the size of the largest component found within the window, which is expected to be a cell. In the present implementation, the *minimum component size* is equal to 25 percent of the size of the largest component. However, if the largest component is very small, and is smaller than a predetermined absolute minimum cell size, we set the *minimum component size* equal to the absolute minimum cell size. In this case, we do not expect to find cells within the image window.

The advantage of having a dynamic minimum component threshold is to allow the application to track different cell types of moderately varying sizes and magnifications, and during different stages in cell growth. For example, a cell just before division will be twice the size of its daughter cells. We allow a 75 percent difference between the area of the largest component and the smallest component within an image window.

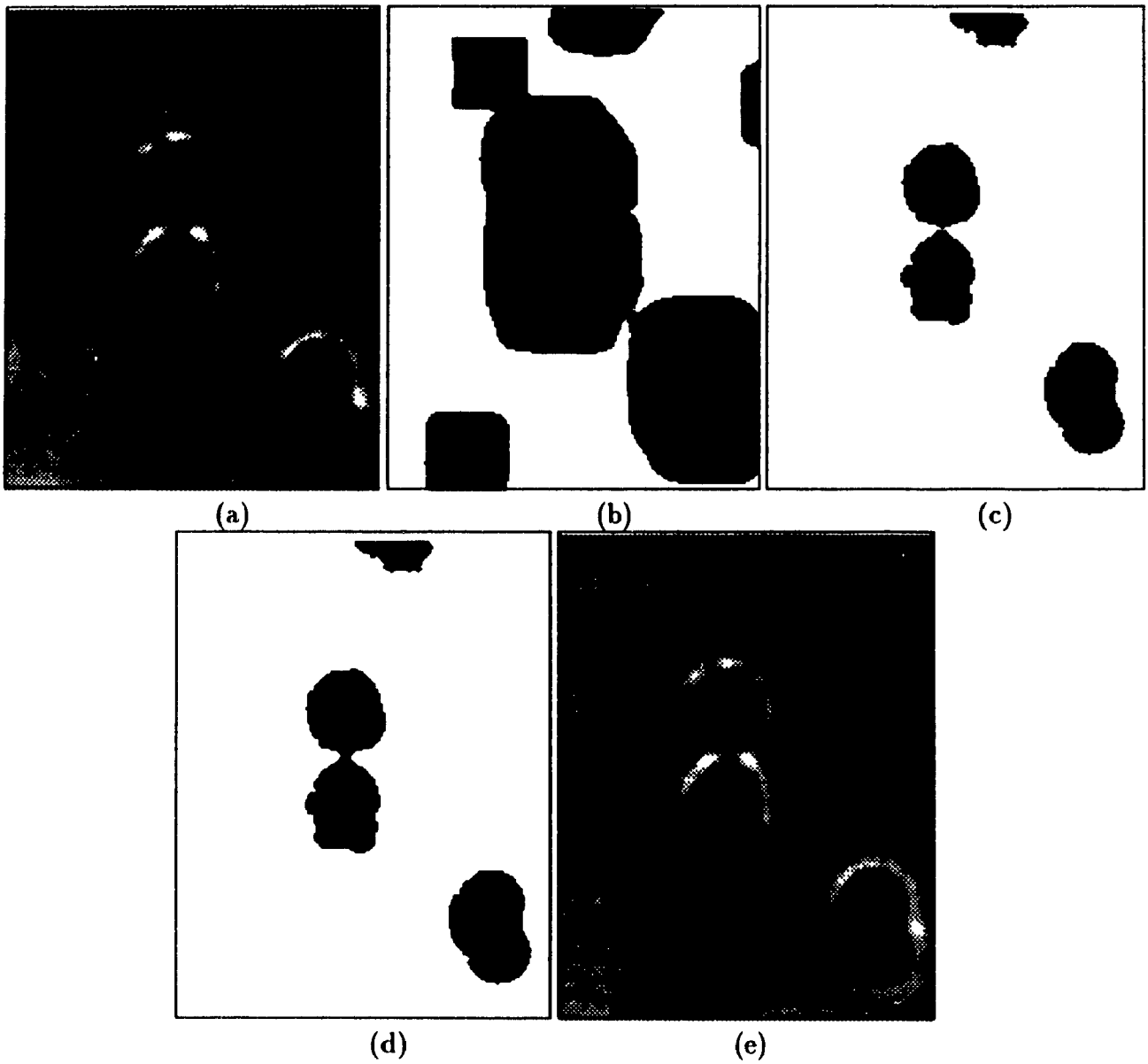

Figure 3.2: Multi-stage segmentation. (a) a cell image; (b) the approximate regions; (c) component extraction; (d) final binary image after expansion, merging, and contraction; and (e) objects on original image.

3.1.4 Component Data

The Mitosis Recognition System differentiates neighbouring cells which have moved together by cell locomotion from two neighbouring daughter cells. As well, the system attempts to recognize the difference between the figure-8 shape formed by two cells which are *in contact* and a cell undergoing mitosis. The segmentation of a cell undergoing mitosis may also falsely separate the cell into two parts *before* the cell has actually divided. It is very important, then, for the system to gather as much information about the neighbourhood of a cell as possible. Information about very close neighbours can be retrieved from further component processing. These conditions must be carefully monitored and recorded as input data to the expert system. This section outlines the extraction of relevant data during the segmentation process, but the analysis of these data is reserved for the chapter on image understanding.

The following data are output from the segmentation processing and used in subsequent operations:

- a binary image consisting of black regions corresponding to objects
- a list of component sizes
- a list of centers of gravity for each component

The set of steps listed below describes the entire segmentation procedure with the emphasis on the extraction of component data needed for the expert system analysis.

Preliminary Segmentation and Filtering

Preliminary segmentation (described by Wu *et al* [60]) is first performed to compute a preliminary binary image of cells, cell parts, and artifacts.

As discussed at the beginning of Section 3.1.2, rough high variance regions are computed, and any holes in these regions are filled. Regions smaller than the *minimum region size* are discarded, and Otsu thresholding [47] is performed on each remaining region.

The thresholded objects are then processed one at a time using a component labeling procedure. [18]. The *sizes* of each component are determined, and components with a size greater than the *minimum component size* (see Section 3.1.3) are *expanded*, *merged*, and then *contracted* [18] to fill in holes.

At this stage each component is a solid object whose boundary can easily be extracted by tracing its contour (Figure 3.2(c)).

The black objects represent cells or parts of dividing cells. The next step expands the objects in their local region so that any parts that should be connected are *merged* and any almost touching cells are joined together (Figure 3.2(d)). This region-based method of determining local neighbours avoids the difficulties of a boundary-based approach which would require computing and evaluating the nearest Euclidean distance between all boundary points of neighbouring objects.

This step is accomplished by *expanding*, *merging*, and *contracting* the entire set of components within the image. The *contraction* is performed at a slightly smaller factor that the expansion, so the components experience a slight increase in area.

The segmentation is now complete, and the resultant image is now used for further shape analysis. It is important to stress that this output binary image may contain *touching* or *merged* components. Subsequent object extraction (by boundary tracing) from this binary image may not match the original list of components since two neighbouring components may have been merged into a single object. A comparison of the com-

ponent data and the final object data is important for the expert system analysis at a later stage. For example, there are *four* components in Figure 3.2(c) and (d) but only *three* separate objects.

3.2 Skeletonization

3.2.1 Introduction

The first step in the image processing of cells is to complete the *segmentation*: the preprocessing required to transform a grey-level microscopic image into a binary image consisting of black objects representing cells and white background. The next step in the goal of describing the shapes and states of the cells is to quantify the position of the cells by exactly locating their boundaries. A linked boundary chain representation describes the location of a boundary point's immediate neighbours along the contour. This chain in itself provides shape and contour information about objects, and certain relevant information is retained at this stage for the expert system. In particular, the curvature extrema of the boundary are evaluated for critical convexities (eg. corresponding to sharp protrusions) and concavities (sharp indentations). The boundary chain and its set of convex critical points are now used to initiate the *skeletonization*.

Several methods with correspondingly different nomenclatures exist to represent what is called *skeletonization* ([13], [39], [40], [61], [54], [44], [43], [8], [4]). Other terms widely used to represent equivalent or almost equivalent shape representations are the SAT (Symmetric Axis Transform) ([7], [49]), and the MAT (Medial Axis Transform) ([55], [29], [22]) as well as several others [39]. The skeletonization method and shape representation used in the Mitosis Recognition System are derived from the skeleton approach of Dill *et al* [13]. Other skeleton-based representations important to a variety of other vision applications are described very well in [40], [39], and [41]. A skeleton consists of thin,

one-dimensional lines which retain the connectivity of its original closed boundary shape [13] and are medial between the boundary points [49]. These lines can also be defined as the locus of the centers of all maximal disks in the object ([49], [40]). The radius of the maximal disk corresponds to the width of the object at the disk's center. Both the radius function and the corresponding skeleton points are required for accurate reconstruction of an object's boundary from its symmetric axis or skeleton. Alternatively, skeleton points can be defined as the points at which propagating wavefronts meet when initialized on the boundary [13]. All of the above definitions of the skeleton are based on a continuous, two-dimensional analog space (the natural environment of biological shapes), rather than on a discrete rectangular grid. The skeleton representation was originally developed because of its particular relevance to biological shapes [7] and processes of growth and change [42]. Practical image processing algorithms, however, must redefine these analog representations to be suitable for digital computations. Skeleton representations have been found to be very useful in the description of the shapes, movements, and pseudopod growth of living cells ([13], [40], [39]). Leymarie uses dynamic skeletons on sequential cell images to take advantage of the shape similarities between successive frames ([40], [39]). Dill *et al* [13] use their skeleton approach to represent pseudopods or protrusions of the cell membrane over several resolution scales. A skeleton representation applied to real, digitized, and less than ideal cell images must be 1) robust in the presence of noise, and 2) guarantee a connected set of pixels in order to be useful for further biological analysis. Real-time processing for tracking applications has the additional constraint that it must be 3) fast and efficient.

The Mitosis Recognition System bases its skeleton approach on the

work of Dill *et al* in order to satisfy these three major requirements and because of its suitability to the description of living cells. However, some modifications were required to suit the application of *mitosis* rather than the study of *pseudopods*. Some features of Dill *et al*'s multiple resolution algorithm were in fact simplified because of the simpler, more roundish shapes which evolve during the life cycle of a cell.

This section first overviews our changes to Dill *et al*'s approach and outlines the general steps involved in producing skeleton data from a binary image. The specific processing steps are then described in detail.

3.2.2 Outline

Dill *et al* [13] state that the representation of a pseudopod must be flexible in order to capture both *locally* convex contour segments as well as more *globally* convex segments. They are not so concerned about spurious skeleton branches because any branch that would be classified as noise would only persist for a short period of time and they do not want to miss small perturbations that could eventually form pseudopods. Dill *et al* propose a construction of a skeleton representation comprising several scales; the scales are based upon the evaluation of the contour at different resolutions using an n-code computation. Spurious or noisy branches that do not persist over time are eventually eliminated.

A significant contribution of Dill *et al*'s algorithm is its ability to combine *boundary* centered information at different scales with *regional* information provided by a skeleton approach. This is achieved by first preprocessing the boundary chain-code to determine significant convexities or critical points, and then initializing the skeleton processing with these feature points. The skeleton processing is likened to a *thinning* or

erosion process, but the boundary to be eroded is always re-evaluated for collapse points at which new skeleton branches may begin. Unlike Arcelli's algorithm [3], significant convexities are only computed for the object's original boundary since Dill *et al*'s original motivation was the capture and representation of *protrusions* of the cell membrane or boundary.

Our mitosis application, however, is not dedicated to finding and accurately representing *all* potential pseudopods. We are concerned mainly with representing *general* shape types over the life cycle of a cell. Thus, protrusion extremes, such as either very large protrusions or very small protrusions, must be carefully analyzed as follows. Firstly, the figure-8 shape which is often indicative of mitosis could be mistaken for a large pseudopod. Some means should be established to help differentiate large pseudopods from a figure-8 shape. Secondly, small segmentation artifacts (or noise) could be mistaken for very small, local pseudopods that would make the skeleton unreasonably complex. Unlike the pseudopod application of Dill's, we are not concerned with catching the formation of pseudopods as soon as they occur; we are more concerned with avoiding noisy artifacts. Local protrusions that are *not* noise will be processed when they become more prominent.

These two points emphasize the two extremes of scale. We do not want skeleton branches ending on critical points to represent very global protrusions, nor do we want spurious skeleton branches caused by a noisy, discretized boundary. A significant convexity that is found should have its associated critical point, hence skeleton *end* point, lie on a *local* convexity if one can be found in the region. Significant convexities are generally attributed to protrusions which represent pseudopods.

The algorithm used for contour processing in this thesis resolves these

two issues, respectively, by considering convexities at two scales:

1. an extremely *local* scale that evaluates small protrusions or noise by examining a boundary point's local neighbourhood.

2. a more global scale that is evaluated using an n-code on arcs of the boundary.

The *local* scale would find isolated critical points (called *multiple points* in Section 3.2.3), whereas the n-code method used for the *global* protrusions would find significantly convex arcs along the boundary. The mid-point of the arc is taken to be the skeleton end-point, unless a local protrusion also exists along the arc. If more than one local protrusion is found along the convex arc, only the center one is chosen to begin the skeleton branch. Other local protrusions (ie. ones not also within the global arcs) are discounted as noise. The significant critical points along the boundary are now marked as skeleton points, and the erosion of the object can begin.

Large protrusions which were not found to have significant convexities will have corresponding skeleton branches once the erosion boundary collapses in the interior of the cell. This includes the protrusions which occur during the figure-8 mitosis state.

Figure 3.3 illustrates the critical points and skeleton end point results for different types of protrusions. The figure-8 shape of mitosis should not contain significantly convex points because of the very smooth, global shape of the protrusions. Similarly, other large, global protrusions have skeletons which do not end on a boundary critical point (Figure 3.3 (a)). Only protrusions which end with significantly convex boundary points are candidates for having a skeleton which begins with a critical point

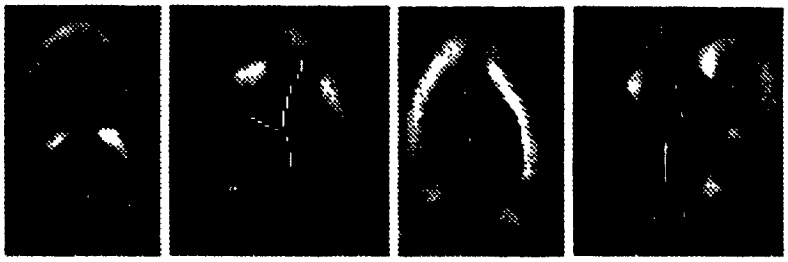

Figure 3.3: Skeleton end points and protrusions. (a) very large knobby protrusions characteristic of mitosis; (b) two significant pseudopods; (c) a protrusion ending in a significantly convex critical point; (d) a complex skeleton.

(Figure 3.3 (b),(c)). Multiple pixel noise on the boundary is filtered so that spurious skeleton branches are not created for very local protrusions.

An input binary image of a cell is transformed into data to be used by the image understanding processing module (Chapter 4) in the following steps, detailed in the following sections in this chapter.

First, the *boundary* is preprocessed in order to extract significant critical points. A linked boundary array must be extracted from the binary image, the n-code computed, and the significant convexities extracted.

Secondly, the object is eroded according to certain criteria, leaving only its skeleton behind.

Finally, the skeleton itself is evaluated to extract data pertinent to the expert system processing of the MRS.

3.2.3 Boundary Preprocessing

Linked Boundary Extraction

After segmentation, the resulting binary image consists of object regions and background regions. The first step in the subsequent analysis is to extract and represent all of the individual objects or connected regions

in the image. Because our skeletonization procedure first analyzes the boundary before erosion, a boundary-based representation must be used for object extraction.

The algorithm to extract the objects is straightforward. The binary image is scanned pixel by pixel until an *object* pixel is found which neighbours a *background* pixel. This object pixel is recorded as a boundary point so that we now have three types of points: object points, background points, and boundary points. The remaining boundary of the object is then traced in a clockwise direction. The search for the next boundary pixel is started using the orientation of the previous boundary point as an indicator for the next search starting position. The contour tracing continues until the original starting position is reached. Boundary pixels may be counted in the contour more than once, such as for protrusions of a single pixel width (see Figure 3.4). Boundary points which are counted more than once are are called *multiple boundary points*.

Once the boundary tracing for a single object is complete, remaining objects are found and traced in the image. If an object has holes, its *hole* boundary will also be traced, however, the segmentation and preprocessing procedures in the previous section eliminate all possibilities of holes. This simplifies the algorithm; however, others have explored the issues of object holes in this context [40] [61].

The boundary-finding algorithm concludes when the entire image window has been scanned, and there are n boundary lists, one for each object found. Note that this step only requires a *single scan* of the image window plus the tracing of n object boundaries.

The total number of objects found in this step is recorded for later evaluation in the expert system. This number will be compared to the to-

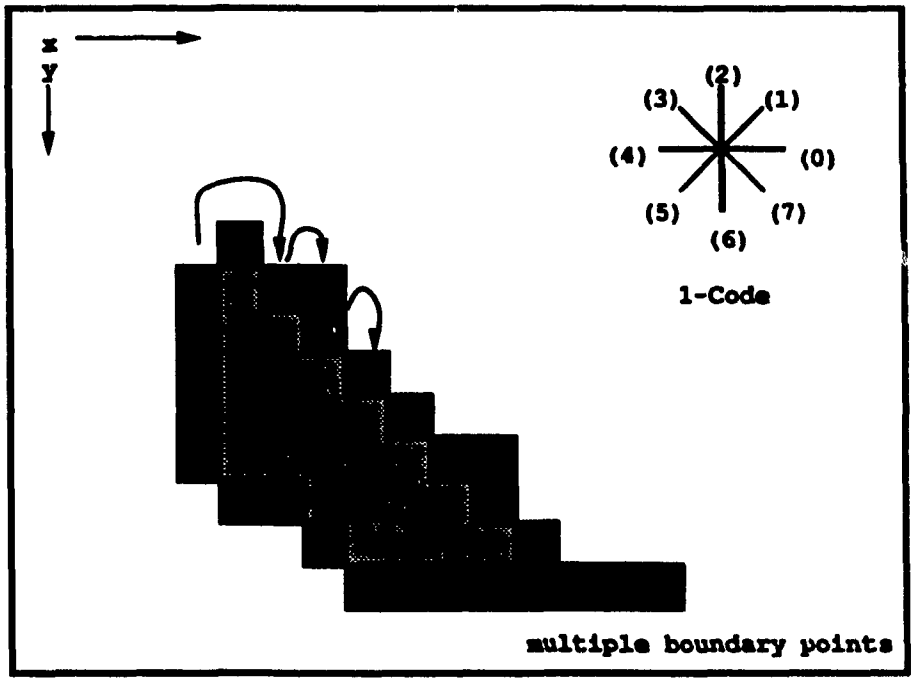

Figure 3.4: Finding the linked boundary chain. The boundary is extracted in a clockwise direction, assuming the image is scanned from top to bottom, row by row. Notice the single pixel-width boundary protrusion which results in the boundary pixels 10 and 11 being represented *again* as entries 14 and 13 in the boundary chain. A chain code or 1-code is assigned to each boundary pixel based on the relative positions of a pixel's immediate boundary neighbours in the chain. This number represents a discrete local curvature at the boundary pixel.

tal number of *components* found in the previous segmentation step. The *length* of each boundary is also recorded, and objects whose boundaries are shorter than a previously set parameter are not processed further.

The contour tracing algorithm described above is similar to the methods used by Leymarie in [40] and by Xia in [61]. However, the main difference is the recording of the multiple boundary pixels during the tracing. The next steps in the skeletonization process analyze only one object at a time.

Boundary Multiple Points

Before the boundary is evaluated using an n-code procedure, boundary points and their immediate neighbours are studied for very *local* protrusions that could either be noise, or be at the end-point of a potential skeleton branch.

A boundary pixel is a candidate for this type of protrusion (termed a *multiple point*) if the pixel [13]:

1. is traversed more than once during contour tracing,
2. has no neighbours which are objects, and
3. has at least one direct neighbour which belongs to the contour but is not one of the two direct neighbours along the contour.

Boundary points which are traced more than once are already marked with values indicating their multiple status (see Figure 3.4).

The second condition is easily checked by examining the local neighbourhood of a boundary pixel for object points. If the point does not have an object neighbour its status is recorded as a multiple point.

The final condition is also determined easily by checking for boundary neighbours directly North, East, South, or West of it which are not immediately before the pixel nor immediately after it in the linked boundary list.

Chain-Code and Boundary *N*-Code

An advantage of a ***boundary*** or ***contour*** representation for a shape is the conversion of the two-dimensional information into a periodic one-dimensional signal [51]. There are several approaches for analyzing an object's shape once the boundary has been obtained [35]. Most commonly, a representation of *local* curvature is computed in order to classify regions of constant or sloping curvatures and to find interesting feature points or *critical points* which occur at discontinuities in curvature. It is important for the application of cell shape that the method of contour representation be relatively invariant to translation, rotation, and dilation because living cells change size and move about unpredictably. It is *most* important that the contour representation chosen is appropriate for *discrete* image data and is easy to compute for real-time imaging applications.

A common method for representing discrete contour data in a one-dimensional signal is the *chain-code* approach, first introduced by Freeman in 1961 ([35], [16]). It is a discrete *orientation* representation in that the numbers from 0 to 7 represent orientation angles in multiples of 45 degrees from one boundary pixel to its neighbour (see Figure 3.4). This is a very coarse representation, so in order to be a useful measure of curvature, a symmetric, low-pass filter is often used to smooth out the orientation angle over several pixels, depending upon the desired reso-

lution. The resulting curvature measurement is often referred to as the n-code ([13], [35], [17]).

There are several problems inherent in the chain-code method which are explored in detail by Leymarie *et al* in [40], and [38]. These problems include discrepancies around 1-pixel thick *depressions* and *protrusions*. However, the present application of using the chain code to help determine significant convexities for cell skeletonization is a simple one. Very accurate curvature measures and sophisticated boundary analysis are *not* needed. We also integrate *multiple boundary point* information, (characteristic of single pixel protrusions) with the n-code approach which avoids some of the difficulties of the chain-code approach discussed by Leymarie *et al*.

A triangular mask $f_{\Delta n}(i)$ is used to perform the low-pass filtering. The chain code c_i^1 (or 1-code as it is commonly called) is correlated with the triangular mask $f_{\Delta n}(i)$ to produce the n-code c_i^n as in [13]:

$$c_i^n = f_{\Delta n}(i) \times c_i^1 \tag{3.1}$$

$$= \sum_{k=-n}^{k=n} f_{\Delta n}(k) c_{i+k}^1 \tag{3.2}$$

$$= n c_i^1 + \sum_{k=1}^{k=n-1} (n-k)(c_{i-k}^1 + c_{i+k}^1), \quad n \geq 1 \tag{3.3}$$

where $\times$ denotes correlation. Dill *et al* [13] evaluate the difference between using a triangular mask and a Gaussian kernel in the discrete case, and find that they yield almost equivalent results, though the Gaussian has somewhat better low-pass characteristics.

We have decided to adopt the triangular mask because of its simple, integral formulation. The algorithm we have developed never needs to

compute a non-integral number.

Normalization is necessary to maintain a consistent curvature representation over changes in scale so that scale-independent features can be compared. The normalized n-code is computed by first realizing that n-code values of an arc with constant curvature should be the same for all values of n. In the case of the arc of constant curvature [13]:

$$c = c_j^1 \quad j = 1-n, \cdots, n, \cdots, i+n \ . \tag{3.4}$$

The n-code is evaluated as [13]:

$$c_i^n = nc_i^1 + \sum_{k=1}^{k=n-1} (n-k)(c_{i-k}^1 + c_{i+k}^1) \tag{3.5}$$

$$= nc + \sum_{k=1}^{k=n-1} (n-k)\, 2c \tag{3.6}$$

$$= c(n + 2 \sum_{k=1}^{k=n-1} (n-k)) \tag{3.7}$$

$$= cn^2. \tag{3.8}$$

Since the normalized n-code of an arc of constant curvature must always equal unity, the computation for a normalized n-code becomes [13]:

$$\bar{c}_i^n = \frac{c_i^n}{n^2} \ . \tag{3.9}$$

The value of n for the computation of the n-code was chosen to be a constant *percentage* of the boundary length so that cells of varying magnifications are processed with n-codes representing the same stretches of arc along their boundaries. This resolution n would be chosen for the particular application, or in the case of Dill's multi-resolution approach [13],

several different values of n are chosen for a hierarchical shape analysis. We determined an appropriate value of n by experimentally computing hundreds of skeletons of cell shapes exhibiting a variety of mitotic states. We varied both the n-code and the value of the convex filter (discussed below) to obtain skeletons of varying numbers of branches and branch lengths. The present version of the Mitosis Recognition system uses an n equal to 7 percent of the boundary length of a given shape (rounded to the nearest integral pixel), but values from 4 to 12 percent would also produce good results for the mitosis application. A minimum value of n is set to 2 pixels to avoid possible aliasing effects [13]. Cell boundaries, however, are expected to be much longer than 30 pixels, so the minimum n-code of 2 would rarely be used.

A *large* value of the normalized n-code $\bar{c}_i^n$ corresponds to a point of relatively *low* curvature. *Very large* values correspond to significant *concavities* which may indicate necks of protrusions or the cusps of the mitotic figure-8 shape. Low values of the n-code signal significant convexities, representative of sharp protrusions or pseudopods.

Critical Point Extraction

Convexity and concavity thresholds or filters are defined to find points and connected points of arcs which would be significantly convex or concave for the mitosis application. Significant concavities are not used for the skeleton processing, but the data provide some information to the final cell shape and state evaluations. Significant convexities, however, are crucial to the initialization of the skeleton algorithm. The processing for both are the same, except for the threshold evaluation described below. 218z Many applications, including Dill *et al* [13], use the *normalized*

n-code result (Equation (3.9)) compared with convexity and concavity *constants* to determine critical points for any given size of shape. We, instead, define *dynamic* thresholds which depend on n^2 , rather than *constant* thresholds and use the *unnormalized* n-code computation (Equation (3.8)) in our algorithm. The final critical points are the only data of interest to us; we do not use the n-codes for any further processing. The normalization, however, is still important, and this is actually performed *implicitly* during the filtering since our filter parameters vary as n^2.

The advantage of this approach is the *avoidance of using real numbers* by eliminating the n^2 division of Equation (3.9) during normalization. Hence, the convex labeling procedure completes very quickly, while retaining the advantages of normalization.

The dynamic convex t_{convex} and dynamic concave $t_{concave}$ thresholds are computed as follows:

$$t_{convex} = k_{convex} \times n^2 \tag{3.10}$$

$$t_{concave} = k_{concave} \times n^2. \tag{3.11}$$

The constants k_{convex} and $k_{concave}$ are parameters of the system and need not be changed. [1]

Boundary points are marked as significantly *convex* if the equation

$$c_i^n \times \kappa \leq t_{convex} \tag{3.12}$$

is satisfied. For concave processing, significantly *concave* points are

[1] Appropriate values of k_{convex} and $k_{concave}$ are 840 and 910 respectively.

marked if the equation

$$c_i^n \times \kappa \geq t_{concave} \tag{3.13}$$

is satisfied. The multiplication constant κ is used to increase the value and the precision of the threshold required, so that a large integer number is used instead of a real number. [2]

The next step in the boundary evaluation is to determine the final critical points from both the set of significant convexities and the previously determined multiple points.

The list of significantly convex points is traversed, and any convex *arcs* are reduced to isolated critical points. If no multiple points exist along the arc, and no multiple points exist within n of the arc midpoint, the arc midpoint is chosen to be a critical point.

Otherwise, if one or more multiple points exist within the significantly convex arc, or within n of the arc's midpoint, the closest multiple point is re-labeled as a critical point. If two multiple points are equally close to the arc's midpoint, the midpoint is chosen as the critical point.

Isolated convex points are considered as arcs of length one (1) and undergo the same processing as the arcs above. Multiple points, then, which are quite close to convex points will be chosen as the critical points rather than the convex point itself. This is done in order to more accurately locate global protrusions within the local protrusion it encompasses.

[2] The value of κ presently used is 225. This number is arbitrary, but should be high enough to allow a significant integral range between the chosen convex threshold k_{convex} and the concave threshold $k_{concave}$. Note that the choice of k_{convex} and $k_{concave}$ depends upon the value of κ.

3.2.4 Boundary (Grassfire) Erosion

The object's boundary is now initialized with a list of of critical points. Any remaining points which are marked as multiple are removed for this first iteration only. Note that this means that multiple points which do not lie on significant convexities are discarded. For the first iteration, all boundary points except the critical points are removed from the object.

The boundary is now traversed again to produce a new linked-chain of boundary coordinates. Multiple points are also marked and evaluated as was explained in the previous section (3.2.3). The critical points (carried over from the previous iteration) are marked again on the boundary so that they are not accidentally removed during subsequent iterations. Now all boundary points, except multiple points and critical points, are removed from the object. Boundary extraction, multiple points marking, and erosion, continue until no more pixels can be eroded.

This preceding erosion procedure thins the object down to a one or two pixel thick skeleton. It is important to perform post-processing to thin the object to a single pixel width so that the length and area of the final skeleton can be compared to skeletons of other cells. Any fast, thinning algorithm can be used as long as a single pixel width is guaranteed and the skeleton remains connected along any pixel's eight neighbours (for instance, see [56], [53], [64], or [65]). Our algorithm is loosely based on that of Zhang and Wang *et al* [65] and Zhang and Suen [64].

3.2.5 Skeleton Analysis

Our boundary extraction algorithm (Section 3.2.3) is also used to traverse the final skeleton. The boundary of the skeleton, however, will have all of its points (except end-points) traversed more than once because the

skeleton has a single pixel width. The resulting skeleton is analyzed to provide pertinent data to the expert system evaluation of an object's shape.

Firstly, the number of exterior skeleton branches is computed. This is usually different from the number of original critical points because some skeleton branches begin where the erosion collapses in the interior of the object. This is very easy to compute because of the nature of the thin skeleton: skeleton end-points are simply determined as boundary points which *are not* traversed more than once. Interior branches are determined after external branches are removed.

In addition to evaluating the size and number of skeleton branches, an approximate *orientation* is computed for each branch by approximating individual branches as straight segments. This approximation is valid in most cases. However, situations where this approximation is not good occur for cells with complex shapes. These complex cells have other features which are more significant than the orientation of the skeleton branches (see Chapter 4), so a poor orientation measurement has little overall effect on the final computation of state.

Adjoining exterior branches are added together as if they were *force vectors* exhibiting a *magnitude* and a *direction* pointing away from the center of the cell (Figure 3.5). In this manner, the whole skeleton is reduced to a single line segment of a particular magnitude and orientation. The skeleton vector provides valuable information about the nature of the cell's shape.

If the cell is quite round, or very circularly symmetric, the skeleton vector length will be very short.

If the cell has several significant pseudopods pointing in some general direction, the resultant skeleton vector will be quite long, and oriented

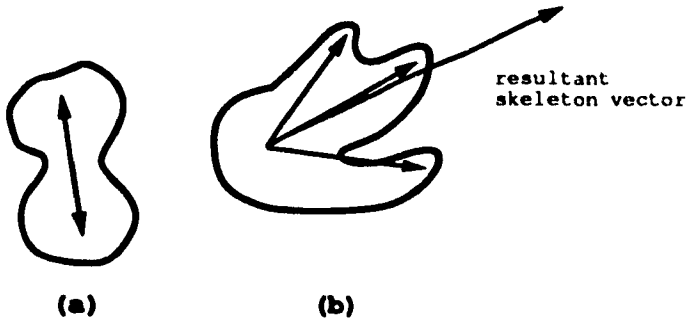

Figure 3.5: Skeleton Branches as Force Vectors. A stationary cell would have a net zero force, but the skeleton branch of a mitotic shape (a) demonstrates the opposing forces between the two parts. A cell undergoing motion has pseudopods characteristic of its *direction*. In fact, if the approximated skeleton branches are summed as vectors, a net force is computed, pointing in the direction of possible motion (b).

in the general direction of expected cell motion (Figure 3.5 (b)).

During *mitosis*, the simple elongated shapes naturally produce vectors which indicate the splitting forces of the two halves (Figure 3.5 (a)). The end-points of a resultant vectors during mitosis could also be used to predict the positions of the centers of the potential daughter cells. However, there is no means as yet to define a consistent method of determining the origin of the force vector for an object like the cell illustrated in Figure 3.5 (a). More complex shapes, such as in Figure 3.5 (b) are processed such that the origin is found at the intersection the skeleton branches.

The analogy between skeleton branches and force vectors needs to be explored further, but for the application of cells, the analogy could be very useful. Pseudopod behaviour is used by cell biologists to visually indicate the direction of cell motion. These are also indications of *physical forces* between the cell membrane and its environment, so the representation of skeleton branches as force shape vectors may be a valid one. Force vectors which are *balanced* (ie. the vector sums are zero)

indicate an object which is stationary. Unbalanced forces require that there is an acceleration or movement in a direction to balance the sum of the applied forces. The movement may be hampered by frictional forces external to the cell. Certainly in the case of cell division, there exists opposing forces in the effort to separate the two parts as the cell remains stationary. This thesis, however, only uses the skeleton information to provide data for the representation of cell shape.

3.2.6 Centroid and the Tracking Center

Cell physiologists who study the paths of moving cells must represent the center of cells under study in order to track their movements. The standard representation is the cell's *center of gravity* based on the segmented area of the cell being tracked [45]. This method produces very good results, except when the cell has very large or very many pseudopods clustered away from the main body of the cell. The large area taken up by the protrusions shifts the center of gravity away from what would be the cell nucleus.

An alternative definition would be a center equal to the center of the largest disc which fits inside the boundary of the cell. This definition would then capture a more appropriate center if a cell had several unbalanced pseudopods, but still had a reasonably-sized cell body. However, if the cell's pseudopods are more significant than the body of the cell, the true center would be displaced.

The skeleton of a cell is a set of all discs which are maximal in the object (see Section 3.2.1). The point or set of points which are the last ones to be eroded during our skeletonization process is equivalent to the center of the largest disc that could be contained in the cell.

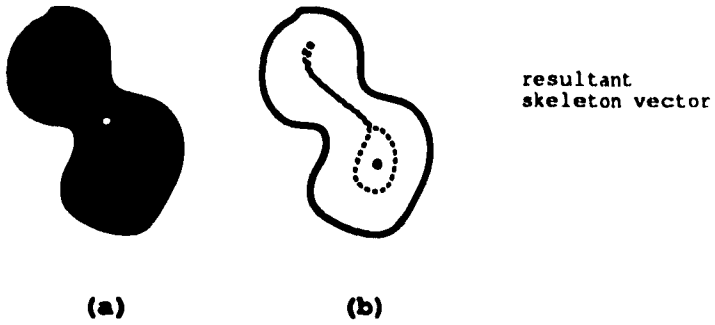

Figure 3.6: Two Cell Center Results. (a) Center of gravity for figure-8 cell shape is in the neck. (b) Erosion Center for figure-8 cell shape is in the middle of one of the knobs. The dotted contour represents an intermediate stage in the erosion process.

The Mitosis Recognition System uses this alternative definition of the cell center during tracking. However, *both* methods are computed and the centers stored for later comparison. An interesting result is that during cell division, the *erosion algorithm* finds the center to be in the middle of the largest figure-8 knob (Figure 3.6 (b)), whereas a *center of gravity* measure would put the center on the cleavage furrow 3.6 (a)). (Figure 3.6 is an approximation of actual results).

3.2.7 Data Summary

The output of the skeletonization procedure provides the following input data to the *understanding* module of the Mitosis Recognition System:

- number of significant concavities
- number of significant convexities
- number of skeleton branches
- skeleton resultant vector length
- true total skeleton length

- cell radius.

The cell's *radius* is taken to be the actual radius of a circle if the cell's perimeter were that of a perfect circle:

$$radius = \frac{boundary\_length}{2\pi}. \tag{3.14}$$

This number is quite large for cells with many boundary protrusions, so should not be used as a very accurate measurement of a cell's linear dimension. It is used in further computations to help normalize the data to provide some consistency across cells of varying sizes. For example, a measurement called the *normalized skeleton length* is computed by dividing the *true skeleton length* by the *cell radius*.

The skeleton coordinates, the boundary coordinates, and the cell centers are also recorded after each cell frame is processed. These data are used for off-line analysis.

Chapter 4

Cell Image Understanding

4.1 Introduction

4.1.1 Overview

Image understanding in the context of this thesis consists of determining the mitotic state of a segmented cell given a "significant" number of image frames in sequence taken over "short" time intervals. The fuzzy words "significant" and "short" depend upon the experimental context for quantification but already we begin to see the benefit of *fuzzy language* to help describe general processes which are difficult to represent precisely.

An expert system approach to solving problems first involves the *description* of human knowledge and experience towards the accomplishment of some task. This subjective expertise is put in the form of conditions and consequences in an attempt to automate similar tasks by computer. Certain expert system approaches, such as those employing fuzzy logic, use heuristic weights or measures of possibility or probability for the possible consequences, rather than Boolean *if-then* logic. Whatever the method, the attempt is to mimic the devices and knowledge used by experts to solve a particular problem or set of problems.

In our problem domain, cell biologists observe cells as they move and change shape over time. A cell's life cycle state can be easily inferred from general rules of shape changes and processes known by the cell biologist. Cell state inference is the primary task of the Mitosis Recognition

System (MRS). An accurate determination of cell state then provides the necessary information for the automatic tracking of a newly created cell once division has been recognized.

The *image understanding* section of the MRS consists of the steps required to transform the data retrieved from the skeleton and image processing of the cell images into general descriptions of the shape and process state of the cells being tracked. The data computed from the initial image processing and skeletonization (eg. number of skeleton branches) are first transformed into fuzzy data to linguistically describe the magnitude of the data in a consistent form. A fuzzy number in the context of this thesis is defined as a number defined from 0 to 1 which represents an approximate degree of belief in an associated predicate. For example, the predicate *number of skeleton branches is HIGH* would have the fuzzy number 1 associated with it if dozens of skeleton branches were computed for the cell under study.

Values for fuzzy *shape* are then computed by combining the evidence supplied by the fuzzy data. Fuzzy *shape* refers to the set of three fuzzy numbers associated with the predicates *shape is ROUND*, *shape is OBLONG*, and *shape is COMPLEX*.

Similarly, the fuzzy *state* consists of the set of fuzzy numbers and their predicates describing every possible cell state defined in the system. First, a preliminary fuzzy state is computed solely on the results of the fuzzy shape computation and other data measured for the cell under study. A final fuzzy state is then determined after considering the shape cycle process rules, the previous fuzzy state, and the new preliminary fuzzy state values.

4.1.2 Baseline Data and If-Then Rules

Rules describing the process of mitosis and cell movement in *if-then* format have been described previously by both Ferrie [14] and and Wheeler [59].

Ferrie ([14] [15]) uses a combination of *strategy rules* providing temporal descriptions and hypothetical models to represent the best estimate of potential daughter cells. The models, consisting of feature descriptions, are compared to the features of the objects in the image until an optimal match is found. The rules used in this method include tolerance comparisons directly in the antecedents to allow a flexible range of data to trigger conclusions. The rules, however, are not adaptable to changing experimental conditions. In the MRS, rules are at a level of abstraction *separate* from the measured data. Flexible data ranges are used in the MRS by mapping the measured data into fuzzy belief values for each evidence predicate.

In Wheeler's previously unpublished exploration of a cell's life cycle processes, [59], an interesting foundation to the present thesis is constructed. A hierarchy of rules is used, with the baseline data (such as *total number of segments*) integrated at the lowest level. These *baseline* data are translated into more linguistically appealing sentences if they trigger a true condition. *Baseline* data is defined here as the data from which all other inferences are derived; usually representing physical and measurable characteristics of the system.

For example, the following condition is evaluated:

Rule 4.1.1 `If the cell's data` total number of segments `is TRUE then the cell` *`has few segments`*.

The features computed from the baseline data are then used to determine

the *shape* of the cell, as in the following condition:

Rule 4.1.2 **If the cell is** *radial symmetric*
and the cell *has few segments*
then the cell is *round*.

Additional rules, consisting of rigid *if-then* conditions describing the allowable mitotic state transitions similar to those depicted in Figure 2.2, are used in conjunction with the *shape* results to determine a final cell state.

A downfall of this method is that a final conclusion cannot be reached, nor even estimated if there are any missing input data or unresolved facts. Bad or very erroneous input data result in a either a wrong conclusion, or no conclusion for that particular image frame.

It must be noted that both Ferrie's and Wheeler's previous methods of determining cell state are hard, binary rules, typical of many practical rule-based approaches. It is found that this Boolean method is not adequate for our application for the following reasons. Firstly, there is no notion of having a cell in an intermediate state (for example between anaphase and telophase). Secondly, estimation of the possibility or probability of different states with unclear or ambiguous data is not addressed.

Without these abilities a binary method becomes extremely brittle when used with less than ideal data. Ad hoc methods must be used to overcome problems of poor or ambiguous data. This makes such systems both overly complex and confined to limited experimental environments, characteristics which were ostensibly to be avoided by adopting a rule-based approach.

The algorithm presented in this thesis is extremely powerful in that

it resolves both of these issues using concepts of multi-valued and fuzzy logic within a simple method of evidence combination. Unlike Ferrie's method and Wheeler's original rule-based approach, time-consuming feature searching [15] and forward chaining [59] is avoided. These previous methods, however, were intended for *off-line* analysis of previously stored image data and are not used in conjunction with a real-time tracking system.

4.1.3 Fuzzy Cell Shapes and States

Instead of determining a *single* cell state, the Mitosis Recognition System computes a state vector referred to as a *fuzzy state* consisting of membership values from each previously defined state type. The state type having the largest membership value for the cell in question (at a particular instant in time) would correspond approximately to the single cell state produced by the binary rule-based systems of Ferrie and Wheeler [15] [59]. However, the fuzzy state representation in the MRS provides intermediate state representations by allowing memberships in more than one state for any given instant. Changes in cell state occur gradually in nature; this continuous type of process is more naturally represented using multiple values in the form of fuzzy sets.

The Mitosis Recognition System actually consists of three main rule-bases of conditions and weights used in the determination of the cell's fuzzy state. These are:

1. A *shape* rule-base to determine a cell's fuzzy shape from skeleton and other measurements.

2. A *state* rule-base to determine a cell's preliminary fuzzy state from the cell's shape and other measurements.

3. A *process* rule-base to determine a cell's predicted fuzzy state from previous state knowledge using only the shape cycle rules.

Shapes

A fuzzy *shape* is also computed in our thesis and is computed using the *shape* rule-base. The fuzzy shape or shape vector assigned to a cell is a set of three numbers which indicate the membership values in each of the three ideal shape categories we have defined: *ROUND, OBLONG,* and *COMPLEX*. The categories of shape are based on the number of skeleton branches and the features of the skeletons. The skeleton processing of a cell (a solid enclosed object with no holes) computes a skeleton having from zero branches (in the case of a perfect circle) to an infinite number of branches. The shape categories *ROUND, OBLONG* and *COMPLEX* encompass all possible skeleton variations, so we say that a cell's shape is completely defined under our fuzzy shape set. Because a cell's shape is defined completely in our fuzzy shape set, the sum of membership values of the three categories must equal unity.

For example, Figure 4.1 shows three examples of cells, one is very round, the second is peanut-shaped, and the third has several protrusions. The shape vector assigned to the roundish cell (a) consists of the following values: 0.9 *ROUND*, 0.1 *OBLONG*, and 0.0 *COMPLEX*. Note that the cell is *not* perfectly round, but is slightly oval, so has a small membership in the *OBLONG* shape type. Likewise, cell (b) is *mostly OBLONG* and cell (c) is *mostly COMPLEX*.

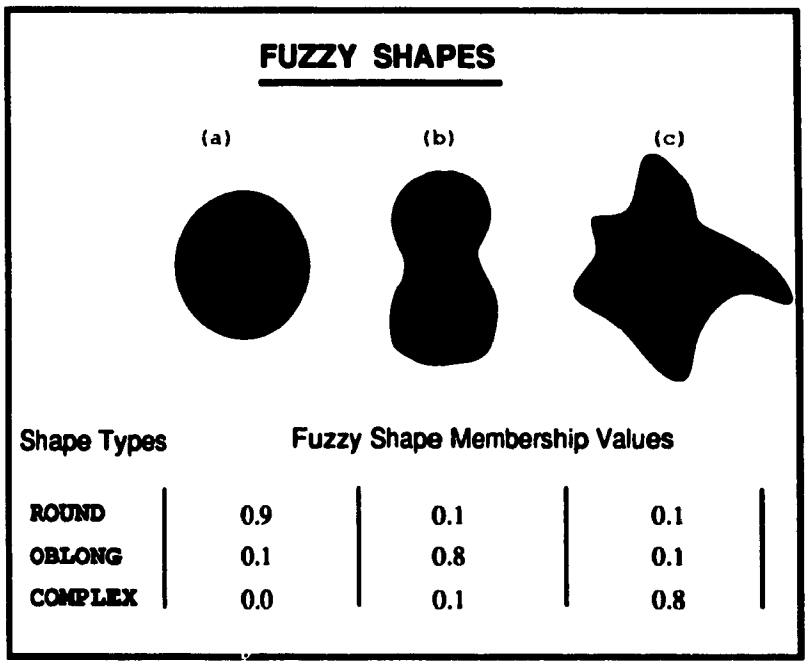

Figure 4.1: Examples of Fuzzy Shapes. Three cell examples: (a) roundish, (b) peanut-shaped, and (c) with protrusions. Each cell has membership values in each of the 3 fuzzy shape categories: *ROUND*, *OBLONG*, and *COMPLEX*.

States

The fuzzy state or state vector is analogous to the fuzzy shape in that the state vector contains a set of values equal to the number of possible state types defined in the system, and the sum of these values equals one (1) for a processed cell object. The *state* rule-base is used to compute a preliminary fuzzy state in the same manner as the fuzzy shape.

The valid state types are listed in Figure 4.2. The time-dependent relationships of the major state types over the cell cycle process are depicted in Figure 2.2. The additional states of **CONTACT, OVERLAP,** and **SEPARATION** depicted in Figure 4.2 are necessary for the possible occurrence of cells moving so close together as to form a single segmented object. This may occur when two cells collide or pass by each other at slightly different depths. Segmented objects in these situations often exhibit figure-8 shape features similar to that of mitosis. Hence the true state must be distinguished from a mitotic state.

A cell's state is automatically initialized with a high membership in the **UNDETERMINED** state at the beginning of a tracking experiment. This value will decrease and other states will become significant when evidence over several frames has been evaluated.

The **EXTINGUISHED** state may become significant if the accumulated error or uncertainty of a cell becomes high. If the **EXTINGUISHED** state value becomes higher than the other state categories, tracking of the cell discontinues.

Evidence Predicates

Both the *shape* and the *state* rule-bases in this system actually consist of a matrix of *evidence predicates* versus all possible *shape* or *state*

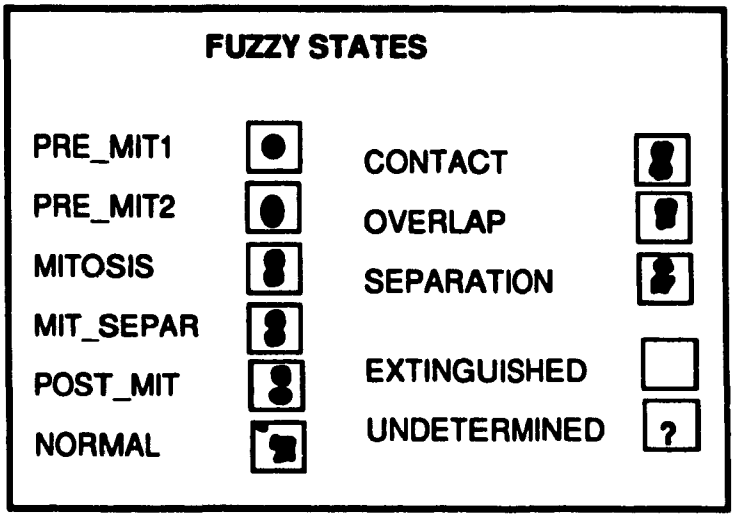

Figure 4.2: Fuzzy State Types. The fuzzy state type categories are listed, with examples depicting possible cell configurations associated with these categories.

types. Every piece of evidence is related to each shape or state type, in terms of how and to what degree the belief value of the evidence predicate affects the given type. For example, the evidence predicate *the cell shape is round* provides different information for the state outcomes of **NORMAL** and **PRE_MIT1**. A cell's roundness would provide evidence *against* a **NORMAL** state outcome, whereas it would provide strong evidence *for* a state outcome of **PRE_MIT1**.

The actual relationships between the evidence predicates and the state types are determined beforehand using linguistic weights (such as *important* or *very important*), and are stored in a static database to be used for the mitosis experiments. The *values* of the evidence predicates, however, are determined dynamically by the system as data is gathered.

In normal Boolean logic, the values of the evidence predicate would be either *true* or *false* (1 or 0). In this system we allow the predicate to take on a range of values between 0 and 1 to indicate how well the

data actually fit the evidence predicate. Most likely the cell data for a roundish cell do not indicate that the predicate *cell shape is round* is true as a certainty, but also indicate that the cell shape is slightly oblong. In this case, the value of the evidence predicate the *cell shape is round* will be close but not equal to 1, and the value of the evidence predicate the *cell shape is oblong* will be slightly greater than 0. These predicate values are multiplied by the predicate weights corresponding to the **PRE_MIT1** and **PRE_MIT2** states so that *strong* evidence is accumulated for the state type **PRE_MIT1** and *weak* evidence is accumulated for the state type **PRE_MIT2**. The preliminary fuzzy state or preliminary state vector outcome would contain a large **PRE_MIT1** membership and a small **PRE_MIT2** membership. We would then, as a preliminary estimate, consider that the cell be somewhere *in between* these two ideal states.

After a *preliminary* estimate of the fuzzy state, a *final* fuzzy state is computed by considering the fuzzy state of the cell from its previous image frame. Certain information collected from previous image frames is incorporated within *some* of the state evidence predicates (eg. previous cell size), but the information contributed by the time-dependent *shape cycle* process rules is used in a separate, final computation.

The shape cycle process rules, like the state rule-base, take the form of a matrix with a list of evidence predicates versus the list of state types. However, the evidence predicates in this case are a list of the previous state possibilities. The present implementation of this process matrix is like a truth-table in that it is an array of all *possible* and *impossible* transitions from a previous ideal state to the next possible state. The final fuzzy state of a cell for a given frame is computed by using this shape cycle matrix, the cell's previous fuzzy state, and the cell's present

preliminary fuzzy state.

In summary, the *image understanding* section of the Mitosis Recognition System consists of the following steps:

1. Convert the *image processing* data into fuzzy data for each *shape*-related evidence predicate.
2. Compute fuzzy *shape* values by summing the product of the evidence values weight factors for each shape type.
3. Collect the *state*-related evidence (including fuzzy shape) and compute the value for each state-related evidence predicate.
4. Compute preliminary fuzzy state values by summing the product of the evidence values and weight factors for each state type.
5. Compute the final fuzzy state by considering the shape cycle process rules, the previous fuzzy state, and the new preliminary fuzzy state values.
6. Determine if cell division has occurred by examining the membership value corresponding to the **POST_MIT** state type.

These steps are discussed in detail later in the chapter (Sections 4.3, 4.4, and 4.5). We first explore the knowledge and information represented by the shape and state rule-bases, and the method used to combine the evidence for a fuzzy state and shape solution.

4.2 Knowledge Representation, Uncertainty, and Fuzzy Sets

4.2.1 Background

The terms "fuzzy sets" and "fuzzy logic" have only been in use since the late 1960's to deal with data uncertainty and inexact reasoning ([11], [31]), however, mathematicians had been studying measures of information and uncertainty for many years prior to the theory of fuzzy sets [25].

Two principle measures of uncertainty were recognized before the theory of fuzzy sets became popular. Hartley proposed a method in 1928 based on classical set theory which measures an important aspect of ambiguity [25]. Shannon, in 1948, created a measurement pertaining to conflict or dissonance in evidence, which is formulated in terms of probability theory [25]. The purpose of both measures is to evaluate information in terms of uncertainty; they are generally referred to as *measures of information*. Specifically, these two classical measures are called *Hartley information* and *Shannon entropy*, respectively.

Shannon entropy can be justified as a significant measure of uncertainty and information on intuitive grounds. An event which has a very high probability of occurrence is *expected* to occur very often, so when it does occur, it is not generally noticed as providing any information. However, when it does *not* occur, the event is noticed, and seems to provide very important information. The information content, then, of an event x should be described by a decreasing function of the probability $p(x)$. The higher the probability of the occurrence of x, the less informative the observation.

The *Dempster-Shafer* theory of evidence ([32], [52]) describes a method of decomposing a set of evidence into unrelated pieces so that separate probabilities can be assigned to compute a final belief function, or a measurement of degree of belief. The resulting belief function becomes very complex with even a moderate number of evidence sets. The *Bayesian* approach, often mentioned in the literature, can be considered as a special case of the *Dempster-Shafer* theory [52]. With several assumptions, such as data independence, the Dempster-Shafer method becomes a practical tool. [1]

Uncertainty in evidence and rules can also be represented by *certainty* factors, such as in the MYCIN system [32] [19]. Uncertain reasoning in expert systems is usually accomplished by methods similar to probabilistic reasoning. Subjective probabilities are provided by experts to estimate the relationship between pieces of evidence, in the form of sentences, and conclusions that can be reached from these sentences. An expert system then computes solutions with estimates of certainty based upon these subjective factors. An *inference net* approach [19] is often used to propagate uncertainty or probability values through a tree-like network of hierarchical rules.

The Mitosis Recognition System uses this type of subjective reasoning in part, but, because of its very different types of evidence and their intricate relationships, certainty factors alone are insufficient. It is also difficult to formulate the time-dependent states and shape data into rules which form the tree-like network of an inference net. However, an analogous result is accomplished by the integration of our matrix-like rule-

[1] In [19] (p. 205) it is noted that Grosof ([21], [20]) generalizes probabilistic logic to encompass Dempster-Shafer theory, Bayesian updating in inference networks, and certainty factor methods.

bases.

A disadvantage of the standard probabilistic approach is its inability to represent the concept of *ignorance* [52]. *Lack of belief* and *disbelief* cannot be distinguished by this method; they are *both* represented by the negation of the proposition. There are practical situations which may occur in which there is not enough evidence to make either a true or false claim, or the evidence gathered may not even be very relevant to the particular problem. The Mitosis Recognition System avoids this difficulty in part by having strong negative weights to represent *disbelief*, large positive weights to represent *belief* and near-zero weights to represent *ignorance* or *irrelevance*.

In the limited domain covered by the Mitosis Recognition System, one must determine what types of evidence or clues can be measured, the reliability of these measurements, and how these can be used and combined to determine a cell's shape or state. For practical reasons, *intuitive* definitions of information components are described and then later used to build a method of evidence combination. The method is *not* justified on rigid theoretical grounds at present, but like many other systems which employ fuzzy techniques, is justified by producing valid and useful results under real experimental conditions [19]. Theoretical methods require a detailed mathematical understanding of the facts, their effects and relationships to each other, and the relationship of the facts on the final belief measurement. These mathematical relationships would be extremely difficult to establish thoroughly for a real imaging and biological shape problem such as the one posed in this thesis.

The method of evidence combination and evaluation of fuzzy subsets can be termed *decision making* [62]. Using linguistic terms in the formulation of decision functions can be an effective way of simplifying

complicated relationships of data towards imprecise objectives. Fuzzy sets with linguistic grades of membership are referred to as *fuzzy sets of type II* [62]. Linguistic terms are used to develop appropriate weights to describe the information content and the effect a piece of data has on a particular outcome. Ramakrishnan and Rao have developed a fuzzy weighted additive rule specifically for fuzzy sets of type II to approximate expert opinion [50].

The relevant factors and linguistic descriptions associated with the information content of the input data and the data's relationship to the final state and shape classifications are described in the next section.

4.2.2 Information Content of the Evidence

Each piece of data is a measurement with some uncertainty associated with it. Likewise, each piece of evidence may be more important for one state [2] than for another so will have a different *information content* for each state. The piece of data may provide *supporting* evidence or *conclusive* evidence, relative to each state. Another factor considers the type of evidence in a more global context. A particular evidence predicate may have more significance than another piece of evidence in the evaluation of the overall state of the cell. Data from the more significant predicate should be valued higher and so be given a stronger weight.

Four important factors are involved in describing the information content from a piece of evidence applied to a particular state. The first factor represents accuracy of the computed data, while the rest describe the relationships of the evidence to the ideal state type conclusions. The

[2]The procedure for evaluating the evidence is the same for both the *state* and the *shape*. For simplicity, this section refers only to the *state*.

numerical values of the evidence factors are established before the system is used.

Data Confidence

The *first* factor is due to uncertainties in data measurements. Imaging noise, artifacts, and other data problems, result in measurements with error. The measure of cell area, for example, is not the *true* area of the cell, since the boundary extraction process is not perfectly accurate. Either a confidence factor, or an uncertainty measure can be associated with the area which may be based on certain heuristics, or on past performance, such as the expected stability or expected area growth over a period of time. As well, statistics could be gathered comparing the area measurement achieved by the Mitosis Recognition System with a more accurate area measurement made by a cell physiologist, though this would be impractical. An area measure with a low confidence factor should not be emphasized in a determination of cell state: it would have *low* information content for all states.

Evidence Certainty

The *second* factor considers how an evidence predicate influences the outcome of each state type. To what degree does this evidence (if considered to be true) support or prove an outcome of the ideal state s? This factor, dealing with *certainty* or *proof* is illustrated as follows. In standard logic, the following rule would be acceptable as a true sentence:

Rule 4.2.1 `If the cell shape is` *COMPLEX*
`then the state is` **NORMAL.**

What about for the other shapes?

Rule 4.2.2 **If the cell shape is** *OBLONG*
then the state is NORMAL.

This statement (Rule 4.2.2) would usually not be true.

Rule 4.2.3 **If the cell shape is** *ROUND*
then the state is NORMAL.

This statement (Rule 4.2.3) would almost never be true.

What is needed is a measurement or qualifier of truth to answer the following question: If the antecedent is true, how does this affect the consequence? The three sentences above are modified to reflect a more realistic world:

Rule 4.2.4 **If the cell shape is** *COMPLEX*
then we are *very certain* **that the state is NORMAL.**

Rule 4.2.5 **If the cell shape is** *OBLONG*
then we are *quite certain* **that the state is** *not* **NORMAL.**

Rule 4.2.6 **If the cell shape is** *ROUND*
then we are *very certain* **that the state is** *not* **NORMAL.**

The modifiers are converted into *weights* or *certainty factors* which mark the degree of truth or belief of the consequence if the antecedent is true. A negative factor indicates the degree to which the consequence is believed *not* to occur if the antecedent is true. A factor of "don't know" or zero indicates that a relevant conclusion cannot be drawn from the belief of the antecedent.

Certainty Factors	Value	Importancy Factors
very certain	8	crucial
almost certain	6	very important
quite certain	4	important
somewhat certain	2	somewhat important
don't know	0	doesn't matter
somewhat certain it's *not*	-2	somewhat important it's *not*
quite certain it's *not*	-4	important it's *not*
almost certain it's *not*	-6	very important it's *not*
very certain it's *not*	-8	crucial it's *not*

Table 4.1: Certainty and Importancy Factors. Linguistic weights are mapped into numerical values for computing information content of evidence.

The translations between the modifiers and the certainty factors are shown in Table 4.1. Conclusive evidence [3] is given if the modifier is either *very certain* or *very certain it's NOT*, assuming that the first data confidence factor is negligible.

Evidence Importancy

The third factor can be considered as measuring *supporting* or perhaps *circumstantial* evidence, rather than measuring degrees of *conclusive* evidence.

Consider the reverse implication of Rule 4.2.1:

Rule 4.2.7 `If the state is` **NORMAL**

`then the cell shape is` *COMPLEX*.

[3] We say "conclusive evidence" meaning that given an *ideal* system, we would expect that this evidence would *really* be conclusive. In practice, "conclusive evidence" means as conclusive as possible considering the single, isolated piece of evidence.

We understand that it is *important* that the state have the feature *the cell shape is COMPLEX* but it does not in any way *prove* that the state is **NORMAL** under these conditions. However, the *Modus Tolens* of Rule 4.2.4 provides a way of proving that the state is *not* **NORMAL**:

Rule 4.2.8 `If the cell shape is` *not COMPLEX*
`then the state is` *not* **NORMAL**.

Using a qualifier, Rules 4.2.7 and 4.2.8 can be restated to emphasize this type of supporting evidence:

Rule 4.2.9 `It is` *very important* `that the cell shape is` *COMPLEX*
`for the state to be` **NORMAL**.

Similarly, for the other shapes:

Rule 4.2.10 `It is` *very important* `that the cell shape is` *not OBLONG*
`for the state to be` **NORMAL**.

Rule 4.2.11 `It is` *crucial* `that the cell shape is` *not ROUND*
`for the state to be` **NORMAL**.

The translations between these linguistic qualifiers (eg. *crucial* and *very important*) and the *importancy* factors are listed in Table 4.1.

The importancy factor also represents the relevance of evidence to the final determination of cell state. For example, an area measure is not relevant when considering its influence on determining a **NORMAL** cell state (when the cell is not undergoing mitosis) and does not provide much information for this state. The importancy factor would then be 0 for "doesn't matter" so as not to have an effect on the **NORMAL** state computation. However, the area of the cell may be important evidence for determining the states of cell **CONTACT** and cell **MITOSIS**.

Note the difference between a negative *importancy* factor and a negative *certainty* factor. A negative *importancy* factor indicates that it is important that the evidence predicate is not true for a particular state or shape outcome to be true, whereas a negative *certainty* factor indicates the degree to which a particular state or shape outcome is believed not to occur if the the antecedent is true.

Evidence Significance

The *fourth* factor represents the *significance* of an evidence predicate relative to other evidence predicates, independent of the measured data and the type of state. For example, the fuzzy evidence predicate: *previous size equals present size* is not as significant as the fuzzy evidence predicate: *shape is round*; thus it should have a smaller weight factor. Certain pieces of information are more significant, in terms of information content, to the final fuzzy state outcome than other pieces of information. The evidence significance factor differs from the importancy factor in that its weight is defined with respect to other pieces of evidence and is not evaluated for each individual state type. Furthermore, it is also used to adjust the effect of very similar contributing (ie. not independent) pieces of evidence.

The problem of deciding how a piece of evidence or data helps support or confirm a hypothesis is a difficult one. The four influencing factors described above are meant as guidelines in establishing the effect of the evidence on the possible conclusions. These factors are summarized as follows:

- *Data Confidence*: the relative *confidence* in the data itself, when considering measurement errors.

- *Evidence Certainty*: the factor representing how *certain* a particular conclusion is given that the individual piece of evidence is TRUE.

- *Evidence Importancy*: the factor representing how *important* a piece of evidence be TRUE given an expected conclusion.

- *Evidence Significance*: the factor representing the *significance* of a piece of evidence to the final determination of a conclusion, with respect to other pieces of evidence available and irrespective of the final conclusion.

4.3 Shape and State: Method of Evidence Combination

4.3.1 Definition of Variables

The method of evidence combination for the fuzzy shape and preliminary fuzzy state are identical. The method used is described in a mathematical context independent of the application in this section; the term *state* is used to represent either the fuzzy *shape* or the preliminary fuzzy *state*.

The variables and constants used in the evaluation of evidence in this section are described below.

Data Dimensions

- n_b: The total number of different baseline measurements.

- n_p: The number of evidence predicates (rule antecedents) to be used in the determination of state.

- n_s: The total number of defined or allowable state types in the system.

Weighting Factors

The data confidence measurement factors may or may not be determined before an experiment. The method of setting these factors depends upon the nature of the individual piece of data.

- m_i: the data confidence factors, where i is a corresponding baseline data measurement (from 1 to n_b).

The following three types of factors, described previously, are determined by an expert *prior* to an experiment and can be considered as constants *during* an experiment. In general, once these factors are determined, they never have to be changed unless the application or experimental conditions change considerably.

- c_{kj}: the evidence certainty factors, where k is the evidence predicate number (from 1 to num_p) and j is the state type (from 1 to num_s).

- r_{kj}: the evidence importancy factors, where k is the evidence predicate number (from 1 to n_p) and j is the state type (from 1 to n_s).

- g_k: the evidence significance factors, where k is the evidence predicate number (from 1 to n_p).

Positive and negative sums of the importancy and certainty factors for each state are also computed for the normalization function.

- $\mathcal{F}_{c(max)_j}$: the sum of the positive certainty factors for each state j.

- $\mathcal{F}_{c(min)j}$: the sum of the negative certainty factors for each state j.
- $\mathcal{F}_{r(max)j}$: the sum of the positive importancy factors for each state j.
- $\mathcal{F}_{r(min)j}$: the sum of the negative importancy factors for each state j.

Dynamic Variables

The following variables are computed dynamically by the system during an experiment. They depend upon the values of the baseline data and the input weight factors described above, and are defined in the next sections.

- y_i: the baseline data elements, with a total of n_b.
- p_k: the predicate belief value elements with values from 0 to 1 computed from the baseline data y_i, where k is the evidence predicate number (from 1 to n_p).
- $\mathcal{C}_j$: the total evidence certainty for each state type j.
- $\mathcal{R}_j$: the total evidence importancy for each state type j.
- $\mathcal{M}_{bel_j}$: the *modified* total belief for each state type j (including both certainty and importancy totals).
- $\mathcal{M}_{cer_j}$: the *modified* total certainty belief for each state type j.
- $\mathcal{M}_{imp_j}$: the *modified* total importancy belief for each state type j.
- $\mathcal{M}_{tot}$: the *sum* of all modified total beliefs over all states.

- s_j: the normalized state value for each state type j, with a total sum of 1 for all s_j's (over all j's).

4.3.2 Mapping Data to Evidence Predicate Values

The mapping of the data to the evidence predicates is *not* one-to-one. There exists a function $\mathcal{F}(y_i, p_k)$ for each predicate k which retrieves the appropriate baseline data element y_i and maps the measurement into a number from 0 to 1 onto the predicate value p_k:

$$y_i \longrightarrow p_k : 0 \leq p_k \leq 1 \tag{4.1}$$

A value of 0 indicates zero belief in the evidence predicate, whereas a value of 1 indicates total belief.

The data confidence factor m_i is applied *after* the mapping to reduce the belief in the evidence if the data measurement is not accurate. Alternatively, the m_i factors could be saved to compute an *overall* confidence measurement of the combined evidence and final state result. For simplicity, we assume *total* confidence in our data, setting all of the m_i's to 1. In practice this works reasonably well because we are concerned mainly with a general maximum cell state but not with the accuracy of the state values.

Computing the Total Evidence

We first compute importancy evidence sum $\mathcal{R}_j$, and certainty evidence sum C_j for each state j:

$$\mathcal{R}_j = \sum_{k=1}^{n_p} p_k \times r_{kj} \times g_k \tag{4.2}$$

$$C_j = \sum_{k=1}^{n_p} p_k \times c_{kj} \times g_k \tag{4.3}$$

It is at this point that we also take into account the effect of the significance factor g_k, so as to weigh pieces of evidence with greater significance more heavily.

Note that *formal* methods of combining evidence take into account *joint* probabilities and measures ([25], [52]). This is *not* done here since it is difficult to determine all the dependencies with each piece of cell data. Many practical applications use intuitive, ad hoc methods to evaluate the total evidence, supporting their practices by experimentation [19]. In the MRS, joint probabilities are not considered which result in some data to be "counted twice" during the evaluation. It is reasoned that when this occurs, more significance is then placed on these particular data and the particular pieces of evidence which they influence; in effect, there are *hidden* evidence significance factors g_k. Since an attempt is made to normalize the summed data, and the evidence significance factors are determined through experimentation, the effect of hidden joint measures is not significant on the overall conclusion.

The influence of extra evidence, which depends on data already covered by existing predicates, can be demonstrated by adding a repeated predicate in the total list of predicates, or by slightly increasing the evidence significance factor g_k of an evidence predicate k. If the normalization procedure (described below) is followed, then the duplicate piece of evidence does not have a large effect unless the total sum of the predicate certainty factors and the importancy factors for a particular conclusion is *small* and the piece of data y_i providing the evidence is much worse than the predicted data confidence factor m_i.

4.3.3 Computing and Normalizing the Fuzzy States

The total evidence sums for each state j should not be compared directly with each other before an attempt is made to normalize these values. An intuitive reason can be given by considering the following example.

The state **PRE_MIT2** has the shape feature of being partly round and partly oblong, but no other major attribute, apart from occurring *between* **PRE_MIT1** and **MITOSIS**. Since this state has many characteristics in common with these two neighbouring states, the **PRE_MIT2** state does not have strong weight factors associated with its evidence predicates. In other words, no data is *crucial* and no piece of evidence determines this state with certainty. So even if all the evidence contributes somewhat to this state, the total could easily be out-weighed by the evidence sums of **PRE_MIT1** or **MITOSIS**. Some states may also have more negative factors associated than positive, relying on the falseness of the other states to be established as true. This makes the evidence sum of a state very difficult to compare with the evidence sum of a neighbouring state which may naturally have more positive factors than negative.

To begin the normalization, we compute, for each state j, the largest possible *positive* evidence certainty sum $\mathcal{F}_{c(max)_j}$, the smallest possible *negative* evidence certainty sum $\mathcal{F}_{c(min)_j}$, the largest possible *positive* evidence importancy sum $\mathcal{F}_{r(max)_j}$, and the smallest possible *negative* evidence importancy sum $\mathcal{F}_{r(min)_j}$, given only the evidence factors c_{kj}, r_{kj}, and g_k:

$$\mathcal{F}_{c(max)j} = \sum_{k=1}^{n_p} c_{pkj} \times g_k \qquad \text{where } c_{pkj} = \begin{cases} c_{kj} & \text{if } c_{kj} \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

$$\mathcal{F}_{r(max)j} = \sum_{k=1}^{n_p} r_{pkj} \times g_k \qquad \text{where } r_{pkj} = \begin{cases} r_{kj} & \text{if } r_{kj} \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.5}$$

$$\mathcal{F}_{c(min)j} = \sum_{k=1}^{n_p} c_{pkj} \times g_k \qquad \text{where } c_{nkj} = \begin{cases} c_{kj} & \text{if } c_{kj} \leq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

$$\mathcal{F}_{r(min)j} = \sum_{k=1}^{n_p} r_{pkj} \times g_k \qquad \text{where } r_{nkj} = \begin{cases} r_{kj} & \text{if } r_{kj} \leq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.7}$$

A *cutoff* or *zero* point is defined for the condition in which we have no significant belief in the state. If either the importancy sum or the certainty sum is very negative (defined here as less than 1/3 of the total possible negative importancy or certainty) we set the (modified) total belief of the state to 0.

$$\text{If } \mathcal{R}_j \leq \mathcal{F}_{r(min)j}/3 \text{ then } \mathcal{M}_{belj} = 0 \tag{4.8}$$

or

$$\text{If } \mathcal{C}_j \leq \mathcal{F}_{c(min)j}/3 \text{ then } \mathcal{M}_{belj} = 0 \tag{4.9}$$

Otherwise, the modified total beliefs for the importancy ($\mathcal{M}_{impj}$) and the certainty ($\mathcal{M}_{cerj}$) are computed individually for each state, and then

combined using an Euclidean metric for the total modified belief $\mathcal{M}_{bel_j}$:

$$\mathcal{M}_{bel_j} = \mathcal{M}_{cer_j}^2 + \mathcal{M}_{imp_j}^2 \tag{4.10}$$

$$\mathcal{M}_{imp_j} = \frac{\mathcal{R}_j - \mathcal{F}_{r(min)_j}/3}{\mathcal{F}_{r(max)_j} - \mathcal{F}_{r(min)_j}/3} \tag{4.11}$$

$$\mathcal{M}_{cer_j} = \frac{\mathcal{R}_j - \mathcal{F}_{c(min)_j}/3}{\mathcal{F}_{c(max)_j} - \mathcal{F}_{c(min)_j}/3} \tag{4.12}$$

The modified total certainty and modified total importancy values are squared before they are summed in order to emphasize relatively high values which are found in either category. Some states, for example, may rely more heavily on the computation of importancy than certainty, and a simple sum of the two totals would tend to negate the effect of a very high importancy against a low or mediocre certainty. Hence, importancy or certainty totals which come very close to the ideal total for a state contribute significantly to the modified total belief $\mathcal{M}_{bel_j}$ value.

The *sum* of the modified total beliefs for each state is computed for the final normalization of the fuzzy state elements.

$$\mathcal{M}_{tot} = \sum_{j=1}^{n_s} \mathcal{M}_{bel_j} \tag{4.13}$$

The preliminary fuzzy state values are now normalized:

$$s_j = \frac{\mathcal{M}_{bel_j}}{\mathcal{M}_{tot}}. \tag{4.14}$$

We now have a set of states s_j whose sum is 1 and whose individual values represent the degrees of membership the measured object has in

each state.

4.4 MRS Computation of Shape and Preliminary State

4.4.1 Fuzzy Shape Computation

Computer vision researchers strive to describe objects in a way that is independent of their orientation, position, or motion. The Mitosis Recognition System must do the same so that the description of a changing cell can be analyzed over many different image sequences.

Chapters 2 and 3 discuss shape-related data obtained from various computations, such as from the skeleton processing. The *integration* of all these data to produce a single fuzzy shape set is not a trivial task. The first step in this integration is to convert the data measurements into belief values for the set of shape-related evidence predicates, or in other words, to perform the mapping indicated by Equation (4.1).

The shape-related evidence predicates pick out the relevant features of the measurements for the three shape categories. The left column of Table 4.2 is the complete list of the shape-related evidence predicates presently used in the system. Additional evidence predicates can easily be added as long as a corresponding mapping function $\mathcal{F}(y_i, p_k)$ is also added in the code.

The list of baseline data measurements y_i is shown in Table 4.3. Note that this array of baseline data is used for *both* the shape and the state computations since some of the data is shared. The mapping functions, however, would be different for the shape and state applications. Not all of the data in the array y_i are used for computing the values of the shape

Evidence Predicates p_k	Shape Types s_j		
	ROUND	**OBLONG**	**COMPLEX**
	r_{k1} c_{k1}	r_{k2} c_{k2}	r_{k3} c_{k3}
num of branches is LOW	8 6	8 4	-8 -8
num of branches is MED	-6 -6	0 0	6 6
num of branches is HIG	-8 -8	-8 -8	8 8
normalized skeleton length is LOW	6 8	-4 -8	-8 -8
normalized skeleton length is MED	-2 0	6 8	0 0
normalized skeleton length is HIG	-8 -6	-2 -6	2 8
skeleton vector measure is LOW	0 0	-8 -8	0 0
skeleton vector measure is MED	-2 0	-4 0	0 0
skeleton vector measure is HIG	-8 -8	8 8	-6 -6
ave branch length is LOW	8 8	-8 -8	-4 -4
ave branch length is MED	-4 -2	-4 -4	4 4
ave branch length is HIG	-8 -8	8 0	0 0
num crit convex pts is LOW	8 6	8 2	-6 -6
num crit convex pts is MED	-4 -4	-4 -4	4 4
num crit convex pts is HIG	-8 -8	-8 -8	8 8
num crit concave pts is LOW	8 2	0 0	0 0
num crit concave pts is MED	-8 -8	8 8	0 0
num crit concave pts is HIG	-8 -8	-8 -8	0 8

Table 4.2: Shape Rule-Base Matrix. Importancy factors r_{kj} and certainty factors c_{kj} used by the MRS for the cell division experiments presented in this thesis are listed for each evidence predicate p_k and each shape type s_j, respectively. The variables k and j represent the row number and column number, respectively.

Data Element Number i	Base-Line Data Description
1	cell radius
2	number of skeleton branches
3	true skeleton length
4	skeleton resultant vector length
5	existence of two "touching" components
6	sum of two touching component sizes
7	round shape belief
8	oblong shape belief
9	complex shape belief
10	previous size (in pixels)
11	object - neighbour distance
12	present object size (in pixels)
13	number of objects in window
14	number of significant convexities
15	number of significant concavities
16	closest neighbour size (in pixels)
17	previous average round shape belief
18	previous average oblong shape belief
19	previous average complex shape belief
20	tracking uncertainty factor

Table 4.3: Base-Line Data Elements y_i.

evidence predicates. For example, the entries for *cell shape* are used only by the fuzzy state computation which are updated *after* the fuzzy shape computation.

The mapping functions $\mathcal{F}(y_i, p_k)$, used to determine the evidence belief values for each evidence predicate from the baseline data array y_i, are coded as individual C functions for each evidence predicate *group*. An evidence predicate group can be considered as a fuzzy subset of the total set of evidence predicates. For example, the evidence predicates *num of branches is LOW*, *num of branches is MED*, and *num of branches is HIG* are considered as members of the fuzzy subset *number of branches*. Therefore, a description of the number of a cell's skeleton branches would consist of the set of these three belief numbers. Different evidence predicate groups are separated by a horizontal space in Table 4.2.

Once the array of shape evidence predicate values p_k are determined

from a data set for a particular image frame, the importancy $\mathcal{R}_j$ and certainty $\mathcal{C}_j$ evidence sums are computed for each shape j as in Equations (4.2) and (4.3). The importancy and certainty factors for each shape rule-base element are shown in Table 4.2. The linguistic translations of these weights are listed in Table 4.1. These weights were originally determined by intuitively applying the linguistic weights to each evidence predicate and their expected relationship to the shape type outcome. Some of the factors were tuned experimentally after some of these relationships and their effect on the final shape outcome were better understood under real experimental conditions.

It was found to be sufficient to set the evidence significance factors g_k to 1 for the fuzzy shape computation. Note, however, that a value of 1 for g_k is not always appropriate for the computation of fuzzy state.

The evidence importancy and certainty minimum and maximum sums are determined directly from the shape rule-base (Table 4.2) upon initialization using Equations (4.4), (4.5), (4.6), and (4.7). These constants are used to help filter the importancy and certainty evidence totals for each shape type (Equations (4.8), (4.9), (4.11), and (4.12)) before finally combining the two types of information (Equation (4.10)) for a modified total belief number for each shape.

Normalization of the fuzzy shape values is achieved by computing the sum of the modified total belief numbers for all three shapes (Equation (4.13)) and finding the fraction of belief for each shape type (Equation (4.14)). Each shape value s_j is a number from 0 to 1 representing the degree to which an object is *ROUND*, *OBLONG*, and *COMPLEX*. Examples of three different objects and their corresponding fuzzy shape values are shown in Figure 4.1.

4.4.2 Fuzzy State Computation

The computation of the fuzzy *state* follows the same procedure as the fuzzy *shape*. Baseline data measurements y_i (Table 4.3) are used to compute the belief values of the state evidence predicates listed in Table 4.4. There are considerably more evidence predicates or rules used for computing the cell's *state* than for the cell's *shape*. There are a total of *eleven* state types as compared to only *three* shape types. Some of the state types are very similar, and additional rules are needed to distinguish them. Many of the rules are required to specifically handle neighbouring cell problems of close contact, collision, and separation. As in the shape computations, the functions used to compute the predicate values p_k are coded as C programming functions for each state evidence predicate group.

The evidence importancy r_{kj} and significance g_k factors are multiplied by the evidence predicate values p_k and summed for each shape type for an *importancy* total $\mathcal{R}_j$ (Equation (4.2)). The evidence certainty c_{kj} and significance g_k factors are multiplied by the evidence predicate values p_k and summed for each shape type for a *certainty* total $\mathcal{C}_j$ (Equation (4.3)). The importancy and certainty factors for each state rule-base element are shown in Table 4.5. These weights, like those for the shape rule-base, were originally determined by intuitively deciding the *importance* of each piece of evidence to the individual state outcomes, and to what degree a piece of evidence would *support* or prove a particular state type.

The evidence significance factors, shown in Table 4.4, were determined experimentally after learning the relative impact of each evidence predicate on the final state outcome. The factor of 3 in the case of the shape related predicates was used to place more emphasis on the evi-

State Rule No.	State Evidence Predicate Description	Evidence Significance Factor g_k
1	shape is ROUND	3
2	shape is OBLONG	3
3	shape is COMPLEX	3
4	cells are near contact	1
5	two components in obj	3
6	sum of component sizes < prev obj size	1
7	sum of component sizes > prev obj size	1
8	sum of component sizes = prev obj size	1
9	prev size + neighbour's prev size < present obj size	1
10	prev size + neighbour's prev size > present obj size	1
11	prev size + neighbour's prev size = present obj size	1
12	contact size < present size	1
13	contact size > present size	1
14	contact size = present size	1
15	prev size < present size	1
16	prev size > present size	1
17	prev size = present size	1
18	invalid number of objs	1
19	tracking is uncertain	1
20	prev shape average is ROUND	3
21	prev shape average is OBLONG	3
22	prev shape average is COMPLEX	3
23	num crit convex pts is LOW	1
24	num crit convex pts is MED	1
25	num crit convex pts is HIG	1
26	num crit concave pts is LOW	1
27	num crit concave pts is MED	1
28	num crit concave pts is HIG	1
29	neighbour size + present size < prev size	1
30	neighbour size + present size > prev size	1
31	neighbour size + present size = prev size	1
32	neighbour size < present size	1
33	neighbour size > present size	1
34	neighbour size = present size	1

Table 4.4: State Evidence Predicate List and Evidence Significance Factors. Evidence predicate groups are separated by blank lines.

EVIDENCE PREDICATE	PRE-MIT1		PRE-MIT2		MIT-OSIS		MIT-SEPAR		POST-MIT		CON-TACT		OVER-LAP		SEPAR-ATION		NORM-AL		EXTING-UISHED		UNDETER-MINED	
1	8	8	-4	-4	-8	-8	-8	-8	4	8	-8	-8	0	0	-8	-8	-4	0	0	0	-1	-1
2	-4	1	8	4	8	4	2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
3	-8	-8	-8	-8	-8	-8	0	0	0	0	0	0	0	0	0	0	8	8	0	0	0	0
4	-6	-6	-8	-8	-8	-8	-8	-8	8	8	-8	-8	-8	-8	8	4	0	0	0	0	0	0
5	-8	-8	-8	-8	-8	-8	8	8	-8	-8	2	4	-8	-8	8	4	-8	-8	0	0	0	0
6	0	0	0	0	0	0	-8	-8	0	0	-8	-8	0	0	-8	-8	0	0	0	0	0	0
7	0	0	0	0	0	0	-8	-8	0	0	8	8	0	0	-8	-8	0	0	0	0	0	0
8	0	0	0	0	0	0	8	8	0	0	-8	-8	0	0	8	8	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	-1	-1	-6	-6	8	8	-8	-8	-2	-2	0	0	0	0
10	0	0	0	0	0	0	0	0	-1	-1	-6	-6	-8	-8	8	8	8	8	0	0	0	0
11	0	0	0	0	0	0	0	0	-1	-1	8	8	-8	-8	-8	-8	-8	-8	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	-8	-8	4	4	-8	-8	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	-2	-2	8	8	8	8	8	8	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	8	8	-8	-8	0	0	-8	-8	0	0	0	0
15	1	0	2	0	0	0	-2	-2	-8	-8	8	8	-8	-8	1	1	-2	-2	0	0	1	1
16	-1	0	-3	0	-2	0	-2	-2	8	4	-8	-8	4	4	-1	-1	-2	-2	0	0	1	1
17	0	0	0	0	0	0	3	0	-8	-8	-8	-8	-4	-4	0	0	2	2	0	0	-1	-1
18	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	4	4	0	0
19	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	-8	8	8	0	0
20	4	6	3	2	6	-6	-8	-8	-8	-8	0	0	0	0	0	0	-1	-1	0	0	-1	-1
21	1	1	5	2	8	2	6	2	8	2	0	0	0	0	2	0	-1	-1	0	0	-1	-1
22	-1	-1	8	-8	-8	-8	-1	0	-1	0	0	0	1	0	1	0	6	6	0	0	0	0
23	8	2	8	2	8	0	0	0	0	0	0	0	0	0	0	0	-8	-4	0	0	0	0
24	8	4	-8	-4	-4	-2	0	0	0	0	0	0	0	0	0	0	4	6	0	0	0	0
25	-8	-8	-8	-8	-8	-8	-2	-2	-2	-2	0	0	0	0	0	0	4	8	0	0	0	0
26	8	0	2	0	0	0	-2	-2	2	0	0	0	0	0	0	0	0	0	0	0	0	0
27	-8	-8	-4	-2	2	4	8	6	-2	-2	0	4	0	0	0	0	0	1	0	0	0	0
28	-8	-8	-8	-8	-4	-4	-2	-2	-4	-4	0	0	0	0	0	0	0	8	0	0	0	0
29	0	0	0	0	0	0	0	0	-8	-8	0	0	0	0	-8	-8	-8	-8	1	1	1	1
30	0	0	0	0	0	0	0	0	-8	-8	0	0	0	0	-8	-8	4	0	0	0	0	0
31	0	0	0	0	0	0	0	0	8	4	0	0	0	0	8	4	-8	-8	0	0	0	0
32	0	0	0	0	0	0	0	0	-8	-8	0	0	0	0	0	0	1	1	0	0	0	0
33	0	0	0	0	0	0	0	0	-8	-8	0	0	0	0	0	0	1	1	0	0	0	0
34	0	0	0	0	0	0	0	0	8	4	0	0	0	0	2	2	0	0	0	0	0	0

Table 4.5: States Rule-Base Matrix. Importancy factors r_{kj} and certainty factors c_{kj} used by the MRS for the cell division experiments presented in this thesis are listed for each evidence predicate p_k and each state type p_j, respectively. The variables k and j represent the row number and column number, respectively.

dence directly related to shape. Other evidence types are considered less significant and given a lower value of g_k.

The evidence importancy and certainty sums, computed with the Equations (4.4), (4.5), (4.6), and (4.7) upon initialization, are used to filter the importancy and certainty evidence totals for each state type as was done with the shape type totals (Equations (4.8), (4.11), (4.9), and (4.12)). The two types of information are combined (Equation (4.10)) to produce a modified total belief number for each state.

Finally, normalization of the fuzzy state values is achieved by sum-

ming the modified total belief numbers for all clever state types (Equation (4.13)) and computing the fractions of belief (Equation (4.14)). As with the fuzzy shape, the sum of the state values is 1, and each individual state value is a number from 0 to 1 representing the degree to which an object belongs to that particular state.

At this point the computed fuzzy state is only a *preliminary* one; the cell shape cycle process information (eg. of Figure 2.2) has not yet been introduced.

Most of the data used for the preliminary fuzzy state computation are from the present image frame. However, some data which depend upon previous image frames have also been incorporated into the rules. These data include a running weighted average of the previous shape. This is to add some additional stability in the preliminary state computation if an aberrant shape computation occurs due to imaging artifacts or segmentation errors. In effect, the computation provides a small amount of data smoothing over time. Smoothing across previous image frames is justified *practically* by realizing that changes in a cell's shape occur continuously. The fuzzy shape values should not change drastically from one frame to the next, given a frame rate suitable to catch general changes in pseudopod behaviour and ideal image segmentation. Additional advantages of integrating an object's shape features into a fuzzy shape representation are that a shape average computation is possible, it is quantitatively meaningful, and it can actually be used in further mathematical analysis; namely the computation of the fuzzy state.

The smoothing function used for shape averaging over *time* is:

$$m_k(n) = (s_k(n) + 3 \times m_k(n-1))/4 \qquad (4.15)$$

where m_k is the time-averaged shape at image sequence n, and s_k is the value of one of the fuzzy shape types (*ROUND, OBLONG* or *COMPLEX*). Note that the shape value from the image frame n is added with a weight of 1/4, to the previous average with a weight of 3/4. This fuzzy shape average is then used in the *next* image frame $(n + 1)$ during the preliminary state evaluation.

Other data used from previous image frames include the data representing object pixel size, average neighbour size, and collided object size. These data are used to help monitor changes in cell area for possible growth and for determining drastic changes such as during cell division, cell collision, and segmentation problems.

4.5 Integration of Process Information

The preceding section describes the method used for combining the evidence, at a particular instant in time, to produce a fuzzy measurement set. In practice, this method is first used to produce a fuzzy shape and then applied to compute a preliminary fuzzy state based solely on the measured evidence. Intuitively, this preliminary fuzzy state is like a "guess" of the state, using image processing information from the present frame along with some hints from the previous frames, but with no knowledge or biological expertise about how a cell's states should progress through its life cycle. A *final* fuzzy state is computed from both the preliminary fuzzy state and the time-dependent process rules.

The process rules consist of the possible transitions from one ideal state to another in a matrix representation similar to the state and shape rule-bases (Tables 4.5 and 4.2). The evidence predicate list of the process rule-base (Table 4.6) consists of the previous state type possibilities

PREVIOUS STATE	PRE-MIT1	PRE-MIT2	MIT-OSIS	MIT-SEPAR	POST-MIT	CON-TACT	OVER-LAP	SEPAR-ATION	NORM-AL	EXTING-UISHED	UNDETER-MINED
PRE_MIT1	1	1	0	0	0	1	0	0	1	1	0
PRE_MIT2	1	1	1	0	0	1	0	0	1	1	0
MITOSIS	0	1	1	1	1	0	0	0	1	1	0
MIT_SEPAR	0	0	1	1	1	0	0	0	0	1	0
POST_MIT	0	0	0	0	0	1	0	0	1	1	0
CONTACT	0	0	0	0	0	1	1	1	0	1	0
OVERLAP	0	0	0	0	0	1	1	1	0	1	0
SEPARATION	0	0	0	0	0	1	0	1	1	1	0
NORMAL	1	0	0	0	0	1	0	0	1	1	0
EXTINGUISHED	0	0	0	0	0	0	0	0	0	1	0
UNDETERMINED	1	1	0	0	0	0	0	0	1	1	1

Table 4.6: Process Rule-Base Matrix.

for each row k, for example: *previous cell state is* **PRE_MIT1**.

The columns in Table 4.6 are the same state types used in the preliminary state rule-base. One could use the methodology of the previous sections in the representation of importancy and certainty factors, but these factors cannot be accurately estimated until more data are gathered. The Mitosis Recognition System itself could be used in the gathering of these data, so that we can more accurately apply reasonable information factors to the process rules in a future implementation.

In the present implementation, the process rule-base consists of *true-false* binary factors (Table 4.6). We know for certain which state transitions *may* occur after particular states, but we cannot at present judge the frequency or the probability of these occurrences. The value of 1 (or *true*) in Table 4.6 represents the possibility that given the previous state $s_k(n-1)$ is believed, the present state $s_j(n)$ may follow. The value of 0 indicates that it is *not* possible for a cell to make the transition from $s_k(n-1)$ to $s_j(n)$. The variable n indicates the image frame sequence number, increasing in time. Some states allow several different transitions, but others are more limited. All states except **POST_MIT** allow re-occurrence of the same state.

The combination of the process rules, previous fuzzy state, and preliminary fuzzy state is a simple procedure with only a few operations, hence the resulting final fuzzy state is computed rapidly.

The first step is to compute process transition weights by combining the previous fuzzy state data with the process rule-base. Elements p_{kj} of each row k of the process rule-base are multiplied by the previous fuzzy state values $s_k(n-1)$ for each k to produce new weighted process elements w_{kj} for all rows k and columns j:

$$w_{kj} = s_k(n-1) \times p_{kj} \,. \tag{4.16}$$

A process state vector is then computed for each column j by considering the values of the weighted previous state evidence elements w_{kj} within the column j. In this step we simply take the *maximum value* of all w_{kj} elements within the column to be the unnormalized total predicted belief value, $\mathcal{T}_{bel_j}$, over all rows k:

$$\mathcal{T}_{bel_j} = MAX(w_{kj}) \tag{4.17}$$

This result is analogous to the modified total belief variable $\mathcal{M}_{bel_j}$ during the preliminary state computation of the previous section. Normalization is left for a later step.

The total predicted belief vector consisting of elements $\mathcal{T}_{bel_j}$ can be considered as a present state probability prediction. Given *only* the previous state information and the process rules, it specifies what state is expected for the tracked cell during the present image frame.

The final step simply combines this present state prediction $\mathcal{T}_{bel_j}$ with the preliminary fuzzy state s_j computed in the previous section from the image data. The two arrays are *multiplied* because of their probabilistic

form. For instance, the probability of the occurrence of state j would be computed from the chance that:

1. the state is *predicted* with the belief (or quasi-probability) $\mathcal{T}_{bel_j}$ and
2. the state couid occur from the preliminary state belief s_j.

The joint belief is computed as follows:

$$\Phi_j = \mathcal{T}_{bel_j} \times s_j \tag{4.18}$$

for all columns j.

The result vector Φ_j is the final *unnormalized* fuzzy state. Normalization is required to produce the final fuzzy state Φ_{norm_j}:

$$\Phi_{norm_j} = \frac{\Phi_j}{\sum_{k=1}^{n_s} \Phi_k}. \tag{4.19}$$

We now have a set of values representing a cell's fuzzy state. This fuzzy state can now be evaluated for control decisions, such as the automatic tracking of a second cell after cell division has occurred.

4.6 Cell Division and the Decision Function

Once a final fuzzy state has been computed, it must be analyzed to determine the occurrence of cell division. At what point can we believe, with a high degree of certainty, that a cell has completed mitosis and two daughter cells should be tracked?

Multiobjective tasks can themselves be formulated in fuzzy logic terms [62]. We could also use fuzzy conditions to determine the degree of certainty of cell division. However, since we only have two possible behaviours from any decision (to track or not to track an additional cell) a

single condition is sufficient. This single condition is as follows: when the membership or belief value corresponding to the state type **POST_MIT** is larger than all other state type values in a cell's fuzzy state set *and* is non-zero, we believe that cell division has occurred.

Such a precise condition can only be used for the post mitotic process state; the same rule could *not* be used with great confidence for other states, such as the **NORMAL** or **PRE_MIT1** states. This is because there remains a certain amount of ambiguity in the outcomes of these overlapping states. It can be shown that a precise goal can be obtained using fuzzy descriptions as long as the fuzzy observations become more and more precise when the goal is approached [31]. Our control goal is to detect cell division. Cell division is believed to occur when the precise fuzzy observation indicates a **POST_MIT** maximum state value.

An additional decision function is also required for the **EXTINGUISHED** state. This state is only used if the accumulated data uncertainty is very high or if an unacceptable processing condition has occurred. An unacceptable processing condition could be an invalid number of objects within the window (eg. no objects) or a high, unexpected change in the number of objects. Data uncertainty is accumulated during cell collision and overlap condition. Presently, we do not have a mechanism to guarantee that the cell being tracked is the same one being tracked after collision and separation has occurred. The best we can do is predict that after collision, the tracked cell will retain its general characteristics and position from *before* collision. A degree of uncertainty is estimated and accumulated each time a tracked cell interferes with another cell in the image. When the value for the **EXTINGUISHED** state becomes higher than the other states, tracking and subsequent pro-

cessing of the cell ceases.

Chapter 5

Experiments and Results

5.1 Description of Method

The Mitosis Recognition System (MRS) was tested using cell images previously recorded on computer hard disk. Image sequences were read and processed in real-time to simulate the environment of TRACES, the automated TRAcking CEll System. Several different types of dividing cells were examined: slime moulds, fibroblasts, and cancerous blood cells.

A typical experiment proceeds as follows. The first image of an experimental sequence is displayed using the MRS environment on a colour or grey-level SUN workstation monitor. Individual cells are selected for tracking by enclosing each cell in their own window. For optimal performance, these user-defined windows should be large enough to contain the entire cell after expected growth, but small enough as to not permit too many adjacent cells to be processed. For example, see the first frame in Figure 5.1. Once cell selection is complete, the tracking is initiated, and the next image frame in the experimental sequence is displayed. The MRS proceeds automatically from this point: the cells are segmented, the boundaries are determined, skeletons are computed, the cell center is found, and finally, the fuzzy shapes and states of each cell are resolved. The MRS program environment displays the results of the shape and state computation adjacent to each cell, and the position of the cell's window is adjusted to account for the change in the cell's central position. If the cells do not divide, but simply change shape and move about,

the MRS program automatically tracks the cells across the screen.

The MRS program constantly monitors the changes in cell state for each image frame. If cell division is detected by the expert system, a new cell window is automatically created on one of the newly formed daughter cells (Figure 5.2). The changes in shape, state, and cell position are recorded in output data files for off-line analysis. The cell's skeleton is also recorded for each frame, and can be used for more detailed shape analysis.

5.2 Results

Figures 5.1 and 5.3 are cell sequence examples of a dividing slime mould and dividing fibroblast, respectively. The slime mould frames are displayed in these figures at approximately 30 second intervals, whereas the fibroblast frames are shown at approximately 15 second intervals. The data was actually recorded using a faster frame rate; intermediate frames were removed for illustration purposes.

The segmentation of the cells proceeded very consistently. Occasionally, however, fading of the cell body would result in poor segmentation. For example, Figure 5.3 shows poor results in frame 11. Fortunately, because of the use of previous knowledge in the shape and state analysis, the expert system is able to recover from an isolated poor segmentation.

Skeletonization proceeded quickly and produced consistent results in most cases. Figure 5.1 shows that short, single branched skeletons were computed for roundish cells, but became elongated as mitosis proceeded. Cells with more complex shapes produced multi-branched skeletons.

Cell shapes and cell states were computed automatically for each cell image, and by monitoring the state values for cell separation and post

mitosis, the expert system was able to determine accurately that cell division occurred. Figure 5.1, Figure 5.2, Figure 5.3 and Figure 5.4 are examples of dividing cells and the subsequent tracking of daughter cells.

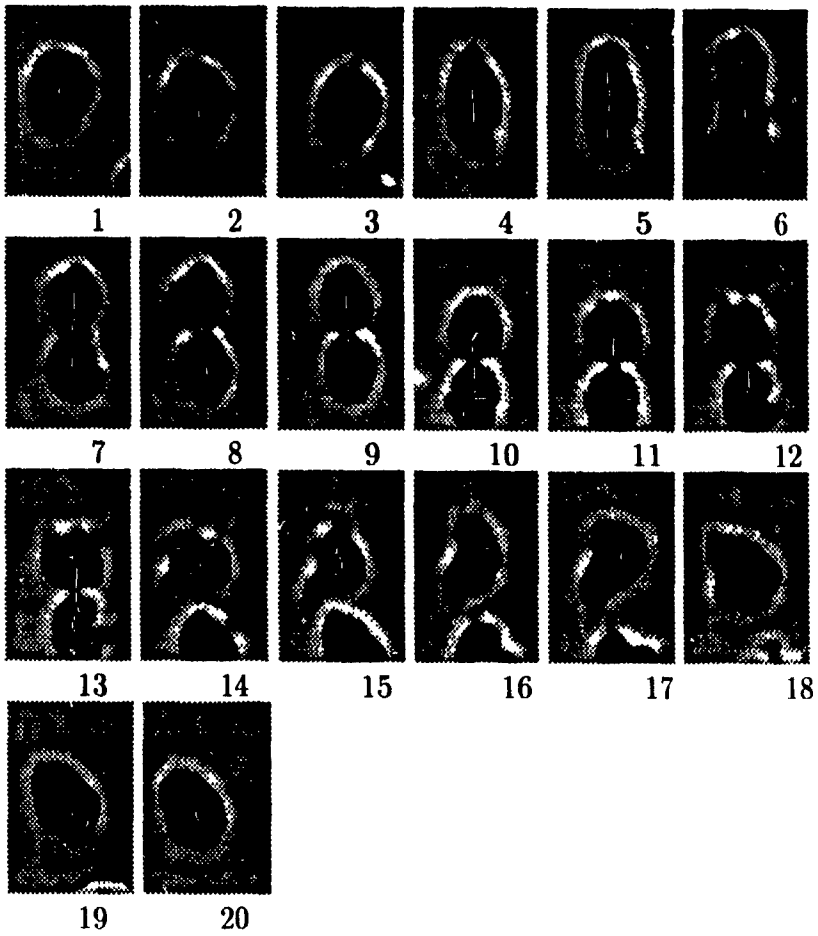

Figure 5.1: Dividing slime mould. Cells are filled in *black* within their grey-level backgrounds. Skeletons are represented in white. After cell separation (frame 14), a daughter cell is subsequently tracked (frames 15 through 20).

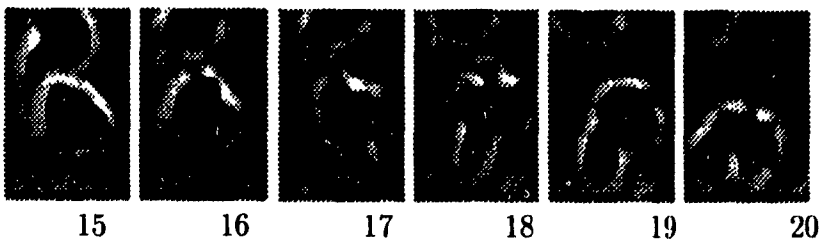

Figure 5.2: Tracking of new cell after division. A new tracking window is created after frame 14 to track the additional daughter cell.

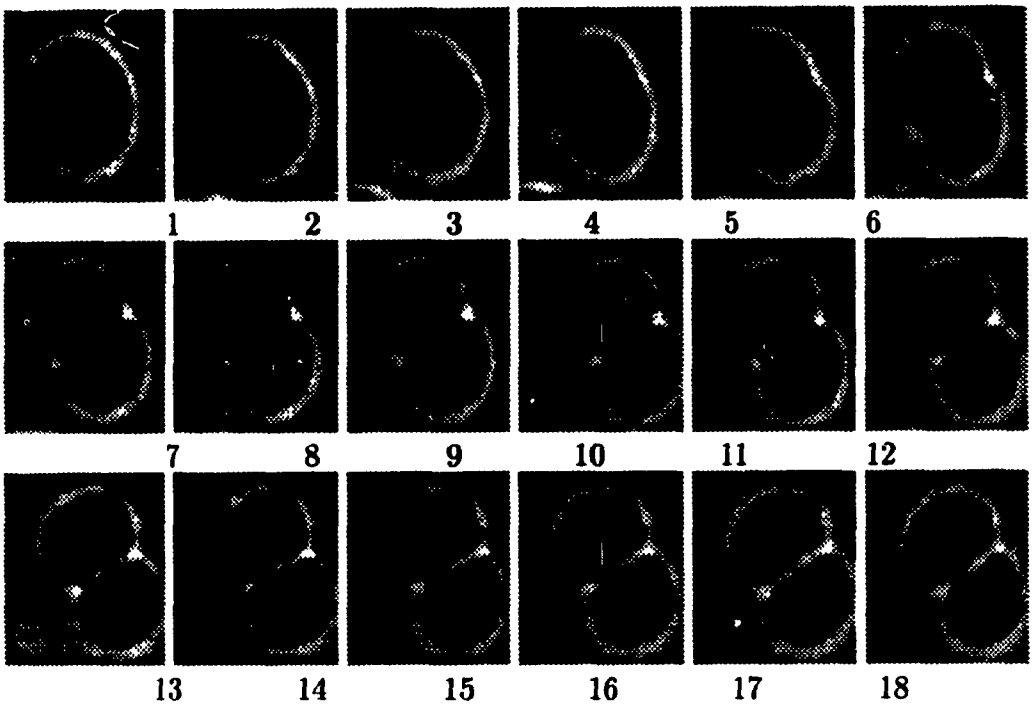

Figure 5.3: Dividing fibroblast. Cells are filled in *black* within their grey-level backgrounds. Skeletons are represented in white. After cell separation (frame 14), a daughter cell is subsequently tracked (frames 13 through 18).

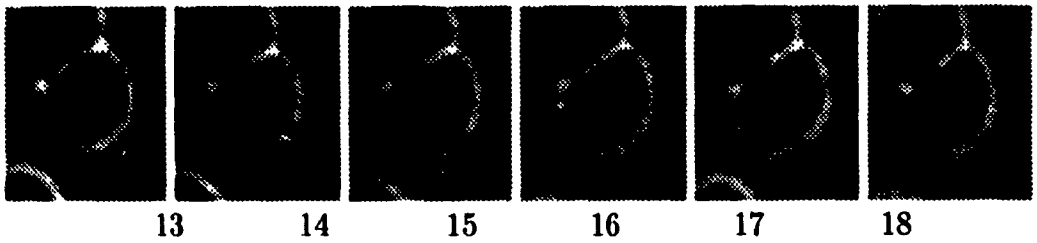

Figure 5.4: Tracking of new cell after division. A new tracking window is created after frame 12 to track the additional daughter cell.

Chapter 6

Conclusions

The application of computer vision techniques to the field of cell biology has tremendous potential. Present experimental methods used by cell biologists to track and monitor microscopic shape changes in cells involve labour-intensive work. Digital image recording, processing, and automatic tracking of cells has obvious benefits in the acquisition of experimental cell data.

In particular, the study of cell proliferation and the inheritance of traits associated with changes in membrane shape and cell locomotion can benefit considerably by a computer vision system that can recognize cell division and automatically monitor changes in cell shape and position. These capabilities were successfully developed and applied in the Mitosis Recognition System (MRS).

Apart from the biological benefits, this thesis explores issues inherent to a complete computer vision system, from image segmentation through a final expert system control decision. Special segmentation techniques are developed specifically for microscopic unstained cell images. Shape analysis is performed using a combination of boundary and region-based analysis methods. In particular, a method of skeletonization is applied to the problem of cells undergoing shape changes characteristic of mitosis. The data retrieved from the segmentation and shape analysis are combined and analyzed to produce fuzzy descriptions of cell shape. Finally, through a method of fuzzy evidence combination, the system integrates past cell state knowledge, present cell state, and cell process rules into

a final fuzzy cell state. The final cell state is monitored for changes in the post-mitosis and cell separation conditions to determine the actual occurrence of cell division.

References

[1] P. Marlene Absher, R. G. Absher, and W. D. Barnes. Genealogies of clones of diploid fibroblasts. *Experimental Cell Research 88*, pages 95–104, 1974.

[2] E. J. Ambrose and Dorothy M. Easty. *Cell Biology.* Thomas Nelson and Sons LTD, 1970.

[3] C. Arcelli. Pattern thinning by contour tracing. *Computer Graphics and Image Processing*, 17:130–144, 1981.

[4] C. Arcelli and G. Sanniti di Baja. A one-pass two-operations process to detect the skeletal pixels on the 4-distance transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-11(4):411–414, April 1989.

[5] Klaus Bayreuther, H. Peter Rodemann, Pal. I. Francz, and Katharina Maier. Differentiation of fibroblast stem cells. *J. Cell Science Suppl.*, 10:110–130, 1988.

[6] H. Blum. Biological shape and visual science (part I). *Journal of Theoretical Biology*, 38:205–287, 1973.

[7] H. Blum and R. Nagel. Shape description using weighted symmetric axis features. *Pattern Recognition*, 10:167–180, 1978.

[8] F. L. Bookstein. The line skeleton. *Computer Graphics and Image Processing*, 11:123–137, 1979.

[9] A. Boyarsky and P. B. Noble. A markov chain characterization of human neutrophil locomotion under neutral and chemotactic conditions. *Canadian Journal of Physiology and Pharmacology*, 55:1–6, 1977.

[10] Marianne Bronner-Fraser and Scott E. Fraser. Cell lineage analysis reveals multipotency of some avian neural crest cells. *Nature*, 335:105–119, 1988.

[11] S. S. L. Chang and L. A. Zadeh. On fuzzy mapping and control. *IEEE Trans. Sysst. Man Cybern*, SMC-2:30–35, 1972.

[12] A. de Boisfleury-Chevance, B. Rapp, and H. Gruler. Locomotion of white blood cells: A biophysical analysis. *Blood Cells*, 15:315–333, 1989.

[13] A. R. Dill, M. D. Levine, and P. B. Noble. Multiple resolution skeletons. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-9(4):495–504, July 1987.

[14] F. P. Ferrie. Experiments in tracking the morphologies of proliferating cell cultures by automatic picture processing. Master's thesis, McGill University, Electrical Engineering Department, Montréal, QC, Canada, September 1979.

[15] F. P. Ferrie, M. D. Levine, and S. W. Zucker. Cell tracking: a modeling and minimization approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-4(3):277–291, May 1982.

[16] H. Freeman. On the encoding of arbitrary geometric configurations. *IRE Transactions on Electronic Computers*, EC-10(2):260–268, June 1961.

[17] G. Gallus and P. W. Neurath. Improved computer chromosome analysis incorporating preprocessing and boundary analysis. *Phys. Med. Biol. (GB)*, 15(3):435–445, July 1970.

[18] D. Gauthier, M. D. Levine, and P. B. Noble. Principles of object detection for an automated cell tracking system. In D.-P. Häder, editor, *Image Analysis in Biology*, chapter 2. CRC, 1991.

[19] M.R. Genesereth and N.J. Nilsson. *Logical Foundations of Artificial Intelligence*. Morgan Kaufmann, Los Altos, CA, USA, 1987.

[20] B. N. Grosof. Evidential confirmation as transformed probability. In L. N. Kanal and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence*, pages 153–166. North-Holland, New York, 1986.

[21] B. N. Grosof. An inequality paradigm for probabilistic knowledge. In L. N. Kanal and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence*, pages 259–275. North-Holland, New York, 1986.

[22] S.-B. Ho and C. R. Dyer. Shape smoothing using medial axis transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(4):512–520, July 1986.

[23] R. S. Kerbel. Tumor heterogeneity, invasion and metastasis. *Invasion Metastasis*, 2:61–75, 1982.

[24] J. Kittler and J. Illingworth. Minimum error thresholding. *Pattern Recognition*, 19(1):41–47, 1986.

[25] George J. Klir and Tina Folger. *Fuzzy Sets, Uncertainty and Information.* Prentice Hall, 1988.

[26] J. J. Koenderink. Scale-time. *Biological Cybernetics*, 58:159–162, 1988.

[27] J. J. Koenderink and A. J. van Doorn. Dynamic shape. *Biological Cybernetics*, 53:383–396, 1986.

[28] J. M. Lackie. *Cell Movement and Cell Behaviour.* Allen and Unwin, London, England, 1986.

[29] D. T. Lee. Medial axis transformation of a planar shape. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(4):363–369, July 1982.

[30] E. T. Lee. Shape-oriented chromosome classification. *IEEE Transactions on Systems, Man, and Cybernetics*, pages 629–632, November 1975.

[31] Edward T. Lee. Applications of fuzzy set theory to image sciences. *Journal of Cybernetics*, 10:127–136, 1980.

[32] K. S. Leung and W. Lam. Fuzzy concepts in expert systems. *IEEE Computer*, pages 43–56, sept 1988.

[33] M. D. Levine, P. B. Noble, and Y. M. Youssef. Understanding blood cell motion. *Computer Vision, Graphics and Image Processing*, 21:58–84, 1983.

[34] M. D. Levine, Y. M. Youssef, P. B. Noble, and A. Boyarsky. The quantification of blood cell motion by a method of automatic digital picture processing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-2(5):444–450, September 1980.

[35] Martin D. Levine. *Vision in Man and Machine.* McGraw-Hill Book Company, 1985.

[36] Martin D. Levine, Peter B. Noble, and Youssry M. Youssef. A rule-based system for characterizing blood cell motion. In T. S. Huang, editor, *Image Sequence Processing and Dynamic Scene Analysis*, volume F2, pages 663–709. Springer-Verlag, 1983. NATO ASI Series.

[37] M. Levinstone, M. Eden, and E. Bell. Similarity of sister-cell trajectories in fibroblast clones. *J. Cell Science*, 59:105–119, 1983.

[38] F. Leymarie and M. D. Levine. Shape features using curvature morphology. In *Proceedings of the SPIE "Intelligent Robots and Computer Vision VIII: Algorithms and Techniques" Conference*, volume 1192, part 2, pages 536–547, Philadelphia, Penn., U.S.A., November 1989.

[39] F. Leymarie and M. D. Levine. Skeletons from snakes. In V. Cantoni et al., editors, *Progress in Image Analysis and Processing*, pages 186–193. World Scientific, Singapore, 1990. Proceedings of the 5^{th} International Conference on Image Analysis and Processing, held in Positano, Italy, 20–22 September, 1989.

[40] Frédéric Leymarie. Tracking and describing deformable objects using active contour models. Master's thesis, McGIll University, Montreal, Canada, February 1990.

[41] Frédéric Leymarie and Martin D. Levine. Simulation of the grassfire transform using an active contour model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-14:56–75, January 1992.

[42] M. Leyton. A process grammar for shape. *Artificial Intelligence Journal*, 34(2):213–247, March 1988.

[43] F. Meyer. Skeletons in digital spaces. In J. Serra, editor, *Image Analysis and Mathematical Morphology; Volume 2: Theoretical Advances*, pages 258–296 (Chapter 13). Academic Press, London, England, 1988.

[44] U. Montanari. A method for obtaining skeletons using a quasi-euclidean distance. *Journal of the Association for Computing Machinery*, 15(4):600-624, October 1968.

[45] P. B. Noble and M. D. Levine. *Computer-Assisted Analyses of Cell Locomotion and Chemotaxis.* CRC Press, Boca Raton, Florida, U.S.A., 1986.

[46] W. R. Oliver. Cell locomotion as shape change. *Blood Cells*, 15:334–342, 1989.

[47] N. Otsu. A threshold selection method from gray level histograms. *IEEE Trans. on Systems, Man and Cybernetics*, SMC-9:62–66, 1979.

[48] Jana Henriette Pavlovitch, Marthe Rizk-Rabin, Michel Gervaise, Philippe Metezeau, and Didier Grunwald. Cell subpopulations within proliferative and differentiating compartments of epidermis. *Am. J. Physiol.*, 256:C977–C986, 1989.

[49] Stephen M. Pizer, William R. Oliver, and Sandra H. Bloomberg. Hierarchical shape description via the multiresolution symmetric axis transform. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-9(4):505–511, July 1987.

[50] R. Ramakrishnan and C. J. M. Rao. The fuzzy weighted additive rule. *Fuzzy Sets and Systems*, 46:177–187, 1992.

[51] Robert J. Schalkoff. *Digital Image Processing and Computer Vision*. John Wiley and Sons, Inc., 1989.

[52] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, N.J., 1976.

[53] K. S. Shanmugan and Craig Paul. A fast edge thinning operator. *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-12(4):567–570, July/August 1982.

[54] B. Shapiro, J. Pisa, and J. Sklansky. Skeleton generation from x,y boundary sequences. *Computer Graphics and Image Processing*, 15:136–153, 1981.

[55] F. Y. Shih and C. C. Pu. Medial axis transformation with single-pixel and connectivity preservation using Euclidean distance computation. In *Proceedings of the* 10^{th} *International Conference on Pattern Recognition*, volume 1, pages 723–725, Atlantic City, NJ, U.S.A., June 1990. IEEE Computer Society Press.

[56] R. Stefanelli and A. Rosenfeld. Some parallel thinning algorithms for digital pictures. *Journal of the Association for Computing Machinery*, 18(2):255–264, April 1971.

[57] Masaaki Tatsuka, Shigeki Jinno, M. Koji Owada, and Takeo Kakunaga. A digital image processing system for quantitating dynamic morphology in cultured mammalian cells. *Experimental Cell Research*, 185:342–352, 1989.

[58] M. Thonnat and M. H. Gandelin. An expert system for the automatic classification and description of zooplanktons from monocular images. In *International Conference on Pattern Recognition*, volume 1, pages 114–118. IEEE, June 1988.

[59] Kim M. Wheeler. Expert system for determining cell states of mitosis. For Course 304-625B Artificial Intelligence, Dept. of Elec. Eng., McGill University, Montreal, Canada, April 1990.

[60] Kenong Wu, David Gauthier, and M. D. Levine. Cell segmentation revisited. Technical Report TR-CIM-91-4, McGill Research Centre for Intelligent Machines, McGill University, Montreal, Canada, October 1991.

[61] Yun Xia. Skeletonization via the realization of the fire front's propagation and extinction in digital binary shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, PAMI-11(10):1076–1086, October 1989.

[62] Ronald R. Yager. Fuzzy subsets of type 2 in decisions. *Journal of Cybernetics*, 10:137–159, 1980.

[63] Y. M. Youssef. *Quantification and Characterization of the Motion and Shape of a Moving Cell.* PhD thesis, McGill University, Electrical Engineering Department, Montréal, QC, Canada, May 1982.

[64] T. Y. Zhang and C. Y. Suen. A fast parallel algorithm for thinning digital pictures. *Communications of the ACM*, 27(3):236–239, March 1984.

[65] Y. Y. Zhang and P. S. P. Wang. A modified parallel thinning algorithm. In *International Conference on Pattern Recognition*, volume 2, pages 1023-1025. IEEE, 1988.