# The Development of a Vision-Based Flotation Froth Analysis System



Benedict Anson Wright

A dissertation submitted to the Faculty of Engineering and the Built Environment, University of Cape Town, in fulfilment of the requirements for the degree of Master of Science in Engineering.

Cape Town, September 1999

# Declaration

I declare that this dissertation is my own work. It is being submitted for the degree of Master of Science in Engineering at the University of Cape Town. It has not been submitted before for any degree or examination at this or any other university.



_____

Benedict Anson Wright

To my parents, with love and thanks for the last 24 years.

# Acknowledgements

Although many people have contributed to this project, I would particularly like to thank the following people and institutions:

- Professor Gerhard de Jager, for his supervision of the project and for his continued assistance.
- Dr. Dee Bradshaw, Department of Chemical Engineering, UCT, for her assistance and unfailing support.
- Dr. Jan Cilliers, Department of Chemical Engineering, UMIST, UK, for his valuable input.
- My "partners in crime"; Jerome Francis, Esther Ventura-Medina and Craig Sweet, for their assistance, expertise, encouragement and generous donations of liquid refreshments during the plant trials.
- Amplats Limited, in particular Mr. Sandy Lambert, for financial and material support.
- The FRD/NRF, for financial support.
- The metallurgical and engineering staff at Rustenburg Platinum Mines Amandelbult Section, for their assistance during the plant trials. Particular thanks must go to Sarel Pelser and Hubert Mostert for their enthusiasm and belief in this project.
- The other members of the DIP Group, for providing such an entertaining and interesting working environment. In particular, I am greatly indebted to Fred Nicolls for his expertise in matters LaTeXand UNIX, and for his excellent proofreading.

# Abstract

This dissertation describes the development of a machine vision system for the on-line analysis of flotation froth images. The size and shape of bubbles that constitute the flotation froth convey considerable information on the performance of the flotation process. A method whereby this size and shape information can be automatically extracted from froth images is highly desirable.

In this research, a system was developed which acquires froth image using a video camera, and then rapidly identifies the bubbles in the froth by segmenting the image using a morphological operation known as the Fast Watershed Transform. Bubble size and shape information is extracted from the segmented images and can be correlated with metallurgical and other data from concentrator plants in order to elucidate relationships between froth appearance and plant performance.

The machine vision system developed was tested on a platinum concentrator plant, and is able to identify and characterise variations in flotation froth appearance, which occur in response to changes in process inputs. The ability of the system to detect changes in bubble size distribution has been found to be particularly useful in detecting process input variations.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Over the last few years, the use of machine vision systems in industrial applications has increased dramatically. Computer-based image analysis and control systems are commonplace in processes as diverse as bottle top manufacturing, fruit sorting and textile production. Recently there has been much interest in the application of machine vision systems to the mining and mineral processing industries. The use of machine vision to examine fragmentation of rocks from crushing and milling plants is well known [16].

A topic that has been the subject of much discussion and research is the use of machine vision techniques for the inspection and characterisation of flotation froths in mineral processing plants. It is envisaged that the data obtained from such a system could be used as a means of controlling the flotation plant in a systematic manner. Of particular interest is the analysis, by means of image processing techniques, of the size and shape of the bubbles that make up the froths. By segmenting the froth image into individual bubbles, the size and shape information can easily be extracted from the images.

This dissertation describes the development of a machine vision system for the analysis, using bubble size and shape information, of flotation froths in a precious metals flotation plant. The research was done in tandem with work in the Departments of Chemical Engineering at the University of Cape Town and the University of Manchester Institute of Science and Technology. The goal of the project is the development of advanced automatic control strategies to optimise the performance of the flotation process. More consistent performance will generally mean less loss of valuable end product, and thus higher revenues.

The format of this dissertation is as follows:

- Chapter 2 introduces the concepts of froth flotation and the control of the flotation process.
- Chapter 3 presents a comprehensive review of the application of image processing techniques to different aspects of flotation modelling and control.
- In Chapter 4, flotation froth images and techniques for their segmentation are discussed.
- Chapter 5 introduces Mathematical Morphology for image processing, which include several very useful operators used for segmenting flotation froth images. These operators include the Watershed Transform, which is the cornerstone of the image segmentation in this research.
- Preprocessing techniques that were developed for use on froth images are extensively described in Chapter 6
- Chapter 7 presents efficient implementations of some of the key components of the froth image segmentation algorithm.
- The descriptive parameters that are extracted from segmented froth images, and used to characterise the froth, are defined in Chapter 8
- The application of the image processing algorithm to small-scale flotation work is described in Chapter 9. The on-line image analysis system that was developed and tested on a flotation plant, and a summary of the tests performed on the plant, is also presented.
- Chapter 10 gives the results of analyses of the image segmentation performance and of the plant testwork.
- Finally, Chapter 11 offers conclusions and suggestions for further work.

# Chapter 2

# Froth Flotation

Flotation is a mineral beneficiation process that was originally patented in 1906 [73] and has grown to become arguably the most important mineral processing technique in use today. Originally developed to treat sulphides of copper, lead and zinc, flotation now includes treatment of oxides, such as hematite (iron oxide) and cassiterite (tin oxide) and other oxidised minerals, as well as non-metallic ores, such as fluorite, phosphates and fine coal [73]. In South Africa, flotation is the basis for the extraction of the economically and strategically important platinum group metals (PGMs) [30]. It is a complex physico-chemical process that depends on many different factors that are sometimes difficult to isolate. Indeed, even after over 100 years of use it is still not well understood.



## 2.1 The Process of Flotation

The flotation process begins with the grinding of ore, with water and certain chemical reagents, down to a particular average particle size, which ensures liberation of the valuable mineral species from the host rock [34]. The resultant mineral-water slurry (or *pulp*) is usually passed through a separating device known as a hydrocyclone. The cyclone *overflow*, containing the fine particles, is passed to the flotation section, while the *underflow*, consisting of coarse particles, returns to the mills for re-grinding. Following the hydrocyclone stage, further reagents may be added to the pulp. A certain amount of *conditioning time* may be needed to achieve even distribution of reagents and to allow give-and-take competition between different mineral particles. Once

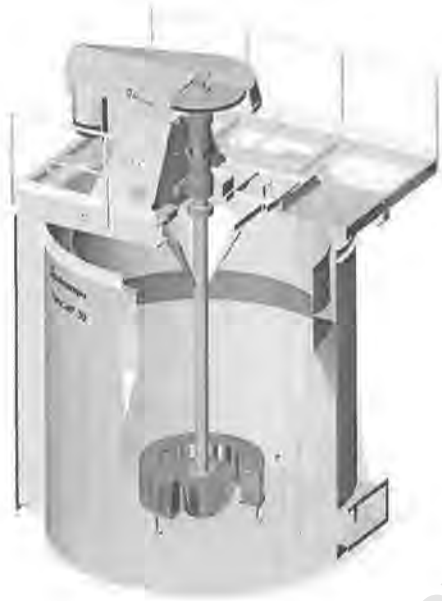


Figure 2.1: A modern TankCell flotation machine (from [45])



conditioning is complete, the pulp is then introduced to the flotation cells. These can crudely be described as large boxes or tanks containing stirrers (Figure 2.1). The agitation of the pulp and the addition of air to the cells causes the formation of bubbles, and the action of the reagents causes the desired minerals within the pulp (the *values*) to attach onto the bubbles and collect on the surface of the pulp as a mineral-rich froth, leaving the undesirable minerals (the *gangue*) in the pulp. The froth on the surface reports over the edge of the flotation cell and is collected in a launder as *concentrate*. Modern flotation plants typically employ several banks of flotation cells operating in parallel to increase plant throughput. Figure 2.2 shows a typical flotation plant. Flotation is generally a multi-stage process. The concentrate from the first (or *roughing*) stage is re-floated in a *cleaner* section (and in fact cleaner concentrate is often itself re-floated in a *recleaner* section) [73]. The non-floated pulp (or *tails*) from the rougher section is re-floated in a *scavenger* section, with the scavenger concentrate being passed to the cleaning section. The purpose of this multi-stage flotation is to improve the total separation of minerals (the *grade*) and the proportion of valuable minerals obtained (the *recovery*) [34]. Figure 2.3 shows a PGM plant flowsheet.

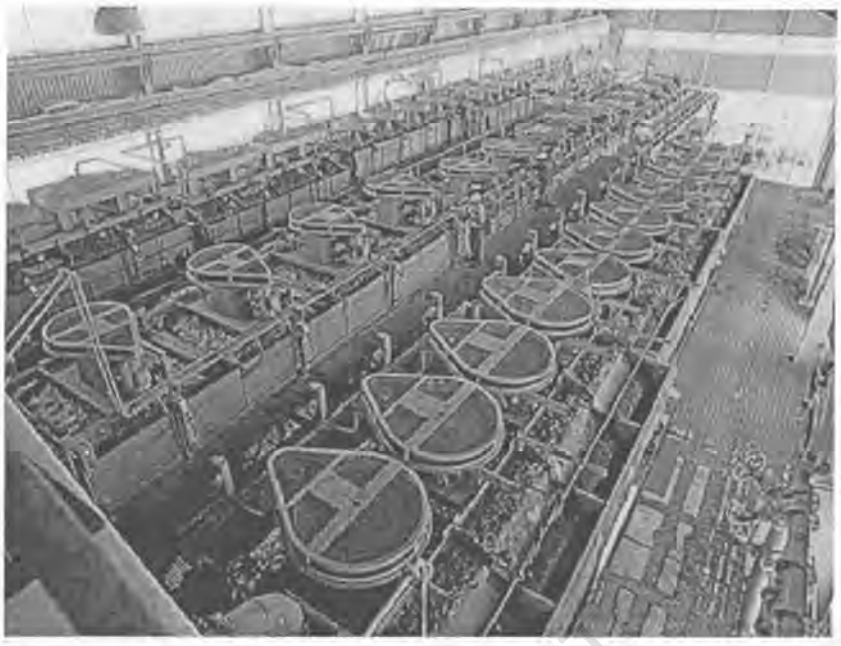


Figure 2.2: An industrial flotation plant

## 2.2 Flotation Reagents

The role of reagents in the flotation process is critical. The generated air bubbles will only attach to the minerals if they are able to displace water from the mineral surface. This will only happen if the mineral particles are to some extent water repellent (or *hydrophobic*). Once the air bubbles reach the surface of the pulp, they can only continue to support the attached minerals if they are sufficiently stable, otherwise they will burst and the mineral particles will drop back into the froth. To achieve these conditions, it is necessary to use various flotation reagents [73].

***Collectors*** are the most important reagents. They adsorb onto mineral surfaces and render them hydrophobic (or *aerophilic*, having an affinity for air). This facilitates attachment to the air bubbles.

***Frothers*** assist in maintaining a stable froth. The froth should be stable enough to allow gangue that has become entrained in the surface froth to fall back into the pulp, but not so stable as to prevent efficient transport of the froth into the launder [49].

***Regulators*** perform various functions such as activating or depressing mineral attachment to air bubbles, as well as controlling the pH of the system [73].

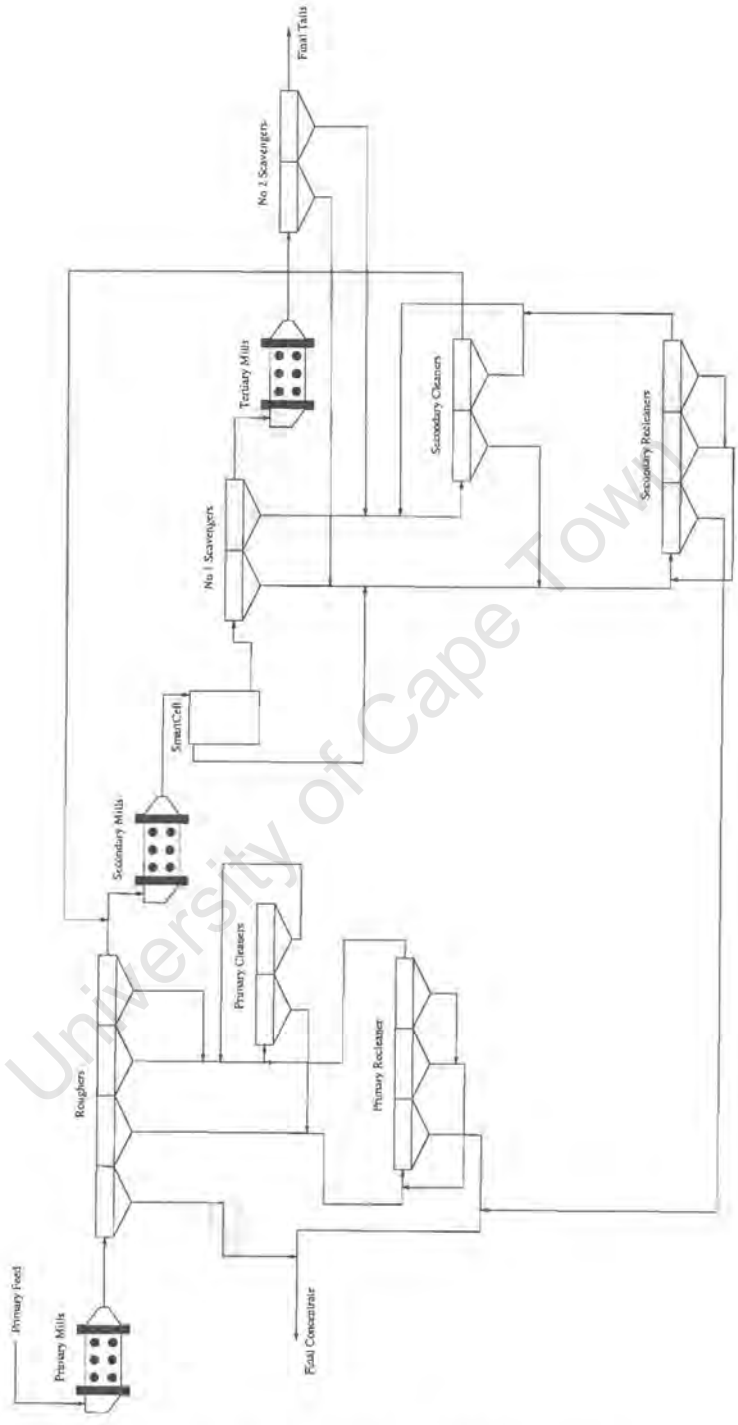



Figure 2.3: Flowsheet of a PGM flotation plant

## 2.3 Control of Froth Flotation

Despite the fact that it is so widely used, froth flotation is one of the least well understood operations within the mineral processing industry. In particular, it is believed that poor understanding of the froth phase contributes significantly to the incomplete understanding of the process as a whole [44]. In general, for effective control to be performed, a comprehensive process model must exist. The pulp phase of flotation can be adequately described by first-order kinetics, but no appropriate general model exists for the froth phase [44]. An in-depth discussion of flotation models is beyond the scope of this dissertation, however mention of froth-based kinetic models in the context of image processing is made in Chapter 3.

### 2.3.1 Difficulties with Flotation Control

The variables that affect the flotation process are numerous and varied [68]. They include:

- Feed characteristics (mass flowrate, mineral composition, liberation size, particle size distribution, specific gravity etc.)
- Physico-chemical factors (water quality, temperature, reagent types and concentrations, interactions between reagents and particles etc.)
- Hydrodynamics (flotation circuit design, cell type, aeration rate, spatial distribution of bubbles and particles etc.)

This plethora of variables makes control of the flotation process an extremely difficult task. As a result, control is usually effected manually by human operators, who base most of their decisions on the visual appearance of the froth (sometimes referred to as the "calibrated eyeball" or, in the context of this research, the "Mark 1 Image Processing Toolkit" method). This often leads to sub-optimal control, owing to factors such as operator inexperience or the inability to act promptly at the first sign of aberrant plant behaviour [48]. Tools such as XRF (X-ray fluorescence) and XRD (X-ray diffraction) exist for on-line measurement of mineral compositions, and can assist in plant control. However, monitoring every step of a large and complex flotation process with on-stream analysers would be prohibitively expensive [54] and is unsuitable for certain types of mineral flotation (such as PGM flotation), where head grades may be too low to be detected [36].

### 2.3.2 Vision-based Control of Flotation

The appearance of the froth on the surface of a flotation cell has great significance, in that it contains information which may describe the grade and recovery of valuable minerals in the concentrate [49]. This is hardly surprising, given that visual inspection of the froth is used as the basis for plant control. In the past, little attention has been given to the relationship between flotation performance and froth appearance, mainly because more value is attached to fundamental systematic investigations rather than observations that are difficult to define quantitatively. Indeed, different flotation plants can exhibit different visual characteristics, which may depend on factors such as the type of flotation cells used, ore mineralogy, reagents used, etc. [49]

In 1972 Glembotskii [22] proposed a description of froths and froth "quality" based on visual parameters:

- A good froth consists of small mineralised air bubbles that break down on the froth surface without forming larger tough-skinned bubbles. The froth should be mobile and not excessively viscous.
- Efficient flotation results in bubbles of between 1 and 3 centimetres in size appearing on the froth surface. These bubbles are partially covered with a mineral particle film, which leads to free areas remaining on the bubble tops.
- The froth should not be so stable as to not break down readily after reporting over the weir.
- An excessively "dry" froth consisting of small, mineral-laden bubbles becomes very stable and hinders further concentration.
- Froth characteristics are determined by the following physical characteristics: froth layer thickness; speed and method of froth removal from the flotation apparatus; amount of bubbles and mineral particles present and their dispersal in the froth; and intensity of pulp movement in the layer directly under the froth.
- The observation of spray above the froth surface indicates instability, possibly due to an excess of frother. Froths of this type carry little mineral and are symptomatic of incorrect reagent feed levels.

Glembotskii [22] also proposes a method of froth classification based on structure:

1. **Film-structural froths (most common)**
   - Air bubbles in the upper froth layers are larger than those in the lower layers.
   - The water layers separating the froth air bubbles decrease in thickness as the froth surface is approached.
   - The froth layer is thick (from 5 to 20 cm).
   - The larger bubbles are usually deformed.
   - The amount of water contained in this type is greater than other types. The mobility is high and the froth stability varies over a wide range.
2. **Aggregate froths**
   - These froths are composed of large particles that are attached to each other by many air bubbles.
   - The bubbles are smaller than those of film-structural froths, however the size distribution of bubbles at different froth levels is similar.
   - The froths contain relatively little water and are reasonably stable, but readily break down when falling into the launder.
3. **Film froths**
   - These froths are similar to, but thinner than, aggregate froths.
   - The mineral particles in the froth are very large, are strongly hydrophobic and have a low specific gravity.



Moolman et al. [50] argue that the above classification method is unsuitable for a machine vision froth inspection system, because the descriptors are not designed with vision-based classification in mind. It is postulated that classification methods should be based not only on simple observation of the froth, but also on a detailed analysis of the relationship between froth appearance and process performance. Based on this argument, three main froth classes are proposed, on the assumption that froth mobility, froth stability and average bubble size are the most important froth features to be identified [47]:

1. **Ideal froth:** The specific froth appearance of an "ideal" froth will vary from plant to plant, according to factors such as ore type, reagent types, cell geometry etc. This froth type will

represent a balance between too runny and too viscous a froth, and can in most cases be identified by a froth composed of closely packed bubbles, with clear areas on top of some of the bubbles. The features of this froth should serve as a reference point for other froths.

2. **Runny froth:** The froth loading (i.e. mineralisation) is too low, the froth is watery and excessively mobile. The average bubble size is too low and the froth is unstable.

3. **Sticky or viscous froth:** The degree of froth mineralisation is too high, the froth is overly viscous and the mobility is lower than that of an ideal froth. The bubbles are large and elliptical and the froth is too stable.

Moolman et al. [50] compiled a comprehensive review of literature concerning the relationship between flotation froth appearance and fundamental flotation principles. The paper serves mainly as a motivation for machine-based inspection of flotation froths. Of particular interest is a study relating flotation process deviations and the effect these deviations have on the surface froth structure. As can be seen in table 2.1, different process deviations can cause similar changes in

Table 2.1: Table of process deviations and resultant change in froth appearance for a platinum flotation operation (from [50])

| *Process deviation* | *Froth appearance* |
|---|---|
| Frother addition rate too high | froth too stable, **bubbles too small** |
| Frother addition rate too low | froth less stable, **large bubbles** formed by coalescence |
| Depressant flow rate too high | froth watery & runny, low mineral loading, **small bubbles** |
| Depressant flow rate too low | froth too viscous, low mobility, **large bubbles** |
| Activator flow rate too high | brittle, mobile froth |
| Pulp level too high | fast, watery froth |
| Pulp level too low | sticky, viscous froth with low mobility |
| Pulp density too high | viscous, low mobility froth |
| Pulp density too low | watery, runny, unstable froth |
| Particle feed too coarse | brittle froth, **small bubbles** |
| Particle feed too fine | stable, sticky froth, **large bubbles** |
| Aeration rate too high | low grade, fast froth, **large bubbles** |
| Aeration rate too low | low recovery, slow froth, **small bubbles** |

surface froth appearance. This serves to highlight the complex nature of the flotation process, as well as the difficulties associated with effective control of the process.

In 1989 Wooodburn et al. [75] described work relating to the flotation of low-rank coals. It is reported that the froth water content is related to the performance of the flotation process, with high frother doses producing high recoveries and lower frother doses producing high grades. The water content also correlated with froth stability, in that high water content was characterised by small bubbles with low rates of coalescence, and low water content producing the opposite effect. An optimal froth structure (corresponding to maximum beneficiation) was described as being the point at which the froth changed from close-packed spherical bubbles to a cellular form of polyhedral bubbles. It is asserted that the optimal structure can be identified visually and characterised by image analysis techniques and used to form the basis of a coal flotation control mechanism.

Woodburn et al. [75] also note the importance of bubble size and shape distributions in giving a sensitive measure of the appearance of an overflowing froth. This point is taken up by many of the authors whose work is described in the literature review. Using image processing techniques to determine these distributions thus seems highly attractive.

It is clear from the above discussion that there is a sound basis for the development of machine vision systems for the control and modelling of flotation plants using computer-based inspection and analysis of images from surface froths. Chapter 3 contains a review of various studies undertaken in this regard, and will also introduce the work undertaken by the author and presented in this dissertation.

# Chapter 3

# Image Processing as applied to Flotation — A Review



The interest in machine-based inspection of flotation froths is evidenced by the number of groups around the world that have been or are engaged in the study of the subject. This chapter presents a review of the work of the various groups, and attempts to define the current "state of the art" in machine-based inspection of flotation froths. The review has been compiled by examining the work of each group in turn, as it is felt this is a more logical way of presenting the information.

## 3.1 University of Mining and Metallurgy in Cracow, Poland

As early as 1988, Kordek and Lenczowski presented work on analysis of froth images in an attempt to correlate the appearance of the froth with the metal content [38]. The authors obtain Optical Fourier Transforms (using a diffractometer) from images of both laboratory and plant froths. The authors state that knowledge of the focal length of the transforming lens and the geometric dimensions of the detector allows the size of structures (in this case bubbles) in the analysed images to be related to light intensity distribution in the diffraction pattern. Results obtained concluded that the froth optical density decreases (i.e. the froth becomes darker) with increased metal content, and the bubble size decreases with increasing metal content.

In a later paper [37], Kordek and Kulig report on further optical diffractogram analysis of

froth images. Froth images were also subjected to digital image analysis by means of a digital computer-based Fourier Transforms. Once in the Fourier domain, filters were applied to the image in order to determine the total area occupied by the froth and the average area of a single bubble. Results obtained seem to correlate well with the useful content of metal in the froths.

A further paper [39] again presents research using optical and digital Fourier analysis of still froth images. Once again, useful metal content was the main parameter extracted by the image analysis. The development of a method based on discriminant analysis to classify froth images according to inferred metal content is described. In both laboratory and plant work, correct classifications of the order of 68% were obtained. The plant work differed from the laboratory scale work in that instead of froths being photographed for later off-line analysis, images were grabbed in real-time from the plant and subjected to various 1-dimensional Fourier Transforms, the results of which were equivalent to the useful sections of a 2-dimensional Fourier Transform. On-line froth classification was performed, but no plant control based on the classification results was performed.



## 3.2 University of Nottingham

Work done at the University of Nottingham has predominantly been concerned with the use of image processing techniques for flotation modelling and performance prediction. Early work in correlating the ash content of coal froths with average gray level measurements of froth images produced encouraging results [29]. Simple linear relationships for predicting mass flowrate and ash content using the average gray level measurements were tested. The mass flowrate predictor gave satisfactory results, while the ash content predictor was less accurate (predictions were correct to within 10-15%).

Research into predicting tin concentrate grade and mass flowrate was also performed at Nottingham [27]. Tin flotation images were analysed and a "relative redness" (the difference between the mean gray and red levels of the image, normalised with respect to the gray level) measure was calculated. It is claimed that this method eliminates the effect of lighting variations. A linear relationship between the relative redness and tin concentrate grade was noted for froths from the cleaner and rougher section of a tin concentrator plant. In addition to the colour information, bubble size distributions and bubble distortion profiles (a measure of the predominant direction of bubble alignment) were extracted from the froth images. These parameters, together with the

chromatic measurements, were used to model tin concentrate grade and mass flowrate.

Modelling the tin grade using a neural network with the colour information and a 7-class bubble size distribution produced an $R^2$ of 0.99 for the model fit[1] and an $R^2$ of 0.98 for the test data fit. The mass flowrate neural network predictor using colour, mean bubble size, standard deviation of bubble sizes and bubble distortion profile as inputs produced $R^2$ values of 0.95 and 0.92 respectively.

An interesting technique introduced by this research is the use of fractal dimensions to describe froth structures. The fractal descriptor consists of a break point and the gradients of a straight line either side of the break point. It was postulated the gradients of the two straight lines represented small bubbles within the froth and the larger bubbles formed by coalescence of smaller bubbles and hence represents unstable elements in the froth. The break point represents the proportion of unstable bubbles in the froth.

Later work [28] made extensive use of the fractal dimensions described above as a tool for describing froth structure. Neural network models linking concentrate grade, mass flow rate and water content with froth fractal dimensions were developed for cassiterite flotation. The result of this modelling was the ability to predict the required froth bubble size distribution for specified grade, flow rate and water content values. The $R^2$ of the linear regression between the predicted and measured bubble size distributions was 0.995. The neural net was also used to generate an "ideal" bubble size distribution when presented the highest experimentally measured metallurgical values as input. No on-line modelling, classification or control in a flotation plant was attempted.



## 3.3 University of Stellenbosch / Hatch Africa / Crusader Systems

A plethora of publications exist on work performed by this group of researchers, who began studying the application of machine vision to froth flotation in 1993.

An early paper [48] presents results of analysing froth images from a copper flotation plant. A

[1] $R^2$ measures indicate the relationship or correlation between variables. An $R^2$ of 1 indicates perfect correlation, whilst an $R^2$ of 0 shows that the variables are not related to each other in any way

relationship between the copper content of a froth and the ratio of minor to major modal frequencies of the froth image grayscale histogram (the "copper peak") was noted. Bubble shape and size information was extracted using a Fast Fourier Transform and used for froth classification.

Later work [49] focused on using spatial graylevel dependence matrix (SGLDM) and neighbouring graylevel dependence matrix (NGLDM) [61] methods for the analysis of froth structures from a textural point of view. These matrix analysis methods generate abstract image descriptors that describe the textures present within an image. Initially, the froth images were grouped into five classes. The graylevel dependence matrix methods were then used to extract features from copper froth images, and these features were used as inputs to the LVQ[2] neural networks used to classify the froths. Overall weighted correct classifications ranged from 78% to 89% using SGLDM, NGLDM and a combination of both methods. Image analysis of froths generated from batch tests using a pyritic ore [47] generated relationships between several surface froth features (bubble size, froth stability and froth mobility) and features extracted from the froth images by the NGLDM method. Backpropagation neural networks were used to model the froth grade and recovery using the extracted froth features.

In industrial tests on South African precious metals plants [46], self-organising maps[3] were used to relate changes in froth appearance to reagent level changes and plant performance. Tests of this system on an Australian base metals plant [7, 66] yielded poor correlations between image features and zinc grade over long periods. Short-term perturbations in the zinc grade and reagent addition rates correlated well with changes in the image features. It was not possible to correlate the absolute value of any image parameter with zinc grade, however changes in the froth appearance were well quantified by changes in the froth image features. A relationship was established between perturbations in the plant grinding circuit and changes in the image features.

Images acquired from batch tests conducted on Merensky PGM ore [1] were subject to the NGLDM analysis method. The abstract features extracted from the froth images were found to give a rough indication of average bubble size, as well as froth instability. The average gray level of each froth image was also computed. These parameters were used to quantify the effect of high intensity conditioning on the sulphide ore batch flotation.

One of the problems with using texture-based methods for extracting features from froth images, is that it may be difficult to relate the abstract descriptors generated by the analysis to the physical

[2] Learning Vector Quantisation, a particular form of neural network classifier

[3] A neural network-based method of visualising multi-dimensional feature spaces

appearance and properties of the froth. For example, two of the measurements produced by the NGLDM method are *small number emphasis (SNE)* and *number nonuniformity (NNU)*, which represent the fineness and coarseness respectively of an image. Both of these measurements have been used in an attempt to estimate bubble size:

> *Although SNE has been found to be inversely related to bubble size, it is not necessarily the case. NNU could be directly related to bubble size, although it has been observed at some plants that SNE and NNU can actually be related positively. This means that neither SNE nor NNU can be related unconditionally to bubble size [7].*

Recently, a "new" product from Crusader Systems, Float-MACS, has been presented. Float-MACS is a visual froth flotation imaging and characterisation system [17], which differs radically from previous work by this group in that it makes use of segmentation to analyse the froth images. This paradigm shift by a group that has up to now based much of its research on texture-based analysis serves, in the author's opinion, to indicate the superiority of segmentation-based methods over abstract textural analysis.

## 3.4 University of Manchester Institute of Science and Technology

Extensive work has been undertaken at UMIST into quantifying flotation performance and developing froth-based kinetic models using image processing. A classic paper by Woodburn et al. [75] reports on characterising coal froths using off-line image analysis techniques. Bubble size and ellipticity information was obtained by identifying each bubble at the point of discharge of a batch flotation cell using a threshold-based segmentation method. This allowed different froth structures to be identified and characterised. In addition, the specific surface of bubbles in overflowing froths was estimated and used to predict flotation rates.

The development of a froth-based kinetic model by Woodburn et al. [74] highlighted the usefulness of image processing as a tool for estimating flotation performance. The model estimated the flux of bubble surface overflowing the flotation cell from cell aeration rate and bubble specific surfaces. This allowed the flux of valuable and waste solids to be determined. The specific surfaces were determined off-line by segmentation of froth images into individual bubbles.

Further work [3, 4] made use of the techniques described above to relate reagent addition and cell performance to froth appearance and bubble flux. In the earlier stages of the research, the image processing work was done off-line using a MicroVAX computer performing edge detection using intensity minima location. Recently, the development of faster PC-based routines for segmenting the froth images [58] opened the door to on-line analysis of images using readily available hardware and software. The segmentation methodology used was based on the watershed transform and provided a strong basis for initial work that lead to the research reported on in this dissertation. The algorithm was implemented using an off-the-shelf image processing package and was used for off-line analysis of batch-scale flotation froths.

## 3.5 Julius Kruttschnitt Mineral Research Centre

Research performed at the JKMRC in Australia has produced a machine vision system for analysis of coal flotation froths. In this system, froth images are analysed using a statistical image analysis method known as the Texture Spectrum [53]. The Texture Spectrum is a measure of how much the pixels of a 3x3 neighbourhood within an image vary with respect to the central pixel of the neighbourhood. The froth structure is determined by subtracting the Texture Spectrum of a froth image from the Texture Spectra of seven pre-defined froth types, with the smallest difference indicating the froth's class. Correct classifications of 93% were achieved using this method. The average bubble size of the froth was related to the middle peak of the froth image Texture Spectrum.

Froth velocity was determined by a simple block-matching algorithm known as pixel tracing [52]. A more detailed description of this algorithm can be found in Chapter 8. The development of a "soft-sensor" [51] gave the system the ability to predict on-line the percentage ash and percentage solids being recovered by the coal flotation process.

A recent application of the JKFrothCam (as the commercial system is now known) is at Minera Escondida in Chile, where it has been used to measure froth velocity [31]. It is worth noting that no use has been made of the texture spectrum froth characterisation part of the system in this application — the system is simply used to control the speed of the froth in the flotation cells.

It is considered that a weakness of the approach used by the JKMRC is the fact that a priori knowledge and identification of all possible froth types is required in order for the system to

perform correctly. This may be relatively easy to achieve for a simple flotation process such as coal, but on complicated multi-stage flotation circuits with many process inputs (such as PGM concentrators), identification of all possible process conditions and resultant froth types may be impossible.

## 3.6 Catholic University of Chile (PUC)

The ACEFLOT system developed at PUC appears to be a sophisticated product at an advanced stage of research. The system is a fully developed froth image analysis system, which has received extensive testing in an industrial copper flotation plant.

Froth colour is analysed by computing the average Red, Green and Blue values for a colour froth image. Bubble size and shape information is extracted by identifying the bright tops of bubbles and segmenting the image (using radial intensity minima detection) into the individual bubbles that compose the froth [24]. Bubble mean diameter and ellipticity information is calculated once each bubble has been identified.

Froth velocity and stability are measured by analysing consecutive frames and detecting the movement of bubble centres and the rate of change in the appearance of the images respectively [14]. The results of the analysis are used as inputs to an expert system which relates froth appearance to process deviations, determines the reason for the deviation, and suggests a method of correcting the deviation [15]. It is not know whether this system has been tested on or applied to different types of flotation.

## 3.7 University of Cape Town

Previous work undertaken at UCT has concentrated on the development of methodologies for the segmentation of flotation froth images into their constituent bubbles. Symonds [65] used a morphological "rolling ball" method to segment froth image, while Liu [41] investigated using a hierarchical watershed segmentation algorithm. An undergraduate project by the author [76] implemented the algorithms used at UMIST [58] to perform a real-time (i.e. continuous) segmentation of videotaped froths.

## 3.8 United States Bureau of Mines

Oestreich et al. [54] developed a video-based sensor for measuring mineral concentrations in flotation froths and slurries. The sensor made use of a "colour vector" for estimating the mineral composition in dry mixtures, slurries and flotation froths. Good correlations were observed between the colour vector values and percentage composition of chalcopyrite and molybdenite.

## 3.9 Baker Process

Work was done by Hales et al. [25] to monitor copper flotation froth colour, bubble size and the "copper peak" histogram peak as defined by workers at the University of Stellenbosch. Factorial tests were performed where the change in froth appearance as a function of varying the collector dosage rate, the frother dosage rate and the froth depth was measured. Genetic algorithms and neural networks were used to determine if the changes in appearance correlate with the process input changes and metallurgical grades. Good correlations (between 85 and 95%) were obtained between changes in reagent levels and image parameters (as inputs to the neural network) and predictions of concentrate and tails grades (output from the neural network).

## 3.10 Virginia Polytechnic

A video-based coal slurry ash analyser was developed to predict the ash content of tailings slurry from a coal flotation plant [18]. It was found that the ash content of the slurry was correlated to the mean gray level of an image of the slurry. Accuracy of 73% $\pm$ 4% ash prediction was obtained from the sensor.

## 3.11 European Union (ESPRIT 4) ChaCo Project

The Chaco (Characterisation of Flotation Froth Structure and Colour by Machine Vision) project is a collaborative venture between four European academic institutions (Helsinki University of Technology and the University of Oulo in Finland, the University of Rome "La Sapienza" in Italy

and the Royal Institute of Technology in Sweden) and two mining companies (Boliden Mineral AB of Sweden and Outokumpu Mining Oy, Pyhäsalmi Mine of Finland). This large project (with a value of around US$1.4 million) has very specific objectives, namely [11]:

1. To analyse mineral concentrations of flotation froths from the colour of the froth.
2. To analyse flotation froth structure (speed of the froth, the size and form of the bubbles, their distribution etc.).
3. To classify the flotation froths (e.g. with statistical methods and neural networks).
4. To develop an on-line froth analyser (bubble size, speed etc.).
5. To develop process models for different type of flotation conditions.
6. To simulate the flotation process and develop control strategies for automatic control of flotation plants.
7. To install and test the resulting products at industrial flotation plants.

The project's 12-month progress report [12] contains a fair amount of information (but no specifics) on the methodologies used for image analysis. Some key results are reproduced here.

Parameters extracted from the froth images include:

- Morphological parameters (size and shape characteristics of bubbles).
- Colour parameters (derived from whole images and from each segmented bubble).
- Textural parameters (evaluating characteristics of neighbouring pixels).
- Fractal analysis.

Morphological parameters are extracted from the froth images by segmenting them using the watershed transform. Other approaches to the segmentation have been studied, including "morphology" and edge-based methods.

The colour analysis of froths is performed by evaluating the average colour of froth images in various colour spaces (RGB, HSV and ISV)[4]. A spectroradiometer has also been used to provide

[4]Red-Green-Blue, Hue-Saturation-Value and Intensity-Saturation-Hue.

data for wavelengths of light that are outside the range of video cameras. It has been shown that zinc concentrate grades can be inferred from froth colour alone, while prediction of magnesium grades requires both colour and structural information.

Other work has involved using the morphological and colour parameters generated by the image analysis to develop methods of classifying froths. It is not clear from the report what exactly the criteria for classification are. Some success has been noted in modelling talc and galena grades using the image data.

A prototype on-line image analyser has been installed on a concentrator at Pyhäsalmi mine, and is being used to collect data for use in the development of models of the flotation process.

## 3.12 Summary Table

The following table summarises the work from the various centres as described in this chapter, in an attempt to produce a useful "quick-reference" tool.

Table 3.1: Table summarising work done in vision-based analysis of flotation froths

| *Group* | *Methodology used* | *Features measured* | *Application* |
|---|---|---|---|
| UMM Cracow, Poland | Optical & digital Fourier Transforms | Optical density, inferred bubble size | Estimation of useful metal content (on & offline) |
| Nottingham | Colour, watershed segmentation | Average gray level | Mass flow prediction (offline) |
| | | Relative redness, bubble size distribution, bubble distortion | Concentrate grade & mass flow prediction (offline) |
| | Fractal dimension | Froth structure | Concentrate grade, mass flow & water content prediction (offline) |
| Stellenbosch / Hatch / Crusader | Graylevel histogram, Fast Fourier Transform | "Copper peak" histogram ratio, inferred bubble size and shape | Copper content estimation, froth classification (offline) |

*continued on next page*

*continued from previous page*

| | | | |
|---|---|---|---|
| | SGLDM and NGLDM | Inferred bubble size, froth stability and froth mobility, froth type | Froth grade modelling (offline) |
| | | | Relation of froth appearance to reagent level changes & plant performance (online) |
| | | | Estimation of zinc grade, grinding perturbation detection (online) |
| | | | Quantifying effects of HIC in batch flotation (offline) |
| UMIST | Segmentation by thresholding | Bubble size, specific surface & ellipticity | Froth characterisation, flotation rate prediction (offline) |
| | Segmentation by edge detection using intensity minima | Bubble size & shape distributions, specific surface | Solids flux estimation, relation of reagent addition rates & cell performance to froth appearance (offline) |
| | Segmentation by watersheds | As above | As above |
| JKMRC | Texture Spectrum, Pixel Tracing | Froth type classification, froth speed | Coal flotation control, % ash & solids prediction (online) |
| | | | Copper froth speed control (online) |
| AceFlot | Colour bands, segmentation by minima detection | Average colour information, Average bubble diameter & ellipticity, froth density, froth velocity & instability | Copper flotation control (online) |

*continued on next page*

| *continued from previous page* | | | |
|---|---|---|---|
| UCT | Rolling Ball segmentation | Bubble size & shape | None |
| | Hierarchical watershed segmentation | As above | None |
| | Fast watershed segmentation | Bubble size | None |
| USBM | Colour bands | Colour vector | Mineral composition (online) |
| Baker | Colour bands, proprietary segmentation, Graylevel histogram | Froth colour, bubble size, "colour peak" | Relation of changes in process inputs & image parameters to concentrate & tails grades (offline) |
| Virginia Polytechnic | Graylevels | Mean graylevel | Ash content prediction (online) |
| ChaCo | Colour information, watershed segmentation, textural information | Froth colour, bubble size, shape and colour | Grade estimation, froth type classification, flotation modelling (online) |

# Chapter 4

# The Segmentation of Froth Images



As discussed in Chapter 2, there appears to be a sound basis for the analysis of flotation froth images using a method whereby the image is segmented by identifying the individual bubbles that constitute the froth. This chapter will present a brief discussion of froth images in the context of segmentation, as well as a motivation for the approach to segmentation method used in this research, namely the Watershed Transform.

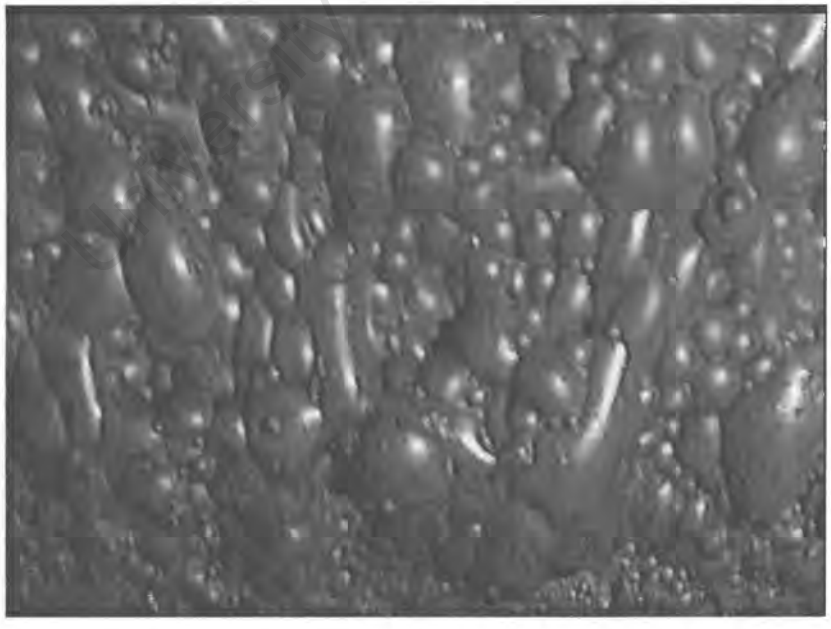


Figure 4.1: Froth image I1

## 4.1 Properties of Flotation Froth Images

Figure 4.1 show an image acquired from a video camera on a rougher flotation bank at Amplats' RPM Amandelbult Section Merensky concentrator plant.

Examination of the image reveals that the bubbles making up the froth are of various sizes and shapes, varying from small circular to large elliptical bubbles. Each bubble has one (or sometimes more than one) bright spot on the top surface due to the illumination of the froth. The boundaries of the bubbles are demarcated by areas that are darker than the surfaces of the bubbles. As observed by Liu [41], the bright spots (or *highlights*) and the darker bubble boundaries are two important visual features that enable the human observer to quickly identify the individual bubbles (i.e. to perform a rapid "segmentation" of the image). It can be argued that a segmentation algorithm that can identify and use the highlights and bubble boundaries may stand a reasonable chance of success.

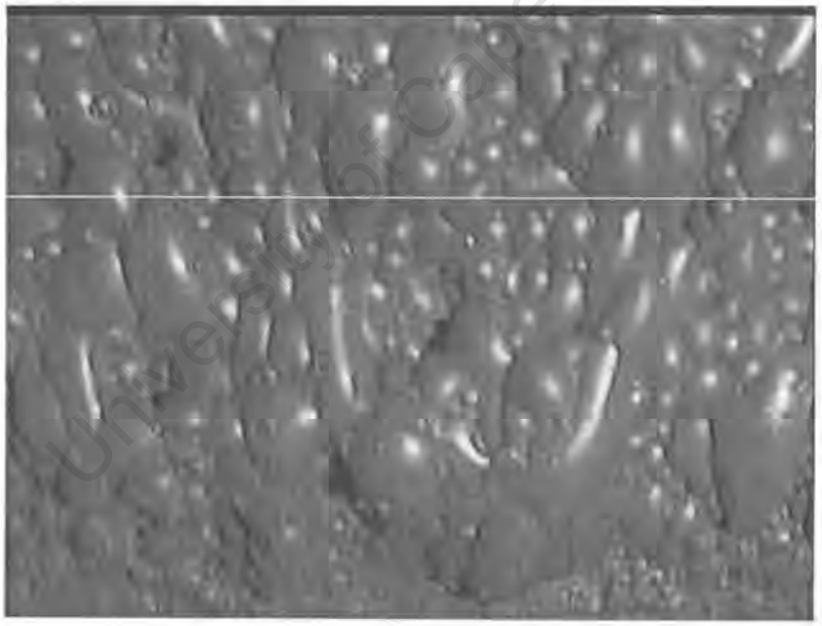


Figure 4.2: Scanline 150 from froth image I1

A better insight into the structure of a froth image may be obtained by plotting the pixel intensity values of a single scanline (i.e. row, shown in Figure 4.2) from a froth image, as is shown in Figure 4.3. The highlights on the top surfaces of the bubbles correspond to the intensity maxima, or peaks, on the graph, whilst the bubble boundaries correspond to the intensity minima, or val-

leys. A segmentation algorithm that can (correctly) detect and label these intensity minima will thus have detected the bubble boundaries, since they are simply the loci of the intensity minima that surround an intensity maximum. The major advantage of boundary-detecting segmentation is that the boundary of a bubble completely specifies the surface area occupied by the region of the bubble on the froth surface and thus provides a direct measure of the size and shape of the surface bubbles [65].

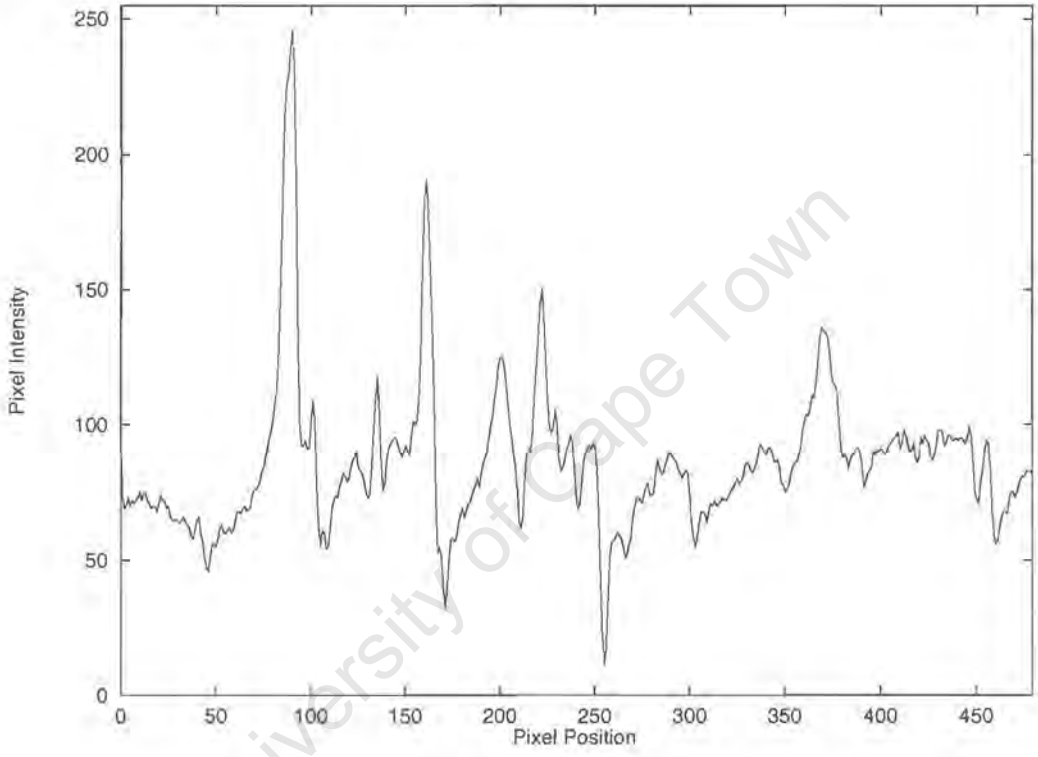



Figure 4.3: Pixel Intensity plot along scanline 150 for image I1



Couched in such simple terms, the problem may seem fairly trivial. This is, however, far from the truth, because the images obtained can never be "perfect" from an intensity profile point of view. As is clearly shown in Figure 4.3, the plot does not vary smoothly, but also contains many small lower intensity peaks. These peaks may correspond to the highlights of small bubbles, but are equally likely to be caused by unwanted noise in the image, and will almost certainly cause problems during image segmentation.

## 4.2 Difficulties with Automatic Image Segmentation

In general, image segmentation is a not an easy task. Often, reliable segmentation is only possible after significant information has been extracted about the objects portrayed [32]. The segmentation of flotation froth images is extremely challenging for many reasons, including the following:

- Due to the dynamic and continuous nature of the process, no two froth images will ever be identical. Thus, whilst the overall appearance of several froth images may be similar, there will always be many subtle (and often not-so-subtle) differences between them.
- Variations in illumination (which are almost certain to occur in flotation plants) may affect the accuracy of segmentation algorithms.
- The flotation of different ore types can yield vastly different types of froth. A classic example is the difference between Merensky (Figure 4.1) and UG2[1] (Figure 4.4) flotation froth images. UG2 froths are characterised by extremely fine bubbles which are difficult to identify with the naked eye.

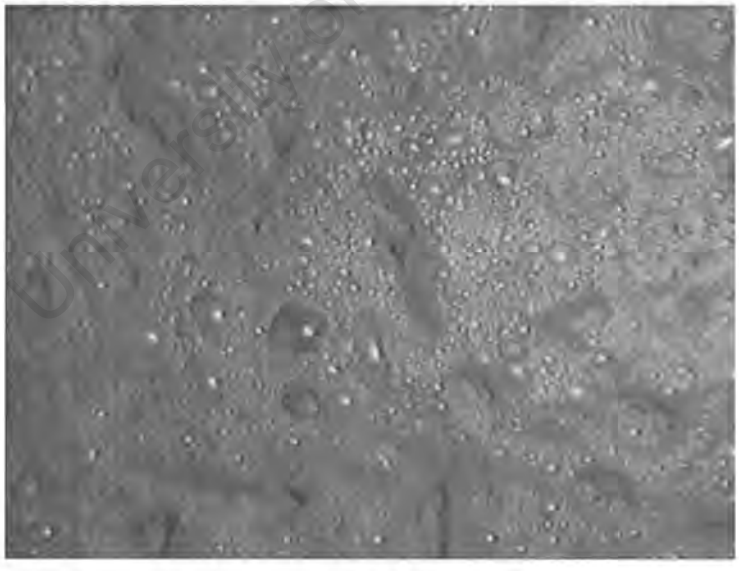


Figure 4.4: Froth from UG2 ore flotation

[1]Merensky and UG2 are the two principal PGM ore types. They have vastly differing metallurgies, and hence different flotation characteristics and appearance.

Mention was made in Section 4.1 of the use of bubble highlights and/or bubble boundaries in the segmentation of froth images. Symonds [65] notes that both these features are dependent on lighting conditions, and that improper lighting can lead to several problems, namely:

- Bubble highlight saturation.
- Multiple bubble highlights.
- Masking of bubble boundaries by shadows.

Depending on the segmentation methodology used, these issues may or may not pose problems. Chapter 6 examines some of these issues in greater depth.

## 4.3 Froth Image Segmentation Techniques

The complexity involved in the segmentation of flotation froth images has been illustrated by the work of Symonds [65] and Liu [41]. Symonds concludes that classical segmentation techniques such as edge detection and thresholding produce unsatisfactory results. Both Symonds and Liu turned to segmentation methods based on mathematical morphology (see Chapter 5). It should be noted that both workers concentrated on basic segmentation algorithm design in their research. For this work, the segmentation algorithm is simply a part (albeit the most important part) of a complete froth image analysis system.

In Section 4.1 reference was made to "peaks" and "valleys" with regard to the intensity profile of a froth image. Figure 4.5 shows a section of Figure 4.1 plotted as a surface. One can clearly see the peaks that represent the bubbles and, to a lesser extent, the boundaries that separate the bubbles. Also clearly visible is the low intensity noise as mentioned previously.

The emphasis placed on the peak and valley analogy for froth images is because there exists a method of image segmentation that is itself based on a topographical analogy. The Watershed Transform, which is introduced in a more rigorous fashion in Chapter 5, requires an image to be interpreted as a series of hills and valleys in order for the segmentation to function correctly. The fact that a froth image can be **directly** considered (and indeed visualised, as in Figure 4.5) as being a landscape makes it an ideal candidate for segmentation using the Watershed Transform.

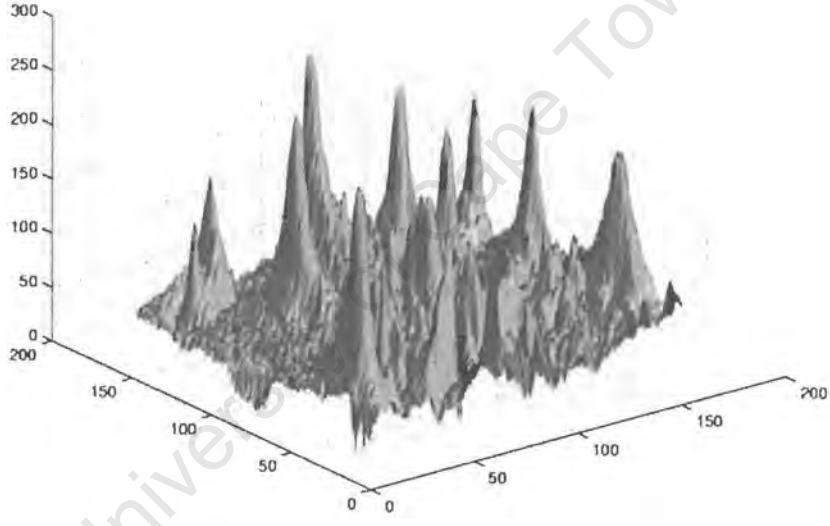





Figure 4.5: Image I1 plotted as a surface

# Chapter 5

# Mathematical Morphology

Mathematical morphology is a general method for image processing based on set theory, where images are considered as sets of points on which operations such as union and intersection are performed [8]. The Greek *morphos*, meaning shape, is the root of the word morphology. Mathematical morphology is thus a method of developing a quantitative description of the geometrical structure of a signal (or an image in this case) [42].

First performed on binary images (i.e. two-dimensional sets), mathematical morphology has been extended to gray-scale images. These can be considered as three-dimensional sets on which morphological operators can be applied.



## 5.1 Basic Operators

The basic transforms of mathematical morphology are *erosion* and *dilation*. Haralick et al. [26] explain how these transformations can be defined for sets in general space and how the definitions can be extended to continuous graytone functions and discrete binary and grayscale images. Here erosion and dilation are directly presented as neighbourhood operations on discrete grayscale images. Binary images can be considered to be a subset of grayscale images, having only two gray levels.

The erosion (resp. dilation) of an image $A$ is the operation of assigning to each pixel $(x, y)$ of the transformed image the minimum (resp. maximum) value found over a possibly modified neigh-

bourhood of the corresponding pixel in the input image. The neighbourhood used is specified by a "structuring element" (s.e.) $B$, which is a function $B_\Phi(i, j)$ defined over a domain $B_\Delta$ around an origin $B_\Theta(i = j = 0)$. The image eroded (dilated) by s.e. $B$, denoted $A \ominus B$ (resp. $A \oplus B$) is then[1]

$$(A \ominus B)(x, y) = \min\{A(x + i, y + j) - B_\Phi(i, j) | (i, j) \in B_\Delta\} \tag{5.1}$$

$$(A \oplus B)(x, y) = \max\{A(x + i, y + j) + B_\Phi(i, j) | (i, j) \in B_\Delta\} \tag{5.2}$$

Structuring elements may be thought of as images with arbitrary domain and origin, although in practice they are much smaller than the image they are applied to. When the s.e. is "flat" (i.e. when $B_\Phi \equiv 0$, Equations 5.1 and 5.2 reduce to

$$(A \ominus B)(x, y) = \min\{A(x + i, y + j) | (i, j) \in B_\Delta\} \tag{5.3}$$

$$(A \oplus B)(x, y) = \max\{A(x + i, y + j) | (i, j) \in B_\Delta\} \tag{5.4}$$

Flat structuring elements are used for binary image processing, because erosion and dilation of binary images by flat structuring elements produce binary images.

The results of erosion and dilation largely depend on the choice of the structuring element. The most commonly used structuring elements are digital equivalents of a disk, such as hexagons on hexagonal lattices and rhombuses in 4-connectivity contexts and squares in 8-connectivity contexts (both on square lattices).

[1] Strictly speaking, the definitions are $A \ominus \check{B}$ and $A \oplus \check{B}$, where $\check{B}$ is the transpose of $B$ about its origin $B_\Theta$. If $B$ is symmetric with respect to the origin (as is often the case), then $B$ and $\check{B}$ are equivalent. For simplicity $B$ will be used here.

Figure 5.1: The elementary disk structuring element in 4, 6 and 8-connectivity

Erosion and dilation are *increasing* transforms, that is they preserve inclusion relations. If $I$ and $J$ are two images such that $I(x, y) \leq J(x, y)$ for every $x$ and $y$, this relation is preserved by erosion and dilation.

If the domain of the structuring element includes the origin, and the value of $B_\Phi(0, 0)$ is greater than zero, then dilation is an *extensive* transform, while erosion is *antiextensive*. This means that the result of dilation is greater than the original image and the result of erosion is smaller than the original image.

Erosion and dilation can be viewed as a kind of special non-linear filtering using a kernel specified by the structuring element; multiplications are replaced by additions or subtractions and summation by the minimum and maximum operator. As with time-domain filtering, the larger the structuring element, the longer the processing time.



## 5.2 More Advanced Operators

Erosions and dilations can easily be cascaded to realize complex transforms. Two simple and well-known combinations of these operations are the "opening" (denoted $A \circ B$) and the closing (denoted $A \bullet B$) of an image $A$ by the structuring element $B$. Assuming a symmetrical structuring element, opening is defined as an erosion followed by a dilation (both operations using the same structuring element):

$$A \circ B = A \ominus B \oplus B \tag{5.5}$$

Similarly, closing is a dilation followed by an erosion:

$$A \bullet B = A \oplus B \ominus B \tag{5.6}$$

Opening and closing are both *idempotent* transforms; once applied to an image, further application will not transform the image any more:

$$A \circ B \circ B = A \circ B \tag{5.7}$$

$$A \bullet B \bullet B = A \bullet B \tag{5.8}$$

Figure 5.2 shows the effect application of the basic morphological operations using a disk shaped structuring element $B$ has on a binary image $X$. Erosion of $X$ by $B$ shrinks the image while dilation expands it. Opening $X$ with $B$ suppresses sharp capes and isthmuses, and closing fills in thin gulfs and small holes ([43], originally in [42]).

The application of a small convex structuring element to a one-dimensional signal (i.e. a one-dimensional gray-scale image) is shown in Figure 5.3. Erosion of the signal reduces sharp peaks and enlarges the signal minima, while dilation decreases the valleys and enlarges the maxima. Opening smooths the signal from below by cutting off peaks, and closing smooths the signal from above by filling in valleys ([43], originally in [42]).



## 5.3 The Watershed Transform

One the newer, and more interesting mathematical morphology-based methods is the Watershed Transform, which is based on geographical features of landscapes. This section presents a formal definition of the watershed transform.

In the physical world, watersheds are one of the classics in the field of topography [71]. If some rain falls on a landscape, the water will follow the swiftest descending path until it reaches

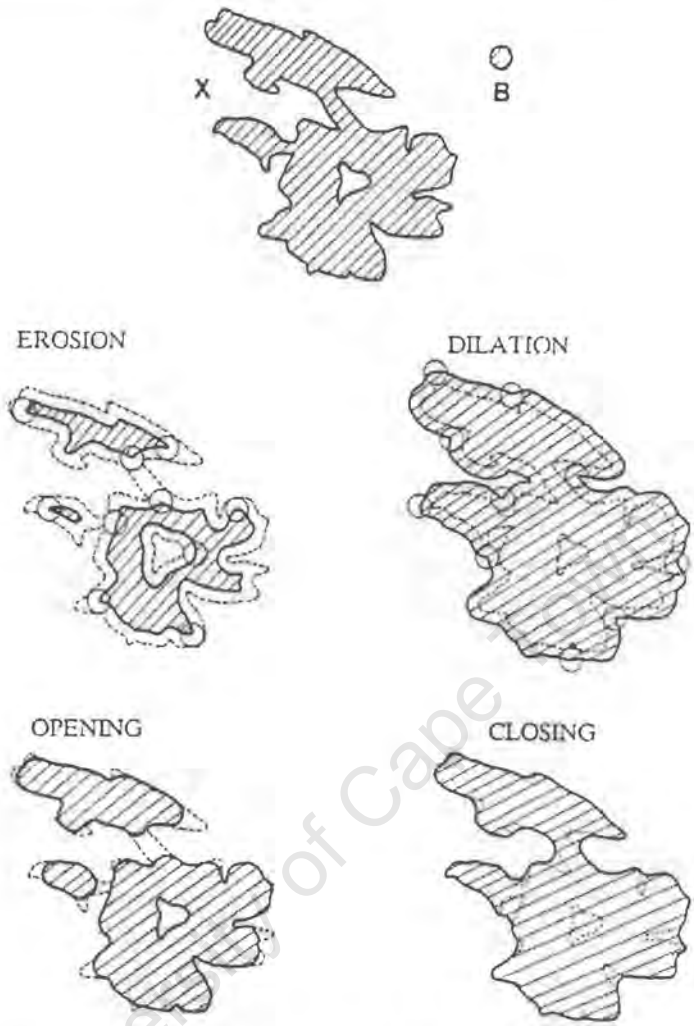



Figure 5.2: Binary Morphological Operations (from [43])



some lake or sea. A geographic region may then be divided according to the different areas of attraction of each lake, which are the watersheds of that region [9]. The lines that separate the different regions are the *watershed lines*, and the regions separated by these lines are also known as *catchment basins*. The lakes or seas are the *minima* associated with the catchment basins and watersheds [71] (Figure 5.4).

In mathematical morphology, grayscale images are often considered as topographic reliefs. In the topographic representation of an image $I$, the grayscale value of each pixel stands for the elevation at each point. The representation allows notions such as minima, catchment basins and watershed to be well defined for grayscale images [71].

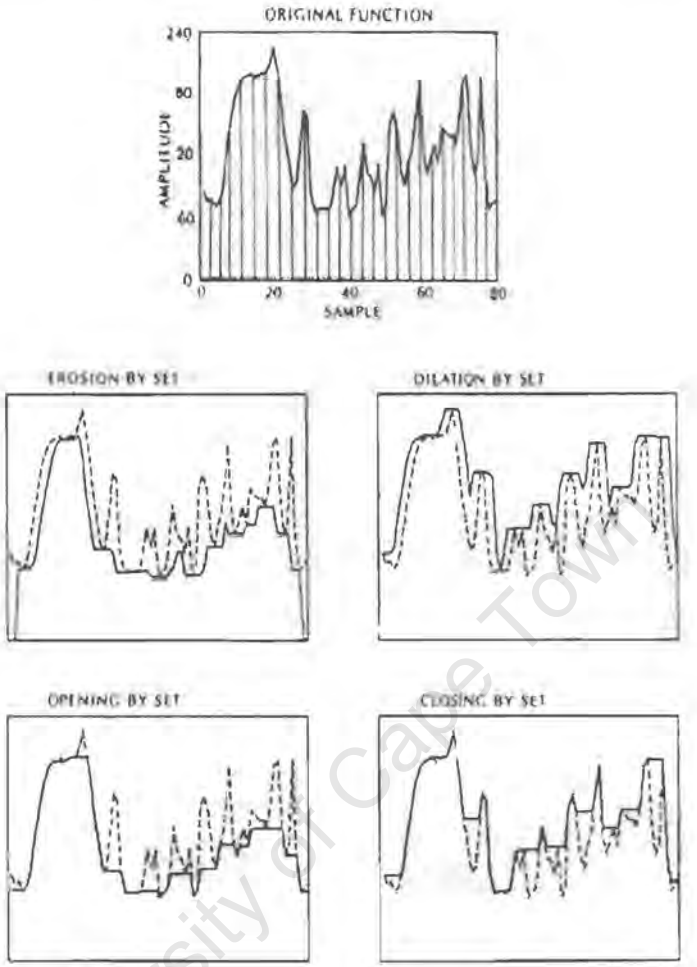



Figure 5.3: Grayscale Morphological Operations (from [43])



The watershed transform as a morphological tool was introduced by Digabel and Lantuéjoul [71]. Later work by Lantuéjoul and Beucher extended the algorithm to grayscale images. A theoretical approach by Maisonneuve used the watershed transform in grayscale segmentation problems, which is the context for its use in this research.

Previously available watershed algorithms have been either very slow or inaccurate. The introduction of an efficient implementation of the transform by Vincent and Soille [71] yielded performance hundreds of times faster than classical watershed transforms running on conventional computers. The (perceived) efficiency of the algorithm was one of the main reasons it has been used as the basis for this research, since real-time analysis is a major requirement.

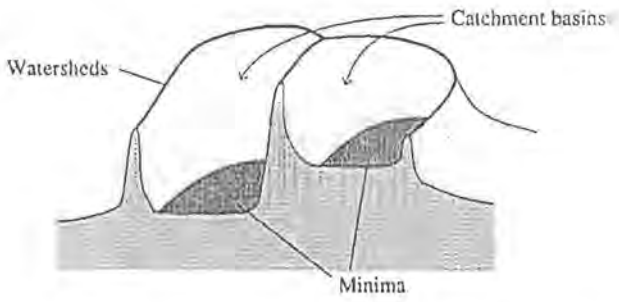



Figure 5.4: Catchment basins, watersheds and minima (from [71])

## 5.4 Definition and Computation of Watersheds

The formal definition of the watershed transform presented here is as given by Vincent and Soille [71]. There are two methods of defining watersheds, namely in terms of catchment basins and by using an "immersion". The catchment basin method relies on the computation of paths taken by drops of water falling on an image. Vincent and Soille argue that this method of watershed computation is not well suited to practical implementation. The immersion analogy requires holes to be pierced in the regional minima of an image. If the image is regarded as a surface, immersing it into a lake will cause water to fill up the catchment basins of the image, beginning with the minima of lowest altitude. Where the water rising from two different minima would merge, dams are constructed (see Figure 5.5). Once the immersion is complete, each catchment basin is surrounded by delimiting dams. The dams correspond to the watershed lines of the image.



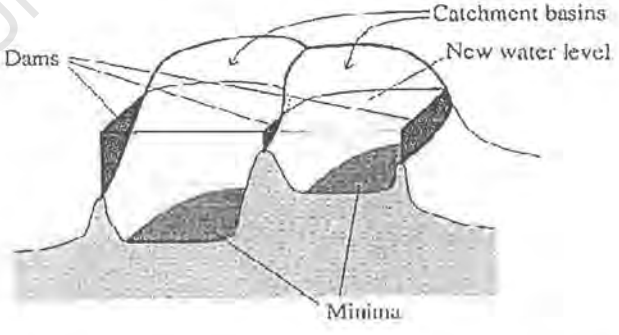



Figure 5.5: Building of dams to prevent merging of water from different minima (from [71])

### 5.4.1 Basic Definitions

Consider a two-dimensional grayscale image $I$ whose domain is denoted $D_I \subset \mathbb{Z}^2$. $I$ takes discrete gray values in a given range $[0, N]$, where $N$ is an arbitrary positive integer. Let $p$ represent any (arbitrary) pixel of $I$:

$$I \begin{cases} D_I \subset \mathbb{Z}^2 & \longrightarrow \{0, 1, \ldots, N\} \\ p & \longmapsto I(p) \end{cases} \tag{5.9}$$

Let $G$ denote the underlying digital grid, which can be a square in four or eight connectivity or a hexagon in six connectivity (see figure 5.6). $G$ is a subset of $\mathbb{Z}^2 \times \mathbb{Z}^2$.

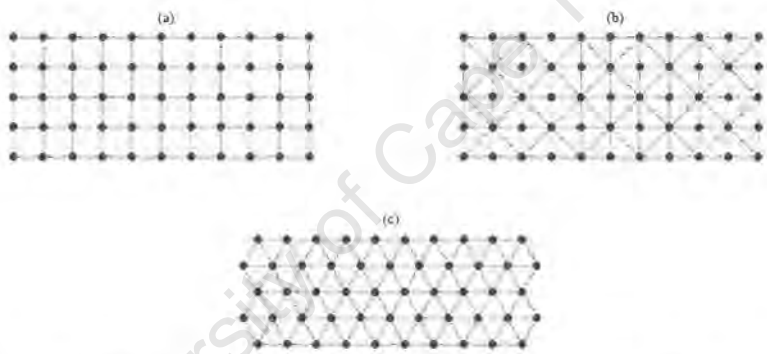



Figure 5.6: Digital grids in 4- (a), 8- (b) and 6- (c) connectivity



**Definition 5.4.1** *A path $P$ of length $l$ between two pixels $p$ and $q$ in image $I$ is a $(l+1)$-tuple of pixels $(p_0, p_1, \cdots, p_{l-1}, p_l)$ such that $p_0 = p$, $p_l = q$ and $\forall i \in [1, l]$, $(p_{i-1}, p_i) \in G$.*

In the following, $l(P)$ will denote the length of path $P$, and $N_G(p)$ will denote the set of neighbours of a pixel $p$ with respect to the grid $G$, i.e. $N_G(p) = \{p' \in \mathbb{Z}^2, (p, p') \in G\}$.

**Definition 5.4.2** *A minimum $M$ of image $I$ is a connected plateau of pixels from which it is impossible to reach a point of lower altitude without climbing:*

$$\begin{gathered} \forall p \in M, \forall q \notin M, \text{ such that } I(q) \leq I(p), \\ \forall P = (p_0, p_1, \cdots, p_l) \text{ such that } p_0 = p \text{ and } p_l = q, \\ \exists i \in [1, l] \text{ such that } I(p_i) > I(p_0) \end{gathered} \tag{5.10}$$

Thus a minimum is a connected area where the gray level is strictly darker than that of all the neighbouring pixels.

### 5.4.2 Definition by Immersion

The immersion process described above can be expressed more formally as follows:

Let $I$ be the grayscale image undergoing immersion, where $h_{min}$ denotes the smallest value taken by $I$ on its domain $D_I$. Similarly, $h_{max}$ denotes the largest value taken by $I$ on $D_I$. $T_h(I)$ stands for the threshold of $I$ at a level $h$:

$$T_h(I) = \{p \in D_I, I(p) \leq h\} \tag{5.11}$$

Now let $C(M)$ denote the catchment basin associated with minimum $M$ and let $C_h(M)$ denote the subset of $C(M)$ composed of the points with an altitude lower than or equal to $h$:

$$C_h(M) = \{p \in C(M), I(p) \leq h\} = C(M) \cap T_h(I) \tag{5.12}$$

With regard to the minima of $I$, $\min_h(I)$ denotes the set of points at altitude $h$ belonging to the minima of $I$.

**Definition 5.4.3** *The* geodesic distance $d_A(x, y)$ *between two pixels $x$ and $y$ in a (simply connected) set $A$ is the infimum of the length of all the paths that join $x$ and $y$ and are totally included in $A$:*

$$d_A(x, y) = \inf\{l(P), P \text{ path between } x \text{ and } y \text{ which is totally included in } A\} \tag{5.13}$$

Figure 5.7 illustrates this definition. Now suppose that $A$ contains a set $B$, which is composed of several connected components $B_1, B_2, \cdots, B_k$.

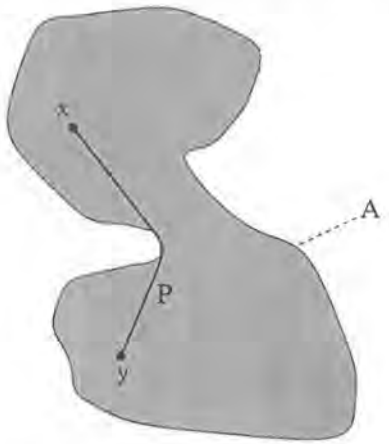



Figure 5.7: The geodesic distance between $x$ and $y$ in $A$

**Definition 5.4.4** *The* geodesic influence zone $iz_A(B_i)$ *of a connected component* $B_i$ *of* $B$ *in* $A$ *is the locus of the points of* $A$ *whose geodesic distance to* $B_i$ *is smaller than their geodesic distance to any other component of* $B$*:*

$$iz_A(B_i) = \{p \in A, \forall j \in [1,k]/\{i\}, d_A(p, B_i) < d_A(p, B_j)\} \tag{5.14}$$

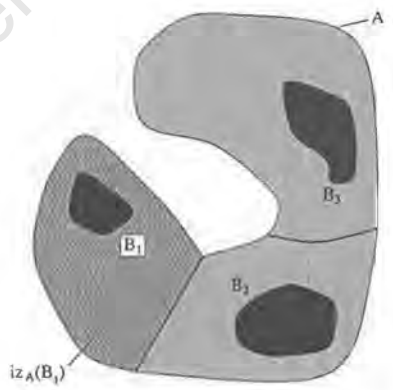



Figure 5.8: The influence zone of $B_1$ in $A$

Figure 5.8 illustrates the concept of influence zones. The *skeleton by influence zones (SKIZ)* of $B$ inside $A$ (denoted $\mathrm{SKIZ}_A(B)$) is the set of points of $A$ that do not belong to any geodesic

influence zone:

$$\mathrm{SKIZ}_A(B) = A/\mathrm{IZ}_A(B) \text{ with } \mathrm{IZ}_A(B) = \bigcup_{i\in[1,k]} iz_A(B_i) \tag{5.15}$$

The simulation of the immersion procedure starts from the set $T_{h_{\min}}(I)$, which are the points first reached by the water. Thus

$$X_{h_{\min}} = T_{h_{\min}}(I) \tag{5.16}$$

i.e. $X_{h_{\min}}$ is made up of the points of $I$ that belong to the lowest altitude minima. Now consider $T_{h_{\min}+1}(I)$ (that is, the threshold of $I$ at level $h_{\min}+1$). Obviously $X_{h_{\min}} \subseteq T_{h_{\min}+1}(I)$. If $Y$ is one of the connected components of $T_{h_{\min}+1}(I)$, then there are three possible inclusion relationships between $Y$ and $Y \cap X_{h_{\min}}$:

1. $Y \cap X_{h_{\min}} = \emptyset$: in this case, $Y$ is a new minimum of $I$. In fact, according to the definitions in Section 5.4.1, $Y$ is a plateau at level $h_{\min}+1$ since

$$\forall p \in Y, \begin{cases} p \notin X_{h_{\min}} & \rightarrow I(p) \geq h_{\min}+1 \\ p \in Y & \rightarrow I(p) \leq h_{\min}+1 \end{cases}$$

   In fact, all of the surrounding pixels do not belong to $T_{h_{\min}+1}(I)$ and have a gray value greater than $h_{\min}+1$. Thus the newly located minimum is "pierced", and its corresponding catchment basin is filled up with water.

2. $Y \cap X_{h_{\min}} \neq \emptyset$ and is connected: in this case, $Y$ exactly corresponds to the pixels belonging to the catchment basin associated with the minimum $Y \cap X_{h_{\min}}$ and having a gray value lower than or equal to $h_{\min}+1$:

$$Y = C_{h_{\min}+1}(Y \cap X_{h_{\min}}) \tag{5.17}$$

3. $Y \cap X_{h_{\min}} \neq \emptyset$ and is not connected: in this case, $Y$ contains different minima of $I$. These various minima are denoted $Z_1, Z_2, \cdots, Z_k$, and $Z_i$ is any one of the minima. At this point, the best possible choice for $C_{h_{\min}+1}(Z_i)$ is given by the geodesic influence zone of

$Z_i$ inside $Y$:

$$C_{h_{\min}+1}(Z_i) = iz_Y(Z_i) \tag{5.18}$$

Figure 5.9 demonstrates the three inclusion relationships as defined above.

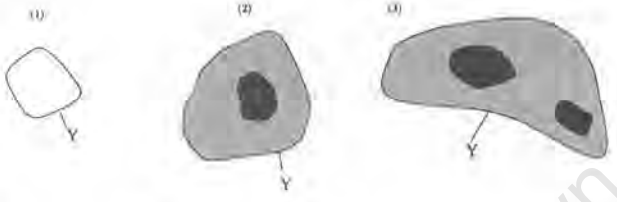



Figure 5.9: The three possible inclusion relationships between $Y$ and $Y \cap X_{h_{\min}}$

Since these relationships cover all possibilities, the second set in the recursion can be defined as follows:

$$X_{h_{\min}+1} = \min_{h_{\min}+1} \cup IZ_{T_{h_{\min}+1}}(X_{h_{\min}}) \tag{5.19}$$

This relation holds for all values of $h$, and allows the following definition to be obtained:

**Definition 5.4.5** *The set of catchment basins of the grayscale image $I$ is equal to the set $X_{h_{\max}}$ obtained after the following recursion:*

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

The set of watersheds of $I$ is simply the complement of the above set in $D_I$, i.e. it is the set of points of $D_I$ that do not belong to any catchment basin.

Figure 5.10 shows the relationship between two successive levels $X_h$ and $X_{h+1}$.

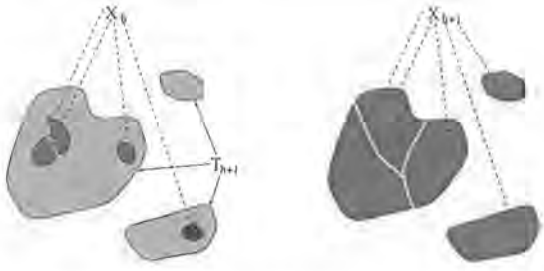


Figure 5.10: The relationship between $X_h$ and $X_{h+1}$

## 5.5 Reconstruction

Reconstruction is another useful, yet little used, operator provided by mathematical morphology [70]. As with most morphological operators, it operates on both binary and grayscale images. The definition of reconstruction presented here is as given by Vincent in [70].



### 5.5.1 Notation

Since reconstruction and the watershed transform are both morphological operations, the definitions for an image $I$, its domain $D_I \subset \mathbb{Z}^2$ and the discrete grid $G$ (Figure 5.6) are identical, and are as defined in 5.4.1. $G$ provides the neighbourhood relationships between pixels. A pixel $p$ is a neighbour of $q$ if and only if $(p, q) \in G$. The elementary ball or disk structuring element on $G$ is denoted $B_G$ or simply $B$ (recall Figure 5.1). $N_G(p)$ denotes the set of neighbours of pixel $p$ on grid $G$. In the following definition of reconstruction, use is made of two disjointed subsets of $N_G(p)$, denoted $N_G^+(p)$ and $N_G^-(p)$. $N_G^+$ consists of the neighbours of $p$ reached *before* $p$ in a raster (that is, left to right and top to bottom) scanning of an image, while $N_G^-(p)$ is the set of neighbours of $p$ reached *after* $p$ (Figure 5.11).


Figure 5.11: $N_G^+(p)$ and $N_G^-(p)$ in 8-connectivity

### 5.5.2 Binary Reconstruction

*1) Definition in terms of Connected Components:* Let $I$ and $J$ be two binary images defined on the same domain $D$ and let $J \subseteq I$. This means that $\forall p \in D, J(p) = 1 \Rightarrow I(p) = 1$. $J$ is the *marker* image and $I$ is the *mask*. Let $I_1, I_2, \cdots, I_n$ be the connected components of $I$.

**Definition 5.5.1** *The reconstruction $\rho_I(J)$ of mask $I$ from marker $J$ is the union of the connected components of $I$ which contain at least a pixel of $J$*

$$\rho_I(J) = \bigcup_{J \cap I_k \neq \emptyset} I_k$$

Binary reconstruction simply extracts the connected components that are "marked" by another image. This concept is illustrated by figure 5.12.

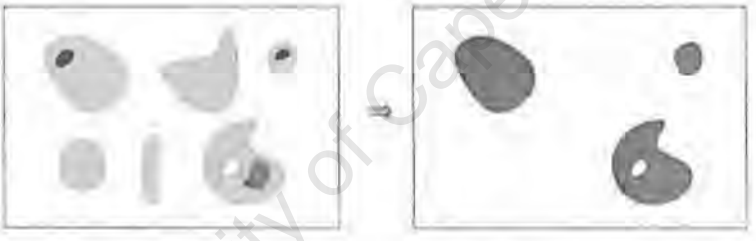


Figure 5.12: Binary Reconstruction from markers

*2) Definition in terms of Geodesic Distance:* Most definitions of binary reconstruction are presented using geodesic distance, which was defined in Section 5.4.2. Geodesic distance is used in the definition of *geodesic dilation* (and erosion):

**Definition 5.5.2** *Let $X \subset \mathbb{Z}^2$ be a discrete subset of $\mathbb{Z}^2$ and $Y \subset X$. The geodesic dilation of size $n \geq 0$ of $Y$ within $X$ is the set of all pixels of $X$ whose geodesic distance to $Y$ is smaller than or equal to $n$:*

$$\delta_X^{(n)}(Y) = \{p \in X | d_X(p, Y) \leq n\}$$

A geodesic dilation of a given size $n$ can be obtained by iterating an elementary ($n = 1$) geodesic dilation $n$ times:

$$\delta_X^{(n)}(Y) = \underbrace{(\delta_X^{(1)} \circ \delta_X^{(1)} \circ \cdots \circ \delta_X^{(1)})(Y)}_{n \text{ times}} \tag{5.20}$$

An elementary geodesic dilation can itself be obtained by performing a standard dilation with the unit disk structuring element $B$, followed by an intersection:

$$\delta_X^{(1)}(Y) = (Y \oplus B) \cap X \tag{5.21}$$

Using this relationship, binary reconstruction can be defined as follows:

**Definition 5.5.3** *The reconstruction $\rho_X(Y)$ of $X$ from $Y \subseteq X$ is obtained by iterating elementary geodesic dilations of $Y$ inside $X$ until stability is reached:*

$$\rho_X(Y) = \bigcup_{n \geq 1} \delta_X^{(n)}(Y)$$

This forms the basis of one of the simplest algorithms for computing geodesics reconstructions in both binary and grayscale cases.



### 5.5.3 Grayscale Reconstruction

Grayscale reconstruction can be defined in a similar fashion to binary reconstruction. The most useful definition, which is given here, is based on grayscale geodesic dilations.

*Definition in terms of Geodesic Distance:* The elementary geodesic dilation $\delta_I^{(1)}$ of a grayscale image $J \leq I$ "under" $I$ is given by

$$\delta_I^{(1)} = (J \oplus B) \wedge I \tag{5.22}$$

Here $\wedge$ is the pointwise minimum operator and $J \oplus B$ is the dilation of $J$ by the elementary ball structuring element $B$. These are direct extensions of intersection and dilation in the binary case.

The grayscale geodesic dilation of size $n \geq 0$ is then given by

$$\delta_I^{(n)}(J) = \underbrace{(\delta_I^{(1)} \circ \delta_I^{(1)} \circ \cdots \circ \delta_I^{(1)})(J)}_{n \text{ times}} \tag{5.23}$$

which leads to a definition of grayscale reconstruction:

**Definition 5.5.4** *The grayscale reconstruction $\rho_I(J)$ of $I$ from $J$ is obtained by iterating grayscale geodesic dilations of $J$ "under" $I$ until stability is reached:*

$$\rho_I(J) = \bigvee_{n \geq 1} \delta_I^{(n)}(J)$$

where $\bigvee$ is the *supremum* operator, which is the image processing equivalent of intersection in a complete lattice [5].

In practice, grayscale reconstruction extracts the *peaks* of the marker image $f$ that are marked by the marker image $g$ (figure 5.13) The dual operation of reconstruction (known as *dual reconstruction*) can be defined in terms of geodesic erosions $\epsilon_I^{(n)}(J)$, which themselves are defined in a manner similar to geodesic dilations:

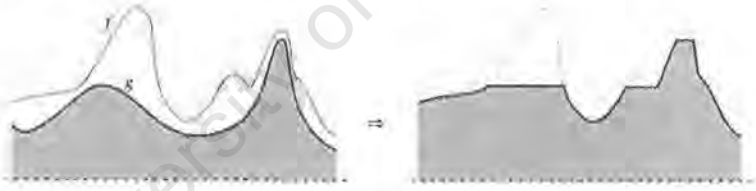


Figure 5.13: Grayscale reconstruction

$$\epsilon_I^{(1)}(J) = (J \ominus B) \vee I \tag{5.24}$$

where $\vee$ is the pointwise maximum operation and $J \ominus B$ is the erosion of $J$ by structuring element $B$. Thus, dual grayscale reconstruction can be defined as follows:

**Definition 5.5.5** *Let $I$ and $J$ be two grayscale images such that $I \leq J$. The dual grayscale reconstruction $\rho_I^*(J)$ of mask $I$ from marker $J$ is obtained by iterating grayscale geodesic erosions of $J$ "above" $I$ until stability is reached:*

$$\rho_I^*(J) = \bigwedge_{n \geq 1} \epsilon_I^{(n)}(J)$$

where $\bigwedge$ is the *infimum* operator, analogous to the union operation [5].

Since reconstruction and dual reconstruction are morphological operations, they can be time-consuming when performed on large images. Chapter 7 presents a fast implementation of these algorithms, as used in this research.

It should be noted that dual reconstruction can be implemented using the reconstruction operator by simply inverting the mask and marker images, applying the reconstruction algorithm and inverting the result. This is due to the complementary nature of the reconstruction and dual reconstruction operators. More formally:

$$\rho_J^*(I) \equiv (\rho_{J'}(I'))'.$$

# Chapter 6

# Preprocessing of Froth Images



For suitable segmentation to be obtained from using the Watershed Transform on froth images, a certain amount of image preprocessing must be performed. This chapter discusses the preprocessing methods that were investigated during this research.

Section 6.1 briefly describes image inversion, which is a mandatory preprocessing step for all froth images. Section 6.2 illustrates why further preprocessing is required for froth images. Sections 6.3 and 6.4 discuss the problem of noise in the image and methods of reducing this noise. Section 6.5 introduces a method for improving the contrast of froth images, and Sections 6.6, 6.7 and 6.8 describe the use of various morphological operators for improving the segmentation accuracy.

## 6.1 Image Inversion

The notion of a froth image as being similar to a topographic surface, with bubble boundaries being valleys and bubble surfaces being peaks, was introduced in Chapter 4. In the formal definition of the Watershed Transform, as given in Chapter 5, it was noted that each watershed region contained an intensity minimum. In order for each bubble in a froth image to correspond to a watershed region, the image must be inverted (i.e. light pixels become dark, and vice versa, see Figure 6.1).

The result of the inversion is that (ideally), each bubble is a catchment basin, with the bubble

boundaries corresponding to watershed lines, and the bubble highlights corresponding to regional minima. Figure 6.2 shows an intensity profile plot of scanline 150 from figure 6.1 (compare with figure 4.3. The minima of the plot correspond to bubble highlights and the maxima correspond to bubble boundaries. The image is now in a form that is appropriate for application of the Watershed Transform.

## 6.2 Direct Application of the Watershed Transform

Figure 6.3 shows the result of the Watershed Transform on the image in figure 6.1. The segmentation is clearly unacceptable, and is due to:

- Noise in the image, caused by various indeterminate sources.
- Interlacing, from the image acquisition.



## 6.3 Image Interlacing

A slight digression is required at this point, in order to define the concept of image interlacing. The images used in this research were acquired with a low-cost **CCD** (Charge Coupled Device) video camera operating on the PAL Television Standard. A CCD camera contains an array of photodetectors, a set of electronic switches and control circuitry. The photodetector array is scanned element by element to produce an image [35].

Each complete scan of the photodetector array is called a *frame*, which contains 625 lines in the case of the PAL system, and is scanned at a rate of 25 frames per second. Each frame is composed of two *interlaced* fields. To avoid flicker, alternate fields are sent at a rate of 50 fields per second. The first field contains the odd lines, and the second field contains the even lines [35].

Considering the above discussion, it is clear that there is a 200 millisecond delay between the odd and even fields in a frame. When acquiring images of a dynamic subject (such as a froth, which is continuously moving), the delay between fields may introduce smearing or blurring when the

Figure 6.1: Inverted froth image I1

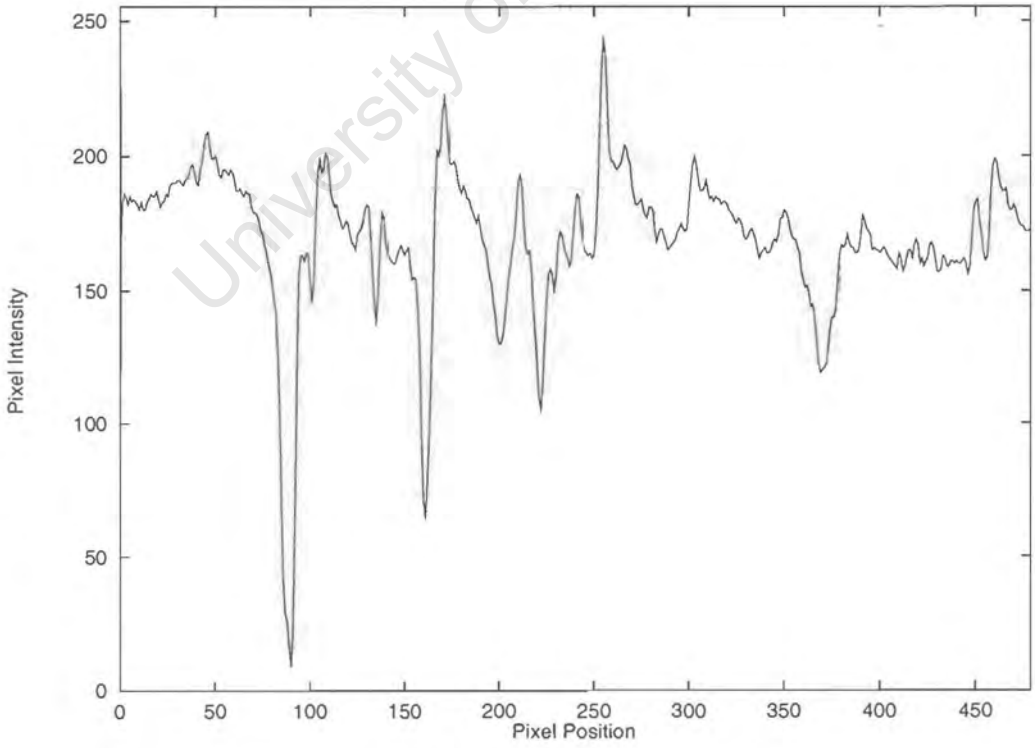



Figure 6.2: Pixel Intensity plot along scanline 150 for inverted image I1

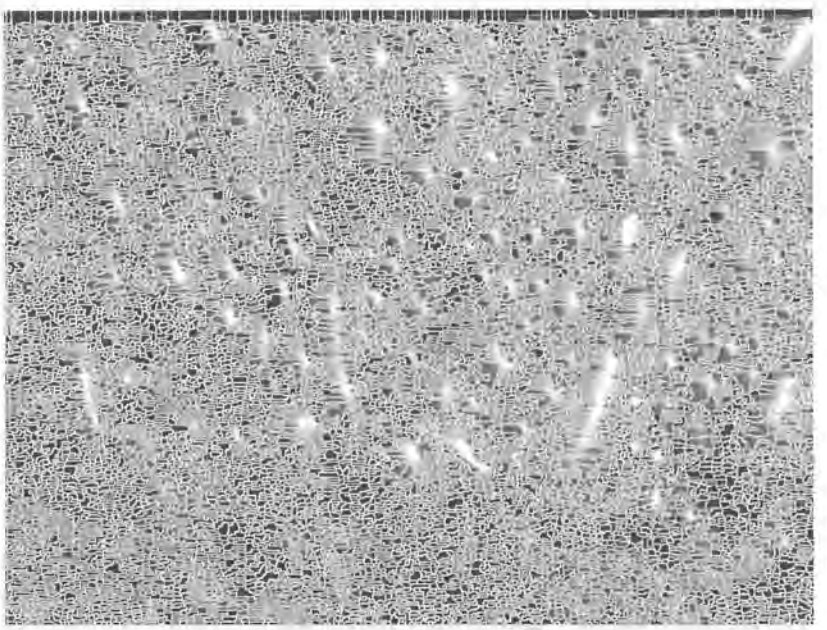


Figure 6.3: Result of Direct Application of the Watershed Transform to image I1

odd and even fields are combined to produce a single frame. This blurring can seriously affect the performance of the watershed segmentation, as is shown in figure 6.3.

It is worth noting that *progressive scan* cameras, which do not use interlacing when acquiring images, are available. These cameras are generally, however, much more expensive than their interlaced counterparts.



## 6.4 Image De-noising

Figure 6.4 shows a pixel intensity plot of column 320 from figure 4.1. A column plot shows the effect of interlacing more than a row (i.e. scanline plot), because both even and odd fields are present in a column plot, whilst a scanline is a single row from either an odd or even field. The high frequency low-intensity noise caused by the interlacing is clearly visible on the plot. A method of lessening the effects of interlacing, using the individual fields of the image frame, is presented in Section 6.4.1. Another method, known as *Anisotropic Diffusion*, which suppresses both interlacing artifacts and other sources of image noise, is defined in Section 6.4.2

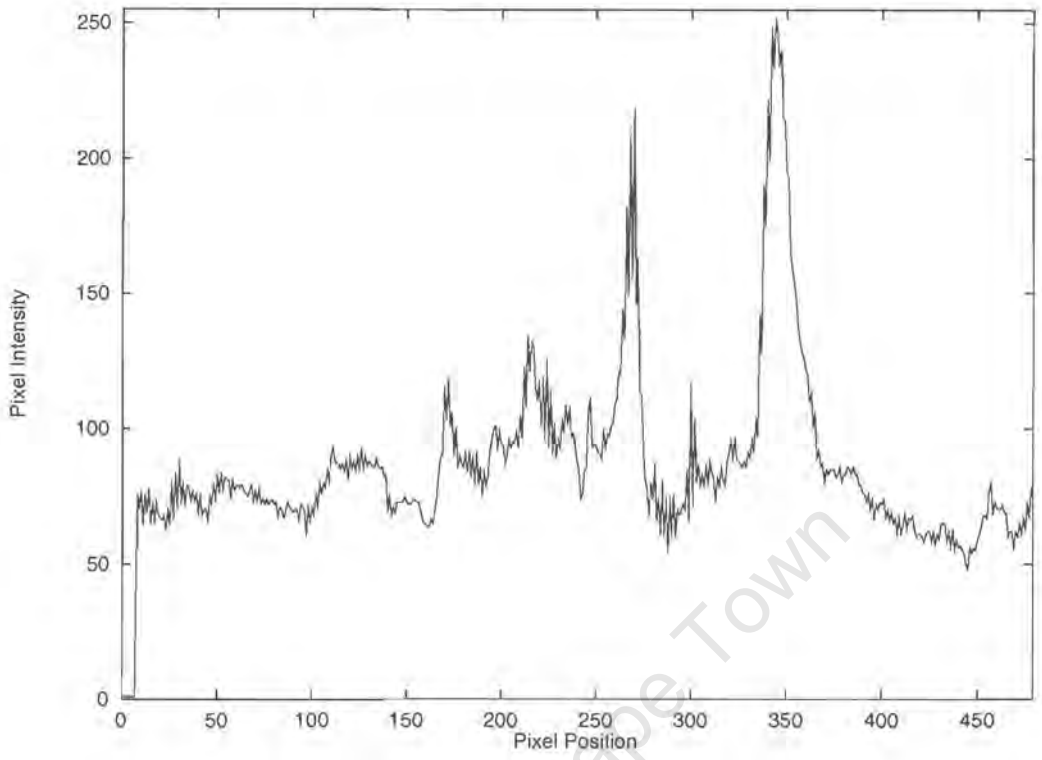



Figure 6.4: Pixel Intensity Plot of column 320 from image I1



### 6.4.1 Frame Splitting and Interpolation

A simple method of eliminating interlacing in an image is to split the image into odd and even fields — that is, to create two new images, one containing only the even numbered lines of an image and one containing the odd numbered lines. The resultant images are the same width as the original image, but are half the height. For size and shape measurement, it is more convenient to work with full-size images, and thus in order to restore the image to its original size, it is necessary to perform *interpolation* on one (or both) of the half-height images.

Linear interpolation, which is satisfactory for most image processing applications, is a first-order hold where a straight line is fitted in between pixels along a row. Then pixels along each column are fitted (i.e. interpolated) along a straight line [35]. For example, linear interpolation along rows gives

$$\left.\begin{aligned} v_1(m,2n) &= u(m,n), 0 \le m \le M-1, 0 \le n \le N-1 \\ v_1(m,2n+1) &= \tfrac{1}{2}[u(m,n)+u(m,n+1)], 0 \le m \le M-1, 0 \le n \le N-1 \end{aligned}\right\} \quad (6.1)$$

Linear interpolation of the preceding gives the first result as [35]

$$\left.\begin{aligned} v(2m, n) &= v_1(m, n) \\ v_(2m+1, n) &= \tfrac{1}{2}[v_1(m, n) + v_1(m+1, n)], 0 \leq m \leq M-1, 0 \leq n \leq 2N-1 \end{aligned}\right\} \quad (6.2)$$

for an original image of size $(M, N)$ and a new image of size $(2M, 2N)$.

In this particular application, only interpolation along the image columns is required, since the rows are still full-width. This amounts to applying Equation 6.2 to the half-height image.

Figure 6.5 shows the equivalent of figure 6.4 after splitting into odd and even fields and subsequent interpolation of the even field. The reduction of high frequency low intensity spikes is clearly visible. Figure 6.6 shows the result of the watershed transform on the interpolated image. While there is less oversegmentation than in figure 6.3, the result is still unacceptable. Thus additional preprocessing is required to further reduce the noise present in the original image. Note that it is assumed that all the preprocessing steps described in the following sections are performed **before** the image is inverted.



### 6.4.2 Anisotropic Diffusion

A common method of reducing noise in images is to perform low-pass filtering on the noisy image. One of the penalties paid for using low-pass filtering is that the edges in an image become blurred, which causes their detection and localisation to become difficult [56]. As far as froth images are concerned, the application of a low-pass filter will cause the bubble boundaries to become blurred, and since well-defined bubble boundaries are required for the watershed transform to work properly, the use of low-pass filtering in this case seems ill advised.

The use of anisotropic diffusion for noise removal in images attempts to solve some of the problems associated with using low-pass filtering and other classical techniques. When applied to froth images, anisotropic diffusion is able to lessen the effects of both image interlacing and other forms of image noise.

Anisotropic diffusion has its roots in the concept of *scale space*. Scale space filtering involves the extraction of features from a family of images that are at different resolutions (or "scales") to the original image under investigation. More formally, a family of derived images $I(x, y, t)$ is

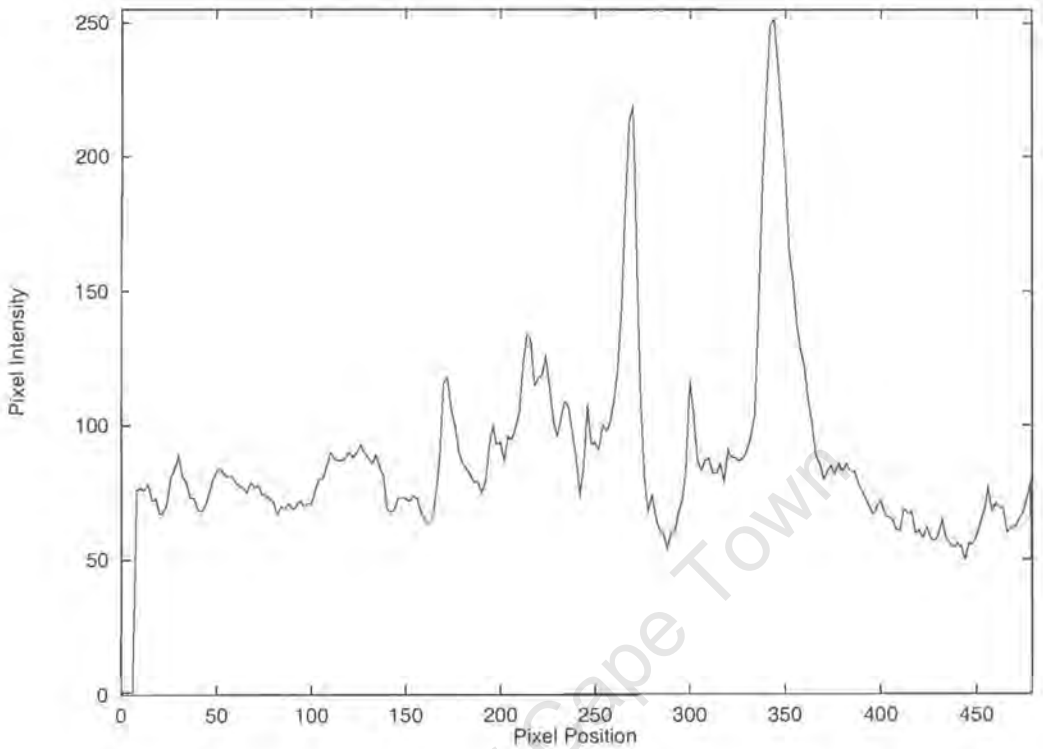



Figure 6.5: Pixel Intensity Plot of column 320 from interpolated image I1

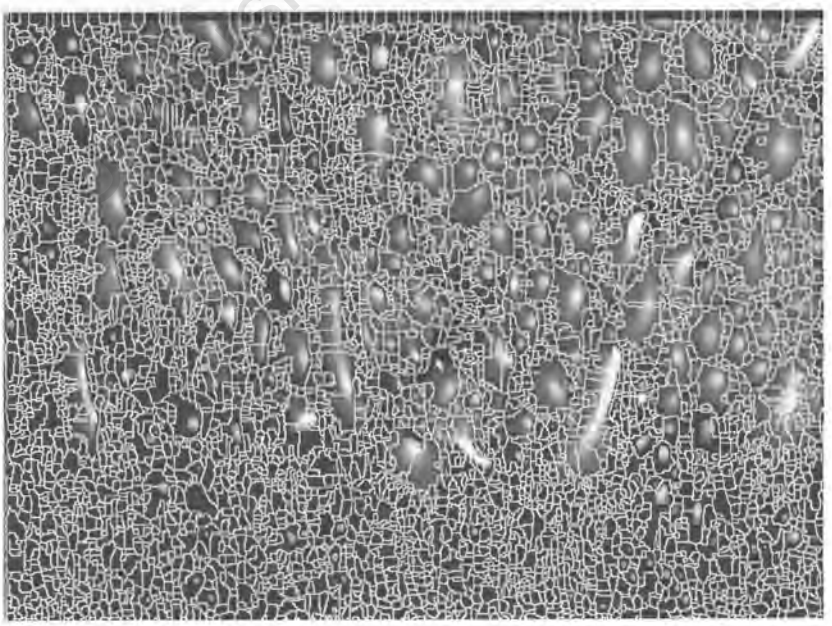


Figure 6.6: Result of application of the Watershed Transform to interpolated image I1

obtained by convolving the original image $I(x, y)$ with a Gaussian $G(x, y, t)$ with variance $t$:

$$I(x, y, t) = I_0(x, y) * G(x, y, t) \tag{6.3}$$

The one parameter family of derived images may also be viewed as the solution of the heat conduction, or diffusion, equation

$$I_t = k(I_{xx} + I_{yy}) \tag{6.4}$$

with the initial condition $I(x, y, 0) = I_0(x, y)$, the original image.

In order for a scale-space paradigm to generate "semantically meaningful" multiscale descriptions of images, it must satisfy three criteria:

1. *Causality:* No spurious detail should be generated when passing from finer to coarser scales.
2. *Immediate Localisation:* At each resolution (i.e. scale), region boundaries should be sharp and coincide with semantically meaningful boundaries at that resolution.
3. *Piecewise Smoothing:* At all scales, *intra*region smoothing should occur preferentially over *inter*region smoothing.

The objectives described above can be met by modifying the linear diffusion equation 6.4 given above. The anisotropic diffusion equation is given by

$$I_t = \mathrm{div}(c(x, y, t)\nabla I) = c(x, y, t)\Delta I + \nabla c \cdot \nabla I \tag{6.5}$$

where $div$ is the divergence operator, and $\nabla$ and $\Delta$ are the gradient and Laplacian operators respectively (with respect to the space variables). A suitable choice of $c(x, y, t)$ will satisfy the second and third criteria listed above without violating the causality criterion. If the conduction coefficient $c(x, y, t)$ is a constant then Equation 6.5 reduces to the isotropic heat diffusion equation $I_t = c\Delta I$.

A froth image can be considered as being composed of many regions (bubbles), with the edges between the regions being the bubble boundaries. To perform noise removal while preserving edges, smoothing should occur within a region in preference to smoothing across boundaries.

Thus the conduction coefficient should be set to 1 in the interior of each region, and 0 at the boundaries. Blurring would then take place within the regions, with the boundaries remaining sharp.

The difficulty in this approach is identifying the region boundaries (edges). What **can** be computed is a current best estimate of the location of the edges at an appropriate scale.

Let $\boldsymbol{E}(x, y, t)$ be such an edge estimate: a vector-valued function defined on the image which ideally should have the following properties:

1. $\boldsymbol{E}(x, y, t) = \mathbf{0}$ in the interior of each region.
2. $\boldsymbol{E}(x, y, t) = K\boldsymbol{e}(x, y, t)$ at each edge point, where $\boldsymbol{e}$ is a unit vector normal to the edge at the point, and $K$ is the local contrast of the edge.

Once an estimate of $\boldsymbol{E}(x, y, t)$ is available, the conduction coefficient $c(x, y, t)$ can be chosen to be a function $c = g(\|\boldsymbol{E}\|)$ of the magnitude of $\boldsymbol{E}$. The function $g(\cdot)$ should be a non-negative monotonically decreasing function with $g(0) = 1$, to ensure that the diffusion will mainly take place in the interior of regions while not affecting boundaries where the magnitude of $\boldsymbol{E}$ is large. As far as "guessing" the edge positions is concerned, Perona and Malik [56] state that a simple gradient of the brightness function, i.e. $\boldsymbol{E} = \nabla I(x, y, t)$ gives excellent results.

If the conduction coefficient is chosen to be an appropriate function of the image gradient, the anisotropic diffusion becomes a stable method of enhancing edges in an image whilst removing noise. A suitable conduction coefficient is a function of the magnitude of the gradient of the brightness function (as mentioned above):

$$c(x, y, t) = g(\|\nabla I(x, y, t)\|). \tag{6.6}$$

For implementation on a square lattice (such as digital image), Equation 6.5 can be discretised, with brightness values being associated to the vertices and conduction coefficients to the arcs (see Figure 6.7). A 4-neighbour discrete version of the Laplacian can be used:

$$I_{i,j}^{t+1} = I_{i,j}^{t} + \lambda[c_N \cdot \nabla_N I + c_S \cdot \nabla_S I + c_E \cdot \nabla_E I + c_W \cdot \nabla_W I]_{i,j}^{t} \tag{6.7}$$

where $0 \leq \lambda \leq 1/4$ to ensure numerical stability, $N$, $S$, $E$, $W$ are the mnemonic subscripts for the cardinal compass directions, the subscripts and superscripts outside the square bracket

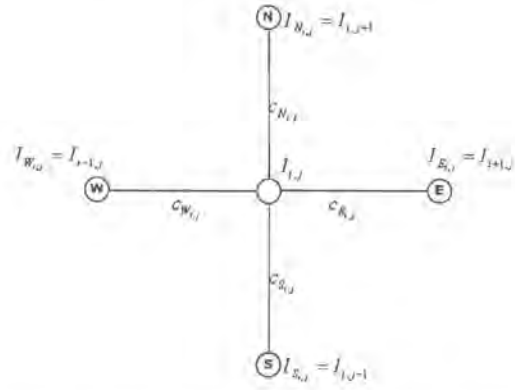



Figure 6.7: Structure of the discrete computational scheme for simulating the diffusion equation

are applied to all terms enclosed by it, and $\nabla$ (**not** the gradient operator $\nabla$) indicates nearest neighbour difference:

$$\begin{aligned} \nabla_N I_{i,j} &\equiv I_{i-1,j} - I_{i,j} \\ \nabla_S I_{i,j} &\equiv I_{i+1,j} - I_{i,j} \\ \nabla_E I_{i,j} &\equiv I_{i,j+1} - I_{i,j} \\ \nabla_W I_{i,j} &\equiv I_{i,j-1} - I_{i,j} \end{aligned} \tag{6.8}$$

The conduction coefficients are updated at every iteration as a function of the brightness gradient (Equation 6.6):

$$\begin{aligned} c^t_{N_{i,j}} &= g(\|(\nabla I)^t_{i+(1/2),j}\|) \\ c^t_{S_{i,j}} &= g(\|(\nabla I)^t_{i-(1/2),j}\|) \\ c^t_{E_{i,j}} &= g(\|(\nabla I)^t_{i,j+(1/2)}\|) \\ c^t_{W_{i,j}} &= g(\|(\nabla I)^t_{i,j-(1/2)}\|) \end{aligned} \tag{6.9}$$

The value of the gradient can be calculated in several ways. The simplest choice is calculated by approximating the norm of the gradient at each arc location with the absolute value of its

projection along the direction of the arc:

$$\begin{aligned} c^t_{N_{i,j}} &= g(\mid \nabla_N I^t_{i,j} \mid) \\ c^t_{S_{i,j}} &= g(\mid \nabla_S I^t_{i,j} \mid) \\ c^t_{E_{i,j}} &= g(\mid \nabla_E I^t_{i,j} \mid) \\ c^t_{W_{i,j}} &= g(\mid \nabla_W I^t_{i,j} \mid) \end{aligned} \tag{6.10}$$

The scheme described above is not an exact discretisation of 6.5, but it is computationally less complex than methods with less crude gradient approximations.

The method implemented for use on the froth images is the one given by equations 6.7, 6.8 and 6.10, with the original image as the initial condition. At first the 4-nearest-neighbour discrete Laplacian (Equation 6.7) was used, but later it was decided to use an 8-nearest neighbour discrete Laplacian ([35],

$$\begin{aligned} I^{t+1}_{i,j} &= I^t_{i,j} + \lambda[c_N \cdot \nabla_N I + c_S \cdot \nabla_S I + c_E \cdot \nabla_E I + c_W \cdot \nabla_W I \\ &+ c_{NW} \cdot \nabla_{NW} I + c_{NE} \cdot \nabla_{NE} I + c_{SE} \cdot \nabla_{SE} I + c_{SW} \cdot \nabla_{SW} I]^t_{i,j} \end{aligned} \tag{6.11}$$

to improve the quality of the diffusion. Equations 6.8 and 6.10 are easily modified for use with 8-nearest-neighbours.

The function used for $g(\cdot)$ is given by

$$g(\nabla I) = e^{(-(\|\nabla I\|/K)^2)} \tag{6.12}$$

According to Perona and Malik [56], this function favours high-contrast edges over low-contrast ones, which seems appropriate for enhancing bubble boundaries, which are much darker than the bubble surfaces. The constant $K$ is determined by use of the Canny "noise estimator" [10]: A histogram of the absolute value of the image gradient is computed, and $K$ is set to 90% of the value of its integral at each iteration.

The smoothing properties of anisotropic diffusion can be seen to good effect in Figure 6.9, as compared to Figures 6.4 and 6.5. Figure 6.8 shows the original image I1 after 6 iterations of anisotropic diffusion have been performed. The image is still recognisable as a froth image, although the blurring introduced is clearly visible (compare with Figure 4.1). Figure 6.9 is the equivalent of figure 6.4 after the anisotropic diffusion has been performed. The basic shape of

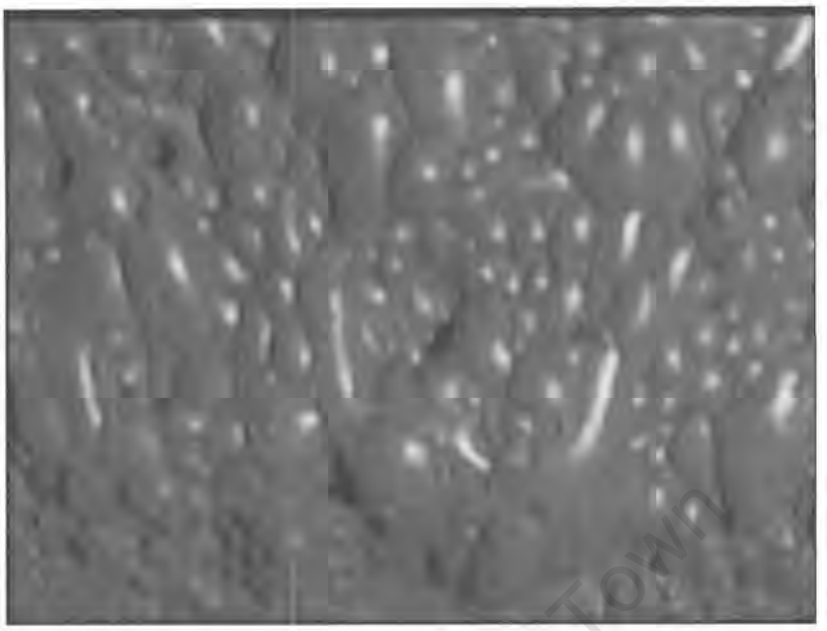


Figure 6.8: Image I1 after 6 anisotropic diffusion iterations



the plot has been retained, but the high-frequency deviations have been removed. Most of these deviations are due to noise, although some may be representative of very small bubbles. This is the main drawback with anisotropic diffusion (and in fact with any de-noising method) - very fine bubbles may be blurred out of existence if too many iterations are performed. On the other hand, too few iterations will not adequately de-noise the image. Additionally, too many iterations will increase the time required to segment an image, which is undesirable if near real-time processing is required. A nominal value of six iterations was chosen for this research. In equation 6.7 the value of $\lambda$ was chosen to be 1/8.

The usefulness of anisotropic diffusion as a preprocessing tool can clearly be seen in Figure 6.10, which shows the result of segmenting the original image after 6 iterations of the anisotropic diffusion algorithm. The result appears vastly superior to the segmentation shown in Figure 6.3, and also appears to be much better than the segmentation of Figure 6.6. The use of the word "appears" here implies that the resulting segmentation has a more pleasing qualitative visual appearance to the human eye.

It can be seen in figure 6.10 that, while small bubbles seem to have been reliably segmented, larger bubbles tend to be fragmented by the segmentation. This is due to the fact that small spots on the bubble surface that may be dimmer than the easily visible main bubble highlight

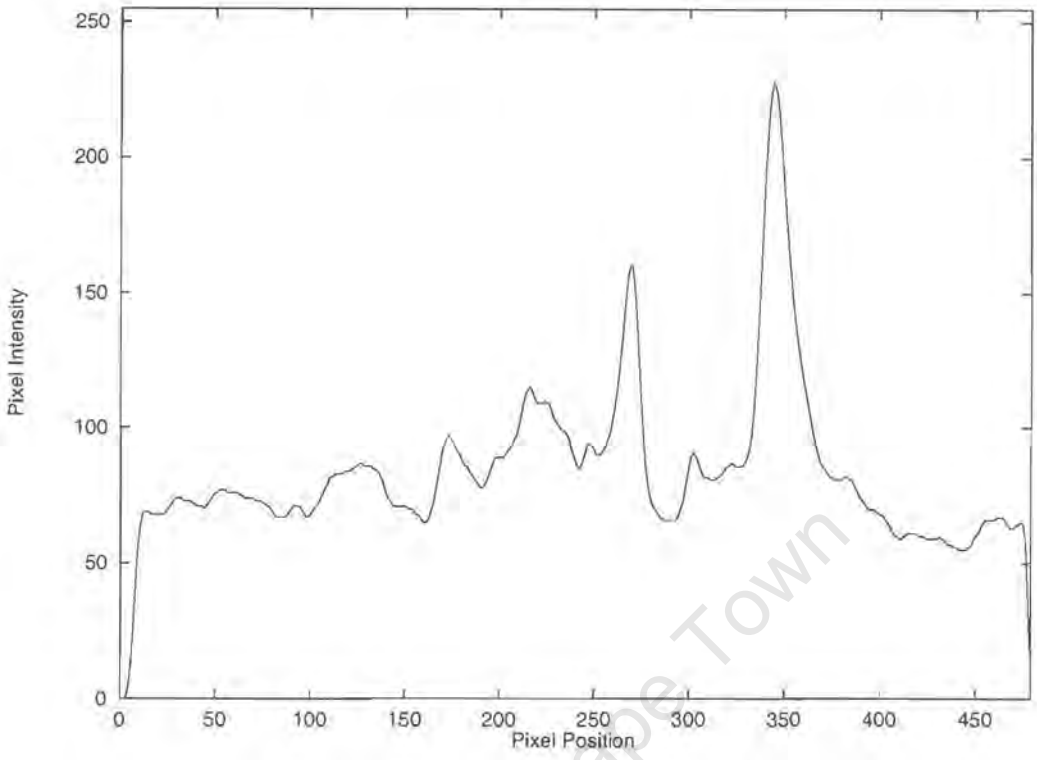



Figure 6.9: Pixel Intensity Plot of column 320 from image I1 after 6 anisotropic diffusion iterations



are large and bright enough to be retained by the anisotropic diffusion and thus be considered a highlight or minimum by the watershed segmentation algorithm. Additionally, if there are multiple highlights on a single bubble (which is most likely to occur on large bubbles in any case) due to ambient light or unwanted shadows, fragmentation will occur. Some simple (and hence computationally inexpensive) techniques that were investigated in an attempt to reduce large bubble fragmentation are presented below. The reader may recall that in Chapter 4, reference was made to bubble highlight saturation as being a factor that can cause poor segmentation of froth images. In fact, saturation of highlights will probably improve segmentation in this case, because saturated highlights can be unambiguously classified as minima by the Watershed Transform.

## 6.5 Histogram Equalisation

When examining froth images, it may sometimes be difficult to distinguish between the surface of the bubbles and the bubble boundaries. This is especially true if the contrast of the image is

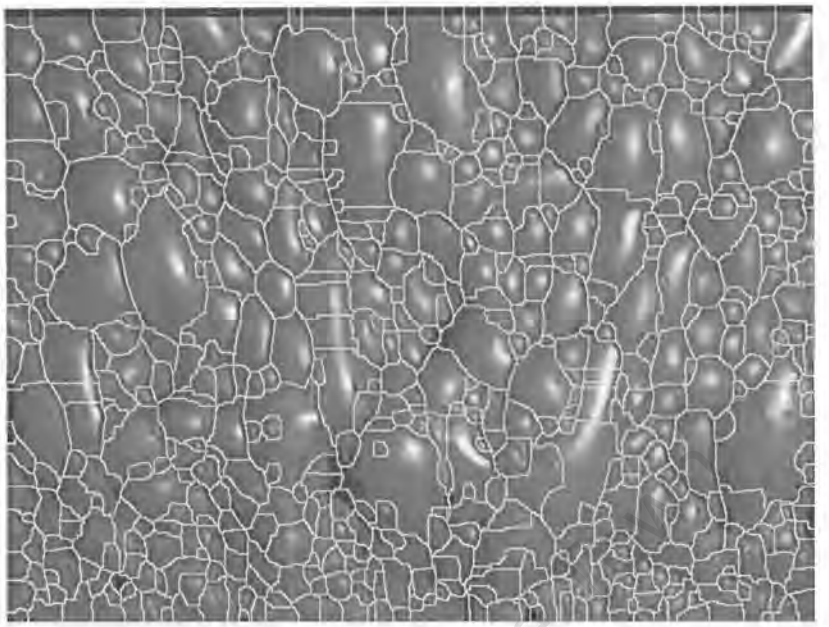


Figure 6.10: Segmented image I1 after anisotropic diffusion



poor.

Pixels in a grayscale image can be represented by any whole number from 0 to $2^8 - 1$ (that is, 255). If the number of grayscale values actually used in an image is small, the result is generally an image with poor contrast. A technique known as *histogram equalisation* is used to "spread out" the grayscale values in an image in such a way that each gray value is taken on by approximately the same number of pixels [57]. This results in improved image contrast. The motivation for applying histogram equalisation to froth images is that it should result in the surfaces of the bubbles being more uniform and there being increased contrast between the lighter bubble surfaces and the darker bubble boundaries.

A grayscale image histogram is simply a histogram in which each bin corresponds to a grayscale value. The number in each bin corresponds to the number of pixels of that particular shade of gray in the image. If the histogram is considered to be a probability density function (PDF), the aim of histogram equalisation is to map the input PDF $p_a(a)$ to an output PDF $p_b(b)$ that is approximately uniformly distributed [72]. This mapping is given by

$$b_i = \frac{G_{\max}}{b_{G_{\max}}} \sum_{i=0}^{j} a_i \tag{6.13}$$

where $a$ is the input histogram (PDF), $b$ is the output PDF, $G_{max}$ is the maximum gray level in the image, and $0 \leq j \leq G_{max}$ [72].


(a) Before equalisation

(b) After equalisation

Figure 6.11: Histograms for image I1 before and after equalisation

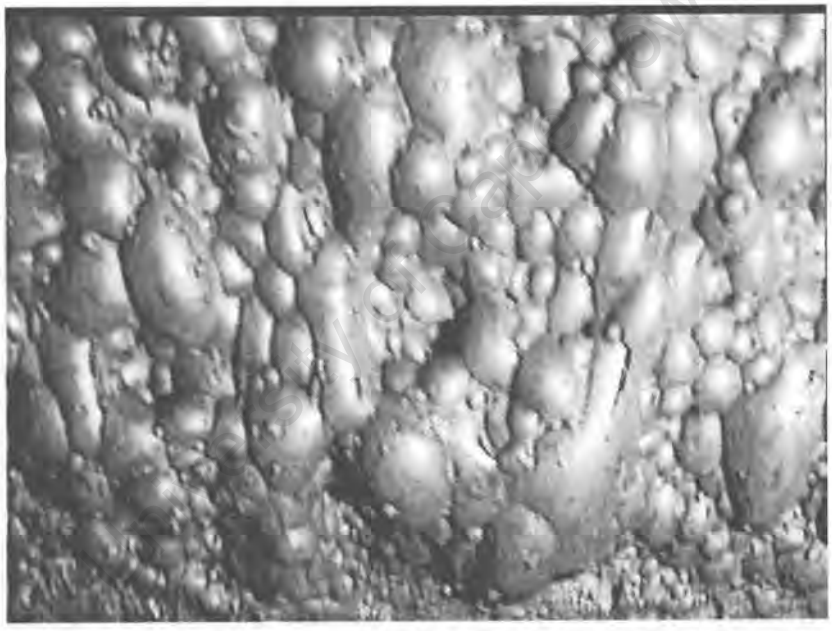


Figure 6.12: Image I1 after histogram equalisation

Figure 6.11 shows the effect equalisation has on the image histogram. Because grayscale values are discrete, the equalized histogram will only be an approximation to a uniform PDF. Figure 6.12 shows the result of the equalisation on image I1. The increased overall brightness of the bubble surfaces and the more prevalent bubble boundaries are clearly visible (compare with Figure 4.1).

## 6.6 Grayscale Dilation

In Chapter 4 and in Section 6.4.2 mention was made of the problems caused by multiple highlights or bright spots on large bubbles. One solution to this problem is to attempt to combine the distinct highlights into a single larger highlight. If the highlights are widely spaced on the bubble surface then it may be difficult to achieve this combination, but highlights that are close together may be easier to handle. A simple method of performing this task is to use a morphological dilation. Recall from Chapter 5 that a grayscale dilation decreases valleys and increases maxima. The action of dilation by a small structuring element on distinct highlights (i.e. maxima) that are close together will lead to the highlights being combined. The decreasing of valleys should not affect the segmentation to any great extent, because the Watershed Transform detects bubble boundaries (which are the "inverse" of valleys) of any height. The choice of size of structuring element (s.e.) is of great importance; if the s.e. is too small then distinct highlights will not be merged, and too large an s.e. will cause small individual bubbles to be merged. After several trials it was decided that a 5 x 5 spherical s.e. gave the best results.

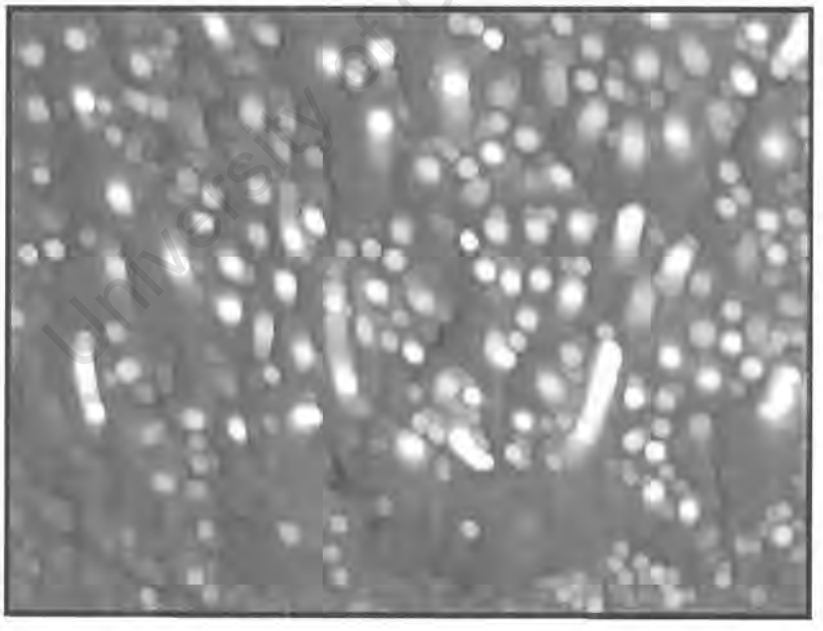


Figure 6.13: Image I1 after dilation with a 5 x 5 structuring element

Figure 6.13 shows the result of dilating image I1 with a 5 x 5 structuring element. The increased brightness of the highlights can be easily seen. Also clearly visible is the enhancement of less bright spots on the larger bubbles (compare with Figure 4.1) that cause fragmentation of the

larger bubbles. Thus histogram equalisation, as described in Section 6.5 is required to render the bubble surfaces as uniform as possible.

On a discrete grid (such as a digital image), grayscale dilation can be defined as [60]

$$D(x, y) = \max_{i,j}[A(x - i, y - j) + B(i, j)] \tag{6.14}$$

where $A$ is the original image, $B$ is the structuring element and $D$ is the resultant dilated image. The following code snippet implements Equation 6.14:

```
void toolbox::dilate(Image &Imin, Image &se, Image &Imout) {

 int rows,cols,max,x,y,i,j,temp,se_R;
rows = Imin.Getyval();
cols = Imin.Getxval();
se_R = (int)(se.Getxval()-1)/2;
for (x = se_R ; x < cols-se_R ; x++)
  for (y = se_R ; y < rows-se_R ; y++) {
  max = -32768;
  for (i = -se_R ; i <= se_R ; i++)
    for (j = -se_R ; j <= se_R; j++) {
                if (se(i+se_R,j+se_R) > 0) {
                temp = Imin(x+i,y+j) + se(i+se_R,j+se_R);
                if (temp > 255)
                        temp = 255;
                if (temp > max)
                        max = temp;
          }
    }
    Imout(x,y) = max;
  }
}
```

## 6.7 Marker Extraction

When using the watershed transform to segment grayscale images, the correct method of use [71, 70] is to extract *markers* that correspond to the objects being segmented, and to begin the watershed flooding (Chapter 5) from these markers. This section presents a method of marker extraction for flotation froth images.

### 6.7.1 Markers for Froth Images

Since the aim of segmenting flotation froth images is to identify the individual bubbles in the froth, there should (optimally) be a one-to-one correspondence between the markers extracted from the image and the bubbles in the image. That is, each bubble in the image should have one, and only one, corresponding marker. While this may be difficult to obtain in practice, is is possible to achieve this to a reasonable extent.

An obvious candidate for marking a bubble is the bubble itself! This may seem circular (to properly detect the bubbles, the bubbles must first be located), but it is in fact possible to extract the upper surfaces of bubbles from an image and use these top surfaces to generate the markers. This process will be most successful if the bubbles are "smooth" (that is, the image is reasonably noise-free), which indicates that marker extraction should be performed after anisotropic diffusion of the images. To perform the marker extraction, a morphological process known as *grayscale reconstruction*, which was formally defined in Chapter 5, is used.



### 6.7.2 Regional Maxima and Dome Extraction

It was stated above that the top surfaces of bubbles can be used as markers to assist in the segmentation process. The concept of *regional maxima* and *dome extraction* [70] are most useful for extracting the tops of the bubbles, and are defined as follows:

**Definition 6.7.1** *A regional maximum $M$ of a grayscale image is a connected component of pixels with a plateau of value $h$, such that every pixel in the neighbourhood of $M$ has a strictly lower value.*

In an (ideal) froth image, the top surface of each bubble will be a regional maximum. These regional maxima can be extracted from an image by using the grayscale reconstruction operator $/rho$:

**Definition 6.7.2** *The (binary) image $M(I)$ of the regional maxima of an image $I$ is given by:*

$$M(I) = I - \rho_I(I - 1)$$

In practice, extracting the regional maxima from a froth image is not always satisfactory, since maxima of small image variations due to noise may be extracted as well as the desired maxima. Now, instead of subtracting a value of 1 in Definition 6.7.2, an arbitrary gray-level constant $h$ can be subtracted from the image. This method extracts "domes" of a given height, which are known as $h$-domes. The following definition can thus be proposed:

**Definition 6.7.3** *The $h$-dome image $D_h(I)$ of the $h$-domes of a grayscale image $i$ is given by*

$$D_h(I) = I - \rho_I(I - h) \tag{6.15}$$

The $h$-dome transformation, as illustrated in figure 6.14, extracts light structures (the top surfaces of bubbles in the case of application to froth images) without involving any size or shape criterion. The only free parameter is $h$, which is related to the height of the structures.

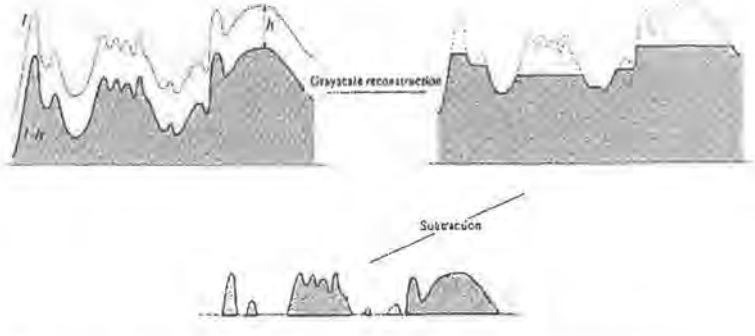



Figure 6.14: Determining the $h$-domes of grayscale image $I$

**Values of $h$ for Froth Images**

The value of $h$ chosen for the extraction process has a significant effect on the resulting domes. Figures 6.15 to 6.18 show the extracted domes of image I1 for $h$ values of 5, 15, 50 and 100[1]. As the value of $h$ increases, more of a bubble's upper surface is included in the corresponding dome. This has an effect on the resultant segmentation, as will be shown later.

### 6.7.3 Image Thresholding

After extracting the domes of an image, it is necessary to produce a binary marker image from the grayscale dome image. This is done by thresholding the dome image such that all pixels with a value greater than or equal to $h - 1$ are given a value of 0, and all other pixels are given a value of 255. Figure 6.19 is an example of such a binary marker image obtained from figure 6.16.

## 6.8 Image Modification

Following the determination of a binary marker image, it is possible to modify the original image in such a way that the *only* regional minima present are those that correspond to minima as specified by the marker image. The removal of unwanted minima from an image should result in better segmentation by the watershed transform, since the flooding will only begin from the specified minima. Modifying the image in this way is known as *homotopy modification*.

### 6.8.1 Homotopy: A Definition

Before describing homotopy modification, it is necessary to define the concept of homotopy. It is a topological property of sets [6], and can be defined practically by considering two sets $X$ and $Y$. These sets are said to be homotopic if the first one can be superimposed onto the second one by means of continuous deformations. In the same manner, a transformation $\Phi$ is said to be homotopic if it transforms $X$ into an homotopic set $\Phi(X)$. By this token, a simply connected set

[1] Note that these images have been modified by histogram stretching, which increases the contrast of the images, and greyscale inversion.

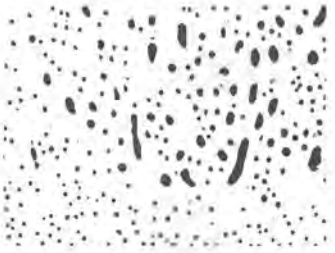


Figure 6.15: Domes of height 5 for image I1

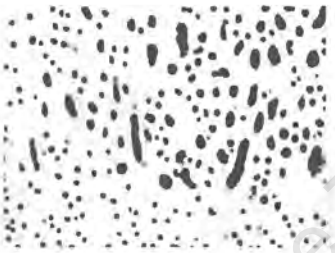


Figure 6.16: Domes of height 15 for image I1

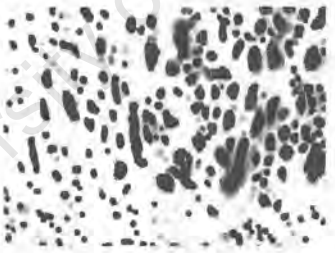


Figure 6.17: Domes of height 50 for image I1

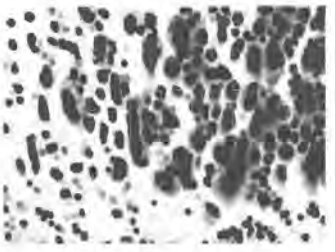


Figure 6.18: Domes of height 100 for image I1

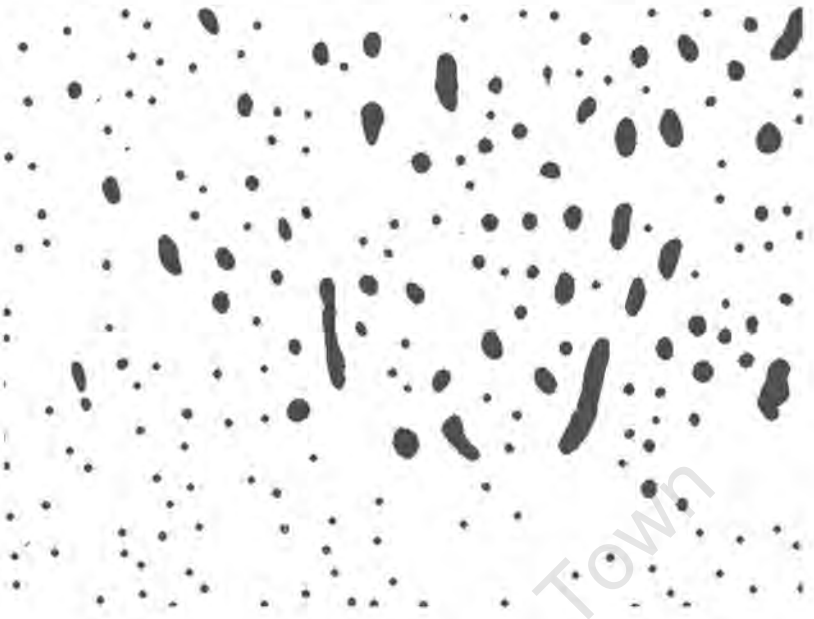


Figure 6.19: The binary marker image obtained from figure 6.16



will be transformed into a simply connected set, a set with one hole it will be transformed into a set with one hole in it, and so on.

Homotopy can be extended to functions (and thus images), although this is not as easy as for sets. It can be shown that an homotopic transformation $\Phi(f)$ of a function $f$ preserves the number and relative positions of the extrema of $f$ [6].

### 6.8.2 Homotopy Modification of Images

The image modification described in 6.8 can be achieved by changing the homotopy of the image under investigation. Suppose $I$ is the image to be segmented (and hence modified), $M$ is the binary marker image and $m$ is the maximal value of the pixels of $I$ [70]. The modified image $I'$ is defined as the dual reconstruction of $\min(I+1, (m+1)M)$ from $(m+1)M$ (recall Definition 5.5.5):

$$I' = \rho^*_{\min(I+1,(m+1)M)}((m+1)M) \tag{6.16}$$

The result of this process is that pixels located on markers are given value a value of 0 in $I$, and any non-marked catchment basins in $I$ are filled up. Thus, when the watershed transform is used

to segment the image, flooding will start from these new minima. The effectiveness of this image modification process is clearly demonstrated by Figures 6.20 and 6.21. Figure 6.20 shows an (arbitrary) set of markers, and Figure 6.21 shows the segmentation that results from modifying image I1 with these arbitrary markers. The image has been segmented into four regions, each of which is marked by one of the black squares in the marker image[2].

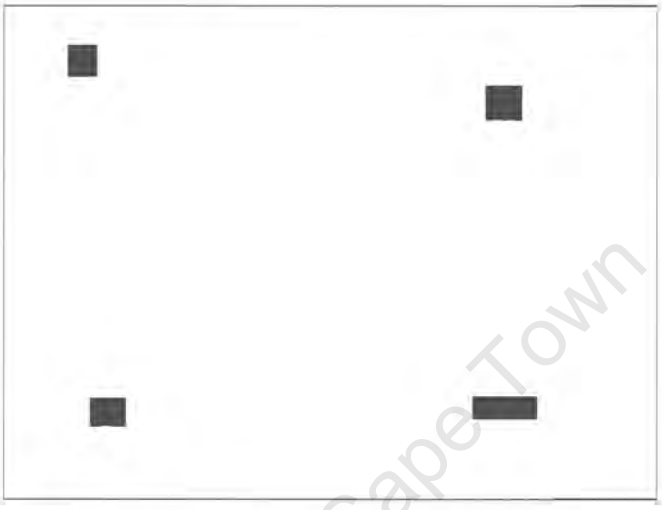


Figure 6.20: An arbitrary set of markers

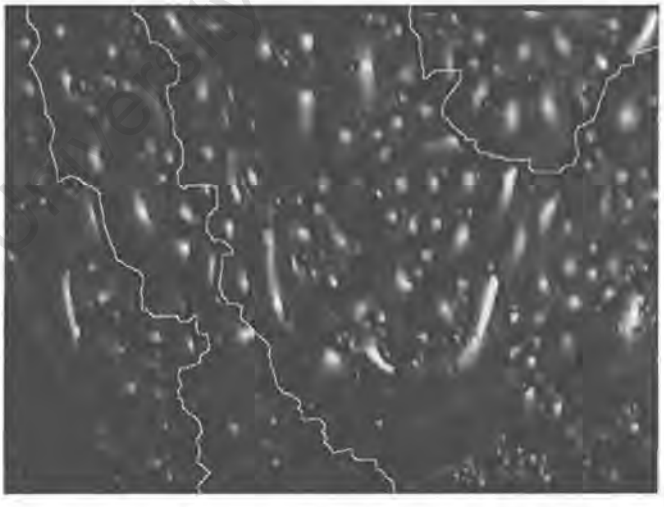


Figure 6.21: Segmentation of image I1 with arbitrary markers imposed as minima

[2]The border in figure 6.20 is **not** a marker; it simply serves to show the size of the marker image.

### 6.8.3 The Effect of the $h$ Parameter on Froth Image Segmentation

Figures 6.22 to 6.25 show the segmentation of image I1 using markers based on domes of various heights (in addition to the other preprocessing methods described previously).

Examining the images, it is clear that small values of $h$ (for example $h = 5$, Figure 6.22) enable small bubbles to be accurately detected, while large bubbles are oversegmented. Large values of $h$ (for example $h = 100$, figure 6.25), cause large bubbles to be properly segmented, while small bubbles are incorporated as part of larger bubbles. This is due to the fact that when the extracted domes (Figure 6.18) are thresholded to create the binary marker image, the domes from small bubbles merge together into a single marker.

The value of $h$ used for the preprocessing was set at 15 in this research, as a compromise between accurate segmentation of large and small bubbles. The best value of $h$ to use may well vary depending on the type of froth that is being analysed.

## 6.9 Complete Preprocessing Algorithm

The final preprocessing algorithm contains most of the processes described above. The algorithm operates in the following way on a froth image:

- Histogram Equalisation.
- 6 iterations of Anisotropic Diffusion.
- A further Histogram Equalisation.
- Dilation with a 5 x 5 structuring element.
- Marker extraction using domes of height 15.
- Inversion of the image.
- Image modification using previously generated markers.

The resultant image is then presented to the Watershed Transform to be segmented.

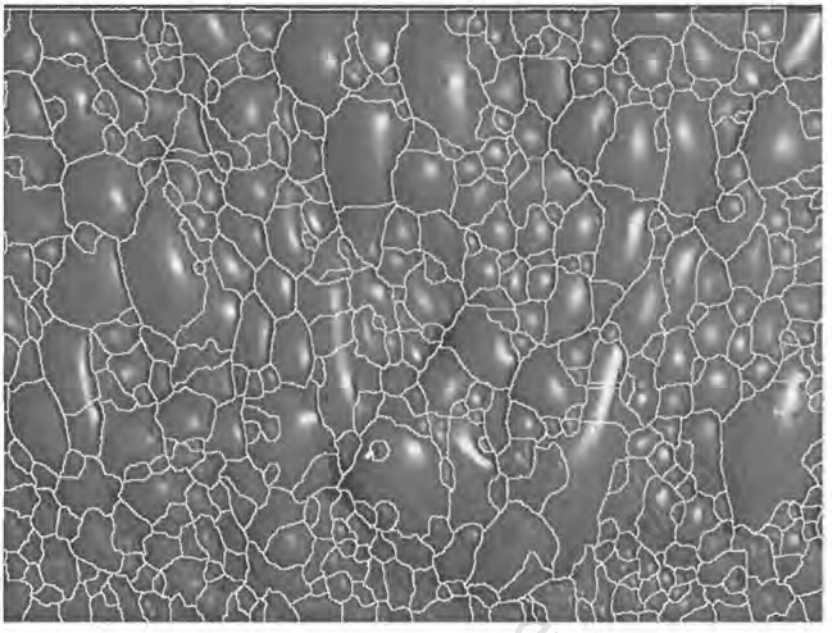


Figure 6.22: Segmentation of image I1 using domes of height 5

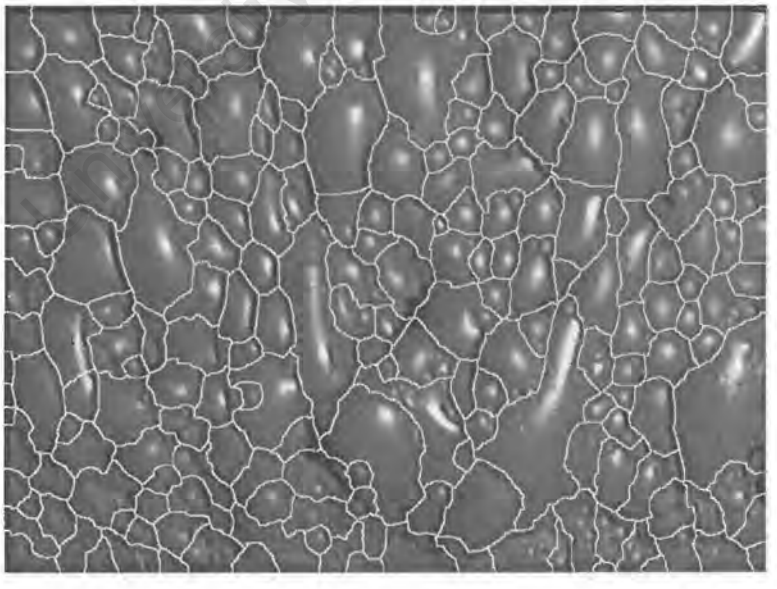


Figure 6.23: Segmentation of image I1 using domes of height 15

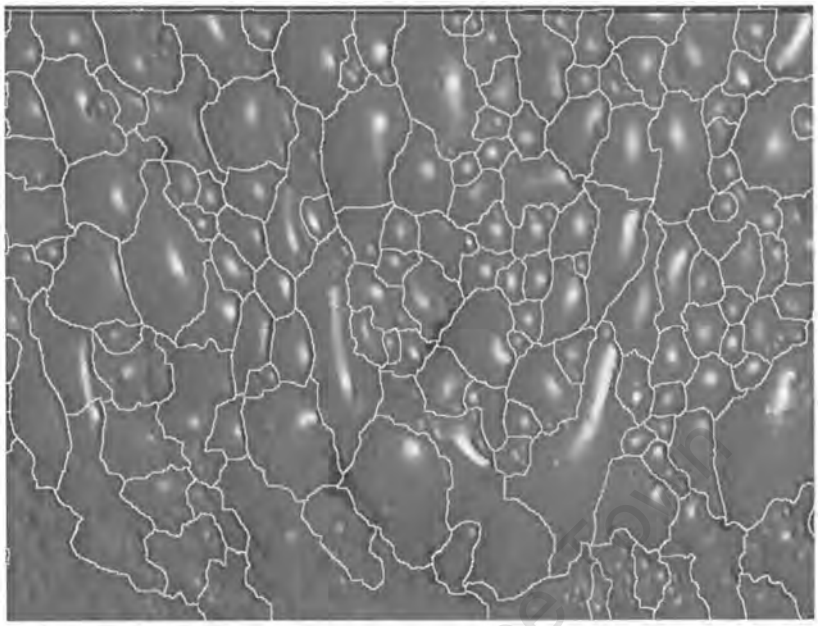


Figure 6.24: Segmentation of image I1 using domes of height 50

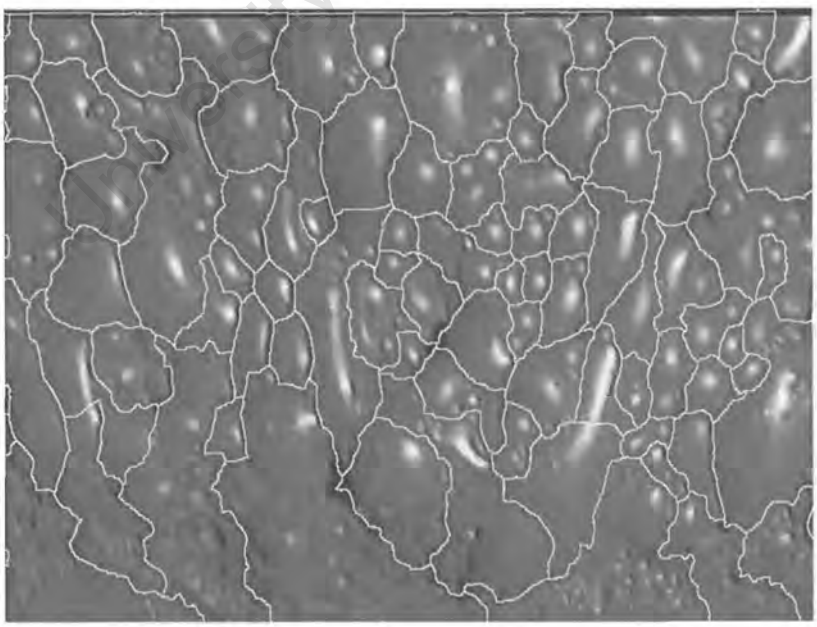


Figure 6.25: Segmentation of image I1 using domes of height 100

# Chapter 7

# Morphological Algorithm Descriptions and General Implementation Details



Much mention has been made up to this point of the use of the Fast Watershed Transform for segmenting flotation froth images. Chapter 5 presented the theoretical foundation of an immersion based watershed segmentation algorithm. In this chapter a fast algorithm developed by Vincent and Soille [71] for implementing the immersion-based Watershed Transform is presented.

The use of grayscale reconstruction as a preprocessing tool was presented in chapter 6. There exist several methods of computing reconstructions on grayscale images, with varying levels of performance. An efficient algorithm developed by Vincent [70] was used in this research, and is described in this chapter.

The final section of this chapter briefly discusses some of the issues pertaining to implementation of the image processing algorithms in this work.

## 7.1 Watershed Algorithms

The performance of watershed algorithms depends to a large extent on the design of the algorithm and the architectures of the computers on which they are implemented. This section briefly describes "classical" watershed algorithms with poor performance, and then presents an efficient watershed algorithm designed for implementation on standard desktop computer architectures.

### 7.1.1 "Classical" Watershed Algorithms

In their review of immersion based watershed algorithms, Vincent and Soille [71] cite several methods of determining watersheds based on iterations of thickenings and thinnings (which are morphological operators based on dilations and erosions, as defined in chapter 5) with special structuring elements until idempotence is reached. These algorithms are very inefficient because often many iterations (during which time every pixel in the image is examined) are required until convergence occurs. Acceptable performance of these algorithms can be obtained on specialised parallel computer architectures rather than conventional computers.

### 7.1.2 An Efficient Watershed Algorithm

It was mentioned above that "classical" watershed algorithms are required to examine every pixel in an image several times, yet often only a few pixels need to be modified per iteration. The obvious result of this inefficiency is poor algorithm performance. If the algorithm is able to directly access the pixels that require modification, the performance will be improved. The design of efficient morphological algorithms requires that, assuming the image pixels are stored in simple array, the following conditions be satisfied:

1. ***Random access*** to the pixels of an image.
2. ***Direct access to the neighbours of a pixel.*** The neighbourhood will vary according to the connectivity used, e.g. 4-,8- or hexagonal connectivity.

The algorithm proposed in [71] is based on the definition of watersheds as given in Chapter 5. Based on this definition it is necessary to consider successive thresholds of the image under analysis, and to compute the geodesic zones of influence of each threshold inside the next as quickly as possible. The algorithm can be easily decomposed into two basic steps. In the first step (the *sorting step*), the pixels of the image must be sorted into increasing grayscale order, in order to allow for direct access to pixels at a particular threshold value. The second step (the *flooding step*) involves fast computation of geodesic influence zones by performing breadth-first scanning of each grayscale threshold level. A *first-in-first-out* (FIFO) queue data structure is used to implement the pixel scanning.

### The Sorting Step

The method described in [71] for sorting the image pixels by increasing gray levels was introduced by E.J. Isaac and R.C Singleton in 1956. The sorting is performed by first computing the histogram of the image and then integrating the histogram to produce the cumulative frequency distribution of the gray levels in the image. The cumulative distribution is then used to populate two arrays with the x and y co-ordinates of the pixels in the image. This results in the pixel co-ordinates being sorted according to increasing grayscale values. The following C++ code fragment implements the sorting procedure:

```
hmax = 0;
hmin = 255;

for (y = 1 ; y < height-1 ; y++)
  for (x = 1 ; x < width-1 ; x++) {
    if (Imi(x,y) > hmax) hmax = Imi(x,y);
    if (Imi(x,y) < hmin) hmin = Imi(x,y);
    hist[Imi(x,y)]++;          // Create histogram
  }

cumulative[hmin] = 0;

for (x = hmin + 1 ; x <= hmax+1 ; x++)     // Find cumulative
  cumulative[x] = cumulative[x-1] + hist[x-1]; // distribution

for (y = 1 ; y < height-1 ; y++)
  for(x = 1 ; x < width-1 ; x++) {
    xsort[cumulative[Imi(x,y)]] = x;
    ysort[cumulative[Imi(x,y)]] = y; //Sort pixels by increasing
    cumulative[Imi(x,y)]++;          //grayscale value
  }
```

This sorting technique is most efficient, as it requires $n$ examinations to create the histogram and $n$ examinations to sort the pixels, where $n$ is the number of pixels in the image. It also requires $h_{max} - h_{min} - 1$ additions to create the cumulative frequency distribution, where $h_{max}$ and $h_{min}$ are the largest and smallest grayscale values respectively in the image. The result of the sorting

is that pixels at a given grayscale level $h$ are directly accessible. This accessibility is crucial in the flooding step of the watershed algorithm.

### The Flooding Step

After the sorting of the pixels, the determination of the watersheds begins by progressively flooding the catchment basins of the image. If the flooding has been done up to a level $h$, then every catchment basin that has been discovered - that is catchment basins with corresponding minima that are at an altitude of $h$ or lower - should have a unique label. Because the pixels at altitude $h+1$ are directly accessible (due to the sorting step described above), they can be given a special label MASK. The pixels at altitude $h+1$ that already have a labelled pixel as a neighbour are placed in the pixel queue. Beginning from these enqueued pixels, it is possible to extend the labelled catchment basins inside the MASK pixels by computing the geodesic zones of influence (see section 5.4.2). Following this step, the only pixels that have not been reached are the **minima** at level $h+1$. Since these pixels have not been connected to any catchment basins and still carry the MASK label, a second image scanning is required to detect these pixels and label them as new catchment basins.

The first-in-first-out queue data structure used to perform the breadth-first search is simply an array of pointers to image pixels. There are three operations that can be performed on the queue:

- *fifo_add(p)*, which places a pointer to pixel $p$ in the queue.
- *fifo_first(p)*, which returns a pointer to the pixel at the head of the queue and removes the pointer from the queue.
- *fifo_empty()*, which returns *true* if the queue is empty, otherwise *false* is returned.

The use of static arrays for implementing the queue is recommended in [71] because the use of dynamic memory allocation to size the queue automatically tends to degrade the algorithm performance. When using the static implementation, a "circular" queue is an efficient method to use: The array used for queueing the pixels has two indices, namely *ptr_first* and *ptr_last*. When an item is enqueued it is stored at index *ptr_last*, and *ptr_last* is then incremented. When the end of the array is reached, *ptr_last* loops around to the front of the array. In a similar manner, *ptr_first* marks the first element in the queue, and is incremented when the element is dequeued. The *ptr_first* index also loops back to the start of the array when required.

### 7.1.3 The Complete Fast Watershed Algorithm

The complete watershed segmentation algorithm is given below. It is specified in a C-like pseudocode. Note that the algorithm is designed to tessellate the image into different catchment basins, and accordingly only pixels that are "half way between" two catchment basins are given the WSHED label to indicate that they are watershed lines.

*Algorithm:* Fast Watershed Transform:

```
#define MASK -2 /* Initially a threshold has this value */
#define WSHED 0 /* Value of watershed boundary pixels */
#define INIT -1 /* Initial value of output image */
```

- Input: $im_i$, a grayscale image
  Output: $im_o$, an image of labelled watersheds

- Initialisations:
  - Each pixel of $im_o$ is assigned the value INIT
  - *current_label*: integer variable
  - *current_label* $\leftarrow 0$
  - *current_dist*: integer variable

- Sort the pixels of $im_i$ by ascending graylevels, as described previously.
  Let $h_{min}$ and $h_{max}$ be the smallest and largest grayscale values respectively.

- For $h \leftarrow h_{min}$ to $h_{max}$ {
  /* Determine geodesic SKIZ of level $h-1$ inside level $h$ */
  For every pixel $p$ such that $im_i(p) = h$ {
  /* These pixels are accessed through the sorted array */
  $im_o(p) \leftarrow$ MASK
  if there exists $p' \in N_G(p)$ such that $im_o(p') \geq$ WSHED {
  $im_d(p) \leftarrow 1$; *fifo_add(p)*
  }
  }
  *current_dist* $\leftarrow 1$; *fifo_add(ficticious_pixel)*;
  repeat indefinitely {
  $p \leftarrow$ *fifo_first()*

```
    if p = ficticious_pixel {
        if fifo_empty() = true then BREAK;
        else {
            fifo_add(ficticious_pixel);
            current_dist ← current_dist +1;
            p ← fifo_first();
        }
    }
    For every pixel p' ∈ N_G(p) {
        if im_d(p') < current_dist and im_o(p') ≥ WSHED {
                /* i.e. p' belongs to an already labelled basin or to watersheds */
          if im_o(p') > 0 {
            if im_o(p) = MASK or im_o(p) = WSHED then
              im_o(p) ← im_o(p');
            else if im_o(p) ≠ im_o(p') then
              im_o(p) ← WSHED;
          }
          else if im_o p = MASK then im_o(p) ← WSHED
        }   else if im_o(p') = MASK and im_d(p') = 0 {
          im_d(p') ← current_dist + 1; fifo_add(p');
        }
    }
}
        /* Now check if new minima have been discovered */
For every pixel p such that im_i(p) = h {
    im_d(p) ← 0        /* Reset the distance associated with p to zero */
    if im_o(p) = MASK {
      current_label ← current_label + 1;
      fifo_add(p); im_o(p) ← current_label;
      while fifo_empty() = false {
            p' ← fifo_first();
            For every pixel p'' ∈ N_G(p') {
                if im_o(p'') = MASK {
                  fifo_add(p'');
```

```
                    im_o(p'') ← current_label;
                }
            }
        }
    }
}
}
```

At this point, the catchment basins can be separated by giving the value WSHED to labeled pixels that have a pixel of a *smaller* label in their neighbourhood. Separation of the catchment basins enables the watershed boundaries to be extracted and overlaid on the original image to produce visually appealing segmented images (as are shown in Chapter 6). Additionally, since each region has a distinct label, it is easy to scan the segmented image and generate size and shape information for the detected bubbles.

Vincent and Soille claim that the above algorithm runs in linear time with respect the the number of pixels in the image. Actual execution time vary somewhat, according to the clock speed of the computer on which the algorithm is run, and on the type of structures in the image that is being segemented. As an indication of performance, the time taken to segement a 640 x 480 pixel image is approximately 3.5 seconds on a 266 MHz Pentium II based Personal Computer.



## 7.2 Grayscale Reconstruction Algorithms

Grayscale reconstruction is an important part of the preprocessing regime for froth images. It is thus important that the algorithm used to implement the reconstruction is as efficient as possible. There exist several implementations of the reconstruction algorithm, each with different strengths and weaknesses. They will briefly be described below, as a precursor to the algorithm used in this work. All of these algorithms appear in [70].

### 7.2.1 Parallel Reconstruction Algorithms

Recalling Definition 5.5.4 in Chapter 6, grayscale reconstruction can be directly implemented by iterating elementary dilations of each pixel in an image, followed by a pointwise minimisation, until stability is reached (that is, no more pixel values are modified). This is a *parallel* implementation of the reconstruction algorithm, and is extremely efficient on large parallel computer architectures, as the scanning of the image pixels is completely arbitrary. On conventional computers, however, sometimes several hundred iterations of the algorithm are required before stability is obtained, which leads to lengthy execution times.

### 7.2.2 Sequential Reconstruction Algorithms

Sequential algorithms are an attempt to reduce the number of complete scannings of an image, and hence increase the speed of execution. These algorithms rely on two principles:

- The image pixels are scanned in a predefined order.
- The new value of an image pixel is written back into the same image, so that the new value is taken into account when the new values of other pixels are computed.

The order in which the pixels is scanned is essential for sequential algorithms to perform properly. Often, the scanning used is raster (left to right and top to bottom) followed by anti-raster (right to left and bottom to top) scanning, with the dilation and pointwise minimum being performed on each pixel. This method reduces the number of image scannings required to a more manageable figure than for parallel algorithms (typically a dozen iterations are required).

A shortcoming of many sequential algorithms is that they do not perform well on images containing certain types of structures (crest lines in particular) [69]. In cases like this, many image scannings are sometimes required, with only a few pixels being modified in each iteration.

### 7.2.3 Queue-based Reconstruction Algorithms

The performance of reconstruction algorithms can be improved further by applying methods used in the implementation of the fast watershed transform (as described in Section 7.1.2) by

only considering pixels whose value may be modified. An initial scanning detects the pixels that will be modified first, and using breadth-first scannings, changes are propagated from these pixels (using a FIFO queue to perform the scanning).

In order for queue-based algorithms to function correctly on grayscale images, the regional maxima of the image need to be determined, as the propagation step begins from these maxima. The initial scannings required to determine the maxima tend to decrease the performance of the algorithm. Additionaly, certain image regions may be scanned more than once during the breadth-first step.

### 7.2.4 A Hybrid Grayscale Reconstruction Algorithm

The performance penalties incurred by the sequential and queue-based algorithms can to a large extent be nullified by combining the two approaches together into a hybrid algorithm. Initialy, the sequential raster and anti-raster scans are performed. During the second (anti-raster) scan, all pixels $p$ having a value that could cause propagation during the next raster scan, i.e. such that

$$\exists q \in N_G^-(p),\ J(q) < J(p) \text{ and } J(q) < I(q)$$

are put into the queue. Following the initial scanning, the algorithm proceeds with the breadth-first propagation as in the normal queue-based implementation. However, according to Vincent [70], the number of pixels that are considered during this step is smaller than for the normal queue-based method. This hybrid algorithm is presented below in a pseudocode similar to that used in Section 7.1.3 for the fast watershed algorithm.

*Algorithm*: Fast Hybrid Grayscale Reconstruction:

- $I$: mask image (binary or grayscale)
- $J$: marker image defined on domain $D_I$, $J \leq I$. Reconstruction is determined directly in $J$
- Scan $D_I$ in raster order:
  Let $p$ be the current pixel;
  $J(p) \leftarrow (\max\{J(q), q \in N_G^+(p) \cup \{p\}\}) \wedge I(p)$

- Scan $D_I$ in anti-raster order:

  Let $p$ be the current pixel;

  $J(p) \leftarrow (\max\{J(q), q \in N_G^-(p) \cup \{p\}\}) \wedge I(p)$

  If there exists $q \in N_G^-(p)$ such that $J(q) < J(p)$ and $J(q) < I(q)$

  *fifo_add(p)*

- Propagation step: While *fifo_empty()* = *false*

  $p \leftarrow$ *fifo_first()*

  For every pixel $q \in N_G(p)$:

  If $J(q) < J(p)$ and $I(q) \neq J(q)$

  $J(q) = \min\{J(p), I(q)\}$

  *fifo_add(p)*

## 7.3 Implementation of the Segmentation Algorithm

In Chapter 6 the froth image preprocessing algorithm was defined, and Sections 7.1.3 and 7.2.4 of this chapter have presented pseudocode implementations of the Fast Watershed Transform and Grayscale Reconstruction. This section discusses the details of implementing the complete segmentation algorithm as the basis for a froth image analysis system.

### 7.3.1 Software Development Environment

At the very beginning of this research it was decided to use the C++ programming language for the software implementation. C++ was chosen rather than other tools such as **MATLAB** and the Interactive Data Language (**IDL**) because a deliverable stand-alone image processing system is a key aim of the research. MATLAB and IDL are better suited to prototyping work in the laboratory environment.

C++ is an Object-Oriented programming language. The fundamental concepts of object-oriented languages are *objects* and *classes* [77]. A class is a description of a collection of objects, and an object is an instantiation of a class or a member of that class. A class is a unit of modularisation

and an object is a unit of programming.

Classes provide the framework into which the elements of an abstraction or model are placed. A class contains *data members* which contain the attributes of the class, and *member functions* which define the relationships between classes and the functionality of a class. For example a class **person** could have data members ***height***, ***weight***, ***hair_colour***, ***eye_colour*** etc. and member functions ***get_height()***, ***get_weight()***, ***walk()***, ***eat()*** etc.

In general, classes hide the details of how an abstract data type is implemented, and only the interface between the class and other classes is made public [77]. Thus the class is viewable as an abstraction, or representation of a part of the "real world". Consensus is that, once learned, object oriented programming is easier and safer than classical functional programming paradigms.

Microsoft Corporation's Visual C++ 5 environment running under Microsoft Windows NT Workstation 4.0 was used to develop the software for this research. Implementation on a Linux based system was investigated, but lack of support for the framegrabber used for image acquisition led to the choice of the NT platform.

The Fast Watershed and Grayscale Reconstruction algorithms were relatively easy to convert to C++, due to the nature of the pseudocode algorithms provided in [70] and [71]. Pseudocode equivalents for the preprocessing methods described in chapter 6 and shape analysis routines (Chapter 8) are not given here, as their direct conversion to C++ is uncomplicated.



### 7.3.2 Software Description

Since the software was implemented in C++, several classes were developed as the basis of the system. The most important classes are as follows:

- The **Image** class. This class abstracts an image, and handles duties such as reading and writing to disk, allocation of memory for images and providing useful overloaded operators.
- The **Toolbox** class. This class contains all the image processing routines as used in the segmentation algorithm.
- The **Fifo** class. This class implements the FIFO queue data structure as used by the Fast Watershed Transform and Grayscale Reconstruction algorithms.

- The **Blob** class. In this context, a Blob corresponds to a detected bubble. A Blob object contains size and shape information for the bubble it represents.
- The **Data** class. Data objects consist of an array of Blob objects. The Data class keeps track of all the identified bubbles in a segmented image and provides statistics about the whole image as opposed to the individual bubbles.

A complete discussion of the software development issues related to this research will not be given here. Whilst being an interesting and complex subject, it is not relevant to this dissertation.

# Chapter 8

# Segmented Froth Image Descriptors

Once a froth image has been segmented, it is necessary to extract descriptors from the segmented image, in order for the froth to be characterised. As was shown in Chapter 2, the size and shape of the bubbles in the froth can be indicative of process performance, and it is the extraction and processing of this information by automatic means that forms the basis of all froth vision systems (recall the discussion in Chapter 3).



## 8.1 Extraction of Size and Shape information

The result of applying the Watershed Transform (as described in Chapter 7) to a froth image is an image composed of a series of regions or *blobs*. Each of these blobs corresponds to a detected bubble, with the blobs being separated by the watershed lines (recall the definition in Chapter 5). A useful result of the watershed segmentation is that each detected blob is uniquely labelled, which greatly simplifies the characterisation of the blobs. A list of blobs can be constructed by simply scanning through the segmented image and searching for pixels of a particular value, as all pixels belonging to the same blob will carry the same value. Figure 8.1 shows the detected blobs for the image I1.

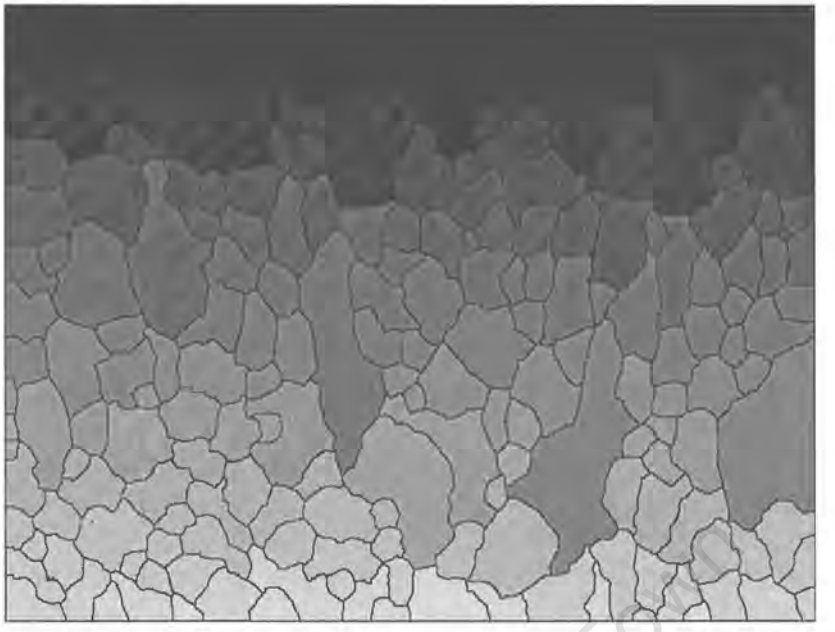


Figure 8.1: Detected blobs for image I1

## 8.2 Basic Size and Shape Measurements

By analysing the detected blobs from the segmented images, it is possible to calculate several simple parameters that describe the extent (size) of a bubble, and also to obtain some measures of the bubble shapes. The derivation of these basic parameters is presented in this section.

### 8.2.1 Bubble Area

Since the watershed lines in the segmented image should correspond to the bubble boundaries, a blob corresponds to the area or *footprint* of a detected bubble. Thus in order to determine the area of a bubble with label $N$, all the pixels of value $N$ in the segmented image simply need to be counted, i.e. to calculate the area $A$ of a bubble with label $N$

$$A_N = \sum_x \sum_y p(x, y), \text{ where } p(x, y) = N \tag{8.1}$$

and $(p(x, y)$ is an image pixel.

Using the bubble size information it is possible to calculate a bubble size distribution for the froth

image under analysis. The importance of bubble size distributions in the identification of froths was noted in chapter 2.

### 8.2.2 Bubble Perimeter

The perimeter $P$ of a bubble is the length of the bubble boundary [23]. It is determined by counting the number of pixels in a blob that have a watershed pixel as a neighbour.

### 8.2.3 Bubble Circularity

In a vision-based froth analysis system it is advantageous to extract descriptive real information rather than abstract shape information. Abstract shape descriptors such as Fourier descriptors may be useful but are difficult to relate to the appearance of the froth as observed by a human operator, whereas a "real" measure such as circularity is much easier to visualise.

The circularity, $C$, of a blob, is defined as [40]

$$C = \frac{P^2}{4\pi A} \tag{8.2}$$

where $P$ is the bubble perimeter and $A$ is the bubble area. For a perfect circle, $C$ is unity (although this is unrealisable when working with discrete shapes). $C$ increases as shapes become less circular.



### 8.2.4 Bubble Ellipticity

The ellipticity or eccentricity of a bubble can give a useful measure of the elongation of the bubble. Ellipticity is a ratio of the principal axes of an equivalent ellipse fitted to a shape and can be calculated from the shape's *moments* [40].

The two-dimensional $(p+q)$th order moments of a digital shape are defined as

$$m_{pq} = \sum_x \sum_y x^p y^q I(x,y) \qquad p,q = 0,1,2,\ldots \tag{8.3}$$

The centre of mass of the shape is then given by

$$\overline{x} = \frac{m_{10}}{m_{00}} \qquad \overline{y} = \frac{m_{01}}{m_{00}} \tag{8.4}$$

Note that $m_{00} = A$, the area of the shape.

If the co-ordinate system being used is shifted such that its origin coincides with $(\overline{x}, \overline{y})$, the result is a set of *central moments* that are translation invariant. Thus equation 8.3 becomes

$$\mu_{pq} = \sum_x \sum_y (x - \overline{x})^p (y - \overline{y})^q I(x, y) \tag{8.5}$$

Central moments are defined in terms of ordinary moments by means of

$$\mu_{pq} = \sum_{r=0}^{p} \sum_{s=0}^{q} C_r^p C_s^q (-\overline{x})^r (-\overline{y})^s m_{p-r,q-s} \tag{8.6}$$

where

$$C_r^p = \frac{p!}{r!(p-r)!} \tag{8.7}$$

Thus,

$$\mu_{00} = m_{00} = \mu \tag{8.8}$$
$$\mu_{10} = \mu_{01} = 0 \tag{8.9}$$
$$\mu_{20} = m_{20} - \mu\overline{x}^2 \tag{8.10}$$
$$\mu_{11} = m_{11} - \mu\overline{xy} \tag{8.11}$$
$$\mu_{02} = m_{02} - \mu\overline{y}^2 \tag{8.12}$$
$$\mu_{30} = m_{30} - 3m_{20}\overline{x} + 2\mu\overline{x}^3 \tag{8.13}$$
$$\mu_{21} = m_{21} - m_{20}\overline{y} - 2m_{11}\overline{x} + 2\mu\overline{x}^2\overline{y} \tag{8.14}$$
$$\mu_{12} = m_{12} - m_{02}\overline{x} - 2m_{11}\overline{y} + 2\mu\overline{x}\,\overline{y}^2 \tag{8.15}$$
$$\mu_{03} = m_{03} - 3m_{02}\overline{y} + 2\mu\overline{y}^3 \tag{8.16}$$

The principal axes $I_a$ (major) and $I_b$ (minor) of the equivalent ellipse for a bubble can be defined

using the above moments and are given as follows:

$$I_a = \frac{1}{2}(\mu_{20} + \mu_{02}) + \sqrt{\mu_{11}^2 + \left(\frac{\mu_{20} - \mu_{02}}{2}\right)^2}$$
$$I_b = \frac{1}{2}(\mu_{20} - \mu_{02}) + \sqrt{\mu_{11}^2 + \left(\frac{\mu_{20} - \mu_{02}}{2}\right)^2} \tag{8.17}$$

The ellipticity $E$ of the bubble is then given by [65]

$$E = \sqrt{\frac{I_a}{I_b}} \tag{8.18}$$

## 8.3 More Advanced Measurements

Apart from the basic measurements described in the previous section, other more descriptive geometric parameters can be calculated. These, and other non-geometric parameters such as froth colour and speed, are presented in this section.

### 8.3.1 Specific Surface Area

The area, perimeter, circularity and ellipticity measurements are the primary descriptors used to characterise flotation froths. However, the collaborative work being done at the University of Manchester Institute of Science and Technology (UMIST) on froth modelling and flotation control strategies requires that more advanced descriptors be extracted from the segmented froth images.

The UMIST froth model is based on an equation for the flux of bubble surface flowing over the weir of a flotation cell:

$$\Psi_a = [\kappa \times \zeta \times v_f \times h \times w] \times S_a \tag{8.19}$$

A detailed discussion of this equation is far beyond the scope of this thesis; the term of interest here is $S_a$, the *specific surface area* of the bubbles. This is in effect the useful surface of the bubbles available to transport solids, and can be estimated by segmentation of images from froths overflowing a weir (currently this is of most interest in batch flotation).

The specific surface area $S_a$ of a froth is given by [67]

$$S_a = \sum_{i=1}^{j} \frac{SA_i^2}{V_i SA_i} \tag{8.20}$$

where $SA_i$ and $V_i$ are the surface area and volume respectively of bubble $i$.

Bubble surface area and volume are calculated by describing each detected bubble as an ellipse and rotating the ellipses about the major axis to produce ellipsoids [13]. If the ellipsoids have major and minor axes of $2a$ and $2b$ respectively, then

$$V = \frac{4\pi ab^2}{3} \tag{8.21}$$

and

$$SA = \frac{2\pi b}{k}\left[\arcsin(ka) + \frac{\sin(2\arcsin(ka))}{2}\right] \tag{8.22}$$

where $k^2 = (a^2 - b^2)/a^4$ when $a \neq b$. If $a = b$ then spherical geometry, where

$$SA = 4\pi a^2, \tag{8.23}$$

is used.

### 8.3.2 Froth Density

Froth density is a measure of the number of bubbles per unit area in the field of view of the image analysis system. It provides another indirect measurement of the average bubble size, and is given by the formula

$$D = \frac{N}{w \cdot h} \tag{8.24}$$

where $N$ is the number of bubbles detected and $w$ and $h$ are the image height and width respectively. Multiplication by an appropriate scaling factor produces a result with units of $bubbles/m^2$.

### 8.3.3 Froth Colour

In certain types of flotation (most notably base metal flotation), the colour of the froth is used as a process indicator. The value of colour information from platinum group metal flotation is uncertain, however because it is an easy parameter to measure, it was implemented in the image analysis system.

The use of colour video cameras allows the froth images to be acquired in full (24-bit) colour, with each image being composed of three colour bands, namely red, green, and blue. A measure of the colour is taken by computing the average of the pixel values in each of the three bands. This in three colour measures — average red, average green and average blue values.

The morphological operations used to segment the froth images are only applicable to grayscale images. Therefore, the colour images acquired by the system were converted to grayscale images using the conversion [55]

$$P(x, y) = 0.3R(x, y) + 0.6G(x, y) + 0.1B(x, y) \tag{8.25}$$

where $P(x, y)$ is a new grayscale pixel and $R(x, y)$, $G(x, y)$ and $B(x, y)$ are the equivalent pixel values in the three colour bands.

Once the image has been converted to grayscale, it is possible to compute an average grayscale value in a manner identical to the computation of the average colour values



## 8.4 Dynamic Froth Information

Since flotation is a dynamic process, the froth is continually moving as the mineral-laden bubbles report to the concentrate launder. Measuring the speed of the froth indicates how quickly the froth is overflowing the cell, or how hard the cell is "pulling". Thus a method of measuring the speed is an important part of a froth image analysis system. Other dynamic parameters that are of interest include bubble breakage, bursting and coalescence rates. Methods for on-line measurement of these dynamic parameters are the subject of separate research within the broader project context, and were not investigated during this research. However, some dynamic information in the form of a froth velocity measurement was required in the prototype analysis system.

### 8.4.1 Velocity Measurement Techniques

Several methods have been proposed for estimating the velocity of flotation froths, including Optical Flow, the generalised Hough Transform, Block Motion Estimation and mobility estimation from textural analysis [52, 19, 20, 50, 2]. Block Motion Estimation appears to give the most promising results, but is computationally very expensive. However, the rapid advances in computing power will almost certainly reduce the computation times. It is envisaged that the work being done by Francis [19, 20] as part of the Amplats project work will be integrated into the analysis system described in this dissertation. As an interim solution, the Pixel Tracing algorithm developed by Nguyen [52] was used for froth velocity measurements in this research.

### 8.4.2 The Pixel Tracing Algorithm

Pixel tracing can be thought of as a block matching algorithm using a single "superblock" of large size [21]. It is based on the assumption that the objects being matched are not distorted during the measurement. This is generally, though by no means always, true for flotation froths.

Consider an object in two consecutive image frames. In the first frame, the object is at a position $(x(0), y(0))$ and has an intensity $I(0)$. To detect the object in the second frame, the frame is scanned *starting at the object's initial position* $(x(0), y(0))$ until the intensity value $I(1)$, where $I(1) = I(0)$ is found. The new co-ordinates $(x(1), y(1))$ give the position of the object in the second frame. Figure 8.2 illustrates this simple example.

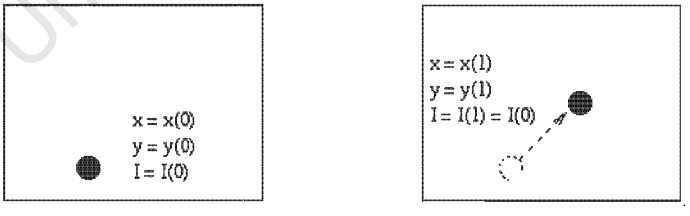



Figure 8.2: The principle of Pixel Tracing

In practice, the "object" is a matrix of pixels taken from the centre of the first frame. This matrix is moved, in the eight cardinal compass directions, a distance from 1 to $n$ ($n$ is typically 20 to 30) pixels over the second frame. The minimum difference in sum of squares between the original

matrix and the image pixels at each of the new positions indicates the best match of the "object". If the time between frames is known then the velocity can be determined, since the position of the best match $p$, where $0 \leq p \leq n$, is known.

The main advantage of the Pixel Tracing algorithm is that it is simple to implement and computationally inexpensive. However it can only provide an estimate of the average velocity of a froth, and cannot detect advanced dynamic features such as bubble breakage and coalescence.

## 8.5 Summary of Measured Parameters

The tables below summarise the information that is extracted from a segmented froth image. Any or all of the measurements can potentially be used to characterise a flotation froth.

Table 8.1: Table of measurements related to bubble size

| |
|---|
| Bubble Area (footprint) |
| Bubble Perimeter |
| Bubble equivalent Volume |
| Bubble equivalent Surface Area |
| Froth Specific Surface Area |

Table 8.2: Table of measurements related to bubble shape

| |
|---|
| Bubble Ellipticity / Eccentricity |
| Bubble Circularity |

Table 8.3: Table of other measurements

| |
|---|
| Froth Speed |
| Froth Direction |
| Froth Average Red Value |
| Froth Average Green Value |
| Froth Average Blue Value |
| Froth Average Gray Value |

# Chapter 9

# Initial Algorithm Application, System Development and Plant Trials



In order for an image analysis system to be useful, it must generate descriptive and meaningful data when applied to the task required. In the case of this research, useful descriptors are required to relate froth morphology to process conditions. In the course of the research analysis was performed on both batch and plant scale flotation froths, with the studies relating the froth appearance to process performance being undertaken as a research project in the Department of Chemical Engineering at UCT.

## 9.1 Batch Flotation Analysis

Early in the research period, a comprehensive study was made of the relationship between froth appearance and variation of process parameters for flotation performed in a 3 litre batch flotation cell. The parameters varied were [63]:

- Depressant type and dosage
- Frother dosage
- Collector dosage
- Froth height

- Airflow rate.

The froths were videotaped and subjected to off-line image analysis to obtain the froth descriptors (area, perimeter, circularity and ellipticity). Figure 9.1 shows the flotation cell and Figure 9.2 the video camera support assembly.

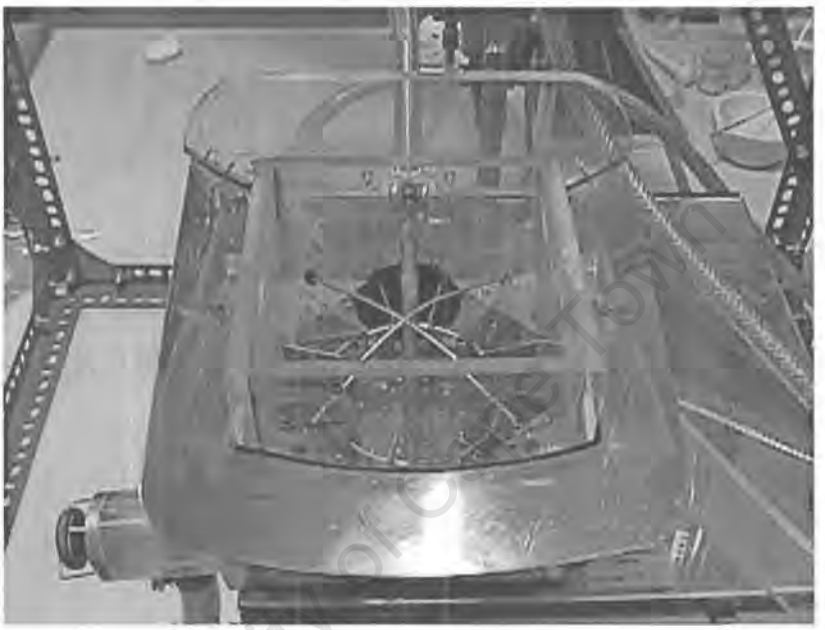


Figure 9.1: The batch flotation cell used for froth characterisation



Figures 9.3 through 9.6 show the segmentation of two different images from the batch flotation tests. The segmentation in Figure 9.4 seems poor compared to Figure 9.6, which is attributed to the fact that:

1. The segmentation algorithm (including preprocessing) was still under development.
2. There are far fewer bubbles in Figure 9.3 than in Figure 9.5 This makes the segmentation much more sensitive to noise and inconsistencies on the bubble surfaces. In addition, the bubbles in figure are transparent, which renders the bubbles below the surface visible.

Despite the perceived segmentation inaccuracy, useful relationships were obtained from the tests, namely [64]:

Figure 9.2: Camera support assembly for batch flotation tests



- Lower depressant doses resulted in coarser bubble size distributions.
- Increased bubble stability at low depressant levels.
- Increased bubble stability at high frother levels. Most differences were observed in metallurgical rather than image data.
- Smaller bubble sizes at higher airflow rates resulting in higher mass recoveries.

During the course of the project research it became apparent that the batch scale flotation testwork was somewhat limiting, mainly due to the poor statistics obtained from having very few bubbles avaliable for image analysis [62]. It was felt that further testwork should be performed on a continuous flotation system. After considering various options, including use of pilot scale plants, it was decided to perform the work on a Merensky Primary Rougher Flotation Bank at Amplats' Amandelbult Concentrator. This was considered to be a good opportunity to test the performance of a fully fledged froth image analysis system in a plant environment, rather than

Figure 9.3: A "poor" batch flotation froth image

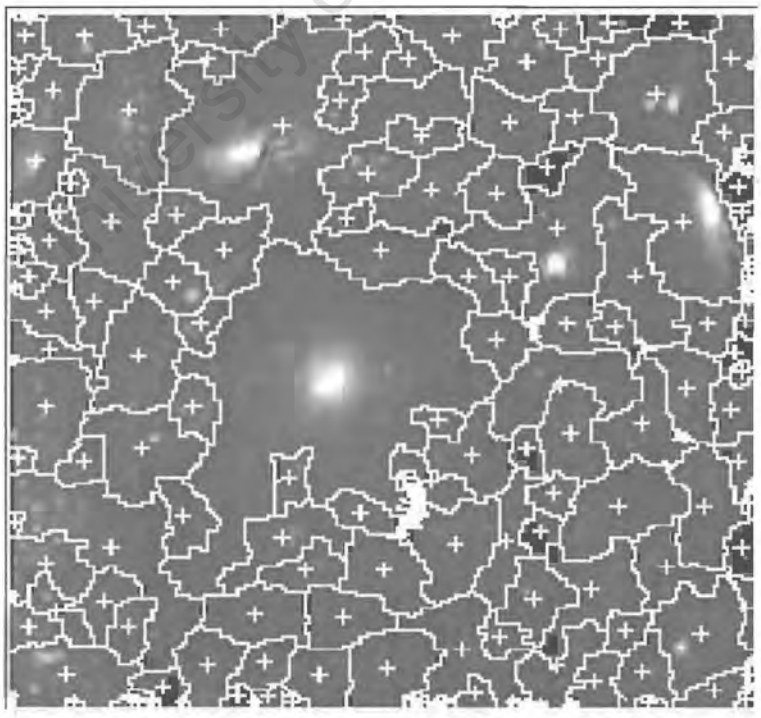


Figure 9.4: The result of segmenting the "poor" batch froth image

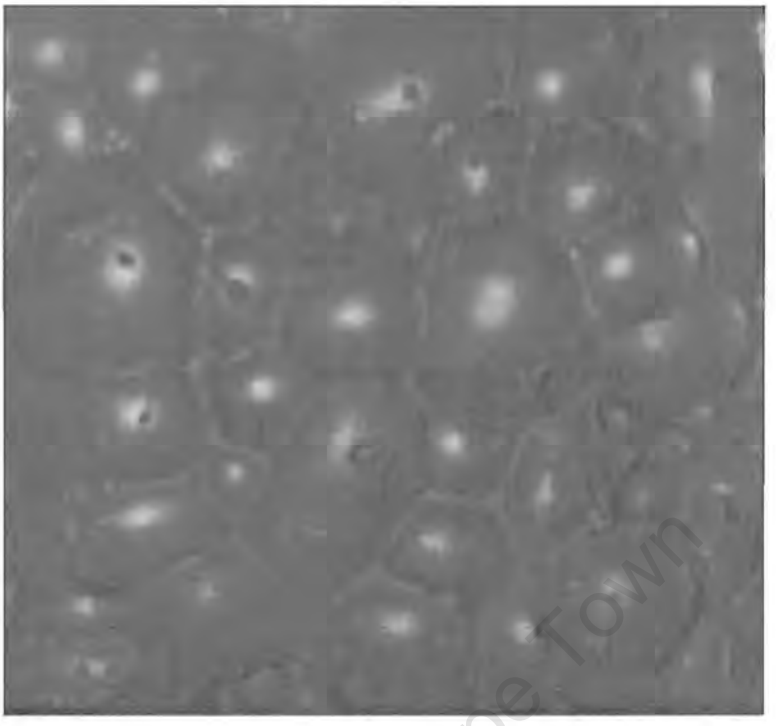


Figure 9.5: A "good" batch flotation froth image

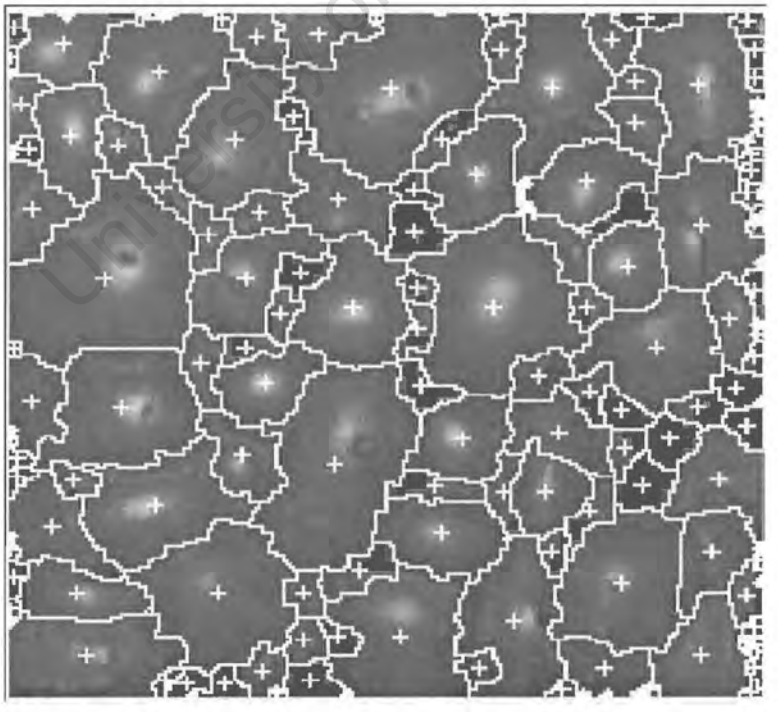


Figure 9.6: The result of segmenting the "good" batch froth image

relying on off-line image processing as was done for the batch flotation tests. To this end, the on-line analysis system as described below was developed.

## 9.2 The Prototype On-Line Froth Image Analysis System

### 9.2.1 Hardware

The three main hardware components of the froth image analysis system are the camera assemblies, the transmission medium and the computer equipment. Each of these will be discussed in detail below

#### Camera Assemblies

An obvious requirement of a machine vision system is a good quality camera. The cameras used in this project were miniature Watec 202-B colour CCD devices, running on 12 volts DC and providing a television quality PAL video signal.

The lenses used with the cameras were Vicon 12.5-75 mm zoom lenses with auto iris function. The auto iris automatically controls the lens aperture to compensate for any variations in light intensity. The zoom and focus were remotely controllable, which meant that adjustments could easily be made when the cameras were in position on the plant.

A concentrator plant is an unsuitable environment for delicate electronic equipment, due to the presence of high levels of dust, moisture and corrosive chemicals. In order to protect the cameras and lenses from the environment, they were placed in water- and dust-proof steel casings. These casings also held the camera power supplies. Grommets at the rear of the casing provided access for 220 volt mains, video signal and lens control cables (Figure 9.7).

500W outdoor halogen floodlights were used to provide illumination, as they are robust, cheap and waterproof.

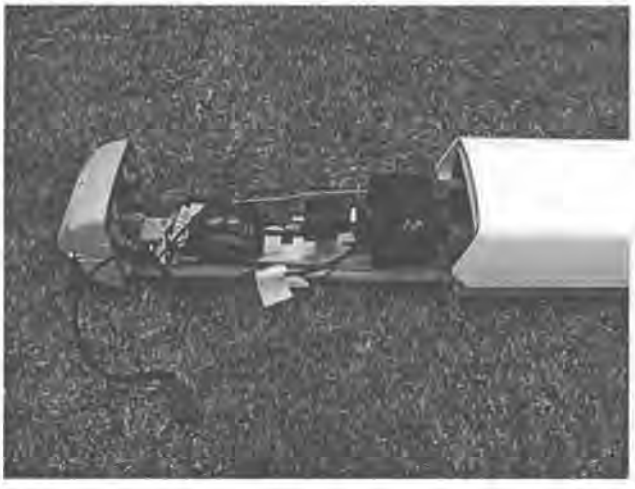


Figure 9.7: The complete camera subassembly



### Transmission medium

The video signal was carried from the cameras to the analysis computers on industrial specification RG-59 B/U coaxial cable. Due to the large number of electric motors present in a concentrator plant (in flotation cells, mills and pumps), a concentrator plant is an electrically noisy environment, and the coaxial cable used is specified for use in such an environment.

### Computer Equipment

The computers used for the prototype system were two Intel Pentium II based personal computers running at a clock speed of 400 megahertz. Each machine was supplied with 256 megabytes of Random Access Memory and an 18 gigabyte Ultra-Wide SCSI disk drive. One computer was used for image acquisition and analysis and was equipped with a Matrox Meteor II framegrabber. This framegrabber allows connection of up to four video input sources, with images from each source being grabbed consecutively. The other computer was used for storing images and the results of the image analysis. Both computers contained 100 Megabit/second Fast Ethernet network cards and were linked by means of a single Category 5 twisted pair network cable.

The hardware components of the system can be divided into a number of different subsystems, which simplifies the interpretation somewhat, as is shown in Figure 9.8.

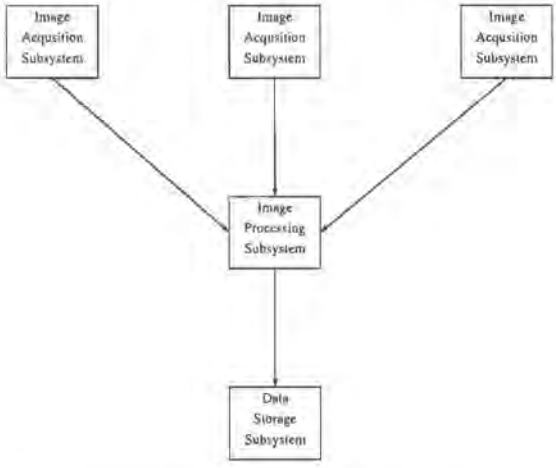



Figure 9.8: Block diagram of subsystems

### 9.2.2 Software

The image analysis methodology used has been described in detail in Chapters 6 and 7. For plant testing purposes, the algorithms were integrated into a more user-friendly software program known as AutoFroth. The program runs under the Microsoft Windows NT Workstation 4.0 operating system.

The user interface of the AutoFroth package (Figure 9.9 shows a screen shot of the main interface) was developed using Microsoft Visual Basic version 6. This was chosen for its ease of use and the tools it provides for rapid application development. The image processing routines were left as C++ code and compiled into a Dynamic Link Library (DLL) that was called when required by the user interface program. Storing the image processing routines as a separate library appeared to introduce no significant performance penalties. The parameters extracted from the froth images were written to a Microsoft SQL Server database which resided on a separate computer. The logical connections between the AutoFroth software and the database were set up using the Windows NT ODBC (Open Database Connectivity) manager.

#### The AutoFroth software - a detailed description

Figure 9.9 shows the main screen of the AutoFroth program. It contains the latest images from the flotation cells being analysed, key numbers from the results of the last image analysis, namely

average bubble size, eccentricity, froth density, average grayscale value of the froth image, froth speed and direction, and the froth specific area. The graphs on the right hand side of the screen display three-point running averages of the parameters generated by the image analysis. Any parameter (as defined in Chapter 8) generated by the image analysis system can be displayed on any of the six graphs. A bubble size distribution is also displayed.

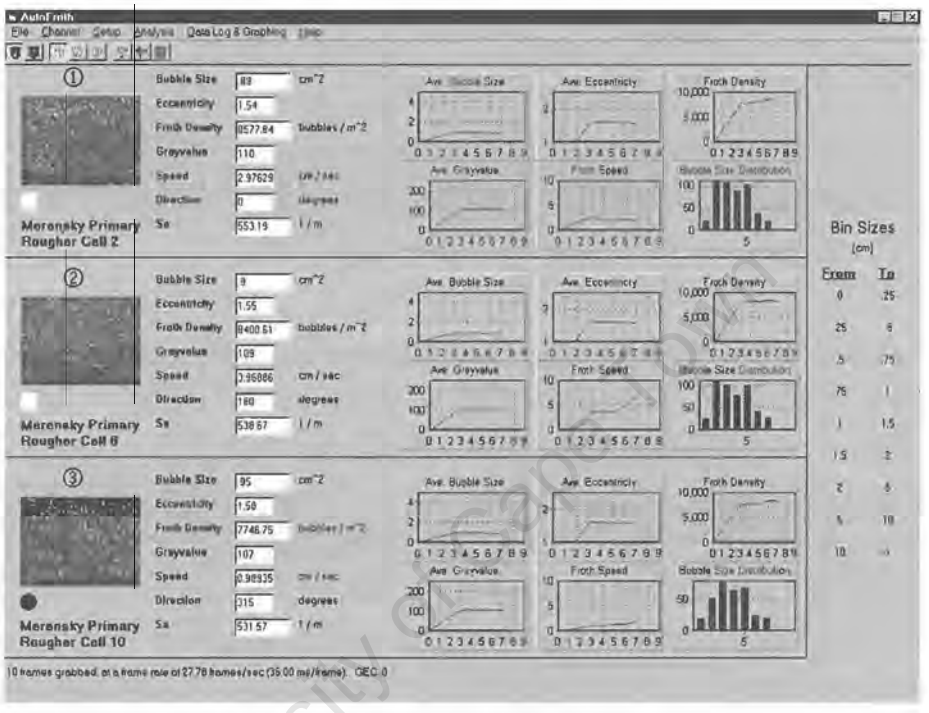



Figure 9.9: Main screen of the AutoFroth system

Before being used for on-line analysis, various parts of the system must be configured. The program allows the user to perform several tasks related to this configuration of the system, as described below.

The Channel menu option allows the user to select which of the three input channels should be made active. It should be noted that this is only used when the image analysis is not running, since the analysis routines automatically sample each input channel in turn. The Setup menu option allows for the modification of digitizer settings, spatial calibration of the image and image cropping for each camera.

Choosing the Digitizer option in the Setup menu allows the user to modify the brightness and contrast levels for each input channel, in order to obtain the best quality picture. A real-time grab from the camera selected is shown in the appropriate video display, and slider bars allow

for changing of the brightness and contrast settings. This mode also provides an opportunity to adjust the focus setting of the lens, if required.

Selection of the Spatial Calibration option allows a mapping from image pixels to a physical area to be defined. This means that measurements such as average bubble size can be expressed in a convenient, meaningful unit, such as $m^2$ or $cm^2$ instead of number of pixels. The calibration is performed by placing some device of known length in the field of view of the particular camera being setup. When the user clicks the "Calibrate" button, a full-size image from the camera is displayed. The user must then use the mouse to draw a line on the picture such that the line is equal to the length of the calibrating device shown in the picture. The user is then prompted to enter the equivalent length of the line in real units, and the mapping from pixels to square metres is calculated automatically.

The Crop option in the Setup menu is similar to the Spatial Calibration option, except it allows the user to select a particular region of the full image for analysis. This is useful for excluding unwanted sections of an image from the analysis. For example, if it is unavoidable that parts of the flotation cell mechanism are in the field of view of a camera, the image can be cropped so that an image containing only froth can be presented to the analysis routines.

The Data Log & Graphing menu allows the user to turn various data storage features on and off. Using this menu, the user is able to activate or deactivate the writing of the image analysis results to the database, the saving of the analysed images to disk and the plotting of data on the system graphs. It is worth noting that the saving of images to disk should be used with caution, since the images are fairly large (approximately 1.33 megabytes).



**Running AutoFroth**

Once the analysis has been started, either by selecting the Start option from the Analysis menu or clicking on the green traffic light button on the toolbar, no further intervention is required from the user. If desired, the user can turn image and data logging on and off while the analysis is running. To stop the analysis, the user either clicks on the red traffic light button or selects Halt from the Analysis menu. The analysis will not halt immediately, but will continue until the third input channel has been sampled and the images from that channel have been processed.

When the analysis is started, the first camera is made active, and 10 image frames are grabbed. This is because when the framegrabber input is switched to a different channel, it takes between

3 and 4 grabbed frames for the image to stabilise. The froth velocity is calculated by applying the pixel tracing algorithm (as defined in Section 8.4.2) to frames 8 and 9 of the 10 frame sequence. Frame 8 is then segmented and the image parameters obtained are written to the system database. The next camera is then selected, and a new set of frames is grabbed. The analysis continues in this manner until stopped by the user. The flow diagram in Figure 9.10 indicates the various steps performed when AutoFroth is analysing images.

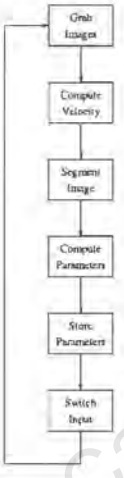



Figure 9.10: AutoFroth algorithm flow diagram



## 9.3 Plant Testwork

In early 1999, the complete prototype analysis system was deployed on the concentrator in preparation for the testwork campaign. Staff from both UCT and UMIST were present for the campaign, assisted by personnel from the concentrator.

The aim of the testwork was to attempt to identify links between the visual appearance of flotation froths (quantified using data from the image analysis system) and metallurgical performance of the system (quantified using mass and assay analyses of slurry samples) [62]. As part of the testing, modifications were to be made to reagent addition levels and froth heights, with the image analysis system being used to detect variations in the froth appearance due to these changes.

Three video cameras placed at the start, middle and end of the bank were used to gather the images for analysis. This decision was made because there is a large and immediately perceivable

difference in the nature and appearance of the froth at the start and end of the bank. Figures 9.11 and 9.12 show one of the cameras installed above a flotation cell on the plant.

Following the installation and commissioning of the image analysis system (which proceeded without any major difficulties being encountered), an intensive metallurgical sampling campaign was undertaken. The image analysis system was initially run only when samples were being taken, but the system proved to be so reliable that towards the end of the campaign it was left running almost 24 hours a day.

The metallurgical part of the testwork proved to be a difficult task, as the plant was very sensitive to changes made upstream in the milling section. These changes were for the most part unexpected and unwelcome, and often totally overwhelmed the changes made as part of the test procedure. In spite of this, vast amounts of useful data were obtained during the testing. Some key results obtained are presented in Section 10.5 of Chapter 10.

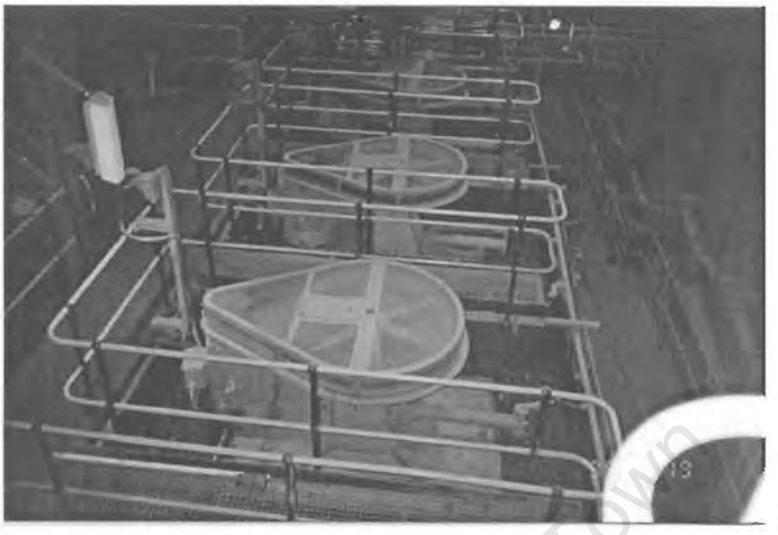


Figure 9.11: A camera installation on a flotation cell



Figure 9.12: A more detailed view of the camera and floodlight assembly

# Chapter 10

# Results

## 10.1 A Discussion of Techniques for Analysing Image Segmentation

Having developed and implemented an algorithm to automatically segment froth images, it is useful to be able to judge the performance of the algorithm. In this context, performance refers to how "well" the algorithm is able to segment images, rather than how quickly the images can be segmented (although this is also a useful parameter to know). From a purely theoretical point of view, optimum segmentation is achieved when every bubble in the froth image has been correctly segmented.

In practice, however, quantifying the accuracy of a segmentation algorithm is a difficult (and contentious) task. The usual approach to this task [65, 16, 41] is to initially perform a "manual" segmentation of an image, that is, a human locates and draws the object boundaries on the image by hand. This segmentation is considered to be "ideal". The same image is then segmented automatically using the appropriate algorithm, and the quality or correctness of this segmentation is evaluated by computing some measure of how close the results of the two methods are to each other.

### 10.1.1 The Chi-Squared Test

In earlier work on analysis of automatically segmented froth images, Symonds [65] and Liu [41] attempted to quantify segmentation accuracy using the *chi-squared ($\chi^2$) goodness of fit* method.

The chi-squared test is a statistic used to compare the fit of experimentally *observed* data to theoretically *expected* data [65], and is defined as:

$$\chi^2 = \sum_{j=1}^{r} \frac{(O_j - E_j)^2}{E_j}, \text{ with } f \text{ degrees of freedom}$$

where:

- $r$ is the total number of cells (divisions).
- $O_j$ is the number of observations occurring in cell $j$.
- $E_j$ is the expected number of observations for cell $j$, based on a known distribution.
- $f$ is the number of degrees of freedom, which in general is equal to $(r - 1)$ minus the number of quantities on which the expected data are based.

Using the *null hypothesis* ($H_0$) and the *alternate hypothesis* ($H_1$):

$H_0$ : the observed distribution is similar to the expected or theoretical distribution.

$H_1$ : the observed distribution is **not** similar to the expected distribution.

and a stated level of significance, it is possible to interpret the results of the chi-squared test using tabulated values for $\chi^2$, where:

$H_0$ is rejected if the calculated sample $\chi^2 >$ the tabulated $\chi^2$

$H_0$ is accepted if the calculated sample $\chi^2 <$ the tabulated $\chi^2$

The chi-squared test was used to determine the accuracy of automatic froth image segmentation by means of the following procedure:

The images under test were segmented both by hand and automatically (as mentioned above). Bubble size and shape distributions were computed for the hand segmented (the "expected" distributions) and the automatically segmented (the "observed" distributions) images. The chi-squared test was performed on the expected and observed distributions. If $H_0$ was accepted for a significance level of 0.05, then the distributions for the two types of segmentation were considered to be similar, and the automatic segmentation could be classified as being "accurate".

During this research, it became apparent that the chi-squared test is not particularly suited for use in quantifying segmentation accuracy. This is largely due to the way in which the distributions of expected and observed values are computed. Multiple distributions of the same set of data were created, by simply choosing different (and arbitrary) values for the number of bins and bin widths of the distributions. Applying the $\chi^2$ test to the different distributions, it was possible to both accept and reject $H_0$ **for the same data set**. This indicates that judicious computation of distributions could quite possibly result in $H_0$ always being accepted, and any segmentation thus being considered "accurate".

### 10.1.2 Other Methods for Segmentation Quantification

Another method of segmentation quantification that was investigated involved classifying the size and shape information (as defined in Chapter 8) into three divisions ("small", "medium" and "large" for bubble sizes, and similar arbitrary divisions for the bubble size information). With regard to the bubble size information, the percentage of the total froth area covered by each size class was calculated for both manual and automatically segmented images. The difference in these percentages for the two segmentation methods is used as a measure of quality, with a small difference being indicative of good segmentation. It was, however, discovered that this method suffers from the same shortcoming as the $\chi^2$ method, in that the choice of boundaries for the three size fractions influences the result that is used to measure the quality of the segmentation. Table 10.1, which shows the results of this method applied twice to a single set of data, but with different values for small, medium and large bubbles, illustrates this problem.

A variation of this method was investigated whereby the size fraction boundaries are related to the size of the image. As an example, if a bubble has an area greater than or equal to one-tenth of the image area, it should be classified as "large". This was done in an attempt to make the choice of class boundaries less arbitrary. It was soon established that this method was also

Table 10.1: Size fraction results for a single data set with different class boundaries

| *Size Fraction* | **Manual** | **Automatic** | **Difference** |
|---|---|---|---|
| *Small* | 2.6259 | 2.6388 | 0.0129 |
| *Medium* | 3.2003 | 6.0376 | 2.8373 |
| *Large* | 94.1738 | 91.3236 | -2.8502 |
| *Size Fraction* | **Manual** | **Automatic** | **Difference** |
| *Small* | 9.0265 | 14.9636 | 5.9370 |
| *Medium* | 9.6923 | 14.3911 | 4.6989 |
| *Large* | 81.2812 | 70.6453 | -10.6359 |

unsuitable, due mainly to the problem of *spatial extent*. Spatial extent refers to how large an area of a physical object is captured in an image. For example an image with a *spatial resolution* of 640 by 480 pixels and a spatial extent of 10 square centimetres has a larger spatial extent than an image at the same resolution with a spatial extent of 5 square centimetres. Returning to the case of froth images, a bubble that can be classified as "large" in an image with large spatial extent will occupy much less of the total image area than a "large" bubble in an image with smaller spatial extent. Since the spatial extent of the images used in this research was unknown, the total area was computed based on the spatial resolution of the images, and this once again renders the concepts of "small", "medium" and "large" bubbles meaningless in the context of quantifying segmentation accuracy.

All of the above methods make use of hand-segmented images as the theoretically ideal case, to which automatic segmentation methods are compared. An implicit assumption here is that all the bubbles in the hand segmented images have been identified and delineated sufficiently accurately. In practice, however, sometimes it is difficult for the human eye to identify **all** the bubble boundaries in the image, as well as for the segmentation lines to be drawn very accurately, especially if the bubbles are very small. These problems are exacerbated if the images are of poor quality, and more particularly when the image contrast is low. It may not be unreasonable to suggest that in some cases, the segmentation algorithm has the potential to do a better job of locating bubble boundaries than a human, especially in low-contrast images, where a difference of one or two grayscale values may indicate a bubble boundary. The central issue in this case is the fact that the hand-segmented images cannot always be relied upon to be an indicator of perfect segmentation.

## 10.2 Analysis of Segmented Froth Images in this Research

Based on the discussion in Section 10.1, it can be argued that there is no immediately obvious method of defining a robust and consistent quantitative measure of image segmentation accuracy. However, some indication of the degree of success of the segmentation algorithm is required. The method applied here makes use of bubble size and shape distributions from hand-segmented images as an *approximation* to accurate image segmentation. Similar distributions are computed for the automatically segmented images, and the perceived quality of segmentation is discussed by referring to the distributions and the segmented images.

In this analysis, distributions are calculated for bubble size (i.e. area), ellipticity, perimeter and circularity. These parameters, as defined in Chapter 8 are extracted from the hand segmented images by a C++ program and the results written to data files. Similar data files are generated for the automatically segmented images.

MATLAB software was used for creating the distributions, with the process being automated by writing appropriate MATLAB scripts and functions. The bin widths of the distributions are computed by applying the Freedman-Diaconis rule [33] to the size and shape data from the hand-segmented images. The Freedman-Diconis rule is given by

$$h = 2(\mathrm{IQ})n^{-\frac{1}{3}} \tag{10.1}$$

where $h$ is the bin width and IQ is the *interquartile range*[1] of the dataset.

Once the bin width has been computed, the number of bins $k$ is calculated using

$$k = \mathrm{range}/h \tag{10.2}$$

where **range** is the range of values in the data set. Using $k$, an equally spaced set of histogram bins can easily be calculated. Following this, the MATLAB ***histc*** function is used to create the distributions of the data using the predetermined bins.

The distributions for the automatically segmented image data are computed using the bin widths obtained from the manual data. This enables the similarity of the distributions to be evaluated. The evaluation is aided by plotting the distributions for both data sets on the same set of axes.

[1]For a set of data arranged in increasing order of magnitude, the *quartiles* $Q_1$, $Q_2$ and $Q_3$ are the values that divide the set into four equal parts. The interquartile range is simply $Q_3 - Q_1$ [59].

Cumulative distributions are also computed for the data, again for use in estimating segmentation quality.

## 10.3 Analysis Results

Five images of five different froths (all from rougher flotation cells) were analysed using the method described above. The results are presented as follows:

- The original image
- The hand-segmented image
- The automatically segmented image
- Distributions and cumulative distributions of size, ellipticity, perimeter and circularity data for both types of segmentation.

It should be noted that the segmentation of each image has been performed within a particular region of interest (ROI). The ROIs were chosen to exclude areas of the froth which were difficult to segment accurately, due to reasons such as the froth breaking down at the edge of the flotation cell (most noticeable in images 1 and 4), and poor illumination in some parts of the image (for example, the far right hand side of image 5).

In section 8.3, several advanced parameters extracted from froth images were defined. These measurements do not from part of the analysis presented here, as they are either not relevant, or can be derived from one or more of the measurements presented here.

### 10.3.1 Image I1

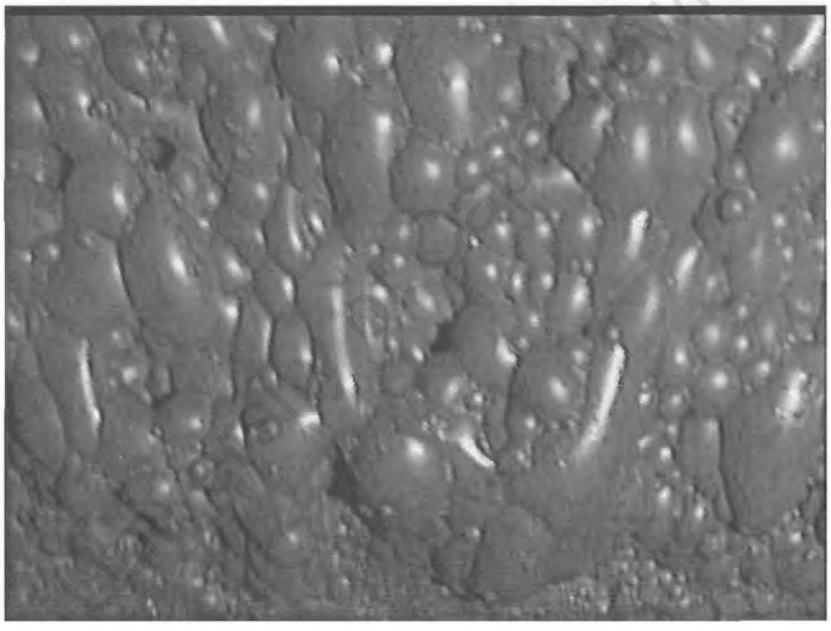


Figure 10.1: Froth image I1

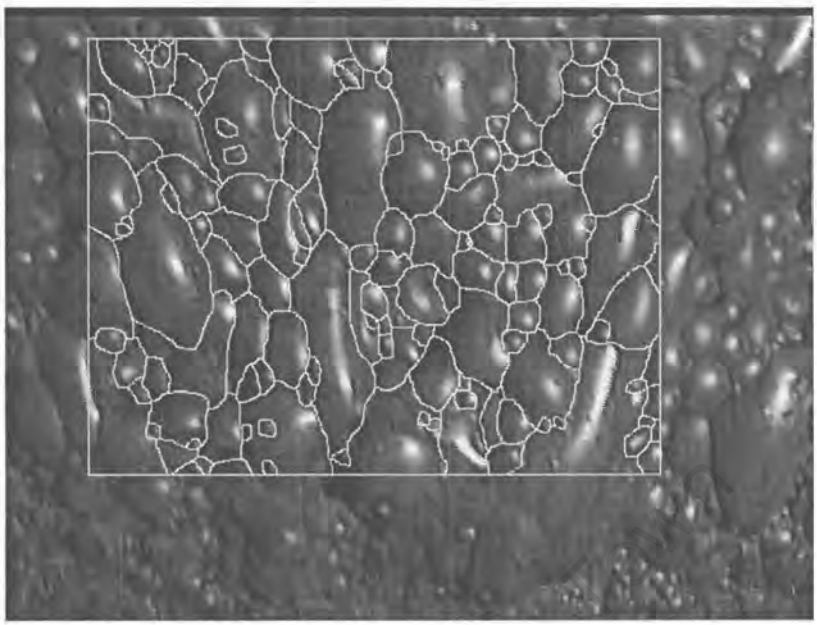


Figure 10.2: Manual segmentation of froth image I1

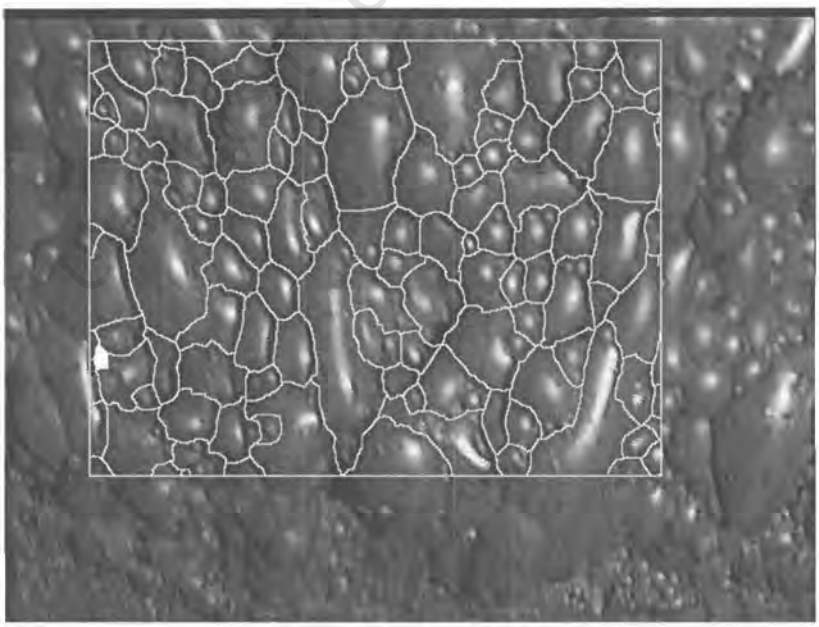
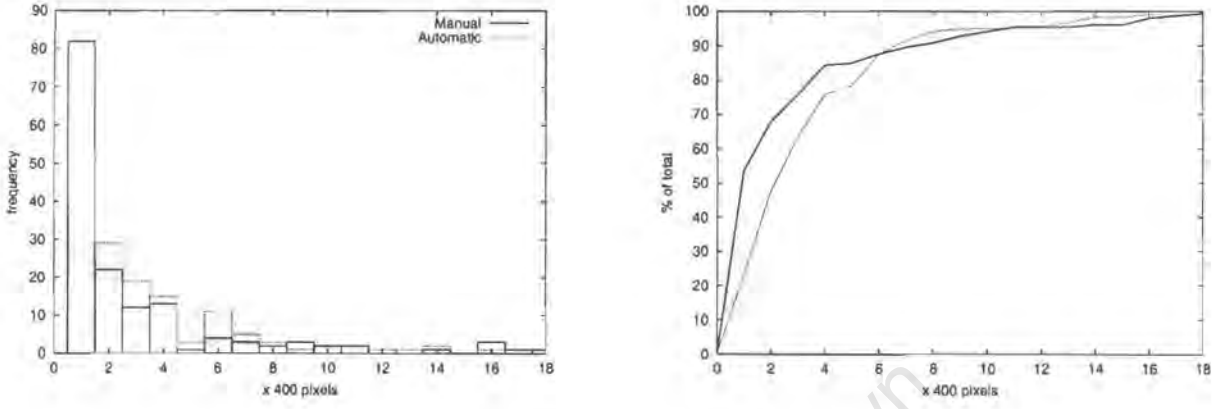
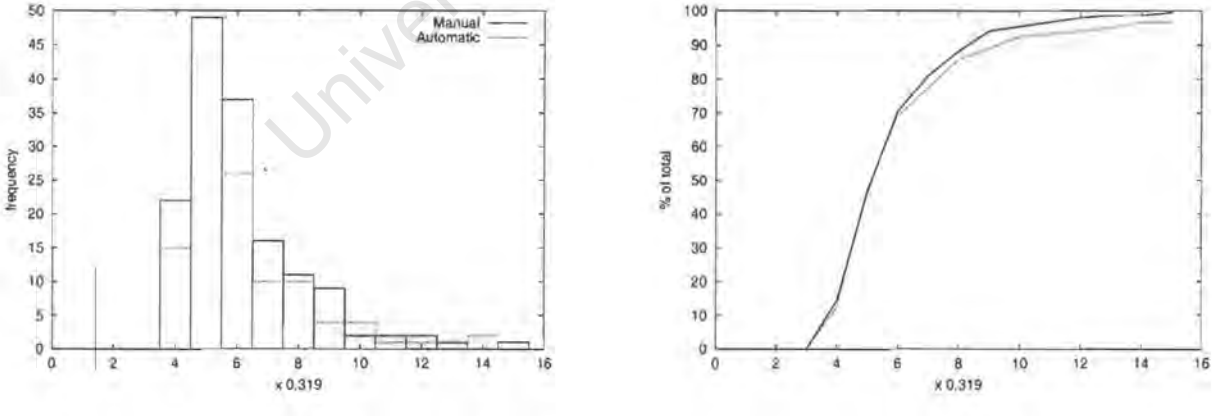


Figure 10.3: Automatic segmentation of froth image I1



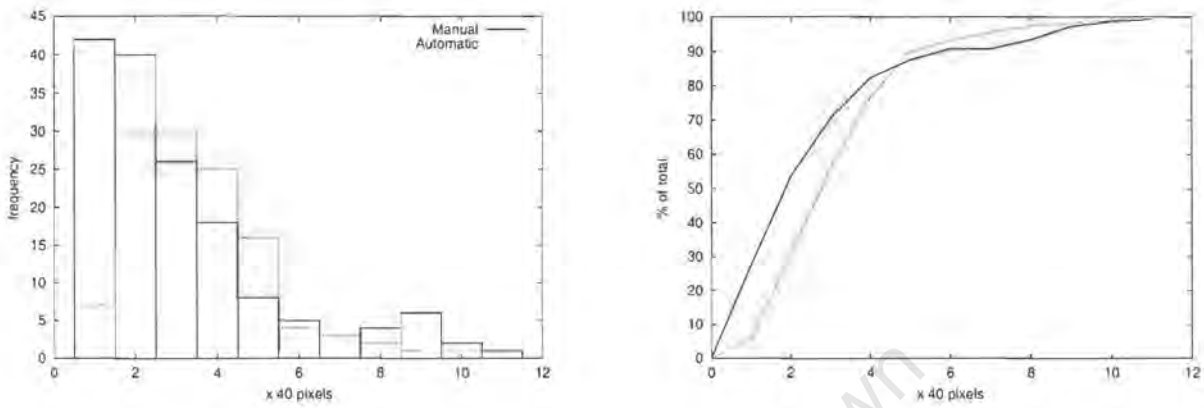

(a) (b)

Figure 10.4: Manual and automatic distributions for bubble area - image I1




(a) (b)

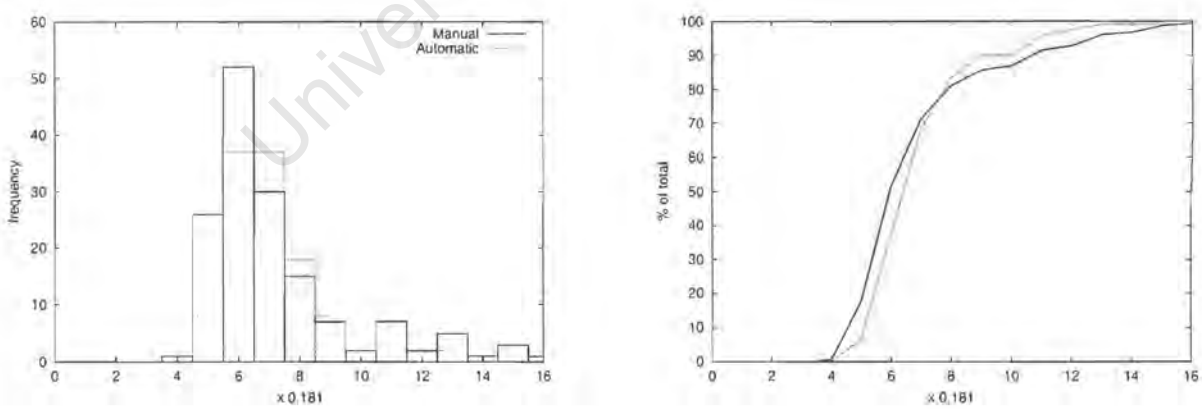
Figure 10.5: Manual and automatic distributions for bubble ellipticity - image I1




(a) (b)

Figure 10.6: Manual and automatic distributions for bubble perimeter - image I1






(a) (b)

Figure 10.7: Manual and automatic distributions for bubble circularity - image I1

### 10.3.2 Image I2

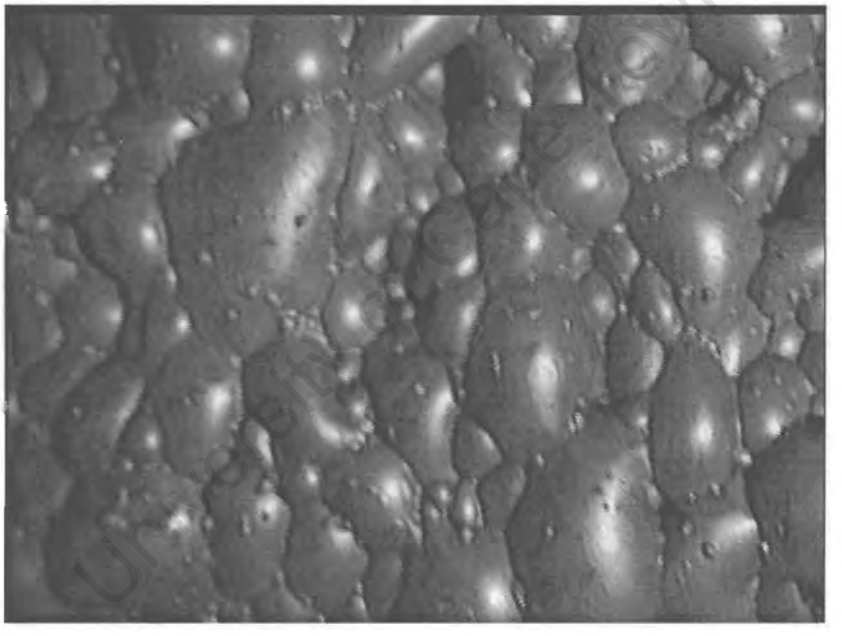


Figure 10.8: Froth image I2

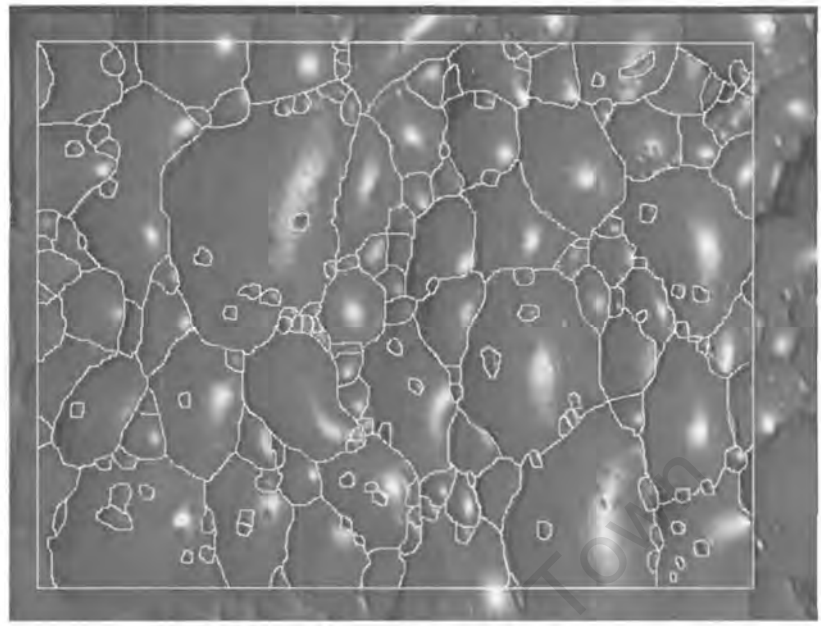


Figure 10.9: Manual segmentation of froth image I2

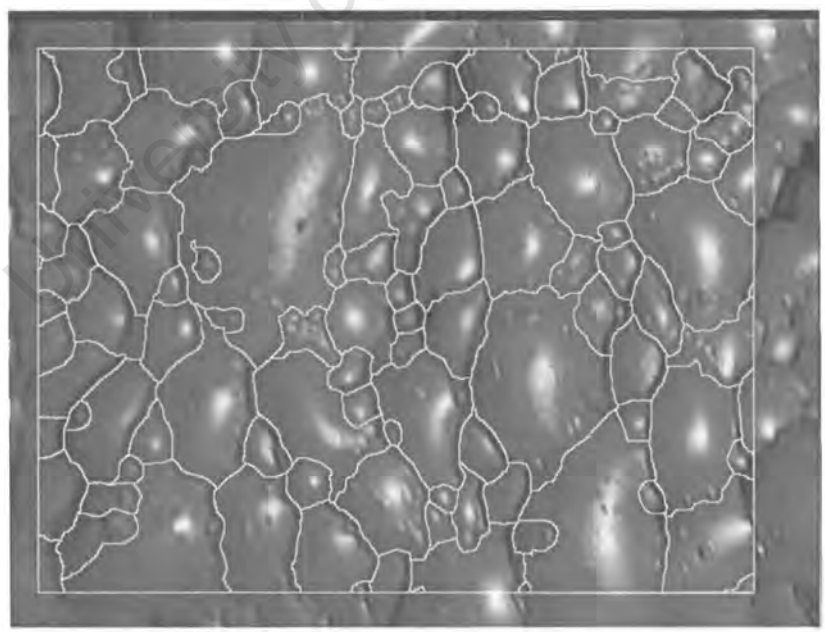
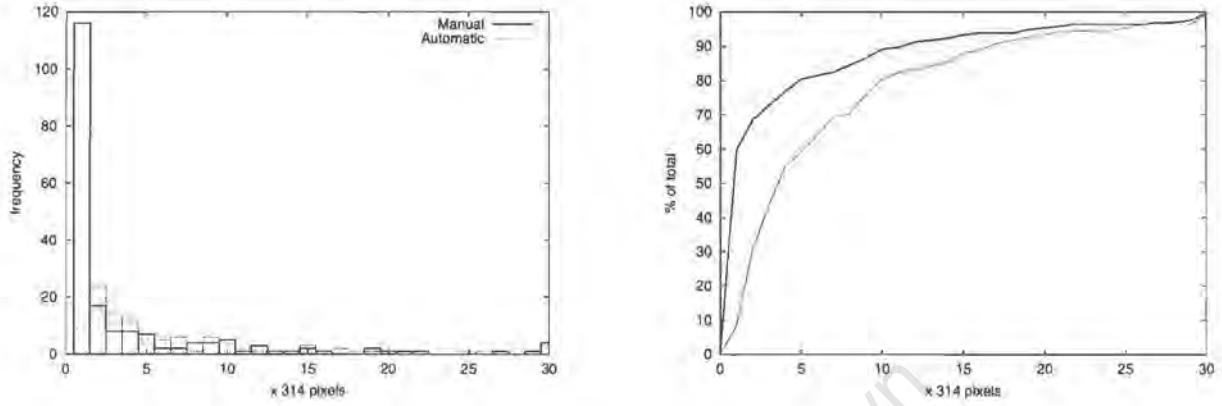
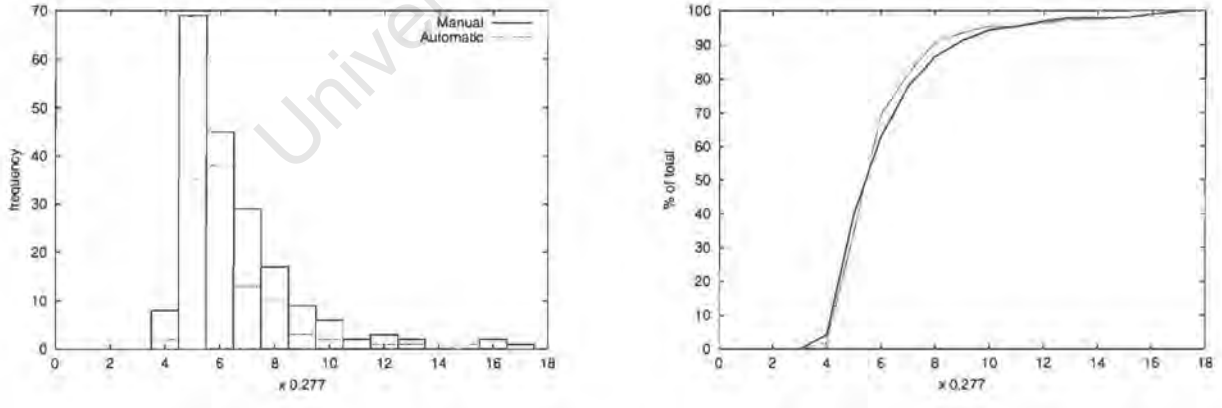


Figure 10.10: Automatic segmentation of froth image I2



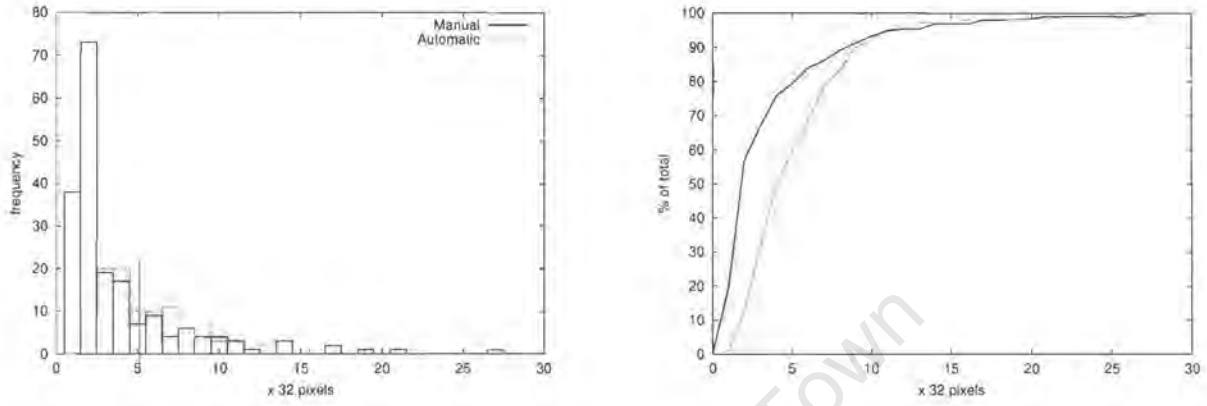

(a) (b)

Figure 10.11: Manual and automatic distributions for bubble area - image I2




(a) (b)

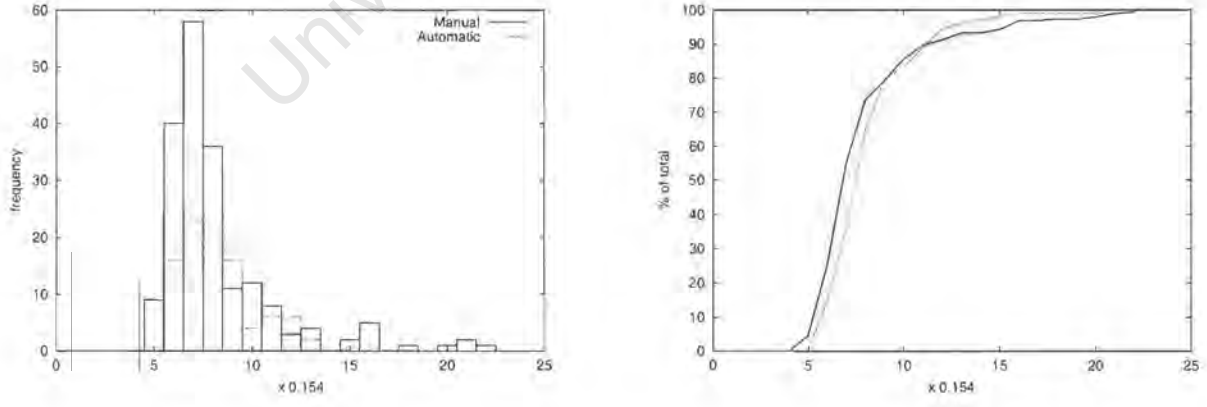
Figure 10.12: Manual and automatic distributions for bubble ellipticity - image I2




(a) (b)

Figure 10.13: Manual and automatic distributions for bubble perimeter - image I2






(a) (b)

Figure 10.14: Manual and automatic distributions for bubble circularity - image I2

### 10.3.3 Image I3

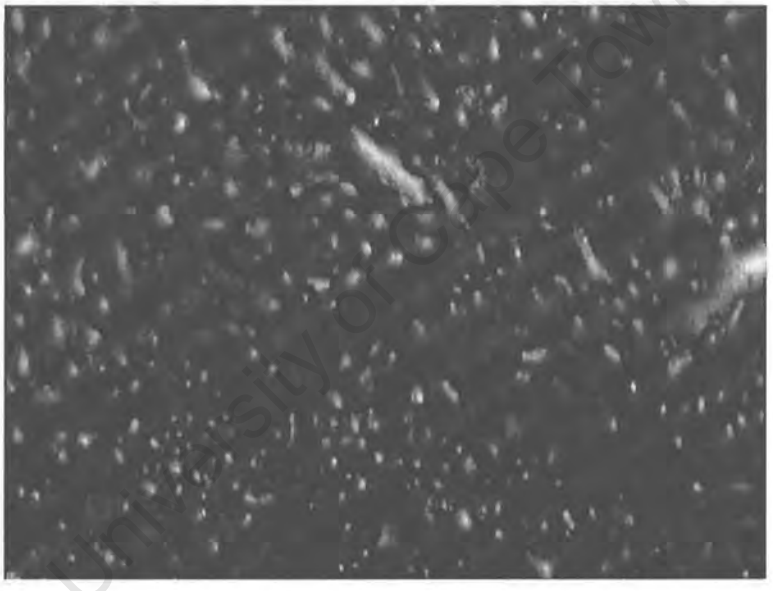


Figure 10.15: Froth image I3

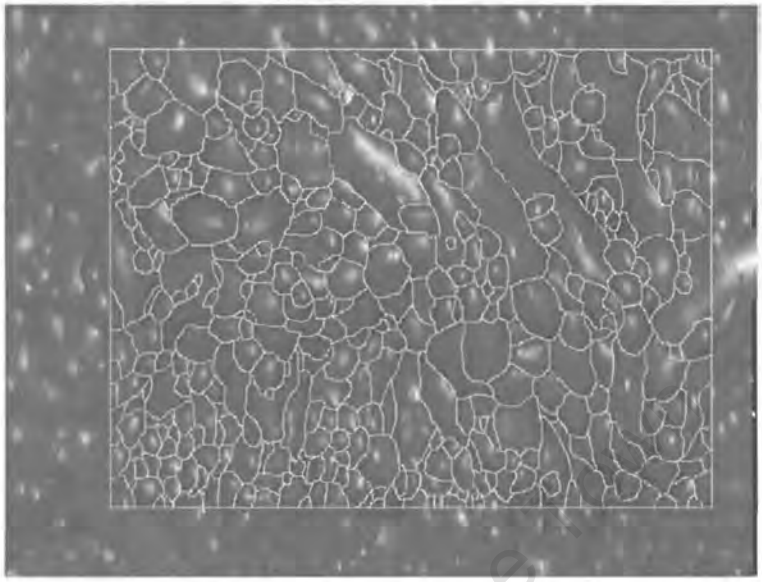


Figure 10.16: Manual segmentation of froth image I3

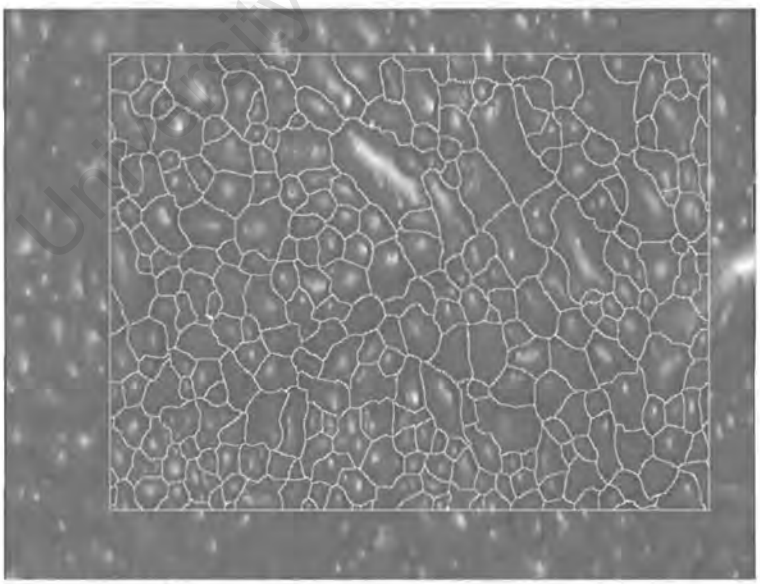
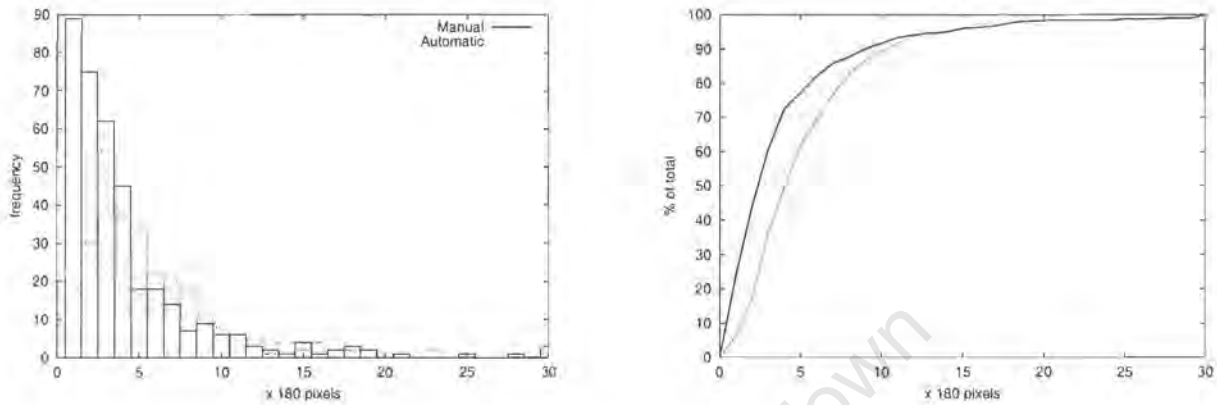
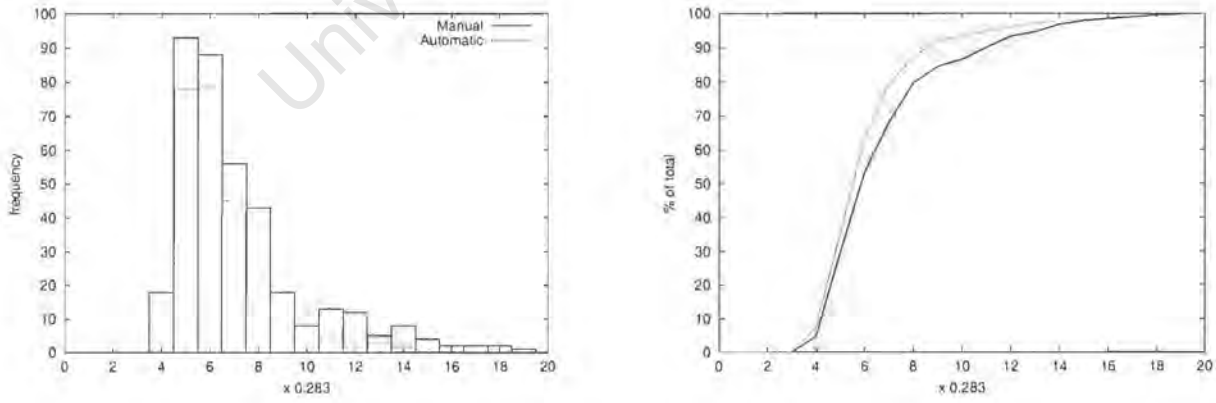


Figure 10.17: Automatic segmentation of froth image I3



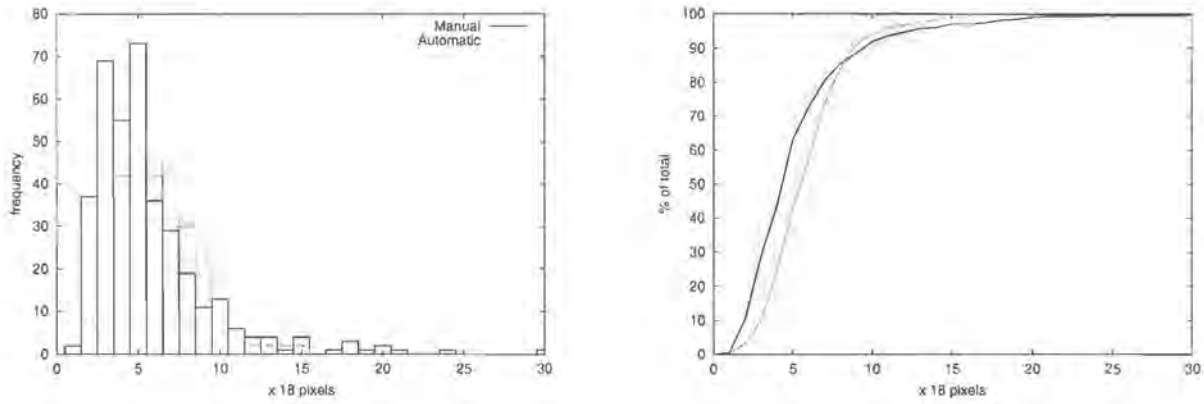

(a) (b)

Figure 10.18: Manual and automatic distributions for bubble area - image I3



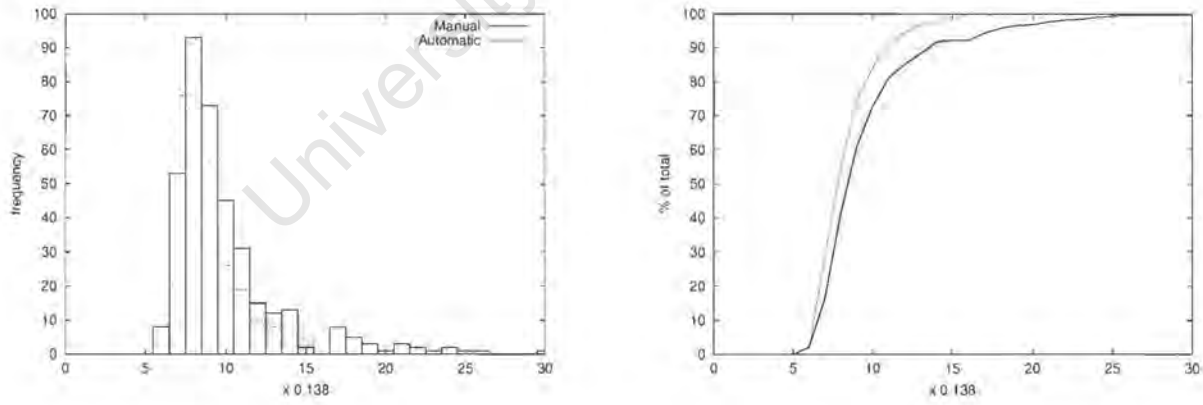

(a) (b)

Figure 10.19: Manual and automatic distributions for bubble ellipticity - image I3




(a) (b)

Figure 10.20: Manual and automatic distributions for bubble perimeter - image I3




(a) (b)

Figure 10.21: Manual and automatic distributions for bubble circularity - image I3

### 10.3.4 Image I4

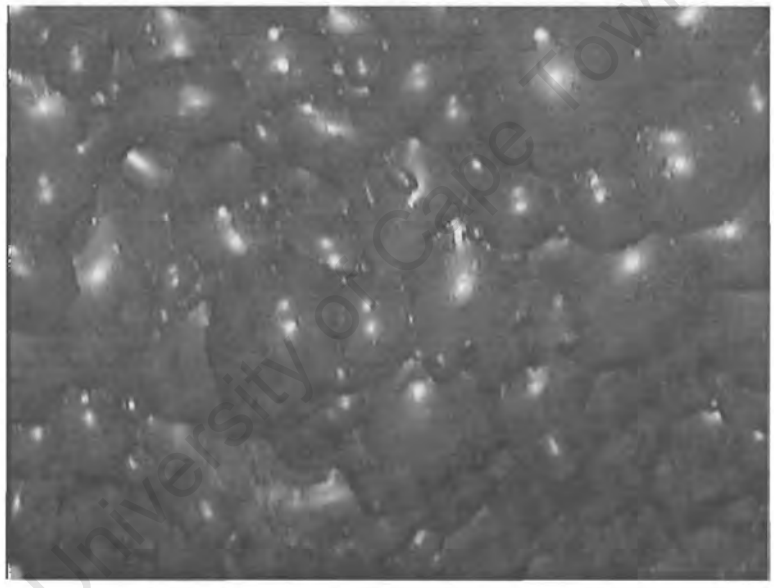




Figure 10.22: Froth image I4

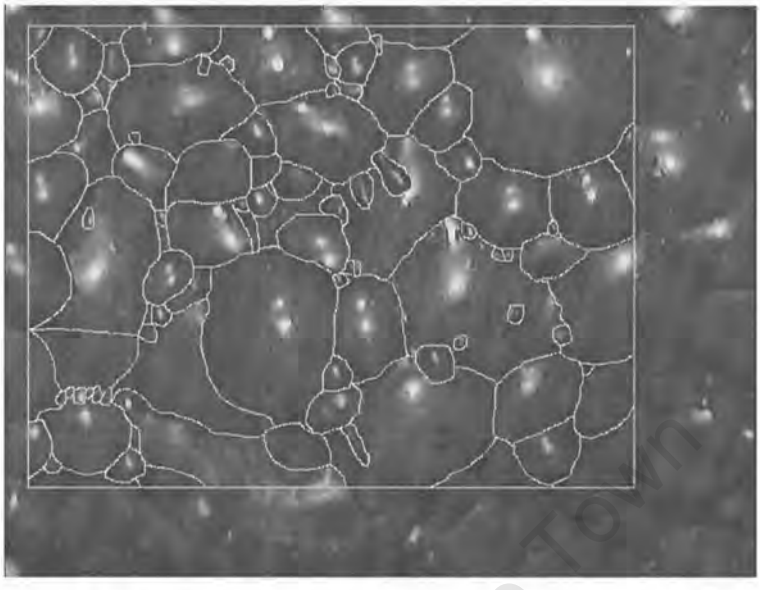


Figure 10.23: Manual segmentation of froth image I4

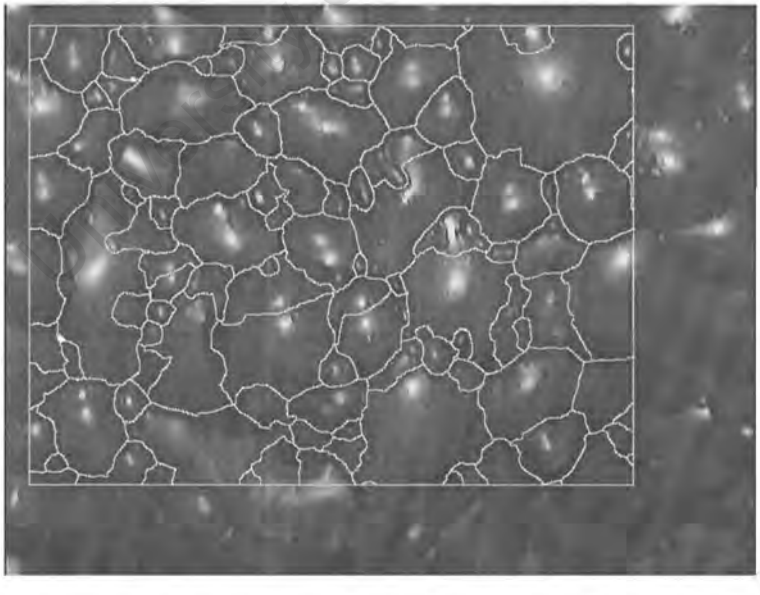
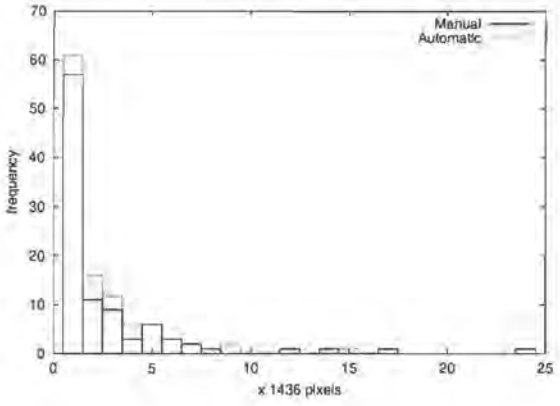
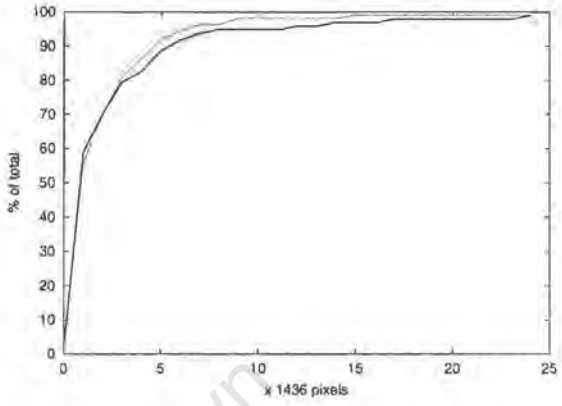


Figure 10.24: Automatic segmentation of froth image I4



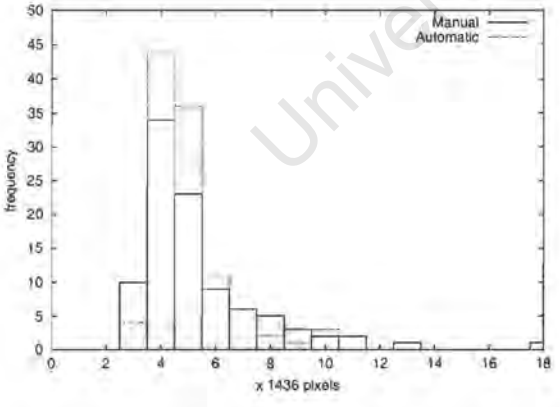

(a)



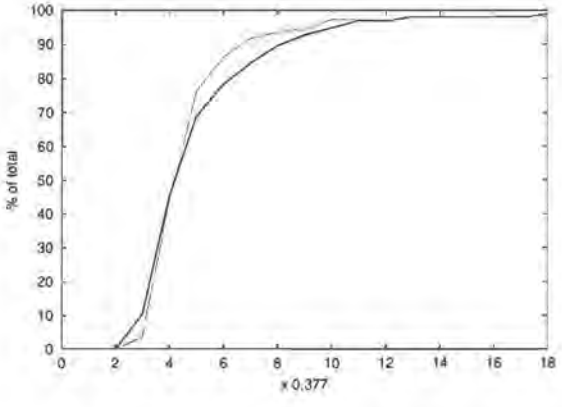

(b)

Figure 10.25: Manual and automatic distributions for bubble area - image I4



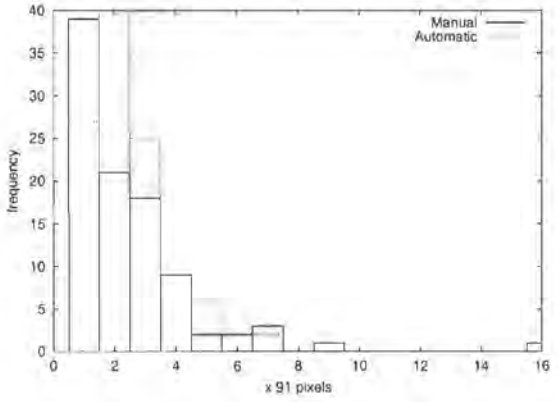



(a)

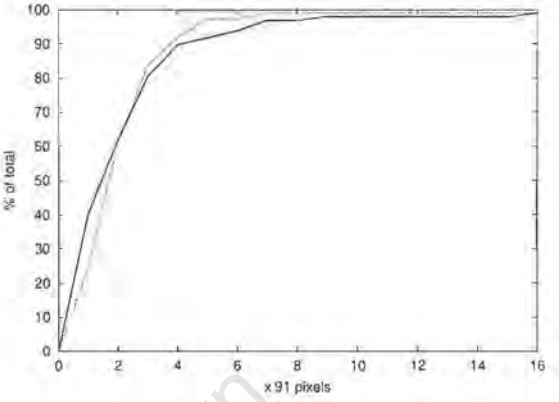



(b)

Figure 10.26: Manual and automatic distributions for bubble ellipticity - image I4

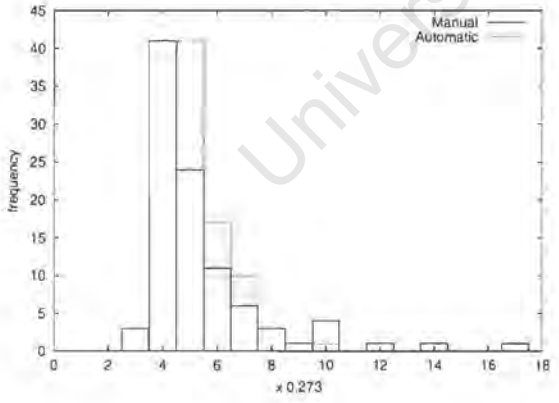



(a)

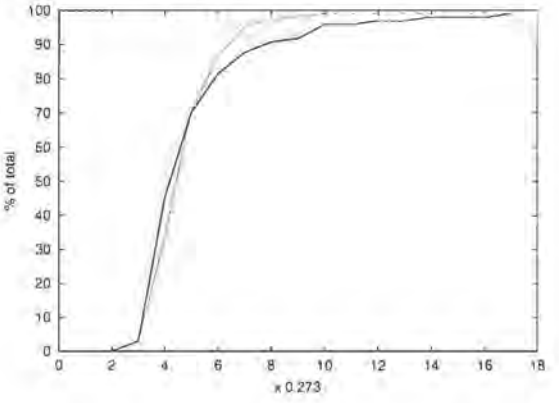



(b)

Figure 10.27: Manual and automatic distributions for bubble perimeter - image I4




(a)




(b)

Figure 10.28: Manual and automatic distributions for bubble circularity - image I4

### 10.3.5 Image I5

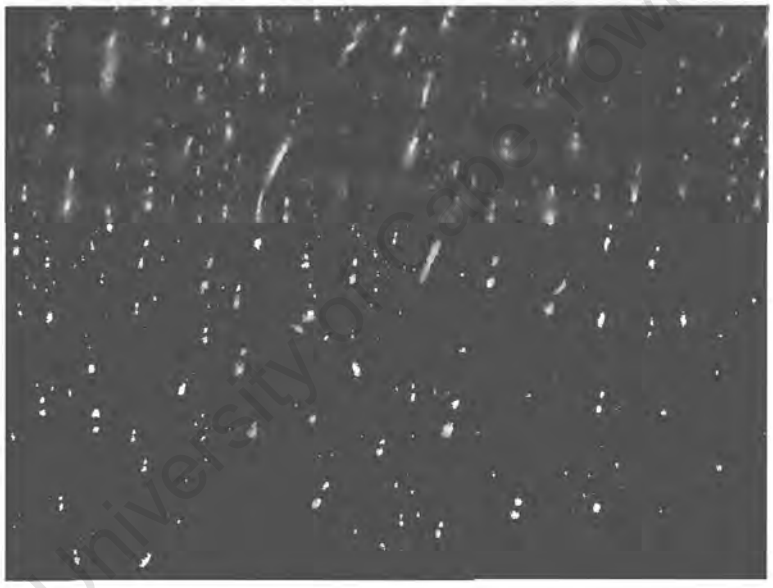




Figure 10.29: Froth image I5

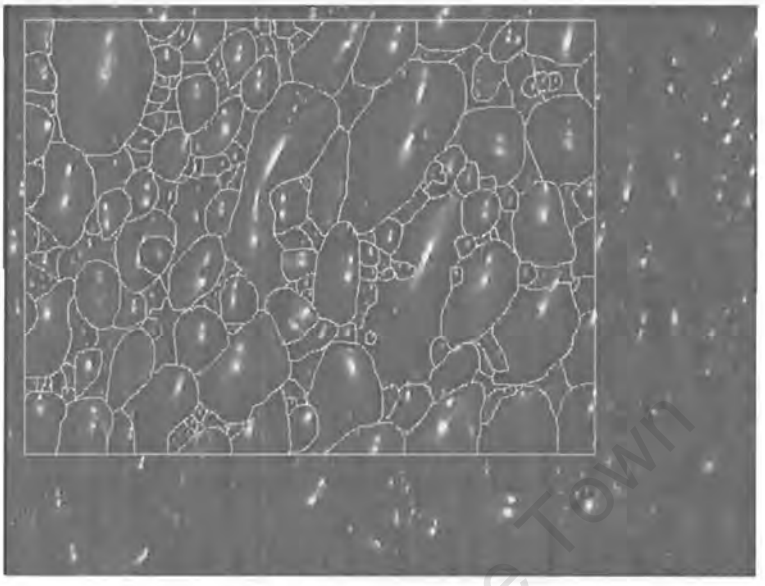


Figure 10.30: Manual segmentation of froth image I5

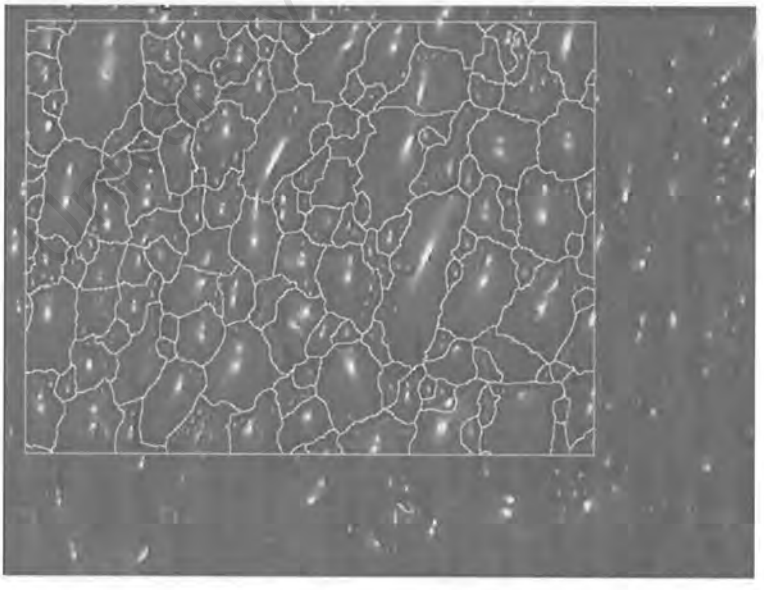
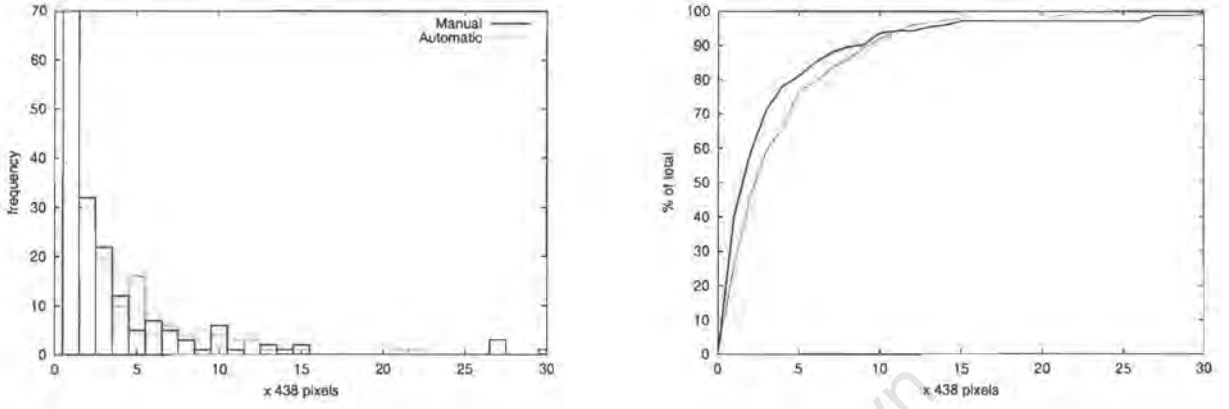
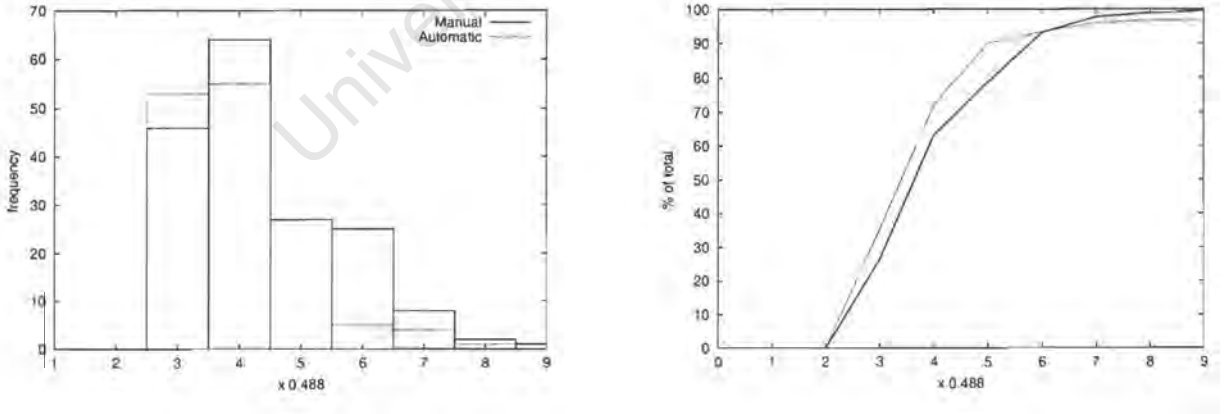


Figure 10.31: Automatic segmentation of froth image I5



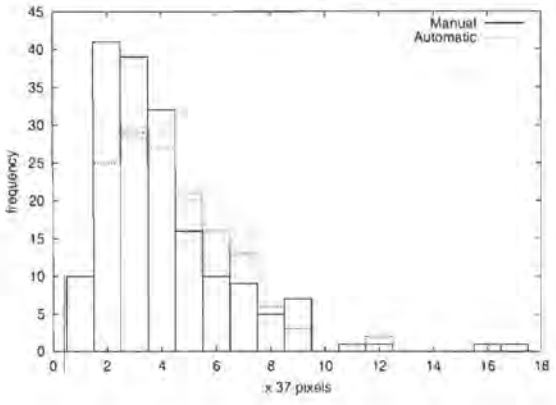

(a) (b)

Figure 10.32: Manual and automatic distributions for bubble area - image I5

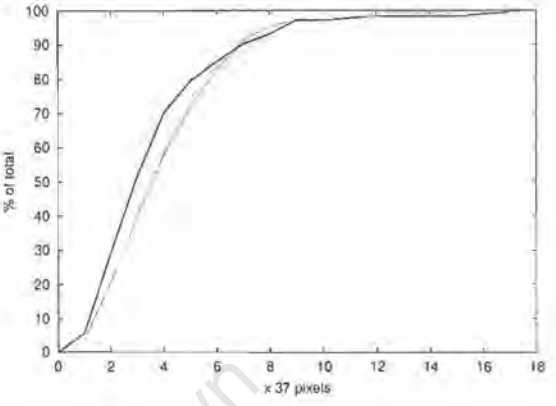



(a) (b)

Figure 10.33: Manual and automatic distributions for bubble ellipticity - image I5

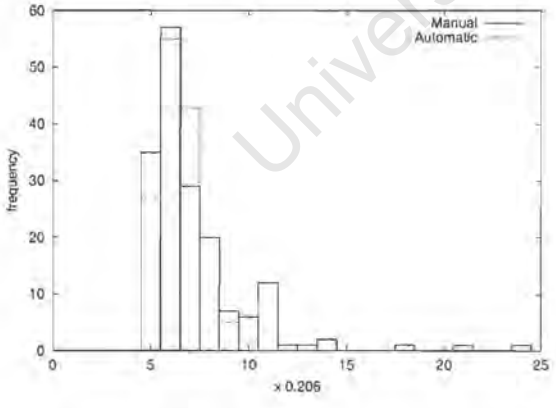



(a)

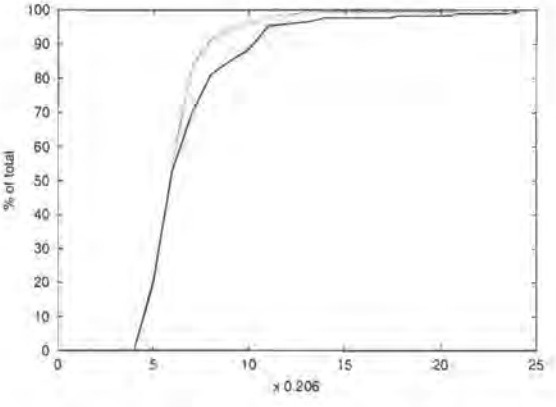



(b)

Figure 10.34: Manual and automatic distributions for bubble perimeter - image I5




(a)




(b)

Figure 10.35: Manual and automatic distributions for bubble circularity - image I5

## 10.4 Discussion of Individual Analyses

### 10.4.1 Image I1

Most of the large bubbles have been correctly identified, although some have been oversegmented, resulting in the increased count of larger bubble sizes for the Automatic distribution in figure 10.4(a). The automatic segmentation has tended to combine multiple small bubbles into larger bubbles, which accounts for the larger value of the smallest size fraction in the Manual distribution of figure 10.4(a), and the slightly higher number of bubbles in the second to seventh bins of the the Manual distribution. The ellipticity distributions (figures 10.5(a) and 10.5(b)) are similar for Manual and Automatic, indicating that the automatic segmentation has accurately determined the shapes of the bubbles. Figures 10.6(a) to 10.7(b) display similar properties to the area distributions of figures 10.4(a) and 10.4(b), which is most probably due to the fact that perimeter and circularity are measurements that depend quite strongly on the area of the the bubble.

### 10.4.2 Image I2

The automatic segmentation has succeeded in accurately identifying most of the the large bubbles in the image, as is evidenced by the similarity of the two cumulative distribution curves in the right hand part of figure 10.11(b). The automatic segmentation has not, however, been able to accurately identify the numerous very small bubbles in the image, which accounts for the disparity in the first size fraction of figure 10.11(a). The reasons for the appearance of the distributions for ellipticity, perimeter and circularity are similar to those for image I1.

### 10.4.3 Image I3

This image is composed of numerous bubble of various different sizes. Once again, the automatic segmentation has been successful in identifying the larger bubbles in the image, and again tending to group small bubbles together. Examining figure 10.18(b), it can be seen that the area occupied by bubbles with an area larger than approximately 2000 pixels is very similar for both manual and automatic segmentation, proving that larger bubbles are identified accurately. Refer-

ring to figure 10.21(b), it can be seen that the automatically segmented bubbles tend to be less circular than those segmented by hand. This is possibly due to the fact that small bubbles tend to be circular, and the grouping together of these small bubbles by the automatic segmentation produces single blobs that are less circular than their component bubbles.

### 10.4.4 Image I4

Examining figures 10.25(a) and 10.25(b), it can be argued that the automatic segmentation of this image is the most accurate of all the images analysed, as both the number and size of the bubbles are almost identical in both manual and automatic distributions. This is almost certainly due to the fact that most of the bubbles in the image are large and easy to identify and segment. Referring to figures 10.23 and 10.24, only two of the large bubble have been badly oversegmented. The differences between the distributions for the ellipticity, perimeter and circularity data can be attributed to the oversegmentation, which can considerably change the shape of the detected bubbles.

### 10.4.5 Image I5

Segmenting this image by hand proved to be very difficult, as the large bubbles are surrounded by numerous extremely small bubbles, whose boundaries were not easily identified and drawn. The automatic segmentation appears to have identified the large bubbles reasonably accurately, although the boundaries of many of these large bubbles tend to incorporate the very small bubbles. Several of the large bubbles have multiple highlights, and, with a few exceptions on the very large bubbles, the preprocessing algorithm has reduced the effect of these highlights. The lower automatic perimeter and higher ellipticity and circularity values are once again due to the oversegmentation of large bubbles and the combining of small bubbles.

The next section of results deals with the testwork program that was described in section 9.3 of chapter 9. The final section of this chapter will discuss issues related to both the fundamental image segmentation and the on-line system as a whole.

## 10.5 Testwork Results

At the completion of the testwork, over a quarter of a million entries had been made to the froth image parameter database. Once the metallurgical analysis of the samples had been completed, the massive task of examining the data for trends and correlations began. To present detailed results of this analysis is well beyond the scope of this work (and in fact is itself the subject of a separate dissertation). However, it is prudent to present some key results that illustrate the performance of the image analysis system. Recall the the purpose of the testwork was twofold [62]:

- To test the performance of the on-line image analysis system in a real plant environment.
- To elucidate relationships between froth appearance, as defined by the image analysis system, and the plant performance, for both steady-state conditions and after process input step changes.



### 10.5.1 Temporal Performance

A fundamental requirement of the on-line image analysis system was that the image segmentation and analysis should be fast enough to provide useful results, given the dynamic nature of the flotation process. In practice, the complete acquisition and analysis of images from a single flotation cell took approximately 10 seconds, which was considered more than adequate. The massive amounts of data generated are mostly due to the high speed at which the analysis was performed.

### 10.5.2 Relation of Image Parameters to Plant Performance

Figures 10.36 to 10.38 show plots of bubble sizes versus time for a test in which the frother addition rate was stepped (in effect, a system identification step test). The four size plots per graph are for average bubble size and the percentage of the total froth area covered by three bubble size fractions, namely small, medium and large. The three graphs are from cells 2, 6 and 10 in the bank (i.e. top, middle and bottom). It can be seen from the graphs that increasing the amount frother decreases the size of the bubbles in the froth at the top and bottom of the bank

[64]. The plots contain a certain amount of noise but the trends (especially in the plots for the three size fractions) are clearly visible. Figures 10.39 and 10.40, which plot the change in bubble size against relative frother dosage, illustrate the trends much more directly.

Figures 10.41 to 10.43 are the equivalents of Figures 10.36 to 10.38 for a test in which the depressant addition rate was stepped. Note that the gaps in the plots are not the result of a system failure, but rather because the analysis system stopped logging results due to the database transaction log being full. A frother modification was also performed, but the changes in the froth due to the depressant are visible before the frother level change is made. Additionally, because the system stopped logging, there is no image data available for this period during the frother test.

At the bottom of the bank (cell number 10, Figure 10.43), increasing the depressant caused the bubble size to decrease. This change can be seen more clearly in figure 10.44. The most convincing evidence that the system has identified the changes in bubble size correctly are Figures 10.46 to 10.49, which show images of the froth in cell 10 under conditions ranging from no depressant to very high depressant.

At the top of the bank (cell 2, Figure 10.41), as the depressant dosage increases, the bubble size increases up to a critical point, due to the froth becoming more stable. After this point is reached, the bubbles appear to become destabilised and rapid bursting occurs, which results in a bimodal bubble size distribution. Figures 10.45 and 10.50 to 10.53 provide further evidence of these trends.

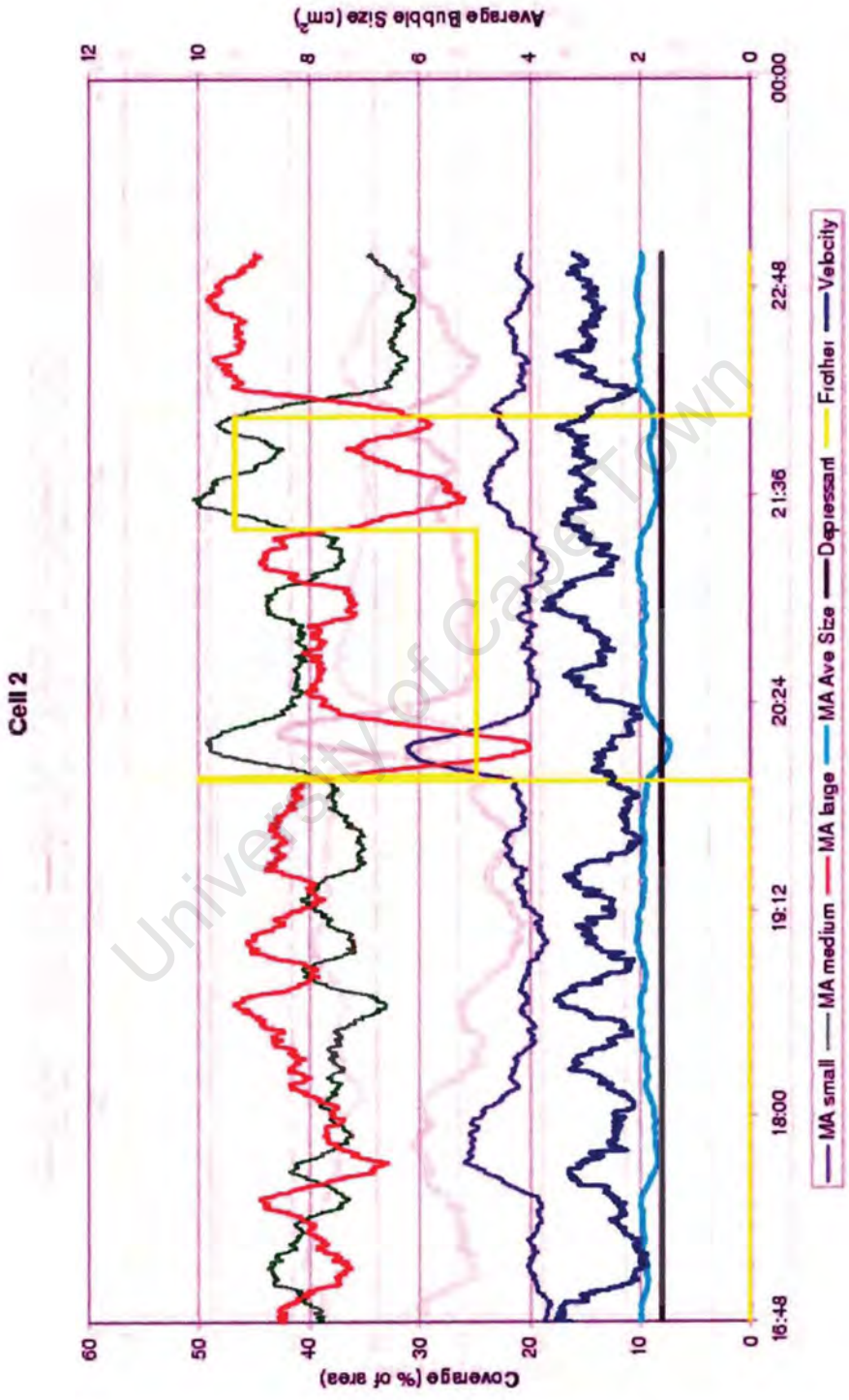



Figure 10.36: Plot of image data for cell 2 during frother test

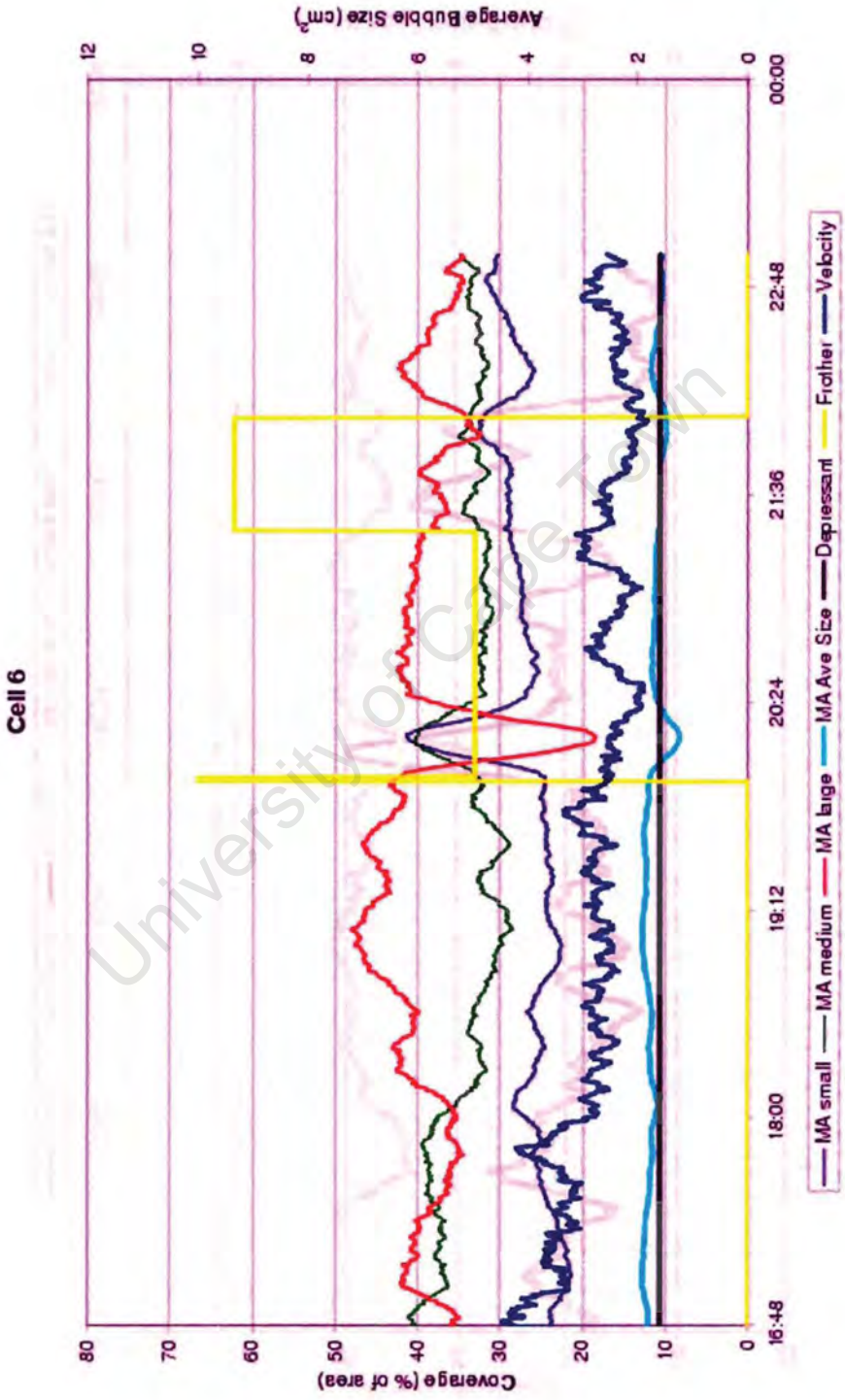



Figure 10.37: Plot of image data for cell 6 during frother test

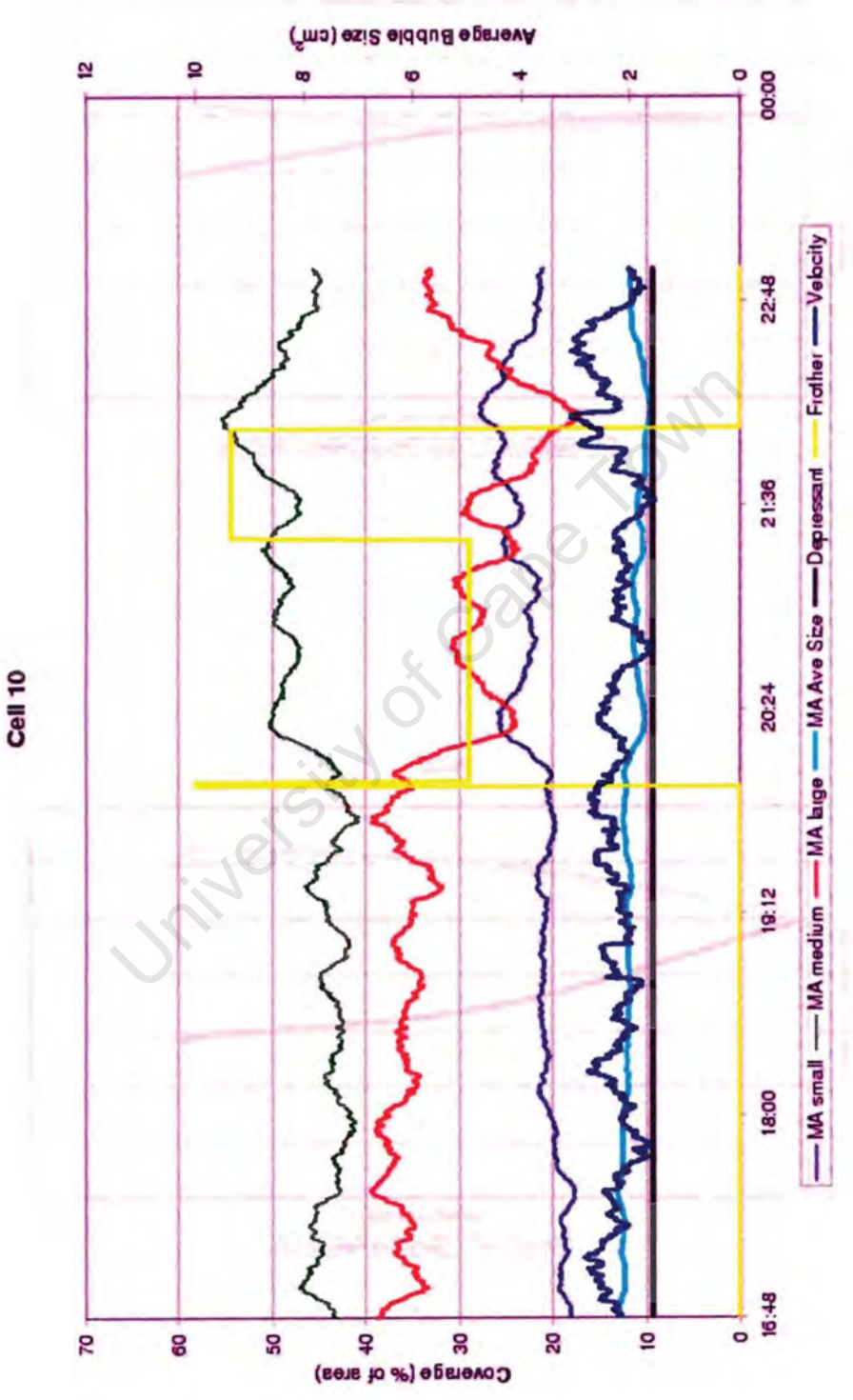





Figure 10.38: Plot of image data for cell 10 during frother test

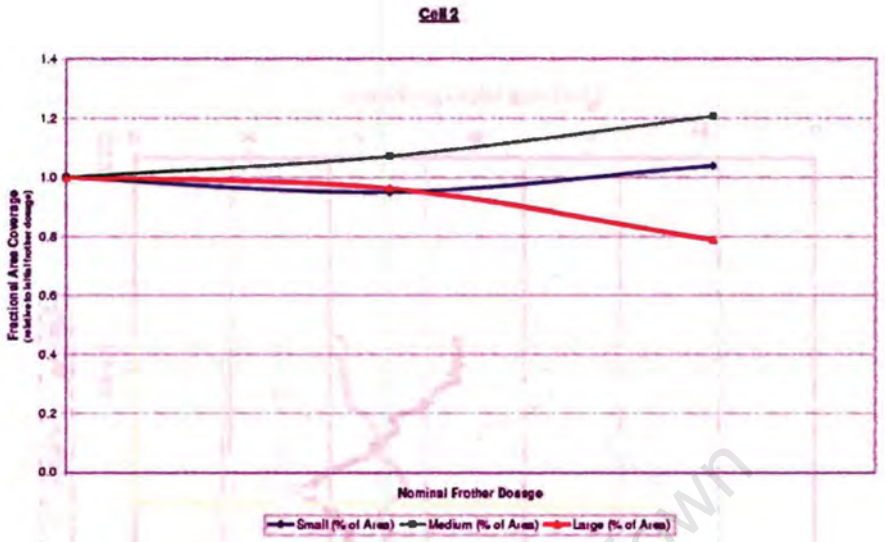



Figure 10.39: Plot of change in bubble size against relative frother dosage for cell 2

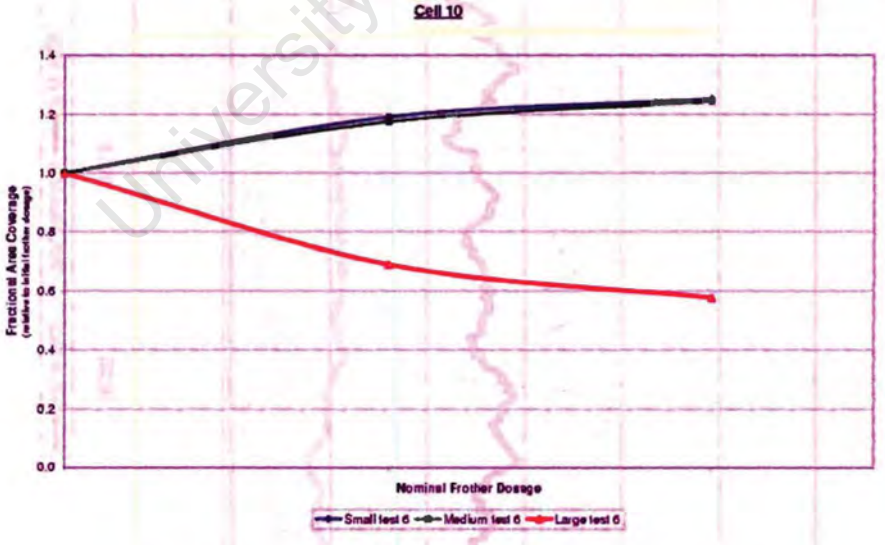



Figure 10.40: Plot of change in bubble size against relative frother dosage for cell 10

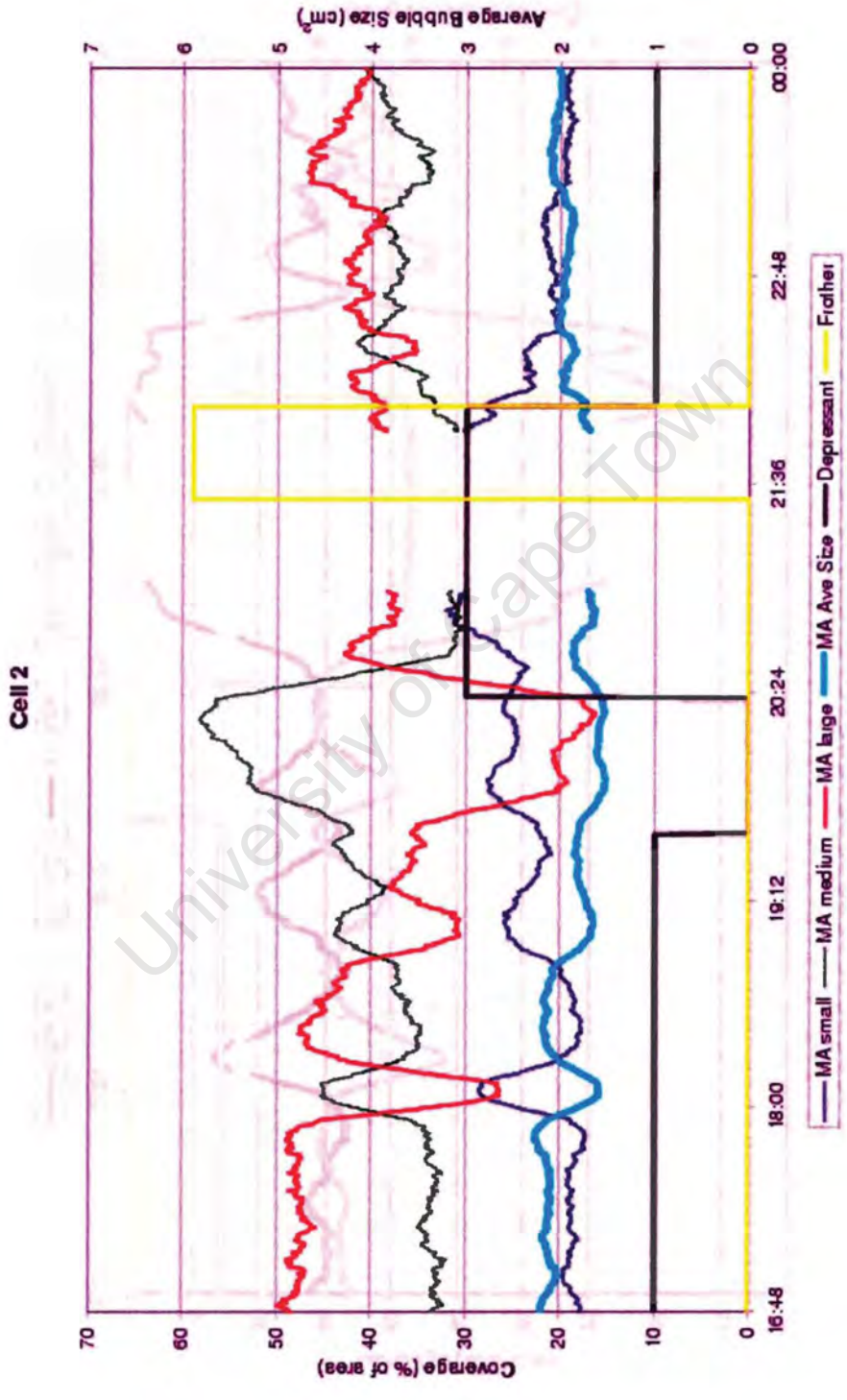



Figure 10.41: Plot of image data for cell 2 during depressant test

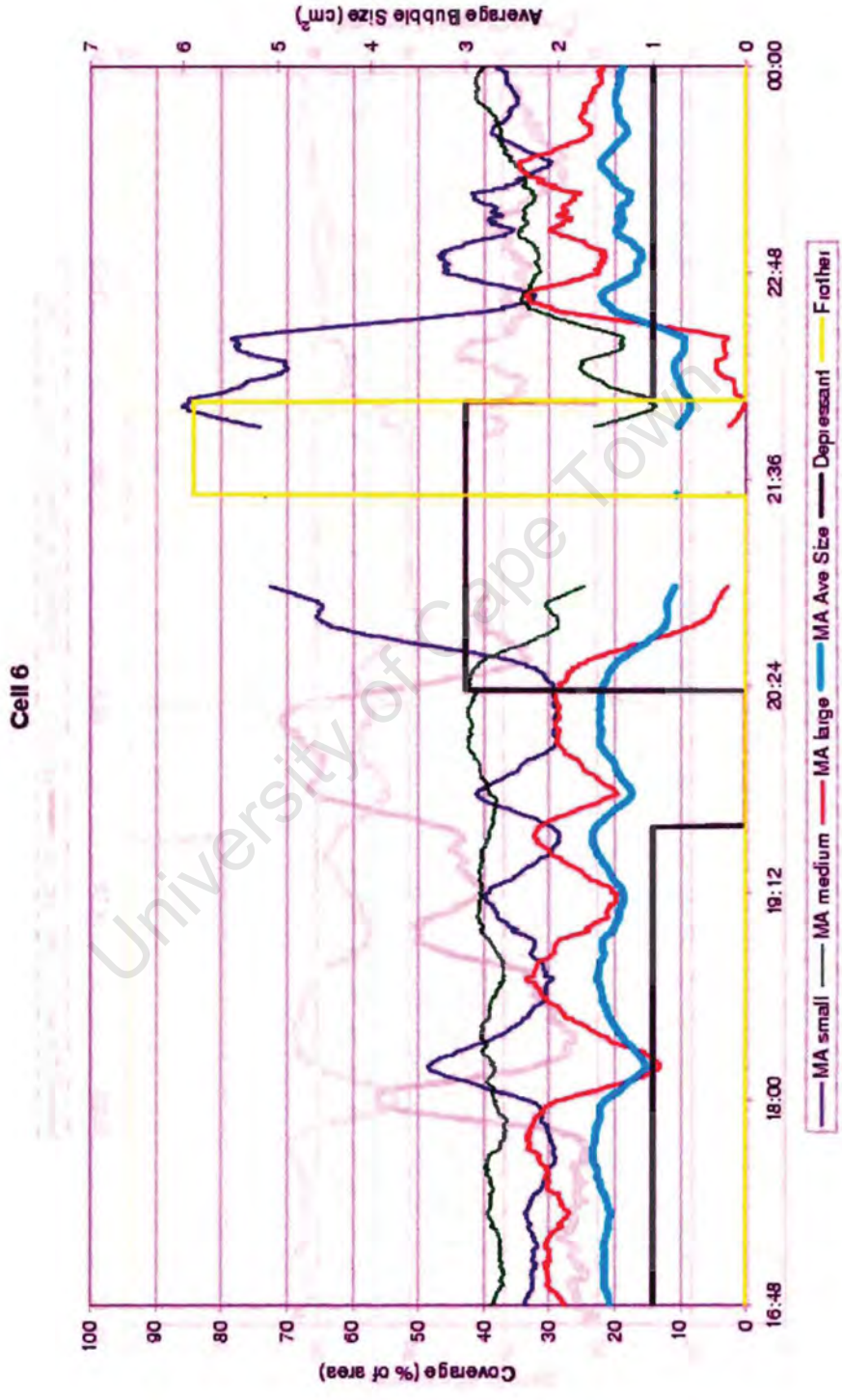



Figure 10.42: Plot of image data for cell 6 during depressant test

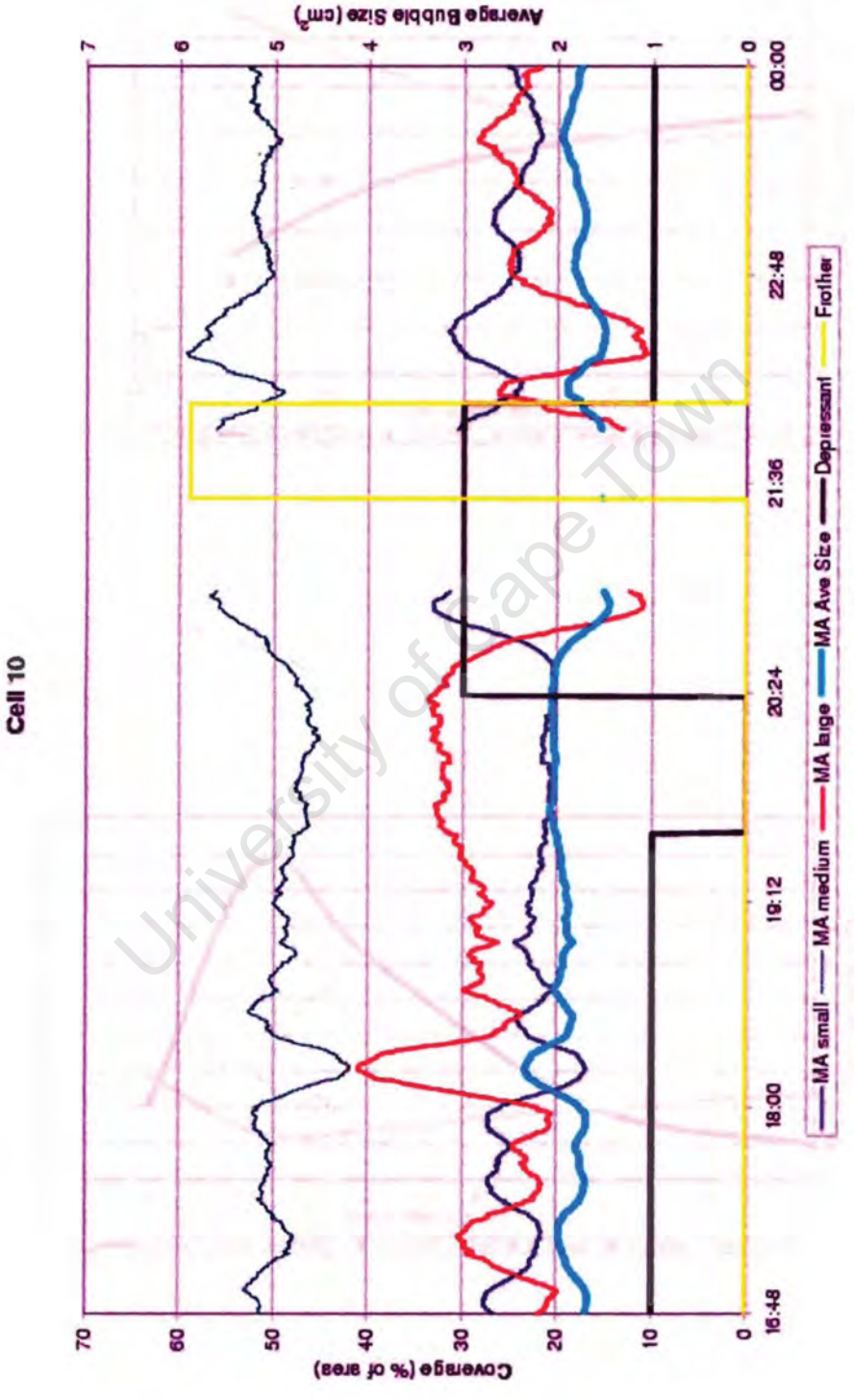





Figure 10.43: Plot of image data for cell 10 during depressant test

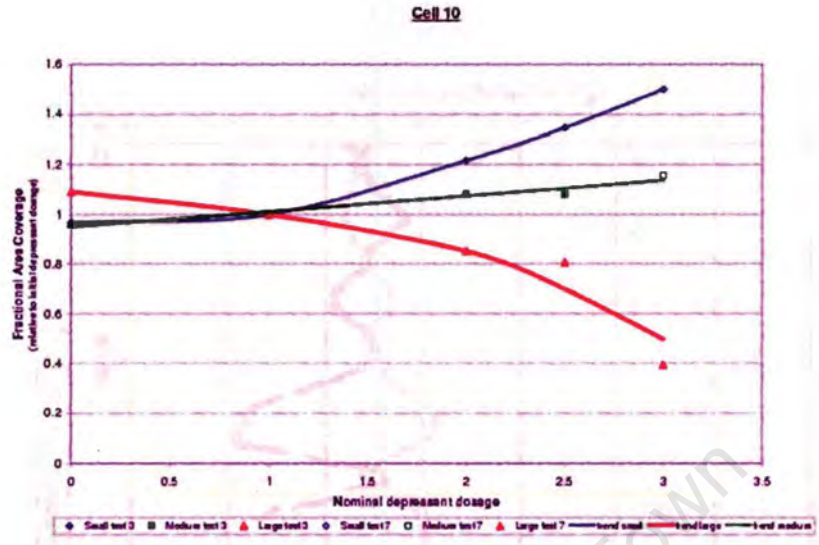



Figure 10.44: Plot of change in bubble size against relative depressant dosage for cell 10

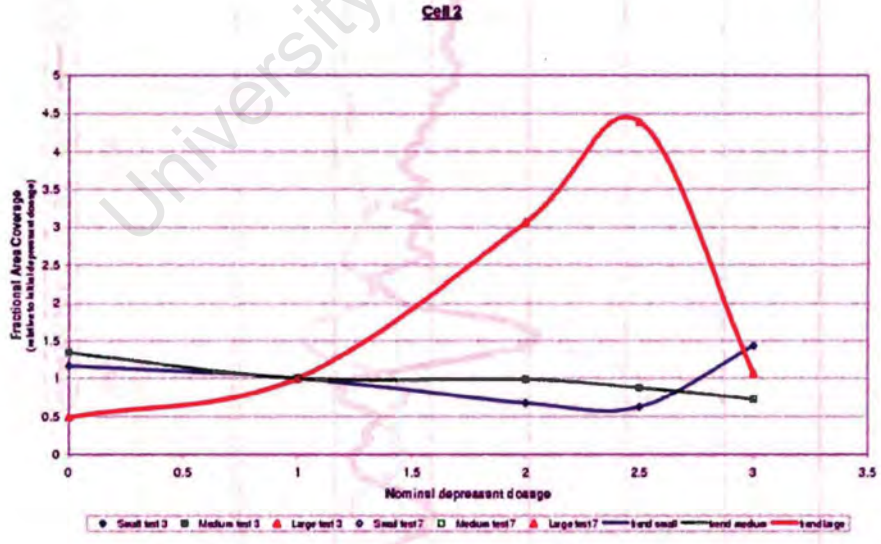



Figure 10.45: Plot of change in bubble size against relative depressant dosage for cell 2

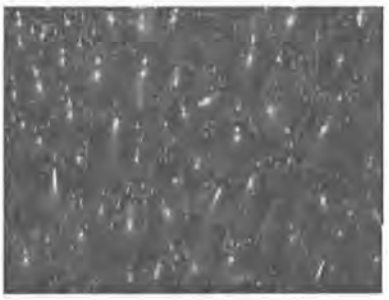


Figure 10.46: Froth from cell 10 - no depressant

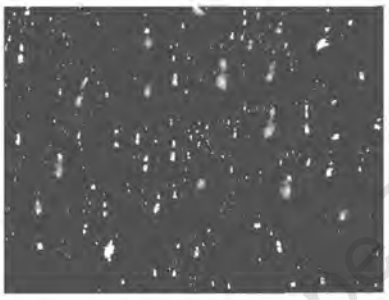


Figure 10.47: Froth from cell 10 - normal depressant

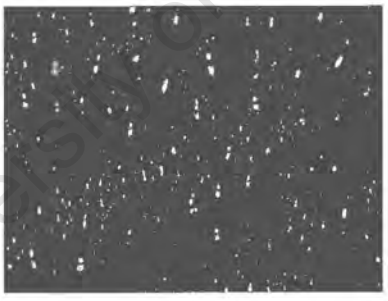


Figure 10.48: Froth from cell 10 - high depressant

Figure 10.49: Froth from cell 10 - very high depressant

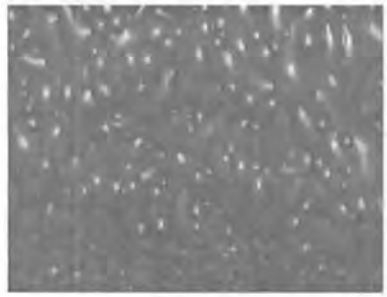


Figure 10.50: Froth from cell 2 - no depressant

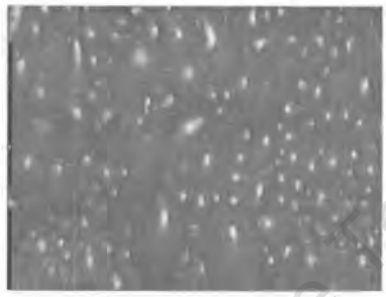


Figure 10.51: Froth from cell 2 - normal depressant

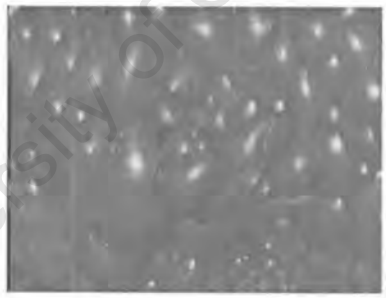


Figure 10.52: Froth from cell 2 - high depressant

Figure 10.53: Froth from cell 2 - very high depressant

## 10.6 Discussion of Testwork Results

The previous section has demonstrated that the on-line image analysis system is able to track changes in bubble size caused by changes in reagent dosage, and that the speed at which the analysis is performed is more than adequate for on-line, real-time application.

The ability of the on-line system to produce bubble size distributions is considered to be the most important factor that contributed to the success of the plant testwork, as it is these distributions, rather than the average bubble size, that change so remarkably in response to reagent variations. At the time of writing, no relationships have been elucidated between bubble shape information (i.e. ellipticity or circularity) and plant performance [64]. It is postulated that this may be because only average values are available for these parameters, as opposed to the distributions that are available for the bubble size. The colour information generated by the system was found to vary according to ambient lighting conditions, and as a result can not, at this stage, be considered to be a useful measurement.

The froth velocity data obtained from the system were found to be very noisy and of questionable value, which is most likely due to:

- The fact that the froth velocity was calculated from a single pair of frames. Using several consecutive frames and averaging the results appears to produce more reliable measurements [51].
- Vibration of the camera.
- A small error in the velocity calculation algorithm that was only discovered after the plant trials were completed.

It is somewhat unfortunate that the velocity data were of poor quality, as velocity information is often used by operators to gauge plant performance. However, as indicated in chapter 8, the emphasis in this research was the development of algorithms for measuring static parameters, with velocity and other dynamic parameters forming the basis of other research. It is envisaged that later versions of the image analysis system will contain more advanced and robust velocity measurement routines.

## 10.7 General Discussion

Sections 10.3 to 10.6 have shown that it is possible to obtain accurate, automatic segmentation of froth images, and to relate the results of this segmentation to changes in plant performance.

With reference to Sections 10.3 and 10.4, it should be clear that the segmentation algorithm identifies large bubbles accurately at the expense of small bubbles. (Note "large" and "small" are of course relative terms.) This is effectively illustrated by comparing the cumulative bubble size distributions for images I2 (Figure 10.11(b)) and I4 (Figure 10.25(b)). Based on this comparison, it could be said that the segmentation of image I2 is the "worst", while that of image I4 is the "best". Both I2 and I4 are composed of large bubbles, but image I2 also contains many very small bubbles that have not been accurately segmented by the algorithm.

This bias towards larger bubbles is almost entirely due to the preprocessing steps (Chapter 6) applied before the watershed segmentation is performed. Factors such as the number of iterations of anisotropic diffusion (Section 6.4.2), the size of the structuring element used for the grayscale dilation (Section 6.6) and the value of $h$ used during the extraction of the markers (Section 6.7) all contribute to the minimum size of bubbles that can be accurately segmented. It must be noted that any detected bubbles with an area of less than 20 pixels were discarded, as the geometric parameters for bubbles smaller than this size were found to be invalid in many cases.

It is of course possible for the segmentation algorithm to detect smaller bubbles than is currently the case, by modifying the preprocessing parameters. This will, however, generally be at the expense of accurate segmentation of larger bubbles. There does not appear to be any systematic method of determining when it is appropriate for small bubbles to be detected more accurately than larger bubbles, or *vice versa*. This decision may well be affected by the relative importance of the different-sized bubbles within the flotation process. This will depend largely on the type of froth that is being analysed. Additionally, the bubble size distribution for a single type of froth is known to vary with changing process conditions (Section 10.5).

Spatial extent (recall the discussion of this concept in Section 10.1) also plays a role in determining the quality of segmentation that can be achieved. Considering image I3 (Figures 10.15 to 10.17), the small bubbles that have not been correctly segmented are probably larger (in **real** terms) than some of the small to medium-sized bubbles which have been reasonably well segmented in the other images. This is because image I3 has a much larger spatial extent than any of the other images.

## 10.8 Summary

The following items have an effect on the quality of automatic segmentation that can be obtained from the algorithm described in this dissertation:

1. The type and amount of preprocessing performed on a froth image prior to application of the Watershed Transform
2. The bubble size distribution of the physical froth.
3. The spatial extent of the froth image.
4. The illumination of the froth in the flotation cell.

Item 2 cannot be considered a parameter that can be controlled in order to obtain accurate segmentation,as it is inherent to the flotation process. Items 3 and 4 will depend to a large extent on the configuration of the flotation cells and the positioning of the cameras relative to the cells. The choices made for item 1 will also be influenced by 2, 3 and 4, and may well vary from plant to plant and even from cell to cell. This is undesirable if the image analysis system is to be an easily installed and used on diverse flotation operations. A single method of preprocessing that performs reasonably well, without modification, on various froth types, seems to be preferable.

# Chapter 11

# Conclusions and Future Work



Based on the work described in this dissertation, the following conclusions can be made:

## 11.1 Conclusions

An algorithm for automatically segmenting flotation froth images has been developed. The Fast Watershed Transform has been used as the basis of the segmentation algorithm and, when combined with suitable image preprocessing methods, is able to segment froth images with a reasonable degree of accuracy.

The segmentation algorithm has been used as the basis of a machine vision system to perform on-line analysis and characterisation of flotation froths on a platinum concentrator plant. Testwork has shown that the system is able to identify and characterise variations in the appearance of a flotation froth caused by changes in reagent addition rates, and that the speed of analysis and characterisation is adequate for use as a real-time source of process information.

The ability of the system to generate bubble size distributions rather than merely a single average bubble size has been found to be of great importance, since it is the bubble size distribution of a froth that changes so markedly when the flotation process is perturbed.

## 11.2 Future Work

The algorithm and system as described in this dissertation form a sound basis for a robust and useful industrial machine vision system. The scope for future research in this field is vast, however in the short term, the following could be investigated:

- Further study of methods for froth image preprocessing.
- A detailed study of the effects of lighting variations on image segmentation, especially in the plant environment.
- Implementation of methods for measuring dynamic froth parameters.
- Investigation of other appropriate froth descriptors.
- Further development of the machine vision system, in order to provide a complete, self-contained "shrink-wrapped" product for deployment onto other concentrator plants.

# Bibliography

[1] C. ALDRICH, D. W. MOOLMAN, S.-J. BUNKELL, M. C. HARRIS, AND D. A. THERON. Relationship between surface froth features and process conditions in the batch flotation of a sulphide ore. *Minerals Engineering*, 10(11):1207–1218, 1997.

[2] R. ASHLEY. Flow measurement of mineral froths. Technical report, University of Cape Town, 1997.

[3] R. A. ASPLIN, N. SADR-KAZEMI, AND J. J. CILLIERS. The effect of surfactant concentration on batch flotation mineral flux and froth structure. *Minerals Engineering*, 11(3):257–269, 1998.

[4] A. W. BANFORD, Z. AKTAS, AND E. T. WOODBURN. Interpretation of the effect of froth structure on the performance of froth flotation using image analysis. *Powder Technology*, (98):61–73, 1998.

[5] J. BARRERA, G. J. F. BANON, AND R. D. A. LOTUFO. A mathematical morphology toolbox for the KHOROS system: Specifications for version 1.2b. Technical report, Universidade de São Paulo, 1994.

[6] S. BEUCHER AND F. MEYER. The morphological approach to segmentation: The watershed transform. In E. R. DOUGHERTY, editor, *Mathematical Morphology in Image Processing*. Marcel-Dekker, 1992.

[7] M. BEZUIDENHOUT, J. S. J. VAN DEVENTER, AND D. W. MOOLMAN. The identification of perturbations in a base metal flotation plant using computer vision of the froth structure. *Minerals Engineering*, 10(10):1057–1073, 1997.

[8] A. BLEAU, J. DE GUISE, AND A. ROBERT LEBLANC. A new set of fast algorithms for mathematical morphology : 1. Idempotent geodesic transforms. *CVGIP: Image Understanding*, 56(2):178–209, September 1992.

[9] A. BLEAU, J. DE GUISE, AND A. ROBERT LEBLANC. A new set of fast algorithms for mathematical morphology : 2. Identification of topographic features on grayscale images. *CVGIP: Image Understanding*, 56(2):210–229, September 1992.

[10] J. CANNY. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.

[11] CHACO PROJECT. Chaco website. http://www.hut.fi/Units/Control/projects/ChaCo/index.htm, 1999.

[12] CHACO PROJECT. ESPRIT LTR Project No. 24931 - ChaCo 12-Months Progress Report. http://www.hut.fi/Units/Control/projects/ChaCo/ChaCo12m.pdf, 1999.

[13] J. J. CILLIERS, R. A. ASPLIN, AND E. T. WOODBURN. Kinetic flotation modelling using froth imaging data. In J. S. LASKOWSKI AND E. T. WOODBURN, editors, *Frothing in Flotation II*. Gordon and Breach Science Publishers, 1998.

[14] A. CIPRIANO, M. GUARINI, A. SOTO, H. BRICENO, AND D. MERY. Expert supervision of flotation cells using digital image processing. In *Proceedings of the XX IMPC, Aaachen, 21-26 September*, volume 1, pages 281–292, 1997.

[15] A. CIPRIANO, M. GUARINI, R. VIDAL, A. SOTO, C. SEPULVEDA, D. MERY, AND H. BRISENO. A real-time visual sensor for supervision of flotation cells. *Minerals Engineering*, 11(6):489–499, 1998.

[16] R. C. CRIDA. *A Machine Vsion Approach to Rock Fragmentation Analysis*. PhD thesis, University of Cape Town, 1995.

[17] D. DE JONGH, D. W. MOOLMAN, AND M. VAN OLST. Current trends in automation and process control using computational intelligence techniques. In *Mineral Processing 1999 Abstracts*, 1999.

[18] P. L. DUNN, E. B. CRUZ, G. H. LUTTRELL, AND G. T. ADEL. Development of a video-based coal slurry ash analyzer. *Minerals & Metallurgical Processing*, 15(2):1–10, May 1998.

[19] J. J. FRANCIS. A sum square error based successive elimination algorithm for block matching. Technical report, University of Cape Town, 1998.

[20] J. J. FRANCIS AND G. DE JAGER. An investigation into the suitability of various motion estimation algorithms for froth imaging. In *Proceedings of the 1998 South African Symposium on Communications and Signal Processing - COMSIG '98*, pages 139–142, 1998.

[21] J. J. FRANCIS AND G. DE JAGER. Watershed segementation based motion estimation of froth image sequences. In *Proceedings of the Ninth Annual South African Workshop on Pattern Recognition*, pages 44–49, 1998.

[22] V. A. GLEMBOTSKII. *Flotation*. Primary Sources, New York, 1972.

[23] R. C. GONZALEZ AND P. WINTZ. *Digital Image Processing*. Addison-Wesley Publishing Company, Reading, Massechusetts, second edition, 1987.

[24] M. GUARINI, A. CIPRIANO, A. SOTO, AND A. GUESALAGA. Using image processing techniques to evaluate the quality of mineral flotation process. In *Proceedings of The International Conference on Signal Processing, Applications & Technology, Boston*, pages 1227–1231, 1995.

[25] L. B. HALES, R. A. YNCHAUSTI, AND D. F. SALMON. Image analysis of froth for improved flotation. In B. K. PAREKH AND J. D. MILLER, editors, *Advances in Flotation Technology*, pages 425–432. SME, 1999.

[26] R. M. HARALICK, S. R. STERNBERG, AND X. ZHUANG. Image analysis using mathematical morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(4):178–209, July 1987.

[27] J. M. HARGARVE AND S. T. HALL. Diagnosis of concentrate grade and mass flowrate in tin flotation from colour and surface texture analysis. *Minerals Engineering*, 10(6):613–621, 1997.

[28] J. M. HARGRAVE, G. J. BROWN, AND S. T. HALL. Use of fractal dimensions in describing froth structure. In *Proceedings of the XX IMPC, Aaachen, 21-26 September*, volume 3, pages 87–93, 1997.

[29] J. M. HARGRAVE, N. J. MILES, AND S. T. HALL. The use of grey level measurement in predicting coal flotation performance. *Minerals Engineering*, 9(6):667–674, 1996.

[30] R. C. HOCHREITER, D. C. KENNEDY, W. MUIR, AND A. I. WOOD. Platinum in South Africa. *J. S. Afr. Inst. Min. Metall.*, 85(6):165–185, June 1985.

[31] P. N. HOLTHAM, K. K. NGUYEN, AND P. M. CAMERON. Recent developments in the application of froth imaging techniques to the control of flotation plants. In *Mineral Processing 1999 Abstracts*, 1999.

[32] B. K. P. HORN. *Robot Vision*. The MIT Press, 1986.

[33] R. HYNDMAN. The problem with Sturges' rule for constructing histograms. Technical report, Monash University, 1995.

[34] K. J. IVES, editor. *The Scientific Basis of Flotation*. NATO Advanced Science Institute Series. Martinus Nijhoff, 1984.

[35] A. K. JAIN. *Fundamentals of Digital Image Processing*. Prentice-Hall International, 1989.

[36] P. JERMAN. Personal communication, 1999.

[37] J. KORDEK AND J. KULIG. The analysis of diffraction patterns of flotation froths as the basis of estimating the content of a useful component. In *Proceedings of the XX IMPC, Aachen, 21-26 September*, volume 3, pages 457–464, 1997.

[38] J. KORDEK AND S. LENCZOWSKI. The evaluation of flotation froth images by the optical method. In E. FORSSBERG, editor, *Proceedings of the XVI IMPC*, 1988.

[39] S. LENCZOWSKI AND J. GALAS. Froth image analysis in a flotation control system. In J. S. LASKOWSKI AND E. T. WOODBURN, editors, *Frothing in Flotation II*. Gordon and Breach Science Publishers, 1998.

[40] M. D. LEVINE. *Vision in Man And Machine*. McGraw-Hill Publishing Company, 1985.

[41] J. LIU. Application of the watershed boundary technique to automatically segment surface froth images. Master's thesis, University of Cape Town, 1995.

[42] P. MARAGOS AND R. W. SCHAFER. Morphological Filters — Part I: Their set-theoretic analysis and r elations to linear shift-invariant filters. *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-35:1153–1169, 1987.

[43] P. MARAGOS AND R. W. SCHAFER. Morphological systems for multidimensional image processing. *Proceedings of the IEEE*, 78:690–710, 1990.

[44] Z. T. MATHE, M. C. HARRIS, C. T. O'CONNOR, AND J.-P. FRANZIDIS. Review of froth modelling in steady state flotation systems. *Minerals Engineering*, 11(5):397–421, 1998.

[45] MINING TECHNOLOGY. Mining Technology. http://www.mining-technology.com/contractors/filtering/outokumpu_mintec/index.html, 1999.

[46] D. W. MOOLMAN, C. ALDRICH, G. P. J. SCHMITZ, AND J. S. J. VAN DEVENTER. The interrelationship between surface froth characteristics and industrial flotation performance. *Minerals Engineering*, 9(8):837–854, 1996.

[47] D. W. MOOLMAN, C. ALDRICH, J. S. J. VAN DEVENTER, AND D. J. BRADSHAW. The interpretation of flotation froth surfaces by using digital image analysis and neural networks. *Chem. Eng. Science*, 50(22):3501–3513, 1995.

[48] D. W. MOOLMAN, C. ALDRICH, J. S. J. VAN DEVENTER, AND W. W. STANGE. Digital image processing as a tool for on-line monitoring of froth in flotation plants. *Minerals Engineering*, 7(9):1149–1164, 1994.

[49] D. W. MOOLMAN, C. ALDRICH, J. S. J. VAN DEVENTER, AND W. W. STANGE. The classification of froth structures in a copper flotation plant by means of a neural net. *Int. J. Miner. Process.*, 43:193–208, 1995.

[50] D. W. MOOLMAN, J. J. EKSTEEN, C. ALDRICH, AND J. S. J. VAN DEVENTER. The significance of flotation froth appearance for machine vision control. *Int. J. Miner. Process*, 48:135–158, 1996.

[51] K. K. NGUYEN. *Flotation Froth Characterisation by using Vision Technology*. PhD thesis, University of Queensland, 1998.

[52] K. K. NGUYEN AND P. HOLTHAM. The application of pixel tracing techniques in the flotation process. In *Proceedings of DICTA '97 - Auckland, New Zealand*, 1997.

[53] K. K. NGUYEN AND A. THORNTON. The application of texture based image analysis techniques in froth flotation. In *Proceedings of DICTA '95 - Brisbane, Australia*, 1995.

[54] J. M. OESTREICH, W. K. TOLLEY, AND D. A. RICE. The development of a color sensor system to measure mineral compositions. *Minerals Engineering*, 8(1/2):31–39, 1995.

[55] W. B. PENNEBAKER AND J. L. MITCHELL. *JPEG Still Image Compression Standard*. Van Nostrand Reinhold, 1993.

[56] P. PERONA AND J. MALIK. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, July 1990.

[57] J. C. RUSS. *The Image Processing Handbook*. CRC Press, 1995.

[58] N. SADR-KAZEMI AND J. J. CILLIERS. An image processing algorithm for measurement of flotation froth bubble size and shape distributions. *Minerals Engineering*, 10(10):1075–1083, 1997.

[59] M. R. SPIEGEL. *Schaum's Outile Series - Theory and Problems of Statistics*. McGraw-Hill Book Company, 1961.

[60] S. R. STERNBERG. Grayscale morphology. *Computer Vision, Graphics, and Image Processing*, (35):333–355, 1986.

[61] C. SUN AND W. G. WEE. Neighbouring gray level dependence matrix for texture classification. *Computer Vision, Graphics and Image Processing*, (23):341–352, 1983.

[62] C. G. SWEET. Amplats froth imaging project plant based 'continuous' test programme. Experimental Proposal, University of Cape Town, 1998.

[63] C. G. SWEET. The use of froth visualisation to monitor metallurgical performance in the flotation of complex sulphide ores. Project Proposal, University of Cape Town, 1998.

[64] C. G. SWEET. Personal communications, 1999.

[65] P. J. SYMONDS. The investigation of the characterisation of flotation froths and design of a machine vision system for monitoring the operation of a flotation cell ore concentrator. Master's thesis, University of Cape Town, 1992.

[66] J. S. J. VAN DEVENTER, M. BEZUIDENHOUT, AND D. W. MOOLMAN. On-line visualisation of flotation performance using neural computer vision of the froth structure. In *Proceedings of the XX IMPC, Aaachen, 21-26 September*, volume 1, pages 315–326, 1997.

[67] E. VENTURA-MEDINA. Personal communications, 1998, 1999.

[68] J. VILLENEUVE, J.-C. GUILLANEAU, AND M. V. DURANCE. Flotation modelling: A wide range of solutions for solving industrial problems. *Minerals Engineering*, 8(4/5):409–420, 1995.

[69] L. VINCENT. Morphological algorithms. In E. R. DOUGHERTY, editor, *Mathematical Morphology in Image Processing*. Marcel-Dekker, 1992.

[70] L. VINCENT. Morphological grayscale reconstruction in image processing: Applications and efficient algorithms. *IEEE Transactions on Image Processing*, 2(2):176–201, 1993.

[71] L. VINCENT AND P. SOILLE. Watersheds in digital spaces: An efficeint algorithm based on immersion simulations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, 1991.

[72] A. R. WEEKS. *Fundamentals of Electronic Image Processing*. SPIE Press, 1996.

[73] B. A. WILLS. *Mineral Processing Technology*. Butterworth-Heinemann, 6th edition, 1997.

[74] E. T. WOODBURN, L. G. AUSTIN, AND J. B. STOCKTON. A froth based flotation kinetic model. *Trans IChemE*, 72(A):211–226, 1994.

[75] E. T. WOODBURN, J. B. STOCKTON, AND D. J. ROBBINS. Vison-based characterisation of 3 phase froths. *International Colloquium: Developments in Froth Flotation, SAIMM, Gordon's Bay, South Africa*, 1:1–30, 1989.

[76] B. A. WRIGHT. An investigation into the use of the watershed transform for the real-time segmentation of flotation froth images. Undergraduate Project Report, University of Cape Town, 1997.

[77] P. YAO AND R. C. LEINECKER. *Visual C++ 5 Bible*. IDG Books, 1997.