

AUTOMATIC DETECTION AND CLASSIFICATION OF LEUKAEMIA CELLS

A thesis submitted for the degree of Doctor of Philosophy

By

Waidah Ismail

Department of Information Systems, Computing and
Mathematics, Brunel University

June 2012

ABSTRACT

Today, there is a substantial number of software and research groups that focus on the development of image processing software to extract useful information from medical images, in order to assist and improve patient diagnosis. The work presented in this thesis is centred on processing of images of blood and bone marrow smears of patients suffering from leukaemia, a common type of cancer. In general, cancer is due to aberrant gene expression, which is caused by either mutations or epigenetic changes in DNA. Poor diet and unhealthy lifestyle may trigger or contribute to these changes, although the underlying mechanism is often unknown. Importantly, many cancer types including leukaemia are curable and patient survival and treatment can be improved, subject to prompt diagnosis.

In particular, this study focuses on Acute Myeloid Leukaemia (AML), which can be of eight distinct types (M0 to M7), with the main objective to develop a methodology to automatically detect and classify leukaemia cells into one of the above types. The data was collected from the Department of Haematology, Universiti Sains Malaysia, in Malaysia. Three main methods, namely Cellular Automata, Heuristic Search and classification using Neural Networks are facilitated. In the case of Cellular Automata, an improved method based on the 8-neighbourhood and rules were developed to remove noise from images and estimate the radius of the potential blast cells contained in them. The proposed methodology selects the starting points, corresponding to potential blast cells, for the subsequent seeded heuristic search. The Seeded Heuristic employs a new fitness function for blast cell detection. Furthermore, the WEKA software is utilised for classification of blast cells and hence images, into AML subtypes. As a result accuracy of 97.22% was achieved in the classification of blasts into M3 and other AML subtypes.

Finally, these algorithms are integrated into an automated system for image processing. In brief, the research presented in this thesis involves the use of advanced computational techniques for processing and classification of medical images, that is, images of blood samples from patients suffering from leukaemia.

TABLE OF CONTENTS

| | |
|--|-----------|
| CHAPTER 1: INTRODUCTION | 1 |
| <hr/> | |
| 1.1 Problem Statement | 4 |
| 1.2 Objectives | 7 |
| 1.3 Benefits | 8 |
| 1.4 Roadmap | 8 |
| 1.5 Summary | 13 |
| CHAPTER 2: LITERATURE REVIEW | 14 |
| <hr/> | |
| 2.1 Introduction | 14 |
| 2.2 Leukaemia | 14 |
| 2.2.1 Blood | 15 |
| 2.2.2 White Blood Cells | 17 |
| 2.2.3 Types of Leukaemia | 19 |
| 2.2.4 Flowchart of patients admitted with leukaemia | 22 |
| 2.2.5 Acute Leukaemia | 27 |
| 2.2.6 Acute Myeloid Leukaemia | 28 |
| 2.2.7 Laboratory Diagnosis of Acute Myeloid Leukaemia | 30 |
| 2.2.7.1 Peripheral Blood Film | 30 |
| 2.2.7.2 Bone Marrow | 31 |
| 2.2.8 Classification of Acute Myeloid Leukaemia (M0 to M7) | 33 |
| 2.2.8.1 Cytogenetics | 33 |

| | | |
|---|---|---------------|
| 2.2.8.2 | Immunophenotyping | 34 |
| 2.2.9 | Treatment | 35 |
| 2.2.10 | Prognosis | 36 |
| 2.3 | Image Processing | 36 |
| 2.3.1 | History of Digital Image Processing | 40 |
| 2.3.2 | Data Collection | 44 |
| 2.3.3 | Image Segmentation | 45 |
| 2.3.4 | Thresholding | 46 |
| 2.3.5 | Otsu | 46 |
| 2.3.6 | Otsu Notation | 49 |
| 2.4 | Cellular Automata | 49 |
| 2.4.1 | Cellular Automata Concepts | 50 |
| 2.4.2 | Manhattan Distance | 52 |
| 2.4.3 | Euclidean Distance | 54 |
| 2.5 | Artificial Intelligent | 54 |
| 2.5.1 | Machine Learning | 56 |
| 2.5.2 | Classification | 57 |
| 2.5.3 | Artificial Neural Network | 59 |
| 2.5.4 | Multilayer Perceptron | 64 |
| 2.6 | Heuristic Search | 65 |
| 2.6.1 | Heuristic Search concepts | 65 |
| 2.6.2 | Hill Climbing (HC) | 69 |
| 2.6.3 | Simulated Annealing (SA) | 72 |
| 2.6.4 | Genetic Algorithms (GA) | 76 |
| 2.6.5 | Genetic Algorithms versus Traditional Methods | 81 |
| 2.7 | Summary | 82 |
| CHAPTER 3: HEURISTIC RANDOM SEARCH | | 84 |

| | | |
|------------|----------------------|-----------|
| 3.1 | Introduction | 84 |
| 3.2 | Previous Work | 85 |
| 3.2.1 | Otsu | 85 |
| 3.2.2 | Heuristic Search | 90 |

| | | |
|------------|--|------------|
| 3.3 | Work Process | 91 |
| 3.4 | Methods | 92 |
| 3.4.1 | Otsu | 92 |
| 3.4.2. | Determining the Fitness Function | 92 |
| 3.4.3 | Fitness Function Notations | 93 |
| 3.4.4 | Fitness Function for Hill Climbing and Simulated Annealing | 93 |
| 3.4.4.1 | Hill Climbing and Simulated Annealing Notation | 94 |
| 3.4.4.2 | Algorithm for Hill Climbing | 94 |
| 3.4.4.3 | Algorithm for Simulated Annealing | 95 |
| 3.4.5 | Circles Overlap Similarity Metric | 97 |
| 3.4.5.1 | Circles Overlap Similarity Metric Notation | 97 |
| 3.4.5.2 | Algorithm for Circles Overlap Similarity Metric | 99 |
| 3.4.6 | Fitness Function for Genetic Algorithm | 99 |
| 3.4.6.1 | Genetic Algorithm Notation | 100 |
| 3.4.6.2 | Algorithm for Genetic Algorithm | 100 |
| 3.5 | Results | 101 |
| 3.5.1 | Choice of Fitness Function | 101 |
| 3.5.2 | Comparison between Hill Climbing, Simulated Annealing and Genetic Algorithms | 105 |
| 3.6 | Summary | 111 |

CHAPTER 4: CELLULAR AUTOMATON FILTERING **113**

| | | |
|------------|---|------------|
| 4.1 | Introduction | 113 |
| 4.2 | Previous Work | 114 |
| 4.3 | Work Process | 115 |
| 4.4 | Methods | 116 |
| 4.4.1 | Cellular Automata | 116 |
| 4.4.1.1 | Cellular Automata Notation | 116 |
| 4.4.1.2 | Algorithm for Cellular Automata | 118 |
| 4.4.2. | Filtering of Cellular Automata | 119 |
| 4.4.2.1 | Algorithms for filtering of Cellular Automata | 120 |
| 4.5 | Results | 121 |

| | | |
|------------|-----------------------------------|------------|
| 4.5.1 | Otsu | 122 |
| 4.5.2 | Cellular Automata | 123 |
| 4.5.3 | Cellular Automata after filtering | 123 |
| 4.6 | Summary | 127 |

CHAPTER 5: COORDINATE DETECTION AND COLOUR 128

IMAGE CLASSIFICATION

| | | |
|------------|---|------------|
| 5.1 | Introduction | 128 |
| 5.2 | Previous Work | 129 |
| 5.2.1 | Image classification | 129 |
| 5.2.3 | Duplicate Coordinates | 130 |
| 5.3 | Work Process | 132 |
| 5.4 | Method | 132 |
| 5.4.1 | Finding Starting Coordinates | 133 |
| 5.4.1.1 | Finding Starting Coordinates Notation | 133 |
| 5.4.1.2 | Algorithm for Finding Starting Coordinate | 134 |
| 5.4.2 | Image Classification | 134 |
| 5.4.2.1 | Image Classification Notation | 134 |
| 5.4.2.2 | Algorithm for Image Classification | 135 |
| 5.4.3 | Duplicate Coordinates | 136 |
| 5.4.3.1 | Duplicate Coordinates Notation | 136 |
| 5.4.3.2 | Algorithm for Duplicate Coordinates | 137 |
| 5.5 | Results | 137 |
| 5.5.1 | Finding the Starting Coordinates | 137 |
| 5.5.2 | Image Classification | 138 |
| 5.5.3 | Duplicate Coordinates | 145 |
| 5.6 | Summary | 148 |

CHAPTER 6: HEURISTIC SEARCH SEEDED **150**

| | | |
|------------|---|------------|
| 6.1 | Introduction | 150 |
| 6.2 | Previous Work | 151 |
| 6.3 | Work Process | 152 |
| 6.4 | Fitness Function | 153 |
| 6.4.1 | Fitness Function for Hill Climbing, Simulated Annealing and Genetic Algorithm | 153 |
| 6.4.2 | Colour Images Fitness Function for Hill Climbing, Simulated Annealing and Genetic Algorithm | 154 |
| 6.4.2.1 | Colour Fitness Function Notation | 154 |
| 6.5 | Result | 154 |
| 6.5.1 | Comparison between Hill Climbing, Simulated Annealing and Genetic Algorithm | 155 |
| 6.5.2 | Processing using colour image | 162 |
| 6.5.3 | Comparison between Hill Climbing, Simulated Annealing and Genetic Algorithm for Seeded with twenty images | 164 |
| 6.5.4 | Circles Overlap Similarity Metric | 170 |
| 6.5.5 | Simulated Annealing Real Images | 171 |
| 8.6 | Computational Complexity | 177 |
| 8.4.1 | Otsu | 178 |
| 8.4.2 | Cellular Automata | 178 |
| 8.4.3 | Heuristic Search | 178 |
| 8.4.4 | Overall Complexity | 179 |
| 6.6 | Summary | 179 |

CHAPTER 7: MULTILAYER PERCEPTRON FOR THE **181**
CLASSIFICATION OF LEUKAEMIA CELLS

| | | |
|------------|----------------------|------------|
| 7.1 | Introduction | 181 |
| 7.2 | Previous Work | 182 |
| 7.3 | Work Process | 184 |

| | | |
|------------|---|------------|
| 7.4 | Methods | 185 |
| 7.4.1 | Sub Images | 186 |
| 7.4.2 | Finding the “best” classifier | 187 |
| 7.4.3 | Finding the “best” method | 188 |
| 7.4.4 | Real Data | 189 |
| 7.5 | Result | 190 |
| 7.5.1 | Sub Images | 190 |
| 7.5.2 | Finding the “best” classifier | 191 |
| 7.5.3 | Further testing for finding the “best” classifier | 192 |
| 7.5.4 | Finding the “best” method for HC, SA and GA | 194 |
| 7.5.5 | Real Data | 195 |
| 7.6 | Summary | 199 |

| | |
|--|------------|
| CHAPTER 8: CONCLUSION AND FUTURE WORK | 201 |
|--|------------|

| | | |
|------------|----------------------------------|------------|
| 8.1 | Introduction | 201 |
| 8.2 | Achievement | 205 |
| 8.3 | Contribution to knowledge | 206 |
| 8.4 | Limitation | 207 |
| 8.5 | Future Research | 207 |
| 8.6 | Process learn of the PhD | 209 |

| | |
|-------------------|------------|
| REFERENCES | 210 |
|-------------------|------------|

LIST OF FIGURES

| | | <i>Pages</i> |
|----------------------|---|--------------|
| Figure 1.1: | Random Heuristic Search | 10 |
| Figure 1.2 | Roadmap | 12 |
| Figure 2.1 | Leukaemia 10 year relative survival rates | 22 |
| Figure 2.2(a) | Steps to confirm Acute Myeloid Leukaemia (AML) – M3 (Step 1 to Step 8) | 23 |
| Figure 2.2(b) | Steps to confirm Acute Myeloid Leukaemia (AML) – M3 (Step 9 to Step 16) | 24 |
| Figure 2.3 | Blood Film | 31 |
| Figure 2.4 | Bone Marrow sample in M2 AML case | 32 |
| Figure 2.5 | Blood taken from Bone Marrow | 32 |
| Figure 2.6 | Normal Karyotype from a Cytogenetic Analysis | 34 |
| Figure 2.7 | Image digitisation | 38 |
| Figure 2.8 | Image in Black and White | 39 |
| Figure 2.9 | Image in Red, Green and Blue | 40 |
| Figure 2.10 | A digital picture produced in 1921 from coded tape by a telegraph printer with special type faces | 41 |
| Figure 2.11 | Example of a microscope | 42 |
| Figure 2.12 | Example of microscope blood images with 100x magnification | 43 |
| Figure 2.13 | Example of microscopic blood image with 40x magnification | 43 |
| Figure 2.14 | Example of a bone marrow image of AML patient (zoom 40x) | 44 |
| Figure 2.15 | (a) Real Image (b) Processed with Otsu Method without threshold (c) Otsu method with threshold | 48 |
| Figure 2.16 | Von Neumann Neighbourhood | 50 |
| Figure 2.17 | Moore Neighbourhood | 50 |
| Figure 2.18 | Manhattan Distance between (x) and (y)(red colour) | 53 |
| Figure 2.19 | Illustration of Euclidean Distance | 54 |
| Figure 2.20 | A neuron | 60 |
| Figure 2.21 | The architecture for Neural Network | 62 |
| Figure 2.22 | The classical process of decision making in an optimisation model | 68 |
| Figure 2.23 | Search space concepts illustration | 69 |
| Figure 2.24 | HC convergence graph | 71 |

| | | |
|-----------------------|---|-----|
| Figure 2.25 | SA convergence graph | 74 |
| Figure 2.26 | Example of a chromosome | 77 |
| Figure 2.27 | Example of crossover | 79 |
| Figure 2.28 | GA convergence graph | 80 |
| Figure 3.1 | Successful topology method for non-overlap blasts cells (a) – (c) | 88 |
| Figure 3.2 | Unsuccessful topology method for overlapping blasts cells (a) – (c) | 90 |
| Figure 3.3 | Work Process for Random Search | 91 |
| Figure 3.4 | Overlapping scenario | 97 |
| Figure 3.5 | Internalised circles | 98 |
| Figure 3.6 | Intersection diagram | 98 |
| Figure 3.7(a) | Coordinate for x,y,r for ten cases images (Image 1 – Image 6) | 102 |
| Figure 3.7(b) | Coordinate for x,y,r for ten cases images (Image 7 – Image 10) | 103 |
| Figure 3.8(a) | Hill Climbing Random Heuristic Search (Image 1 – Image 6) | 107 |
| Figure 3.8(b) | Hill Climbing Random Heuristic Search (Image 7 – Image 10) | 108 |
| Figure 3.9(a) | Simulated Annealing Random Heuristic Search (Image 5 – Image 10) | 108 |
| Figure 3.9(b) | Simulated Annealing Random Heuristic Search (Image 1 – Image 4) | 109 |
| Figure 3.10(a) | Genetic Algorithm Random Heuristic Search (Image 1 – Image 2) | 109 |
| Figure 3.10(b) | Genetic Algorithm Random Heuristic Search (Image 3 – Image 10) | 110 |
| Figure 3.11 | Convergence graph for HC, SA and GA | 111 |
| Figure 4.1 | Work Process – Chapter 4 | 116 |
| Figure 4.2 | Distance using Manhattan Cellular Automata | 118 |
| Figure 4.3 | AML Images | 121 |
| Figure 4.4 | Examples of images processed by the Otsu Method | 122 |
| Figure 4.5 | Examples of images processed by CA | 123 |
| Figure 4.6(a) | Images where CA filtering was unsuccessful (M1 – Image 18) | 124 |
| Figure 4.6(b) | Images where CA filtering was unsuccessful (M5 – Image 34) | 125 |
| Figure 4.7(a) | Examples of successful application of CA filtering (M1) | 125 |
| Figure 4.7(b) | Examples of successful application of CA filtering (M2, M3, M5) | 126 |
| Figure 5.1 | Traditional Venn Diagram | 131 |
| Figure 5.2 | Work Process for chapter 5 | 132 |
| Figure 5.3 | Finding the starting coordinates | 133 |
| Figure 5.4 | Colours corresponding to red, blast cells and background | 135 |
| Figure 5.5 | New coordinate diagram | 136 |
| Figure 5.6 | Examples of blast cells coordinate detection | 138 |
| Figure 5.7(a) | Wrong classification to blast cells (M2) | 139 |

| | | |
|-----------------------|--|-----|
| Figure 5.7(b) | Wrong classification to blast cells (M5) | 140 |
| Figure 5.8(a) | Examples of successful Image Clustering (M1) | 140 |
| Figure 5.8(b) | Examples of successful Image Clustering (M2, M3, M5) | 141 |
| Figure 5.9 | Summary of Image Clustering Efficiency | 142 |
| Figure 5.10 | Wrong classification of blast cells | 143 |
| Figure 5.11 | Wrong classification of blast cells from detecting coordinates | 143 |
| Figure 5.12 | Potential blast cells classified to pink | 144 |
| Figure 5.13 | Overlapping blast cells | 144 |
| Figure 5.14 | Blast cells undetected | 145 |
| Figure 5.15 | Summary of Duplicate Coordinates efficiency | 146 |
| Figure 5.16 | Internalised method applied | 147 |
| Figure 5.17 | Intersection circles duplicate | 147 |
| Figure 5.18 | Same locate duplicate coordinates | 148 |
| Figure 6.1 | Work Process for Seeded Heuristic Search | 152 |
| Figure 6.2 | Work Process for Colour image for Random and Seeded Heuristic Search | 153 |
| Figure 6.3 | Comparison between Seeded HC, SA and GA seeded | 156 |
| Figure 6.4 | Images corresponding to the highest fitness value | 157 |
| Figure 6.5(a) | Hill Climbing Seeded Heuristic Search (Image 1 – Image 8) | 158 |
| Figure 6.5(b) | Hill Climbing Seeded Heuristic Search (Image 9 – Image 10) | 159 |
| Figure 6.6(a) | Simulated Annealing Seeded Heuristic Search (Image 1 – Image 6) | 159 |
| Figure 6.6(b) | Simulated Annealing Seeded Heuristic Search (Image 7 – Image 10) | 160 |
| Figure 6.7(a) | Genetic Algorithm Seeded Heuristic Search (Image 1 – Image 4) | 160 |
| Figure 6.7(b) | Genetic Algorithm Seeded Heuristic Search (Image 5 – Image 10) | 161 |
| Figure 6.8 | Random HC, SA and GA for colour image | 163 |
| Figure 6.9 | Seeded HC, SA and GA for colour image | 164 |
| Figure 6.10 | Fitness values for HC, SA and GA (20 images) | 166 |
| Figure 6.11(a) | Seeded HC, SA and GA (Image 1 – Image 12) | 167 |
| Figure 6.11(b) | Seeded HC, SA and GA (Image 13 – Image 20) | 168 |
| Figure 6.12(a) | Seeded SA (Image 11 – Image 12) | 168 |
| Figure 6.12(b) | Seeded SA (Image 13 – Image 20) | 168 |
| Figure 6.13 | Result before performing seeded SA | 172 |
| Figure 6.14 | Result after performing seeded SA | 173 |
| Figure 6.15 | Result from Seeded Heuristic Search seeded before and after | 174 |
| Figure 6.16 | Targeting in between blast cells before and targeting full blast cells after performing heuristic search | 174 |
| Figure 6.17 | Example of M2 subtype before and after heuristic search | 175 |
| Figure 6.18 | Before heuristic search targeting blast cells but after heuristic search targeting red blood cells | 175 |

| | | |
|----------------------|--|-----|
| Figure 6.19 | Before heuristic search the circle is targeting blast cell and red blood cells, but after heuristic search it is only targeting the blast cell | 176 |
| Figure 6.20 | Before heuristic search the circle is targeting blast cells, but after heuristic search it is targeting blast and red blood cells | 176 |
| Figure 6.21 | Example of M5 subtype before and after heuristic search | 177 |
| Figure 7.1 | Work Process: sub images - classification | 184 |
| Figure 7.2 | Finding the “best” method for classification | 185 |
| Figure 7.3 | Classification for M3 subtype with other subtype using multi-binary classification mode | 186 |
| Figure 7.4 | Real Image of M3 subtype | 190 |
| Figure 7.5 | Image segmentation of M3 subtype | 190 |
| Figure 7.6 | Example of wrong classification of M5 subtype | 199 |
| Figure 8.1(a) | Example of processing steps in Seeded Heuristic Search | 204 |
| Figure 8.1(b) | Example of processing steps in Seeded Heuristic Search - Classification | 205 |

LIST OF TABLES

| | | <i>Pages</i> |
|----------------------|---|--------------|
| Table 1.1(a) | FAB classification of AML (M0 – M3) | 5 |
| Table 1.1(b) | FAB classification of AML (M4 – M7) | 6 |
| Table 2.1(a) | Major elements of Blood (Red blood cells, White blood cells, platelets) | 16 |
| Table 2.1(b) | Major elements of Blood (Plasma) | 17 |
| Table 2.2 (a) | White Blood Cells (Neutrophil, Eosinophil, Basophil) | 18 |
| Table 2.2 (b) | White Blood Cells (Monocyte and Lymphocyte) | 19 |
| Table 2.3 (a) | Types of Leukaemia (ALL and AML) | 20 |
| Table 2.3 (b) | Types of Leukaemia (CML and CLL) | 21 |
| Table 2.4 | Leukaemia cases in UK for 2007 | 22 |
| Table 2.5 (a) | Analytical description of each step in Figure 2.2 step (1) – (3) | 25 |
| Table 2.5 (b) | Analytical description of each step in Figure 2.2 step (4) – (14) | 26 |
| Table 2.6 | WHO classification of AML | 29 |
| Table 2.7 | Data collection for individual subtypes | 45 |
| Table 2.8 | KAPPA Table | 58 |
| Table 3.1 | Test method for ten test cases images | 104 |
| Table 3.2 | Summary of Fitness Functions Evaluation | 105 |
| Table 3.3 | Comparison between HC, SA and GA | 106 |
| Table 4.1 | Mature White Blood Cells and their sizes | 120 |
| Table 4.2 | Percentage of successful converted images using CA Filtering method | 124 |
| Table 5.1 | RGB values for purple and pink for image classification | 134 |
| Table 5.2 | Summary of Image Classification efficiency overall | 139 |
| Table 5.3 | Summary of Image Classification efficiency | 142 |
| Table 5.4 | Summary of Duplicate Coordinates | 146 |
| Table 6.1(a) | Comparison between Seeded HC, SA and GA (Image 1 – Image 9) | 155 |
| Table 6.1(b) | Comparison between Seeded HC, SA and GA (Image 10) | 156 |

| | | |
|---------------------|--|-----|
| Table 6.2 | Comparison between Seeded and Random HC, SA and GA methods | 162 |
| Table 6.3 | Comparison between random HC, SA and GA for colour images in random | 163 |
| Table 6.4 | Comparison between seeded HC, SA and GA for colour images in seeded | 163 |
| Table 6.5(a) | Fitness values for HC, SA and GA (Image 1 – Image 18) | 165 |
| Table 6.5(b) | Fitness values for HC, SA and GA (Image 19 – Image 20) | 166 |
| Table 6.6(a) | Result of the circle overlap similarity metric for HC and SA (Image 1 – Image 9) | 170 |
| Table 6.6(b) | Result of the circle overlap similarity metric for HC and SA (Image 10 – Image 20) | 171 |
| Table 6.7 | Result before performing application seeded SA | 172 |
| Table 6.8 | Result after performing application seeded SA | 173 |
| Table 7.1 | Attributes for classifier | 189 |
| Table 7.2(a) | Summary results from WEKA (Classification Bayes, Function and Lazy) | 191 |
| Table 7.2(b) | Summary results from WEKA (Classification Meta, Rules and Tree) | 192 |
| Table 7.3 | Summary of distribution of the images | 192 |
| Table 7.4 | Summary of further testing for 317 images | 193 |
| Table 7.5 | Summary of HC, SA and GA for twenty images | 195 |
| Table 7.5 | Summary of using SA in the Figure 7.3 | 197 |
| Table 7.6 | Table for the confusion matrix | 198 |

LIST OF ALGORITHMS

| | | <i>Pages</i> |
|-----------------------|---------------------|--------------|
| Algorithms 2.1 | HC Algorithm | 72 |
| Algorithms 2.2 | SA Algorithm | 75 |
| Algorithms 2.3 | GA Algorithm | 81 |
| Algorithms 3.1 | HC Random Algorithm | 95 |
| Algorithms 3.2 | SA Random Algorithm | 96 |

LIST OF APPENDICES

| | <i>Pages</i> |
|-------------------|---|
| Appendix A | Ethical Approval 223 |
| Appendix B | Proof for Cellular Automata 224 |
| Appendix C | Full Results in Detecting Coordinates 226 |
| Appendix D | Full Results in Duplicate Coordinates 242 |
| Appendix E | Full Results in Heuristic Search 250 |
| Appendix F | Summary of the test result from WEKA 259 |
| Appendix G | Execution time for start and end time for twenty images 263 |
| Appendix H | Multilayer Perceptron for sigmoid and nodes used in WEKA for 51 sub-images with 10 folds validation 264 |
| Appendix I | User Interface 275 |

LIST OF NOTATION

| Notation | Descriptions |
|-----------------------------|---|
| B | Black pixel in the image |
| C₀ | Background pixel |
| C₁ | Foreground pixel |
| E | Weight for Multilayer Perceptron |
| G | New image after Cellular Automata filtering |
| H | Hidden layer for input |
| I | Colour images |
| K | Pink that represent red blood cells |
| L | Grey Level images |
| M | Hidden layer for output |
| N | Total number of Pixels |
| <i>g</i> | Pixel in the image |
| P | Purple that represent leukaemia cells |
| q | Number of nodes |
| R | Radius in the image |
| T | Threshold for greyscale |
| w | White pixel in the image |
| X | Coordinate x in the image |
| Y | Coordinate y in the image |
| b | Blue in the colour image |
| d | The different size for white blood cells |
| g | Green in the colour image |
| i | Number of Circle |
| r | Red in the colour image |
| κ | Black and White images after converted into Otsu method |
| Λ | A set of matrix after performing Cellular Automata |
| ω | White blood cells |
| \square | Rules of movement |
| η | Change of point for Cellular Automata |
| X_{max} | The number of pixels in the x-axis |
| Y_{max} | The number of pixels in the y-axis |

ACKNOWLEDGMENTS

A special thanks to all the people who supported me through my years as a PhD student, including my family, friends and colleagues in the United Kingdom and Malaysia. First and foremost, I offer my sincerest gratitude to both my supervisors, Dr. Stephen Swift and Dr. Annette Payne who have supported me throughout my thesis with their patience and knowledge. I would never have completed this without your guidance. I owe my deepest gratitude to Dr. Rosline Hassan, Ms. Selamah Ghazali and Mrs Narisah from the Department of Haematology, Universiti Sains Malaysia, for supplying me with the medical images needed, as well as their guidance and expertise. I would like to thank Dr. Stelios, my friends and lecturers from the Centre for Intelligent Data Analysis (CIDA) lab at Brunel University, who were always willing to answer any questions, advice me and sometimes just tell me to get on with it and stop complaining. Finally, I would like to express my gratitude to the Institute of Higher Education of Malaysia and the Universiti Sains Islam Malaysia (USIM) for providing me with funding to cover my fees and cost of living while working on my PhD.

Thank You.

SUPPORTING PUBLICATIONS

Conference Papers

- Waidah, I, Rosline, H, Payne, A & Swift, S, 6th July 2011 “Classifying of Blast cells from Promyelocytic Leukaemia cells (AML M3) blood samples using Neural Network for clinical decision support. IDAMAP 2011: Intelligent Data Analysis in Biomedicine and Pharmacology (IDAMAP 2011), Lake Bled, Slovenia.
<http://www.biolab.si/idamap/idamap2011/>
- Waidah, I, Rosline, H & Swift, S, 18th May, 2010 – 21st May, 2010 “Detecting Leukaemia (AML) blood cells using Cellular Automata and Heuristic Search” Intelligent Data Analysis (IDA 2010) University of Arizona, USA
<http://hiit.fi/ida2010/>
- Waidah, I, Rosline, H & Swift, S, 5th May – 6th May 2010, “Can Heuristic Search techniques locate Leukemia cell?” Research Student Poster Conference, Brunel University. – Poster
<http://www.brunel.ac.uk/courses/pg/graduateschool/poster>
- Waidah, I, Rosline, H & Swift, S, 16th April, 2010 – 18th April, 2010 “Detecting Leukaemia (AML) blood cells using Genetic Algorithms” 2nd Student Conference Operational Research (SCOR 2010), University Nottingham, UK
<http://www.scor2010.co.uk/>

Chapter 1

INTRODUCTION

The aim of this research is to automate the detection and classification of leukaemia cells. In the scientific language, leukaemia cells are referred to as blast cells. There are two types of Acute Leukaemia, Acute Lymphoblastic leukaemia (ALL) and Acute Myeloid Leukaemia (AML). ALL classification has been previously implemented by researchers in Malaysia. This thesis focuses on Acute Myeloid Leukaemia(AML), which consists of eight subtypes, M0 to M7. Of those only four, M1, M2, M3 and M5 were studied here due to data availability. Given the fact that these subtypes are characterised by substantial morphological similarities even haematologists have difficulties in differentiating between them using leukaemia cell images. Currently, the process of detection and classification is manual, based on the use of a microscope and diagnosis takes up to five days. The motivation behind this research is to improve the diagnostic process by automating it and reducing its time span from five days to a matter of a few hours. The three main methods employed in this research include Cellular Automata, Heuristic Search and Neural Networks.

Nowadays, medical imaging is one of the fastest growing fields in medicine, clinical setting and research and development (R&D). Further research in medical imaging is expected to improve patient care, contributing to areas such as personalised medicine with individually tailored treatment, increasing evidence-based decision making within healthcare, reducing complications during and after surgery, and allowing better understanding of the effects of treatments in various diseases (Brunetti & Haraldseth, 2007).

Image processing in medical research is becoming a subject of prime focus due to its tremendous potential for the public health sector and the scientific community in general. In particular, imaging applications are emerging as a new opportunity for innovation at the meeting point between medicine and computer science. Many software and research groups focus on the development of image processing applications for medical images, for example to improve low-resolution photographic images and produce effective high-quality imagery (Robison et al., 2010). Collaboration with clinicians has allowed the extraction of useful information contributing to more efficient diagnosis, especially in the treatment and study of cancer (Poulsen & Pedron, 1995).

In most people's minds there is no more frightening disease than cancer, often viewed as an untreatable, unbearable, and painful disease with no cure. Indeed it is a serious, potentially life-threatening illness (King & Robin, 2006). Leukaemia is a type of blood cancer. The first accurate description of leukaemia was performed by Velpeau in 1827 (Wiernik et al., 2003). Leukaemia was recognised in 1845, by Bennett in Scotland and Virchow in Germany (Frost, 2003). In their first two patients, it was the post-mortem appearance of blood, which first gave the hint of an abnormal condition. In Virchow's case, the blood vessels contained a "yellowish-white almost greenish mass" (Karp, 2007). Microscopically, there were a few red blood corpuscles and some colourless white bodies, which are found in the blood of a normal person. The relationship between

the red and the colourless corpuscles was the reverse of the normal ratio. Virchow introduced the term, “leukaemia” to describe the condition (Karp, 2007).

Leukaemia is a disease of an unknown cause where the bone marrow produces large numbers of abnormal cells. Most acute leukaemia patients are referred to specialist units for evaluation and treatment. Leukaemia can be diagnosed by blood tests and bone marrow tests. Currently the best available treatment is chemotherapy, which unfortunately also kills normal body cells along with the cancerous ones. A bone marrow and a blood test can help determine the different leukaemia types, allowing doctors to decide on the best choice of treatment. There are four general types of leukaemia, namely Acute Lymphoblastic leukaemia (ALL), Acute Myeloid Leukaemia (AML), Chronic Lymphocytic Leukaemia (CLL) and Chronic Myeloid Leukaemia (CML).

This thesis focuses on AML, a serious illness caused by the abnormal growth and development of early nongranular white blood cells. It starts in the bone marrow blast cells which develop into granulocytes, that is white blood cells containing small particles, or granules. As the blast cells build up they hamper the body's natural ability to fight infection and stop bleeding. Therefore, the disease requires immediate treatment. Moreover, AML has eight subtypes (M0 – M7), all exhibiting similar morphological characteristics.

The early recognition of blast cells in the bone marrow of patients suffering from AML, during the developmental stage of the illness, is extremely important for appropriate treatment (Nipon & Gader, 2002.) Clinicians have to identify these abnormal cells under a microscope, in order to decide if a patient with suspected leukaemia would need a bone marrow transplant. In addition, they have to calculate the number of blast cell to confirm the diagnosis (Yi et al., 2005). Before classification of AML, it is necessary to recognise the types of blast

cells which can be observed, and how they may differ from promyelocytes (M3 subtypes) (Hassan, 1996).

This chapter is organised as follows: section 1.1 presents the problem statement; section 1.2 deals with the objectives of the thesis and section 1.3 depicts the benefits of this work. Section 1.4 portrays the thesis' roadmap, and section 1.5 summarises this chapter.

1.1 Problem Statement

The microscopic study of human blood has led to the conclusion that a set of methods, including microscope colour imaging, segmentation, classification, and clustering can allow the identification of patients suffering from leukaemia (Weinberg, 1996, Uthman, 2008).

Machine learning is one of the methods used in image processing for detection of blast cells. This research focuses on AML and the classification of the eight subtypes mentioned previously, M0 to M7. This is an important issue since they require different treatment. All subtypes share extensive similarities which makes identification under a microscope difficult, time-consuming and physically tiring for the haematologists.

There are two classification schemata for AML, the French-American-British (FAB) and the World Health Organization (WHO) system. The FAB classification of AML describes 8 main types, M0, M1, M2, M3, M4, M5, M6 and M7 shown on Tables 1.1(a) and (b).

Table 1.1(a): FAB classification of AML (M0 – M3)

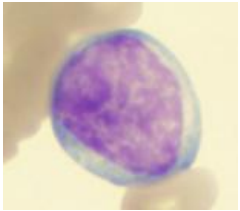
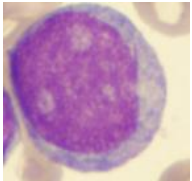
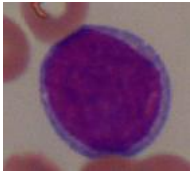
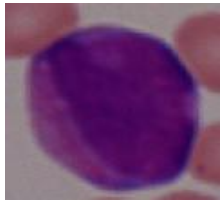
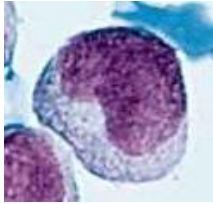
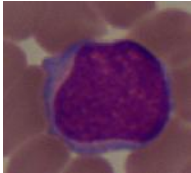
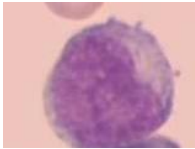
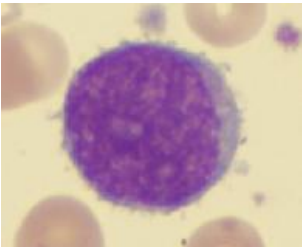
| Category | Morphologic Criteria (Bone Marrow) |
|--|---|
| <p>M0: Minimally differentiated acute myeloid leukaemia (Bell & Sallah, 2005).</p>  | <p>The blast cells are large and quite heterogeneous with the absence of Auer rods (Stasi et al., 1999).</p> |
| <p>M1: Myeloblastic without maturation</p>  | <p>≥ 90% of myeloid cell lines are blasts. The blast cells show few granules but may show Auer rods (Bell & Sallah, 2005).</p> |
| <p>M2: Myeloblastic with maturation</p>  | <p>30-89% of myeloid cell are blasts. >10% are promyelocytes <20% are monocytes. Show multiple cytoplasmic granules (Bell & Sallah, 2005).</p> |
| <p>M3: Promyelocytic</p>  | <p>Hypergranular promyelocytes with heavy to dust like granules, frequent Auer rods, nucleus often blooded; microgranular variant may occur. Blast cells show multiple Auer rods (Bell & Sallah, 2005).</p> |

Table 1.1(b): FAB classification of AML (M4 – M7)

| Category | Morphologic Criteria (Bone Marrow) |
|---|---|
| <p>M4: Myelomonocytic (Bell and Sallah, 2005)</p>  | <p>30-80% of myeloid cell lines are myeloblasts plus maturing neutrophils. 20-80% of myeloid cell lines are monocytic lineage. Blasts have some monocytoid differentiation (Bell & Sallah, 2005).</p> |
| <p>M5: Monoblastic monocytic</p>  | <p>>80% of a myeloid cell line are monoblasts, promonocytes or monocytes. In M5a 80% of myeloid cell lines are monoblasts; in M5b, <80% were monoblast and the remainder are promonocytes or monocytes (Bell & Sallah, 2005).</p> |
| <p>M6: Erythroleukaemia (Bell and Sallah, 2005)</p>  | <p>≥ 50% of bone marrow cells are erythroid precursors. >30% of non-erythroid myeloid cell lines are blasts. Showing preponderance of erythroblast (Bell & Sallah, 2005).</p> |
| <p>M7: Megakaryocytic (Bell and Sallah, 2005)</p>  | <p>Blasts in marrow or blood are identified as megakaryocytic lineage. If marrow is undesirable, biopsy shows large tumour of blasts, frequently increased numbers of megakaryocyte and reticulin (Bell & Sallah, 2005).</p> |

The diagnosis of AML and its subtypes is based on the morphological analysis of peripheral blood and bone marrow (Hassan, 1996). The process strictly follows the FAB classification (Bennett et al., 1976). In the case of M0, M1 and M2 the development of white blood cells stops at the stage of a Neutrophil. In the M3 type, the development of white blood cells stops at the stage of a Promyelocyte. In M4, development stops before the stage of a Monoblast and Myeloblast. In M5, the white blood cells stop developing at the stage of Proerythroblasts. Finally, in subtypes M6 and M7, the development of white blood cells stops at Megakaryoblasts. Each of the subtypes has its own morphological features.

Haematologists may often face difficulties identifying the correct subtype, given the morphological similarities they share. This research aims to aid the haematologists in automatically detecting and classifying AML. In addition it shortens the diagnostic process from five days to a matter of hours.

1.2 Objectives

The objectives of the thesis were the following:-

- a) To develop an automated method of analysis of AML blast cell images. Haematologists often face difficulties identifying the subtypes of AML, due to the similarities of their morphological features.
- b) Following AML detection, blast cells need to be classified into M3 or one of the other subtypes. The reason for targeting M3 is that its treatment differs from the treatment of the rest, requiring All-Trans-Retinoic-Acid (ATRA) to be added to the initial chemotherapy.
- c) The proposed methods are included in image-processing software, which enables the haematologist to diagnose AML more effectively and efficiently.

1.3 Benefits

In current practice, a haematologist uses a microscope to detect blast cells. Given the similarities shared by AML subtypes it is often difficult to distinguish between them. The haematologist performs cytogenetic analysis which can be a time-consuming process, usually lasting around five days. It is also physically tiring, causing suffering to the eyes and back.

However, novel analytical techniques were implemented here, facilitating Cellular Automata, Heuristic Search and Neural Networks for the automatic diagnosis of AML and its subtypes. Each of the methods has a contribution to research in this field, introducing novelties in the analytical approach. Cellular Automata were used to detect potential blast cells, remove image noise and find the radius of the cells. Heuristic search was implemented to optimise the detection of the blast cells before performing classification. Neural Network was used to classify the blast cells subtypes.

As mentioned previously, the development of a fully automated screening system prototype for AML may provide the specialist with significant aid in the effort to detect and classify AML cells more effectively and efficiently. The system requires further enhancements such as being linked to WEKA. User testing would also be beneficial.

1.4 Roadmap

The roadmap followed in this PhD project, which facilitates artificial intelligence in medicine, is shown in Figures 1.1 and 1.2. This section presents a general overview of the thesis. The subsequent chapters provide a deeper insight and a

much greater discussion of the workings of image processing and the steps required to detect and classify blast cells.

Chapter 2 includes a discussion of red, white blood cells and leukaemia cells. It presents the four main types of leukaemia, ALL, AML, CLL and CML. The thesis concentrates on AML of type M0 to M7. It also explains the two classification schemata, FAB and WHO, in more detail. Finally it describes the background of the implemented methods, including Cellular Automata (CA), Heuristic Search and Artificial Intelligence.

Chapter 3 elaborates on the use of Heuristic Search techniques to identify blast cells and the produced results. This is only a preliminary step in the process of detecting the blast cells. In particular, three methods were implemented, that is, Hill Climbing (HC), Simulated Annealing (SA) and a Genetic Algorithm (GA). These are termed Random Heuristic Search techniques given that the starting coordinates (points) are chosen randomly. The work process is shown in Figure 1.1. The Otsu method is applied to a real image in order to extract objects from their background. A grey scale threshold is applied to segment the image into foreground and background, converting the colour image into black and white image prior to performing the Heuristic Search. Following the application of the described techniques it is observed that the HC and SA approach exhibit similar performance. In contrast, the GA search reaches varying fitness values because during crossover coordinates overlap causing a large amount of computational overhead. Instead the degree of overlap is factored into the fitness function so that selective pressure is used to prevent overlaps. Testing was conducted using ten randomly selected images from the data set. However, the results are not satisfactory because blast cells are often not correctly detected. The results show that a random SA search reaches higher fitness values. However, the GA performs better in identifying blast cells.

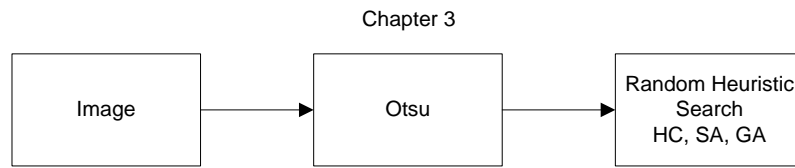


Figure 1.1: Random Heuristic Search

Chapter 4 describes the application of Otsu's method, to extract objects from their background. Following that, Cellular Automata (CA) was used to convert a black and white image to a matrix (of the same size as the input image) where each point in the matrix represents the shortest distance to each point in the image furthest away from the background. CA filtering is applied to remove cells of no interest, such as red blood cells. In this chapter, a new method of filtering is presented based on the size of the five distinct types of white blood cells. This method shows that the biggest areas in the image correspond to potential blast cells.

Chapter 5, elaborates on the process of identifying the coordinates of the starting points for the Seeded Heuristic Search. This consists of determining the centre of the regions for the biggest areas corresponding to potential blast cells, defined in chapter 4, which are used as possible starting coordinates (points) for locating blast cells. Image classification determines whether each point corresponds to a blast cell. The selected cells are placed into categories, based on their colour, where pink corresponds to plasma and red blood cells and purple to blast cells. The colour (three values, Red, Green, Blue scale) of a pixel within the circle is identified based on the starting point, to see how close it is, in terms of Euclidean Distance, to either a pink (red blood cell) or purple (blast cells) vector. In addition the methods checks to ensure that circles do not overlap, otherwise the optimisation process can not proceed.

Chapter 6, the choice of the starting coordinates, is followed by the detection of blast cells, facilitating a Seeded Heuristic Search. Two fitness functions are implemented. The one defined in Chapter 3 and a colour fitness functions. The obtained results reveal that the SA and HC show superior performance in detecting blast cells. Following that Venn Diagrams are used to examine if the resulting coordinates are consistent, based on 10 subsequent runs of the algorithm. The testing is only conducted for HC and SA based on the “best” result. It was observed that the SA shows the “best” performance, with minimal variability in the final results. This method is then applied to analyse all images in the dataset.

Chapter 7 deals with the classification of these images, following the processing steps discussed above. Each sub-image representing the blast cells is identified using the methods outlined in chapter 6 and then extracted. Summary statistics (mean, median, variance, high and low of RGB levels) are produced regarding the RGB colour of each of the pixels in these sub-images and these are used as features for classification. The WEKA application is used to perform the classification. WEKA is a data analysis software tool which implements a set of machine learning algorithms for data mining tasks. The Multilayer Perceptron classifier produced the best results. The testing was conducted for all three search methods (SA, HC and GA). In addition the hierarchical organization of the resulting classification was examined. In particular, images were classified into either M3 or other subtypes, then into M5 and the remaining subtypes, M1 and M2, and finally, into M1 and M2 subtypes. The obtained results are remarkably good. These findings are of important clinical significance given that AML M3 requires different treatment than the other AML subtypes.

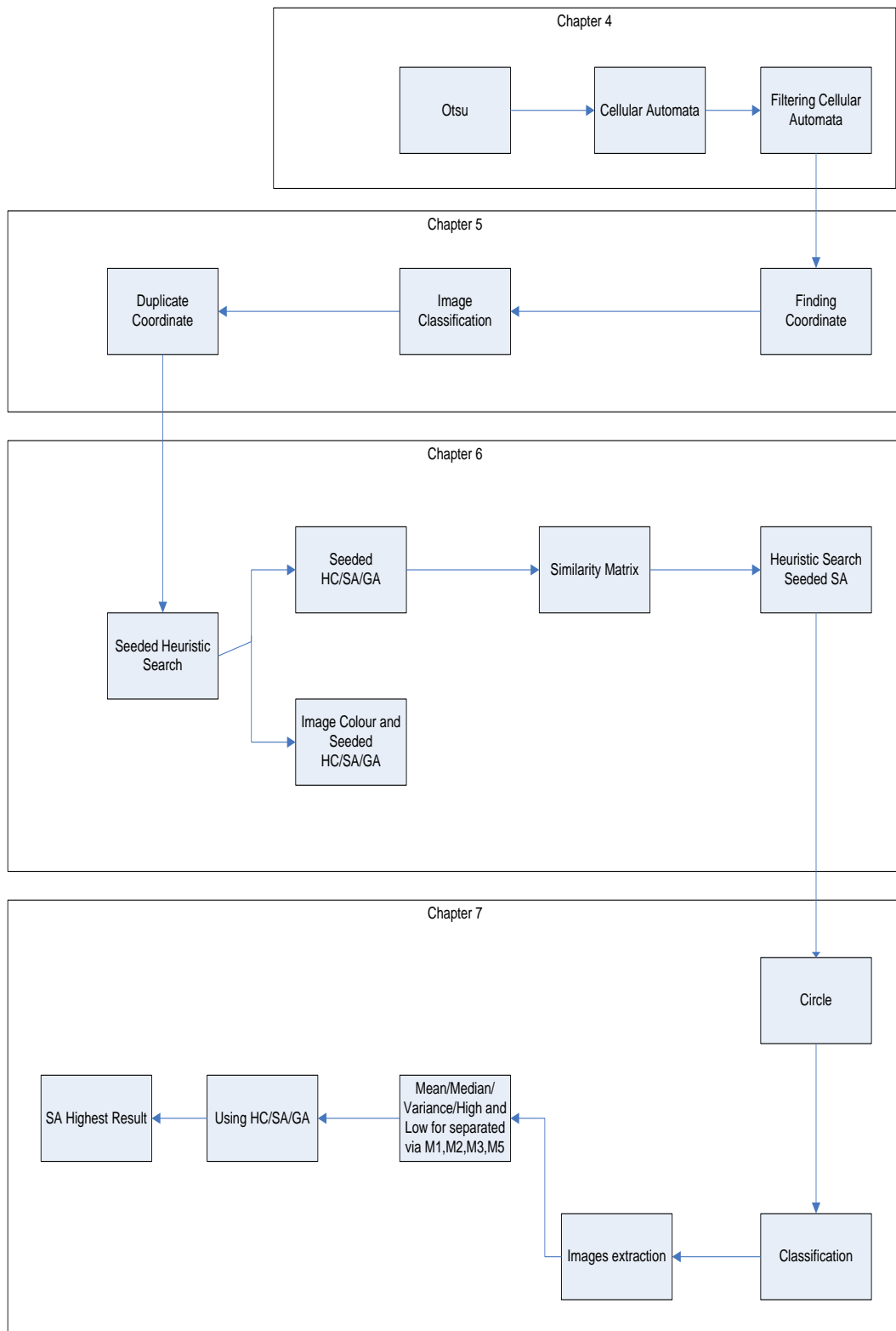


Figure 1.2: Roadmap

1.5 Summary

This chapter presented the motivation behind the research in this thesis, that is, the difficulties faced by haematologists in identifying subtypes of AML by visual inspection of microscopic images. As discussed, there are eight subtypes of AML, with different morphological features. However, given that they also share extensive similarities diagnosis can be time consuming.

The aim of this research is the development of an automated method for the detection of blast cells in images of blood smear and bone marrow microscope slides and their subsequent classification into AML subtypes. The most important issue is the classification of an image into M3 or the group of the remaining subtypes, as AML M3 requires different treatment.

The novel analytical methods proposed in the thesis are based on Artificial Intelligence, and include Cellular Automata, Heuristic Search and Neural Networks for auto-detection and classification of blast cells. A software application was developed to assist haematologists to diagnose AML more effectively and efficiently.

This preliminary chapter discussed the research presented in this thesis. It presented the Otsu method for segmenting an image into foreground and background, using a histogram. It commented on the used heuristic search methods, the promising results obtained, and the classification approach used for AML images. Naturally, further work is required to develop a software application to be used by haematologists.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

This chapter provides a brief overview of leukaemia and a conceptual analysis of the main methods used for the detection and classification of leukaemia cells facilitating Artificial Intelligence, Cellular Automata and Neural Networks.

2.2 Leukaemia

Cancer has become a data-intensive area of research, with increasing rate of developments in data collection technologies and methodologies. In 1895, Wilhelm Roentgen discovered that X-ray tubes, used extensively for imaging bones and then for treating a variety of conditions. The technicians who ran the Radiograph machines, as well as many of the exposed patients, contracted skin tumours and leukaemia (Wienberg, 1996). Accurate diagnosis and classification of blast cells is an

extremely valuable requirement for the precise diagnosis of leukaemia and has a positive impact on treatment and prognosis (Hassan, 1996).

2.2.1 Blood

Blood is fundamental to human life. An average human body is approximately 70 litres of which five litres are blood (Uthman, 2008). Biologically, blood is essential for maintaining homeostasis, that is keeping the body's status stable. This refers to hydration, temperature regulation and ion concentration. The main blood functions include (The Blood, 2011):

- a) Delivery of nutrients from the digestive system to all parts of the body.
- b) Transport of oxygen from the lungs to all parts of a body.
- c) Transport of carbon dioxide from all parts of the body to the lungs.
- d) Transport of waste products from cells to the external environment, especially via the kidneys.
- e) Maintaining an ongoing “discussion” of its components with tissue fluids and keeping electrolyte balance.
- f) Protecting the body against attack from foreign organisms through the white blood cells and antibodies.
- g) Defending the body against injury or illness using the inflammatory response.
- h) Preventing serious haemorrhage by the clotting process.
- i) Maintaining the body's temperature by circulating heat.

Blood has four main elements to ensure it fulfils its functions, shown in Table 2.1. (The Blood, 2011).

Table 2.1 (a): Major elements of Blood (Red blood cells, White blood cells, Platelets).

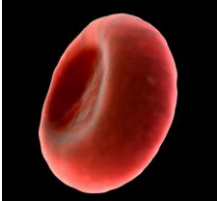
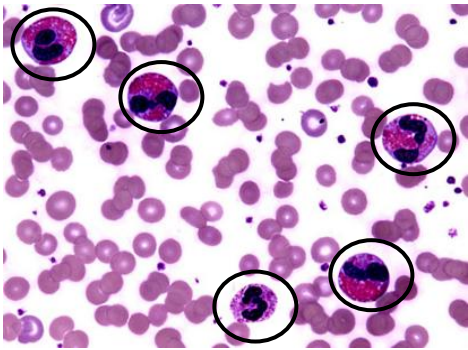
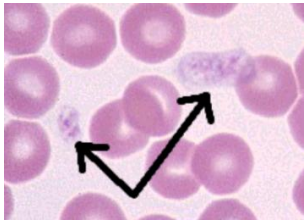
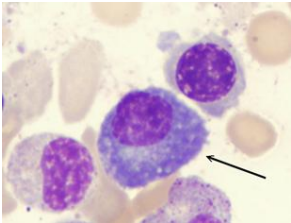
| Elements | Description |
|---|---|
| <p>Red blood cells (RBC's or erythrocytes) (Maya, 2011)</p>  | <p>Transport oxygen from the lungs to organs and peripheral site. (Maya, 2011)</p> |
| <p>White blood cells (Human, 2011)</p>  | <p>Defensive role in destroying invading organisms, e.g. bacteria and viruses and assist in the removal of dead or damaged tissue cells. (The Blood, 2011).</p> |
| <p>Platelets (Immune, 2010)</p>  | <p>Assist in the clotting process. (Blood – CH19, 2011)</p> |

Table 2.1(b): Major elements of Blood (Plasma).

| Elements | Description |
|---|---|
| Plasma (Haken, 2010)  | Carries nutrients, metabolites antibodies and the proteins involved in blood clotting. (Blood – CH19, 2011) |

2.2.2 White Blood Cells

A white blood cell is larger than a red blood cell (Zamani & Safabakhsh, 2006). White blood cell composition and concentration in the blood gives valuable information and plays a crucial role in the diagnosis of different diseases. White blood cells fall into five categories: Neutrophil, Eosinophil, Basophil, Monocyte and Lymphocyte (Zamani & Safabakhsh, 2006), shown in the Table 2.2. These cells provide the greatest defense against infections, and their individual concentrations can help specialists to distinguish between the presence or not of severe pathologies (Piuri & Scotti, 2004, Scotti, 2006, Zamani & Safabakhsh, 2006). The size of white blood cells will be used for the filtering process discussed in the chapter 4.

Table 2.2(a): White Blood Cells (Neutrophil, Eosinophil, Basophil)

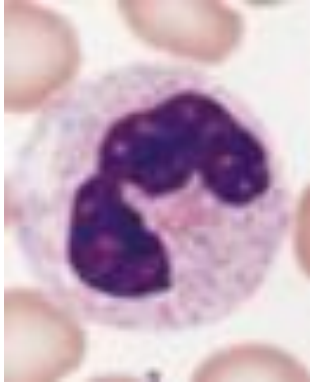
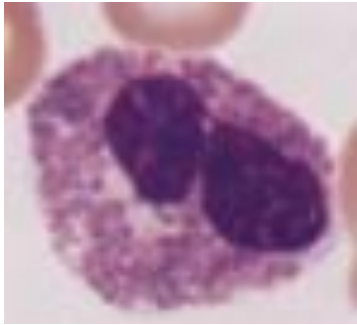

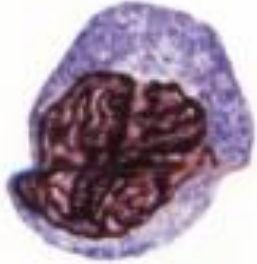

| Types | Descriptions |
|---|--|
| <p data-bbox="451 464 597 495">Neutrophil</p>  | <p data-bbox="764 464 1385 930">This cell has a characteristic nucleus, consisting of between two and five lobes, and a pale cytoplasm. The granules are divided into primary, which appear at the promyelocyte stage, and secondary which appear at the myelocyte stage, and predominant in the mature neutrophil. The lifespan of neutrophils in the blood is only about 10h. Size: 12-15 μm in diameter (Hoffbrand et al., 2001).</p> |
| <p data-bbox="451 957 597 989">Eosinophil</p>  | <p data-bbox="764 957 1385 1371">These cells are similar to neutrophils, except that the cytoplasmic granules are coarser and more deeply red staining. They enter inflammatory exudates and have a special role in allergic responses. They provide defence against parasites and help the removal of fibrin formed during inflammation. Size: 12 to 15 μm in diameter (Hoffbrand et al., 2001).</p> |
| <p data-bbox="467 1396 581 1428">Basophil</p>  | <p data-bbox="764 1396 1385 1701">Basophil cells are only seen in normal peripheral blood. They have many dark cytoplasmic granules, which overlie the nucleus and contain heparin and histamine. Size: 9-10 μm in diameter (Hoffbrand et al., 2001).</p> |

Table 2.2(b): White Blood Cells (Monocyte and Lymphocyte)

| Types | Description |
|---|---|
| <p data-bbox="329 464 743 495">Monocyte (Bell & Sallah, 2005)</p>  | <p data-bbox="781 464 1382 982">These are usually larger than other peripheral blood leucocytes. The monocyte precursors in the marrow (monoblasts and promonocytes) are difficult to distinguish from myeloblasts and monocytes. Monocytes spend only a short time in the marrow and after circulating for 20-40 hours, leave the blood to enter the tissues where they mature and carry out their principal functions. Size: 16-20 μm in diameter (Hoffbrand et al., 2001).</p> |
| <p data-bbox="451 1010 618 1041">Lymphocyte</p>  | <p data-bbox="781 1010 1382 1482">These are the immunologically competent cells which assist the phagocytes in the defence of the body against infection and other foreign invasion. Two unique features characteristic of the immune system are the ability to generate antigenic specificity and the phenomenon of immunological memory. Size: 8-10 μm in diameter (Hoffbrand et al., 2001).</p> |

2.2.3 Types of Leukaemia

Leukaemia is a disease of unknown cause where the bone marrow produces large numbers of abnormal cells white blood cells that stop developing before maturity.

(Stock & Hoffman, 2000). There are four main types of leukaemia, namely Acute Lymphoblastic Leukaemia (ALL), Acute Myeloid Leukaemia (AML), which is used as a case study in the thesis, Chronic Lymphocytic Leukaemia (CLL) and Chronic Myeloid Leukaemia (CML). Most commonly, acute leukaemia patients are referred to specialist units for evaluation. Treatment is based on chemotherapy through the veins, lasting four to six months, which also kills normal body cells. Leukaemia can be diagnosed by blood tests while a bone marrow test serves to decide on the best choice of treatment. Table 2.3 exhibits the main types of leukaemia.

Table 2.3(a): Types of Leukaemia (ALL and AML)

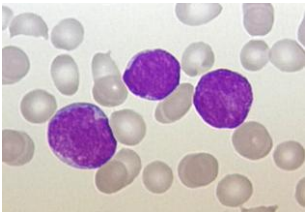
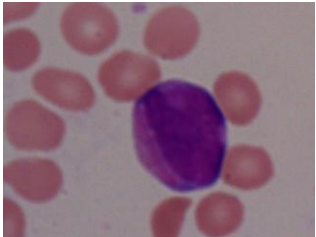
| Types | Descriptions |
|---|--|
| Acute Lymphoblastic Leukaemia (ALL) (Teri & Mccoysca, 2011)  | The most common type of leukaemia in young children. This disease also affects adults, especially over the age of 65 (Hoffbrand et al., 2001). |
| Acute Myeloid Leukaemia (AML)  | It develops in both adults and children (Hoffbrand et al., 2001). |

Table 2.3(b): Types of Leukaemia (CML and CLL)

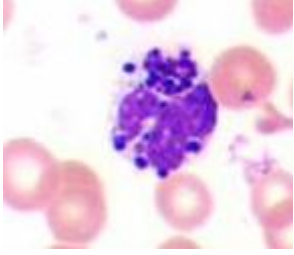
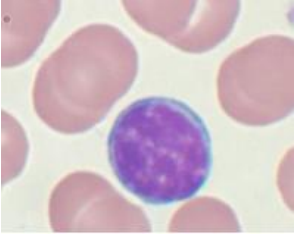
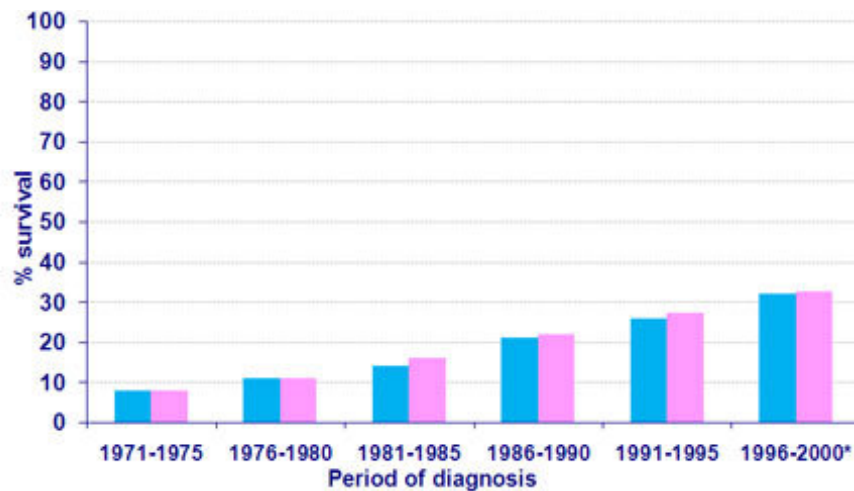
| Types | Descriptions |
|---|--|
| <p>Chronic Myeloid Leukaemia (CML) (Mcgaufin et al., 2005).</p>  | <p>It occurs mainly in adults. A very small number will infects the children (Hoffbrand et al., 2001).</p> |
| <p>Chronic Lymphocytic Leukaemia (CLL) (Chronic Lymphocytic Leukemia/Small Cell Lymphoma (CLL/SLL), 2011)</p>  | <p>Most commonly it affects adults over the age of 55. It sometimes occurs in younger adults, but it almost never affects children (Hoffbrand et al., 2001).</p> |

Table 2.4 shows the UK Leukaemia case statistics for males and females in 2007, revealing that the survival rate has increased from 2001 to 2006. The diagnosis and the medical treatment have improved significantly as shown in Figure 2.1. Automated detecting can contribute to the early diagnosis of patients and survival rates are expected to increase in the future.

Table 2.4: Leukaemia cases in UK for 2007 (Leukaemia Statistics - Key Facts, 2010).

| Leukaemia - UK | Males | Females | Persons |
|---|-------|---------|---------|
| Number of new cases (UK 2007) | 4,069 | 2,932 | 7,001 |
| Rate per 100,000 population* | 11.5 | 6.9 | 9.0 |
| Number of deaths (UK 2008) | 2,483 | 1,884 | 4,367 |
| Rate per 100,000 population | 6.4 | 3.6 | 4.9 |
| One-year survival rate (for patients diagnosed 2004-2006, England) | 61% | 62% | - |
| Five-year survival rate (for patients diagnosed 2001-2006, England) | 40% | 41% | - |
| Ten-year predicted survival rate (for patients diagnosed 2007, England & Wales) | - | - | 33.2% |

**Figure 2.1:** Leukaemia 10-year relative survival rates (Leukaemia Survival Statistics, 2010).

2.2.4 Flow chart of patients admitted with Leukaemia.

Figure 2.2 shows the steps that need to be taken by a haematologist in order to diagnose a patient with acute leukaemia. Table 2.5 provides a more detailed explanation of the individual steps in Figure 2.2.

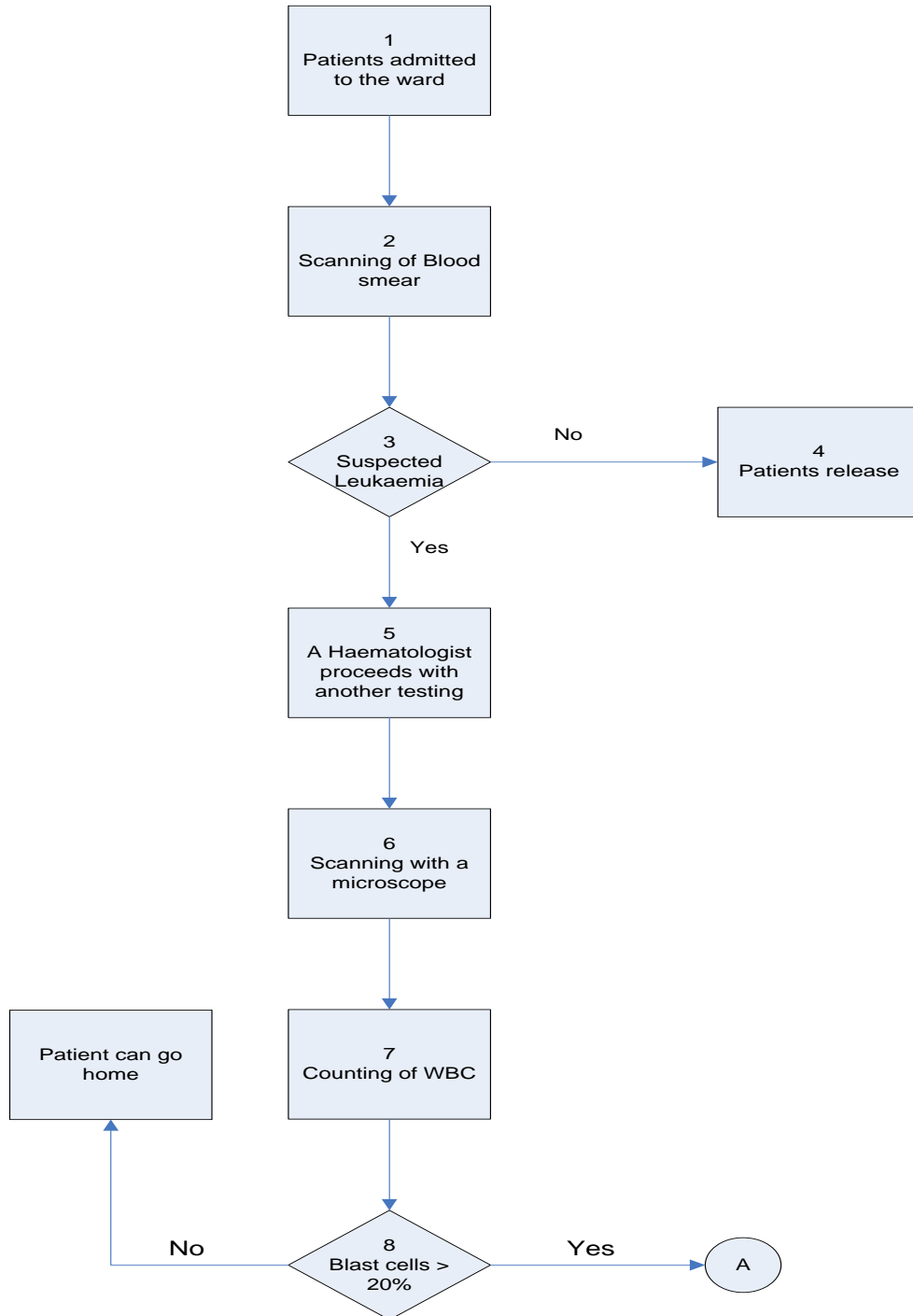


Figure 2.2(a): Steps to confirm AML – M3 (Steps 1 to Steps 8) (14,15)

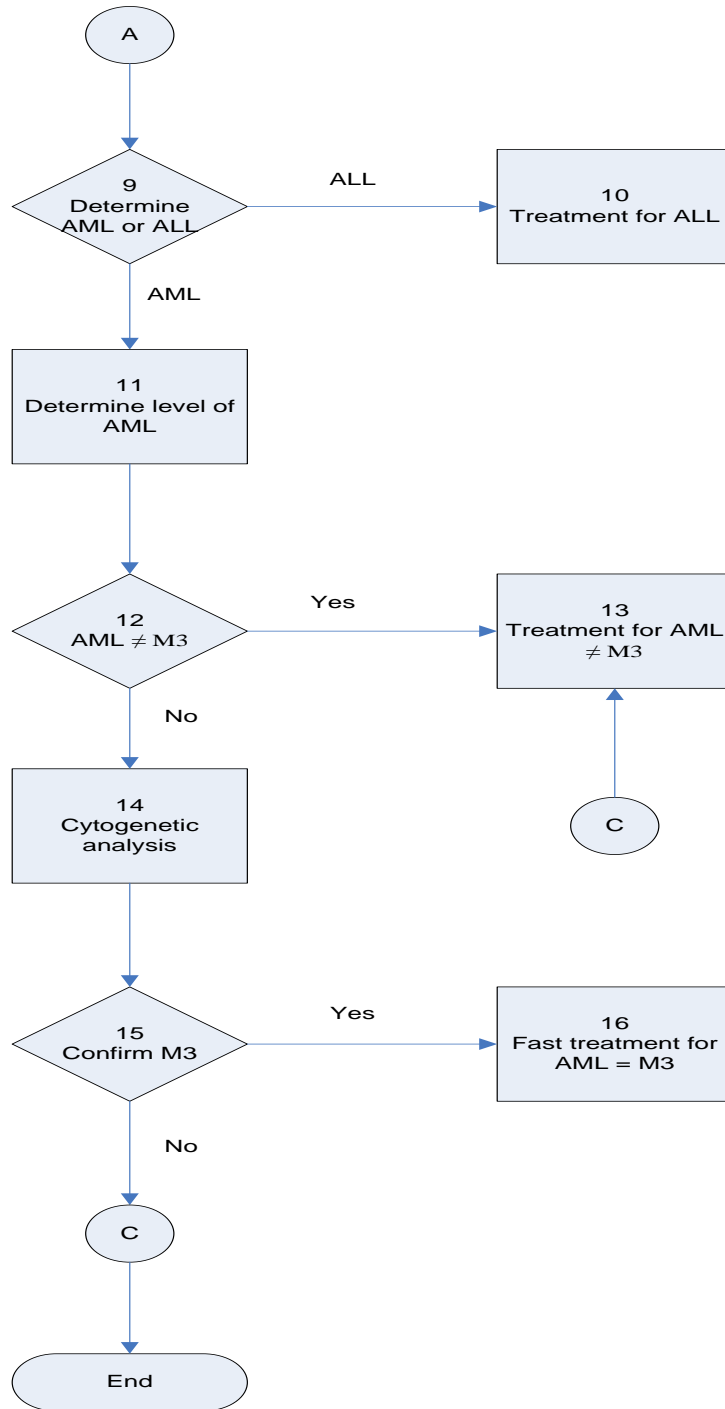


Figure 2.2(b): Steps to confirm AML – M3 (Step 9 to Step 16)

Table 2.5 (a): Analytical description of each step in Figure 2.2

| Steps | Descriptions from Figure 2.2 |
|--------------|---|
| 1 | Patient warded in hospital. |
| 2 | <p>In acute leukaemia patients, the White Blood Cell (WBC) count and morphology will be abnormal.</p> <p>Doctor will suspect that a patient has leukaemia based on:</p> <ol style="list-style-type: none"> a) Clinical presentation: patient presented with hepatosplenomegaly and/or lymphadenopathy. b) Abnormal blood count: haemoglobin and platelet count is low, WBC is normal, low or high. <p>Blood test will be taken from the patient.</p> <p>In a healthy adult the differential WBC test should show the following results:</p> <ul style="list-style-type: none"> • Neutrophil – 40% – 70% • Eosinophil – 5% • Basophil – 1% • Monocyte – 6% - 10% • Lymphocyte – 20% - 50% <p>The blood count will be performed by the Machine.</p> <p>In case of clinically suspicious and abnormal blood count specimens a blood smear is prepared and the slide is referred to a haematologist for examination.</p> |
| 3 | The blood smear is scanned through a microscope by a haematologist. Refer to Table 2.2 on the WBC. |

Table 2.5 (b): Analytical description of each step in Figure 2.2

| Steps | Descriptions |
|-------|---|
| 4 | If the blood smear test reveals no evidence of leukaemia, i.e. normal blood cell count and morphology, other underlying causes need to be investigated. |
| 5 | If the patient is suspected of suffering from leukaemia, bone marrow sample is required to confirm the diagnosis. |
| 6 | The marrow smear is scanned under a microscope by a haematologist with 10x, 40x and 100x magnifying power. |
| 7 | The haematologist will calculate the WBC differential count manually preferably based on 300-500 cells. |
| 8 | Blasts should account for about 20% of cells, based on the WHO classification. Less than 5% is considered normal. |
| 9 | The image allows identification of ALL or AML. Refer to Table 2.3 for differentiating between ALL and AML. |
| 10 | If the patient diagnosed with ALL then required treatment. |
| 11 | In the case of AML, the subtype must be identified as either M3 or non-M3 (M0-M7 except M3). |
| 12 | If the patient is suspected as M3 based on the blood smear and bone marrow smear test, Cytogenetic/molecular follows to confirm the diagnosis. |
| 13 | Confirm M3: Treatment for M3; All-Trans-Retinoic-Acid (ATRA). |
| 14 | If non-M3 <ul style="list-style-type: none"> • Another Chemotherapy |

2.2.5 Acute Leukaemia

Acute leukaemia is cancer of the white blood cells. There are two types of acute leukaemia, Acute Lymphoblastic Leukaemia (ALL) and Acute Myeloid Leukaemia (AML). In 1971, the diagnosis of leukaemia cells was based on their morphology and according to (FW Gunz & AF Burry, 1963) in (Hassan, 2008) it was correct in just 70% of examined cases. The incidence increases over the age of 40 to 50 years, while the median age in different series ranges from 25 to 37 years. In the early 1980, 90% of acute leukaemia cases could be correctly classified as ALL and AML based on the the FAB classification system, prior to development of immunophenotyping (Hassan, 1996).

Acute leukaemia is an aggressive disease in which the malignant transformation causes accumulation of early bone marrow haemopoietic progenitors. Hematopoiesis is the formation of cellular blood components. All cellular blood components are derived from haematopoietic stem cells. The dominant clinical manifestation of leukaemia is usually bone marrow failure caused by accumulation of blast cells, although tissue infiltration also occurs (Hoffbrand et al., 2001).

With modern chemotherapy, better supportive care and central nervous system (CNS) prophylaxis, 70% to 80% of patients achieve complete remission and about one-third of these patients can expect disease-free survival of more than five years. Additionally, advances in treatment have increased the cure rate for AML from 10%-20% in the 1960s to 40-60% in 1980s. This increase is also the result of more accurate diagnosis (Hassan, 1996).

Acute leukaemia is diagnosed in people with more than 20% of blast cells in the bone marrow. It is further subdivided into AML and ALL based on whether the blasts are myeloblasts or lymphoblasts. According to the European cancer registry between 1988 and 1997, 5-year survival of children with leukaemia was 73% and 44% for infants and adolescents respectively (Stiller et al., 1997) in (Hassan, 2008).

2.2.6 Acute Myeloid Leukaemia

Acute Myelogenous Leukaemia (AML) is a serious illness caused by the abnormal growth and development of early nongranular white blood cells. It starts in the bone marrow blast cells which develop to shape granulocytes, that is, white blood cells that contain small particles, or granules. This thesis is centered on the analysis of AML images. The AML blasts do not mature and become too numerous in the blood and bone marrow. As the cells build up, they hamper the body's ability to fight infection and stop bleeding. Therefore, it is necessary to treat this disease within a short time after diagnosis. The recognition of the blast cells in the bone marrow of patients suffering from AML is a very important step in identifying the developmental stage of the illness and choosing an appropriate treatment (Nipon & Gader, 2002). Clinicians need to identify these abnormal cells under a microscope in order to conclude that a patient suffers from leukaemia. The patient's bone marrow is examined to count the blast cells and confirm the diagnosis (Fang et al., 2005). For classification of AML, it is necessary to recognise the types of blast present in the blood smear, and how they differ from promyelocytes (Hassan, 1996).

The established diagnosis of AML is based on morphological examination of peripheral blood and bone marrow. Initially it strictly followed the FAB classification

system (Hassan, 1996) shown in the Table 1.1 (a) – Table 1.1 (b). More recently, the WHO classification system was adopted (Table .2.6) (Hassan, 2008).

Table 2.6: WHO classification of AML

| |
|---|
| AML with recurrent cytogenetic translocation |
| <ul style="list-style-type: none"> • AML with genes of t(8;21) (q22;q22). Translocations are when chromosomes break and rejoin with other chromosomes. |
| <ul style="list-style-type: none"> • AML with abnormal bone marrow eosinophils |
| AML with multilinear dysplasia |
| <ul style="list-style-type: none"> • With a prior myelodysplastic syndrome, |
| <ul style="list-style-type: none"> • Without a prior myelodysplastic syndrome, but with dysplasia in at least 50% of cells in 2 or more myeloid lineages |
| AML not otherwise categorised |
| <ul style="list-style-type: none"> • AML minimally differentiated |
| <ul style="list-style-type: none"> • AML without maturation |
| <ul style="list-style-type: none"> • AML with maturation |
| <ul style="list-style-type: none"> • Acute myelomonocytic leukaemia |
| <ul style="list-style-type: none"> • Acute monocytic leukaemia |
| <ul style="list-style-type: none"> • Acute erythroid leukaemia |
| <ul style="list-style-type: none"> • Acute megakaryocytic leukaemia |
| <ul style="list-style-type: none"> • Acute basophilic leukaemia |
| <ul style="list-style-type: none"> • Acute panmyelosis with myelofibrosis |
| AML Therapy related. |
| <ul style="list-style-type: none"> • An alkylating agent related. |
| <ul style="list-style-type: none"> • Myeloid sarcoma |

AML is a general form of acute leukaemia that is increasingly common with progressing age but may occur in all age groups. It forms only 10%-15% of leukaemia incidents in childhood. Both types of leukaemia are associated with distinct genetic markers and have different prognoses. In addition, cytogenetic abnormalities and response to initial therapy have a significant impact on prognosis (Hoffbrand et al., 2001).

2.2.7 Laboratory Diagnosis of Acute Myeloid Leukaemia

Laboratory diagnosis of acute leukaemia requires morphological examination of peripheral blood film and bone marrow. Additional laboratory testing such as cytochemical staining, immunophenotyping, chromosomal studies and molecular analysis is required in order to improve diagnostic accuracy (Plesa et al. 2008). Haematological diagnosis has traditionally been based on light microscopy supplemented by cytochemistry. However, the development of new techniques in the past two decades, such as immunophenotyping, and cytogenetic and molecular genetic analysis, provided an additional source of information, which has an important impact on patient management (Hassan, 1996).

2.2.7.1 Peripheral Blood Film

A blood film is a thin layer of blood smeared on a microscope slide that contains the blood cells to be examined (Figure 2.3). Blood films are commonly examined to investigate haematological problems. A blood film is prepared by placing a drop of blood on one end of a slide, and using a spreader slide to disperse the blood over the slide's length. The aim is to make sure that cells are spaced far enough apart to be

counted and differentiated. The slide is left to air dry, after which the blood is fixed to the slide by immersing it briefly in methanol (Rapson & Matthews, 2011)



Figure 2.3: Blood Film (Basic Morphology of Blood Film., 2011)

2.2.7.2 Bone Marrow

Bone marrow is a special fatty tissue containing stem cells, located inside a few large bones. These stem cells can transform into white blood cells, red blood cells and platelets that have various roles. AML can compromise the resilience of bone marrow. Inside this special tissue, immature stems cells reside, along with extra iron. Stem cells remain undifferentiated until abnormal, weakened, or damaged cells need to be replaced. A stem cell can transform itself into a platelet, a white blood cell or a red blood cell. This is the only process through which cells get replaced to maintain the body healthy (Orazi et al., 2006). Figure 2.4 shows an example of bone marrow cells corresponding to M2 AML. Figure 2.5, graphically depicts the process of taking a bone marrow sample.

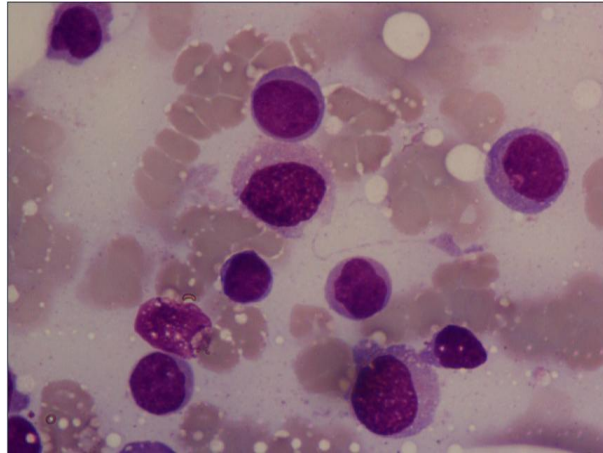


Figure 2.4: Bone Marrow sample in M2 AML case

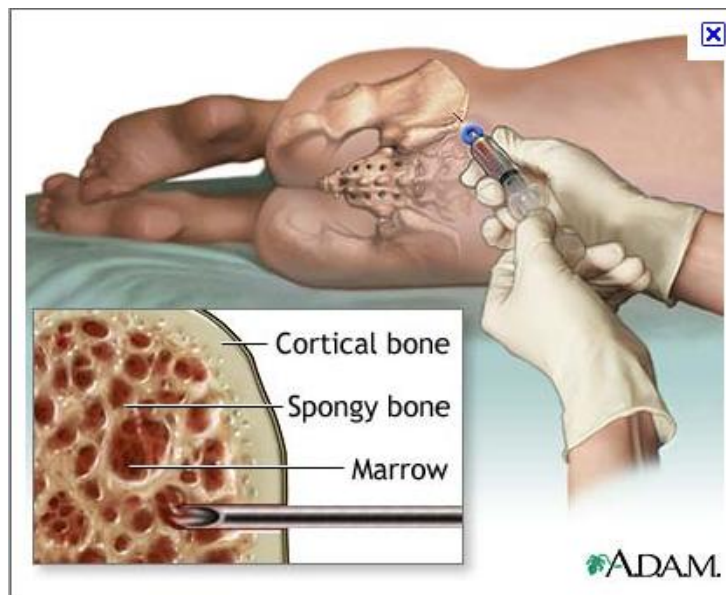


Figure 2.5: Blood taken from Bone Marrow (Dugdale, 2010).

2.2.8 Classification of Acute Myeloid Leukaemia (M0 to M7).

In the last couple of decades, the development of new techniques such as immunophenotyping and cytogenetic analysis have provided additional sources of information, which have had an impact on the treatment of AML patients (Plesa et al., 2008). Diagnosis requires additional cytogenetic testing and molecular recognition. The diagnostic process has become more demanding with respect to considerations of each individual, time and costs, given the expansion of laboratory testing. The molecular genetic approach is faster than cytogenetics with a turnaround time of three to five days, compared to two to four weeks respectively. This is a consequence of technological advances in the molecular recognition and identification of factors that contribute to prognosis (Hassan, 2008). In addition diagnosis can be based on the use immunological markers.

2.2.8.1 Cytogenetics

In AML, Genetic testing focuses on leukaemia-specific clonal and prognostic markers. Standard cytogenetic analysis is a method for identifying clonal aberrations in patients suffering from AML (Hassan, 1996). Cytogenetics serve to:

- a) Confirm diagnosis.
- b) Differentiate between M3 and Non-M3 AML.
- c) Classify acute leukaemia.
- d) Classify AML according to the WHO system.
- e) For prognosis.
- f) For Gene targeted treatment.
- g) Follow up the value of minimal residual disease.

Cytogenetic testing has become a key part of the evaluation of acute leukaemia, especially in assessing prognosis. It has been used to develop the progression or regression of an abnormal cell lineage (Zhang & Le Beau, 2011). Figure 2.6 depicts the standard karyotype resulting from a Cytogenetic analysis. Karyotype is a laboratory test used to study an individual's chromosome make-up.

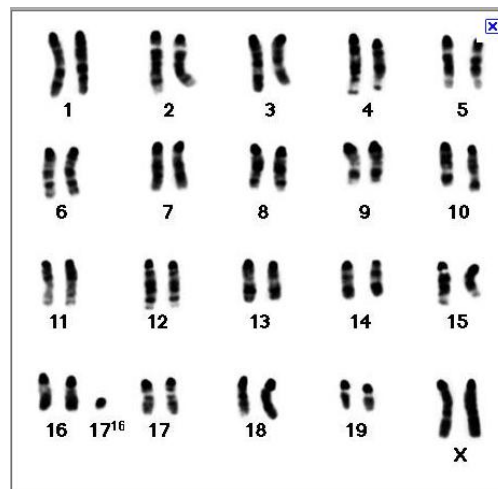


Figure 2.6: Normal Karyotype from a Cytogenetic analysis (The Jackson Laboratory, 2011)

2.2.8.2 Immunophenotyping

Immunophenotyping is a way to study proteins in a cell, which involves the labelling of white blood cells with antibodies. By choosing appropriate antibodies, the differentiation of leukemia cells can be determined. The whole procedure can be performed on cells from the blood, bone marrow or spinal fluid in a matter of a few hours (Immunophenotyping, 2011).

2.2.9 Treatment

It is important to accurately determine the subtype of leukaemia, since treatment may differ. In case of wrong diagnosis, patients face complications and may die. Test for Disseminated Intravascular Coagulopathy (DIC) is beneficial in patients with promyelocytic (M3) variant of AML. The treatment of AML is primarily based on the use of intensive chemotherapy. This is usually administered in four or five doses, each approximately one week apart and the most commonly used drugs include cytosine arabinoside, daunorubicin, idarubicin, 6-thioguanine, mitoxantrone or etoposide (Hoffbrand et al., 2001).

All AML subtypes (FAB M0 – M7) are treated similarly except for the promyelocytic M3 variation associated with the t(15;17) translocation in which ATRA is added to the initial chemotherapy. Since the drugs are myelotoxic, with limited selectivity between leukaemia and normal cells, there is considerable risk of marrow failure, thus prolonged and intensive supportive care is required. A key concept developing in AML therapy is that of basing the treatment program of individual patients on their risk status. Remission after one course of chemotherapy is also beneficial (Hoffbrand et al., 2001).

For patients over 60 years of age, AML treatments are unsuccessful due to principal disease resistance and poor tolerability of intensive treatment protocols. Death from haemorrhage, infection or failure of the heart, kidneys or other organs is more frequent than in younger patients. In elderly patients with serious disease of other organs, the decision may be made to use supportive care with or without single drug chemotherapy. However, combination chemotherapy similar to that used in younger patients may produce long-term remissions (Hoffbrand et al., 2001).

2.2.10 Prognosis

The prognosis for patients with AML has been improving steadily especially for younger patients. Perhaps 50% of children and young adults may require a long-term treatment. Cytogenetic abnormalities and timely response to treatment are predictors of prognosis. In the elderly the situation is worse with, only 5% of those over 65 years of age able to expect long-term remission (Hoffbrand et al., 2001).

Firstly, haematologists examine and count blast cells from blood smear under a microscope and attempt to identify the subtypes of blast cells. They need to diagnose AML based on the cytogenetic testing which takes from three to five days. The process is extremely tiring and time-consuming process. The work in this thesis aims to assist haematologists in detecting and classifying leukaemia cells. A quick diagnosis allows patients to receive treatment properly. Appendix G shows the results of applying this research classification approach, with maximum execution time of two hours.

2.3 Image Processing

Digital image processing has been facilitated in a number of areas in medical practice and applications or research. Naturally, the expression refers to the use of the computer algorithms to perform analysis of digital images. For example digital image processing is used to determine the brightness of stars in pictures from telescopes, to determine the structure of a virus in a microscope image, and to produce highly accurate maps of the earth from satellite-gathered pictures (Niblack, 1985). There are a number of applications in medicine (Scholl et al., 2010), in cartography (Aguilera.

& Lahoz, 2008), printing and publishing, and a host of scientific research fields, including astronomy (Puetter et al., 2005).

Digital images are often subjected to enhancement, for example in order to remove image degradation or to emphasise important image information. At the same time, there is growing demand for computer graphics, where digital images can be combined for a variety of visual effects. There are many different zoom levels at which images can be analysed. Scenes from the world contain objects of varying sizes and features. Moreover, objects can be at different distances from the viewer. As a result, any analysis procedure that is applied only at a single zoom level may miss information. The solution is to carry out analyses at all zoom levels simultaneously (Adelson et al., 1984).

The importance of image processing can be summarised in the popular quote:

"One picture is worth more than ten thousand words."

(Gonzalez & Woods, 2002 pg.1).

Digital images come from several sources, such as the Internet, medical microscopes or satellites. The purpose of digital image processing is to enhance or improve the image or to obtain information from it. Typical tasks to process images include blur removal, noise reduction, improvement of contrast or other visual properties of an image prior to displaying it, segmenting an image into regions such as objects and background, maximisation, minimisation or rotation of an image and so on.

Digital images have two main advantages compared to traditional photographs. First in each generation of a photographic process, there is a loss of

image quality, whereas a digital image can essentially maintain accurate precision. Another advantage is its extreme flexibility in terms of enlargement.

An image may be defined as a two-dimensional function $\kappa(x, y)$, where x and y are special coordinates. The amplitude of κ at any pair of coordinates (x, y) is called grey level of the image at that point. Alternatively an image can be seen as a two-dimensional array of numbers. A digital image is composed of a number of elements each of which has a particular location and value. These elements are referred to as picture elements, image elements and pixels. The term pixel is widely used to denote the elements of a digital image (Figure 2.7).

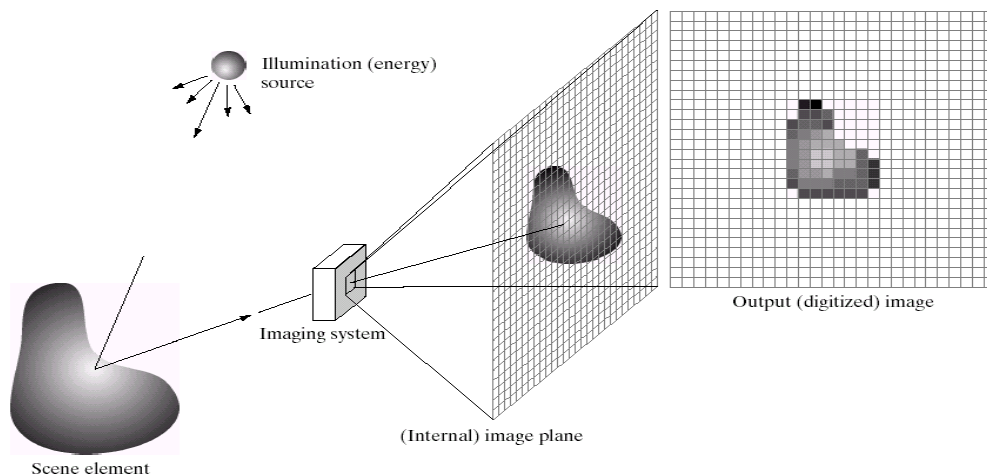


Figure 2.7: Image digitisation (Image Processing and Vision: Introduction, 2011).

The size of the physical area represented by a pixel is called the spatial resolution of the pixel. Each pixel has a value, and coordinates, which reveal its location in the image array. The images used in this thesis are 1280 by 960 pixels. The minimum value a pixel can have is “0”, and the maximum depends on how the

number is stored in the computer. One way is to store each pixel as a single bit, which means it can take only the values of “0” and “1” or black and white as shown in Figure 2.8. Another common way is to store each pixel as a byte, which consists of 8 bits. In this form, the maximum pixel value is 255.

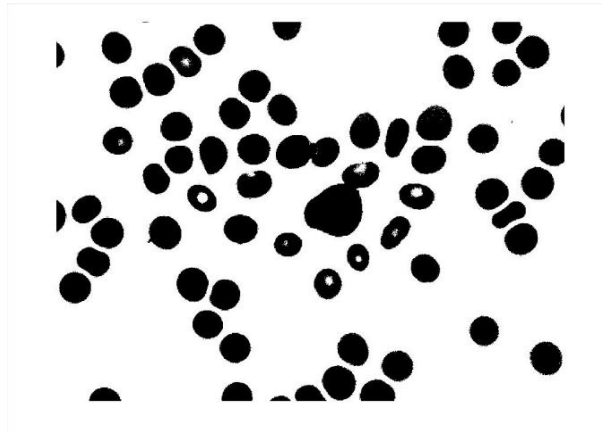


Figure 2.8: Image in Black and White

Colour is an essential and typical representation for images, and a key element for distinguishing objects. In addition, humans perceive various impressions from colour images, such as paintings and photographs. The relationships between colour features and human perception are useful for human-computer interaction. A display of a three band image in which one band is applied to the red, one to the green and one to the blue is called an RGB display, as shown in Figure 2.9 (Niblack, 1985).

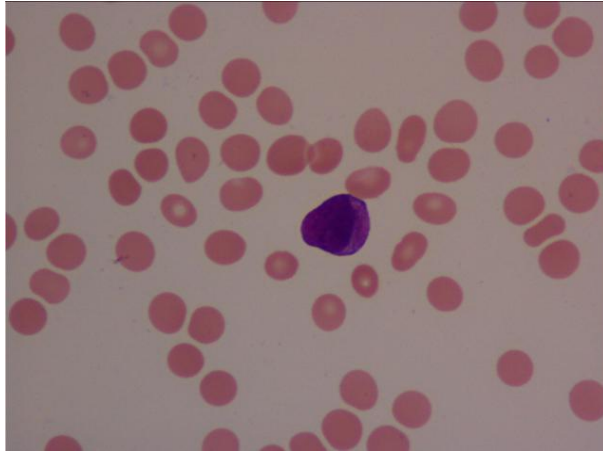


Figure 2.9: Image in Red, Green and Blue

2.3.1 History of Digital Image Processing

The newspaper industry was one of the pioneers in exploiting digital images, when photos were first sent by submarine cable between London and New York. Introduction of the Bartlane cable image transmission system in the early 1920s reduced the time required to transport an image across the Atlantic from more than a week to less than three hours. Specialised printing equipment coded pictures for cable transmission and then reconstructed them at the receiving end. Some of the initial problems in improving the visual quality of these early digital pictures were related to the selection of printing procedures and the distribution of intensity levels in the printing process. Figure 2.10 shows an example of such a picture. The printing method in question was abandoned toward the end of 1921 in favour of a technique based on photographic reproduction made from tapes perforated at the telegraph receiving terminal.



Figure 2.10: A digital picture produced in 1921 from a coded tape by a telegraph printer with special type faces (Liu & Gibbon,. 2010)

In the late 1960s, digital image processing was facilitated in space applications and in the early 1970s in medical imaging, remote earth resources observations and astronomy. The invention of computerised tomography (CT) in the early 1970s is considered a breakthrough in the application of image processing in medical diagnosis. Computational methodologies have improved our ability to analyse and understand images, in medicine and biological science. Typical problems in machine intelligence that routinely utilise image processing techniques are character recognition, military reconnaissance, automatic processing of fingerprints, screening of X-rays and blood samples. The image acquisition stage involves pre-processing, such as scaling. Image enhancement is the simplest step in digital image processing which involves manipulating the image to improve its suitability for a specific application and generate better looking images. Image restoration uses mathematically based techniques or probabilistic models of image degradation to improve the appearance of an image. Compression is dealing with techniques for reducing the storage required to save an image or the bandwidth required to transmit it. Notably, storage technology has improved significantly over the past decade (Gonzalez & Woods, 2002).

As mentioned previously, this thesis is focused on the processing of medical images acquired through the use of a microscope (Figure 2.11). An example of a medical software package used for this purpose is 3D-Doctor. This is advanced 3D modelling, image processing and measurement software for MRI, CT, PET, microscopy, scientific, and industrial imaging applications (Able Software Corp, 2011). Magnification levels typically vary between 10x, 20x, 40x, 60x, 80x and 100x depending on the microscope. Diagnosis of leukaemia is based on the analysis of blood and bone marrow smear under a microscope. Computer technology has made a tremendous impact on medical imaging technology. The recent availability of whole slide digital scanners has made research on pathological image analysis more attractive, by enabling quantitative analysis tools to decrease the evaluation time pathologists spend on each slide. This also reduces the variation in decision-making processes among different pathologists or institutions and introduces reproducibility. The analysis of pathology images is particularly challenging due to the large size of the data (Hartley et al., 2008).



Figure 2.11: Example of a microscope (Natural Biosciences: Autologous Stem Cell Generated Molecular Therapy, 2011)

A blood smear contains five types of white blood cells, plasma and red blood cells. The image in Figure 2.12 shows an example of blood smear image under a microscope with magnification of 100x.

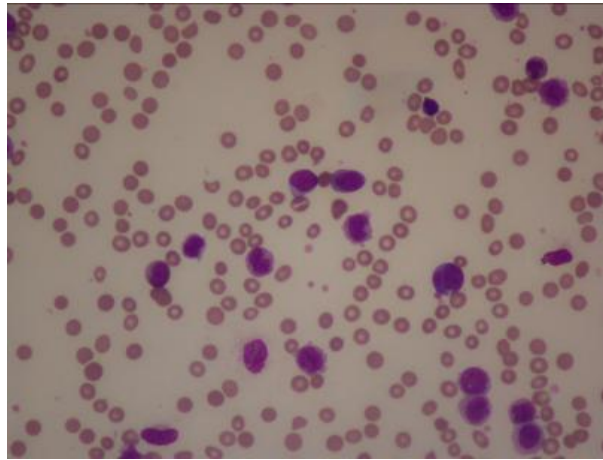


Figure 2.12: Example of microscopic blood image with 100x magnification

Figure 2.13 shows a blood smear image with magnification of 40x from a patient diagnosed with Acute Myeloid Leukaemia (AML).

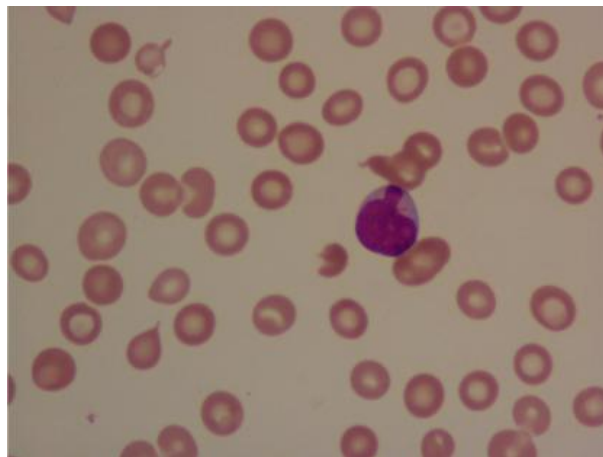


Figure 2.13: Example of microscopic blood image with 40x magnification from an AML patient

Figure 2.14 shows an example of a microscopic image (zoom 40x) of bone marrow smear from an AML patients.

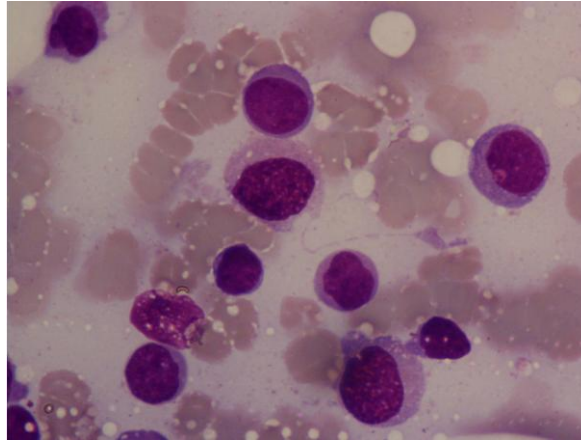


Figure 2.14: Example of a bone marrow image of an AML patient (zoom 40x)

2.3.2 Data Collection

This section provides a discussion of the dataset used in the research presented in the thesis, and the diagnosis made by the haematologist, corresponding to each image. The data consists of 322 real images, 1280 by 960 pixels in size, all from patients suffering from AML. They were provided by the Department of Haematology in the Universiti Sains Malaysia (USM) in Kota Bahru, Kelantan, Malaysia. Ethical approval has been granted by Brunel University (see Appendix A). The subtypes described in the thesis are M1, M2, M3 and M5. Table 2.7 shows the number of available images for each individual subtype. The images were taken with the same lighting and contrast but different microscope zoom magnifications of 100x, 40x and 60x. Examples of images from the dataset are shown in the Figures 2.12, 2.13 and 2.14.

Table 2.7: Data collection for individual subtypes.

| Acute Myeloid Leukaemia | Images collection |
|--------------------------------|--------------------------|
| M1 | 47 images |
| M2 | 129 images |
| M3 | 92 images |
| M4 | 54 images |
| Total | 322 images |

2.3.3 Image Segmentation

Segmentation of images into homogeneous regions is an important field in the research of computer vision. Segmentation can be defined as the partitioning of an image into non-overlapping regions. The goal is to simplify the representation of an image into something more meaningful and easy to analyse. The main image segmentation problem is essentially one of psychophysical perception, and not susceptible to a purely analytical solution. The use of HSL (Hue, Saturation, and Lightness) information consisting of a conversion from a rectangular coordinate system to a cylindrical coordinate system is commonly used. The cylindrical colour coordinate system can represent the brightness and saturation of an image (Hanbury, 2002). Colour image segmentation attracts more attention as firstly, the colour images can provide more information than grey scale images. Secondly, personal computers can be used to process colour images. Unlike monochrome images, segmentation of colour images uses Red, Green and Blue. However, there are not many comprehensive surveys on colour image segmentation. Some areas that have been studied are the application of edge-based and region-based segmentation techniques

to colour images with complex texture (Cheng et al., 2000) and the application of watershed methods (Jiang et al., 2003).

2.3.4 Thresholding

Thresholding is an important technique in image segmentation and machine vision, which is defined as partitioning an image into homogeneous regions. It is considered an analytic image representation method and plays a very important role in many tasks of pattern recognition, computer vision and image and video retrieval (Huang & Chau, 2008).

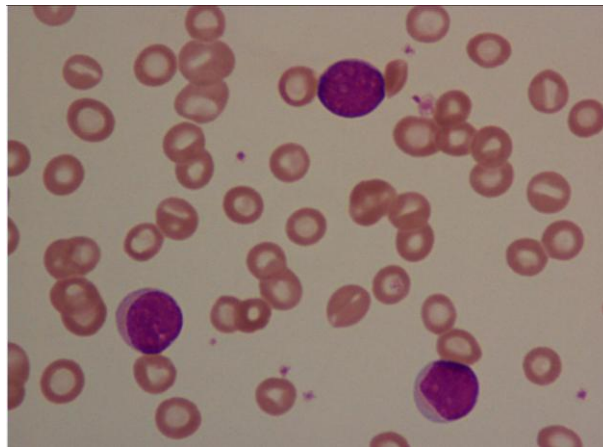
The major problem with thresholding is that it only considers the intensity of pixels, ignoring any relationship between them. There is no guarantee that the pixels identified by the thresholding process are neighbouring. It may include extraneous pixels that are not part of the desired region and at the same time miss isolated pixels within the region (especially near the boundaries of the region). These effects get worse as the noise gets worse, simply because it's more likely that a pixel's intensity does not represent the normal intensity in the region. When thresholding is used, typically have to play with it, sometimes losing too much of the region and sometimes getting many extraneous background pixels (Morse, 2000).

2.3.5 Otsu

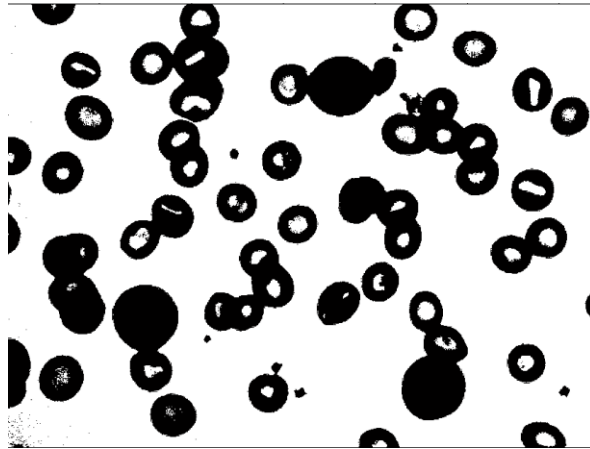
The Otsu method, invented by Nobuyuki Otsu, is well-known in computer vision and image processing. It is used to perform histogram shape-based image thresholding or to reduce of a grey level image to a binary image. The algorithm assumes two classes

of pixels in the analysed image, corresponding to the foreground and background, and looks for the optimal threshold separating those two classes. In image processing, it is often necessary to determine an appropriate threshold of grey level to allow as to extract the foreground from the background of an image. The Otsu method uses a histogram to represent the foreground and background and then uses the valley point as the threshold. This utilises information concerning neighbouring pixels (or edges) in the original image to change the histogram to make it useful for thresholding. The method deals directly with the problem of evaluating the suitability of a threshold from the viewpoint of discriminant analysis, automatically selecting an optimal threshold (Otsu, 1979).

In this thesis, grey level thresholding is implemented for (R,G,B) colour images. Figure 2.15 (a), shows a real image, (b) an image processed with the Otsu method without thresholding, and (c) an image processed with the Otsu method with thresholding using grey level.

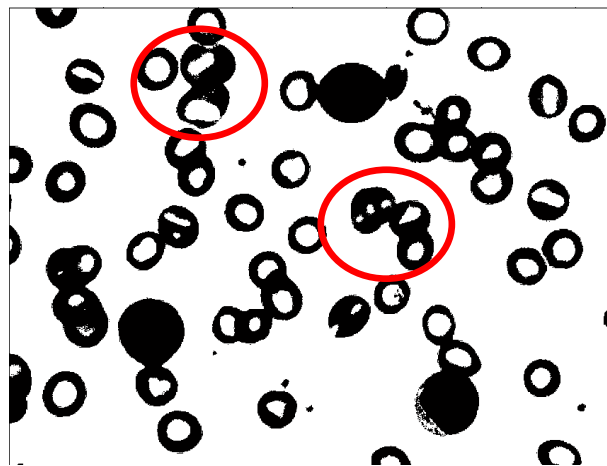


(a) Real Image



(b) Otsu Method without Threshold

Image 2.15 (c) is less noisy and better than image (b). The Otsu method is a smoothing filter, used to reduce the noise, details or “busyness” in an image. Moreover, this study is focusing on detecting and classification of leukaemia cells. The red circle in Figure 2.15 (c) shows an example of how using a threshold produces better results than not doing so (Figure 2.15 (b)).



(c) Otsu Method with Threshold

Figure 2.15: (a) Real Image, (b) Processed with Otsu Method without threshold, (c) Otsu Method with threshold

2.3.6 Otsu Notation

This section provides some useful notation, adapted from the Otsu Method. Here use I colour images. Let $I(x,y)$ be the coordinates of each pixel in an image. Otsu's Method assumes two classes, C_0 and C_1 (background and objects or foreground respectively). The threshold T , allowing the best separation of classes in grey levels, would be the best threshold (Otsu, 1979).

$$I(x, y) = \begin{cases} 1, & \text{if } I(x,y) \geq T \\ 0, & \text{Otherwise} \end{cases} \quad (2.1)$$

Let the pixels of a picture be represented in L grey levels $[1, 2, \dots, L]$. The number of pixels at level g is denoted by n_i and the total number of pixels by $N = n_1 + n_2 + n_3 + \dots + n_L$. Pixels belong to two classes C_0 and C_1 separated by a grey scale threshold T . Pixels with intensity level $[1, \dots, T]$ belong to class C_0 , while pixels with intensity $[T + 1, \dots, L]$ to class C_1 . Hence, pixels with intensity below the threshold T as grey scale defined as partitioning an image between "black", and "white" otherwise.

2.4 Cellular Automata (CA)

Cellular Automata was implemented here for the automatic detection of blast cells. A number of papers relevant to the use of CA in image processing and especially in medical applications are discussed in chapter 4. CA was invented by John Conway. This method is based on a two dimensional grid of binary cells that can be in one of two possible states, 0 or 1 (Griffeath & Moore, 2003).

2.4.1 Cellular Automata concepts

A Cellular Automaton is popular model in natural science, combinatorial mathematics, and computer science. CA is decentralised spatially extended systems of large numbers of simple identical components with local connectivity. A CA consist of two components. The first component is a cellular space where each cell exhibits an identical pattern of local connections to other cells for input and output, along with boundary conditions. The second component is a transition rule or CA rule that gives the update state for each cell. CA are based on the idea of neighbourhood known as Von Neumann neighbourhood, shown in Figure 2.16 and Moore neighbourhood shown in Figure 2.17 (Mitchell,. 1996).

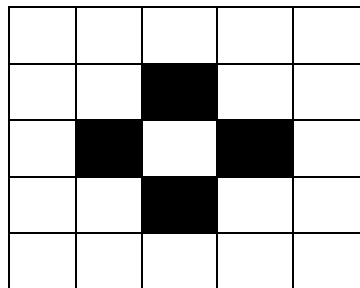


Figure 2.16: Von Neumann Neighbourhood

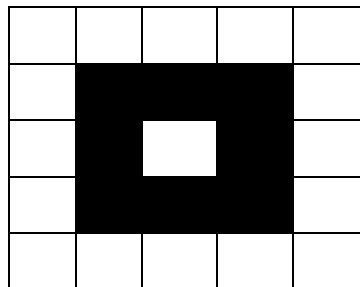


Figure 2.17: Moore Neighbourhood

CA were first perceived by Stan Ulam and further developed by John Von Neumann to provide a workable model of the behaviour of a complex and extended

system. Stan Ulam proposed the idea of self-reproducing structures within the cellular automata. It was then discovered that similar principals are employed by biological systems. Later Von Neumann's continued Stan Ulam's work on self-reproducing automata. In computer science, CA are used to model parallel processing and Von Neumann (self-reproducing) machines. Throughout his life Van Neumann's made two important contributions, one involving the technology of death (weaponry) and the other relative to the technology of life. In particular Von Neumann realised that biology offered the most powerful information-processing system, and that its emulation would be the key to a strong artificial system. In his work on computer design, Von Neumann viewed the different parts of a computer as organs (Russel & Norvig,. 2003).

In 1970, the mathematician John Conway invented a program called LIFE. The game of LIFE may be thought of as describing a population of organisms, developing through time under the effect of counteracting propagation and elimination tendencies. This rationale is based on the idea that a piece on the grid may remain present (in which case the organism "survives"), may be removed from the grid (in which case the organism is considered to have "died") or a new piece may be placed on the grid (or "born"). An individual is represented by a cell that can have a value of 1, corresponding to "alive", or 0, corresponding to "dead" (Toffolli & Margolus, 1987).

CA is based on a physics theory which has been adopted and developed in a computer science. It can also be seen as a stylised universe which is used to model biological systems from the level of cell activity to the levels of clusters of cells and populations of organisms. CA is discrete space-time models that can be used to model many complex systems. Space in a CA model is represented as uniform lattice of

cells containing bits of information. As time advances in discrete steps cells are subjected to the rules of the universe. Each one computes its state from that of its closest neighbours. Thus, the system's law is local and uniform (Toffoli & Margolus, 1987). Wolfram was one of the first investigators of the class of elementary CA and introduced the naming system for one-dimensional CA, known as Wolfram code. In his book "A New Kind of Science" he stated the following:

"Traditional intuition might suggest that to do more sophisticated computations would always require more sophisticated underlying rules. However, what launched the whole computer revolution is the remarkable fact that universal systems with fixed underlying rules can be built that can in effect, perform any possible computation. The threshold for such universality has however generally been assumed to be high, and to be reached only by elaborate and special systems like typical electronic computers. However in fact, there are systems whose rules are simple enough to describe in just one sentence that is, nevertheless, universal. And this immediately shows that the phenomenon of universality is vastly more common and important in both abstract systems and nature than has been ever been imagined before." (Ganguly et al., 2003 pg 10).

2.4.2 Manhattan Distance

The Manhattan Distance metric was facilitated in this work to estimate the radius of potential blast cells. This metric, also known as taxicab or as city block distance, was proposed by Hermann Minkowski in the 19th century when he developed taxicab geometry (Krause, 1987).

The Manhattan distance function computes the distance that would be traveled to get from one data point to the other if a grid like the path in Figure 2.18, indicated in red, is followed. Hence, the Manhattan distance between two points is the sum of the absolute differences of their coordinates.

The formula for this distance between a point $x = (x_1, x_2, \dots, x_n)$ and a point $y = (y_1, y_2, \dots, y_n)$ is $d = \sum_{i=1}^n |x_i - y_i|$ where n is the number of variables, and x_i and y_i are the values of the i th variable, at points x and y respectively. Figure 2.18 shows an example of the Manhattan distance between points (x) and (y) , where the path is indicated in red colour.

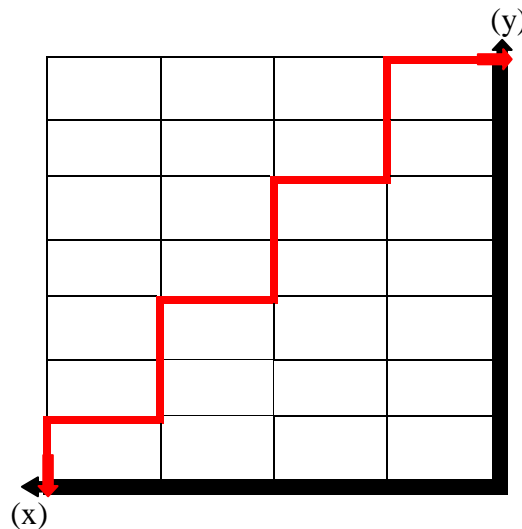


Figure 2.18: Manhatttan Distance between (x) and (y)(red colour)

2.4.3 Euclidean Distance

Euclidean Distance between two points is the length of the straight line connecting them. In the euclidean plane, the distance between points (x_1, x_2) and (y_1, y_2) is given by equation (2.2) (Weisstein, 2011). Figure 2.19 graphically portrays the Euclidean distance between points (x) and (y).

$$ED(x_1, y_1, x_2, y_2) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \quad (2.2)$$

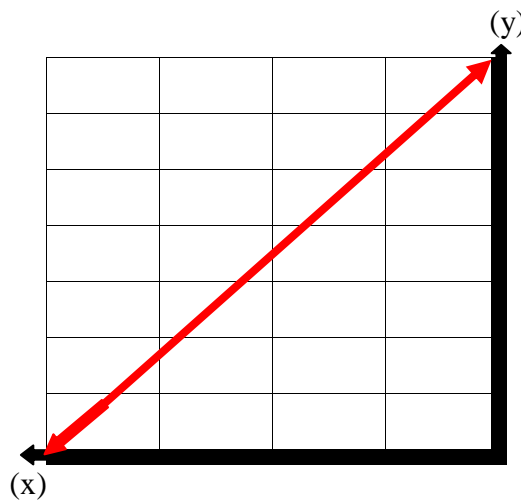


Figure 2.19: Illustration of Euclidean Distance

2.5 Artificial Intelligence

Artificial Intelligence is used for logistics, data mining, medical diagnosis, expert system and the other areas in the technology industry. For example the MYCIN is an early expert system designed to identify bacteria causing severe infections, developed at Stanford University in the mid 1970s. It was designed to aid physicians in the diagnosis and treatment of meningitis and bacteraemia infections and was the first

large expert system to perform at the level of a human expert and to provide users with an explanation of its reasoning (Harmon & King, 1985).

In the 1990s and early 21st centuries, Artificial Intelligence achieved its greatest successes. In theory Artificial Intelligence can solve many problems by searching through a number of possible solutions. But not all AI can be solved using learning algorithms which called as clustering. Some of the solution required learning method which is user decision called as classification. Classification procedure involved formal method which some decision or forecast is made based on the current information. There are several of terms that used in the construction of a classification procedure such as patter recognition, discrimination or supervised learning. Given the huge number of possible solutions in a variety of problems, "heuristics" often constitute the best approach to look for an optimal solution. Search techniques based on the mathematical theory of optimization became popular in the 1990s.

Hill Climbing and Simulated Annealing, implemented in this thesis to optimise the search for blast cells, constitute two characteristic examples of a heuristic search. These techniques are discussed in some details in chapters 3 and 6. Genetic algorithm constitutes another category of optimisation algorithms also used in the thesis. These search approaches are compared in an effort to identify the one exhibiting the best performance for the purposes of this work.

In general, the simplest Artificial Intelligence applications can be divided into two types, that is, classifiers and controllers. Classification is a supervised learning methodology that seeks to match a given pattern to a predefined group, hence classifying objects into known categories or classes. A classifier can be trained in

different ways. Statistical and machine learning approaches are used in training a classifier. Neural networks are perhaps the most widely used classifiers. The classifier's performance depends on the characteristics of the data to be classified. Artificial Neural Network had been studied prior to the emergence of the field of Artificial Intelligence, by Walter Pitts and Warren McCulloch. Such networks have also been facilitated in this thesis, as further discussed in chapter 7 (Russel & Norvig, 2003). There are two main categories of neural networks, the feed forward and the recurrent one.

2.5.1 Machine Learning

In 1956 at the Artificial Intelligence summer conference in Dartmouth (Russel & Norvig., 2003), machine learning was at the forefront of Artificial Intelligence research. Machine Learning is a subfield in artificial intelligence, concerned with the design and development of algorithms that allow computers to evolve their behaviour based on empirical data, such as databases. The field of machine learning is related to information theory, stochastic modelling, classification, optimisation and statistic analysis. A major focus of machine learning research is to automatically learn to recognise complex patterns and make intelligent decisions based on data. The difficulty lies in the fact that the set of all possible behaviours given all possible inputs is too large to be covered by the set of observed examples (training data). The representation of the learned information also plays an extremely crucial role in determining the performance of the learning algorithm. The last prime factor in the design of learning systems is the availability of prior knowledge. Object recognition, is the process of converting features of the image into a model of known objects.

Object recognition consists of three steps including segmentation, orientation to the observer and determining the shape of each object (Russel & Norvig, 2003).

There are many scientific applications. In biology, machine learning is used to help identify the thousands of genes within each new genome. In biomedicine, it is used to predict drug activity by analysing the chemical properties of drugs and their three-dimensional structure. In chemistry, it has been used to predict the structure of certain organic compounds from magnetic resonance spectra (Witten & Frank, 2005).

The work presented in this thesis exploits machine learning concepts to identify the subtypes of blast cells present in the analysed images. At the same time, machine learning is used to automate the process of blast cell detection.

2.5.2 Classification

Nowadays, classification is widely used in pattern recognition. Pattern recognition deals with information-processing applications from speech recognition and the classification of handwritten characters to fault detection technologies and medical diagnosis. Classification is performed by algorithms that attempt to assign a given piece of input into a number of categories. Medical classification (Antonie et al., 2001) and texture analysis (Chang & Jay Kuo, 1993) are just two examples. WEKA is free, publicly available, classification software. It is a data analysis software tool which implements a set of machine learning algorithms for data mining tasks (Mark et al., 2009).

KAPPA Statistic measure interobserver variation in any situation in which two or more independent observers are evaluating and classifying the same data. The calculation is based on the difference between how much agreement is actually present (“observed” agreement) compared to how much agreement would be expected to be present by chance alone (“expected” agreement). Kappa is a measure of this difference, standardized to lie on a -1 to 1 scale, where 1 is perfect agreement, 0 is exactly what would be expected by chance, and negative values indicate disagreement that is not due to chance. Cross-validation is another technique that reduces over fitting implemented in WEKA. The basic idea is to estimate how well each hypothesis will predict unseen data. This is done by setting aside some fraction of the known data and using it to test the prediction performance of a hypothesis induced from the remaining data. A commonly cited scale for KAPPA is represented in Table 2.8 (Viera & Garret, 2005).

The table highlights the Kappa statistic is for two binary variables, sometimes referred to as Cohen’s Kappa. Kappa measures the percentage of data values in the main diagonal of the table and then adjusts these values for the amount of agreement that could be expected due to chance alone. The Kappa statistic is implemented in Chapter 7.

Table 2.8: KAPPA Table (Viera & Garret, 2005)

Interpretation of KAPPA

| Kappa | Agreement |
|--------------|----------------------------|
| < 0 | Less than chance agreement |
| 0.01 – 0.20 | Slight agreement |
| 0.21 – 0.40 | Fair agreement |
| 0.41 – 0.60 | Moderate agreement |
| 0.61 – 0.80 | Substantial agreement |
| 0.80 – 0.99 | Almost perfect agreement |

2.5.3 Artificial Neural Networks (ANN)

Over the past decade, there has been growing interest in Artificial Neural Networks (ANN), mostly in the area of feed forward networks for pattern recognition applications. There are two basic categories of classification methods, which are supervised and unsupervised classification. Neural networks were invented in the early 1940s, by McCulloch and Pitts. They proposed neuron models in the form of binary threshold devices and stochastic algorithms, inspired by the structure and functional aspects of biological neural networks. Subsequently, Hebb proposed mathematical models that attempted to capture the concept of learning by strengthening or association. During the mid-1950s and early 1960s, a discipline called Machine Learning attracted interest among researchers and practitioners of pattern recognition theory. Rosenblatt invented the perceptron, a simple linear classifier. It was shown that when trained with separable training sets, perceptrons converge to a solution in a finite number of iterative steps. This was followed by attempts to consider multiple layers of the devices. Although conceptually appealing, they lacked effective training algorithms such as those that had created interest in the perceptron itself. A few years later, Minsky and Papert presented a discouraging analysis of the limitations of perceptrons. More recent results by Rumelhart, Hinton and Williams, dealing with the development of new training algorithms for multilayer perceptrons, have generalised the delta rule for learning by back propagation, providing an effective training method for multilayer machines. Although the training algorithms cannot be shown to converge to a solution of the analogous proof for the single-layer perceptron, the generalised delta rule was used successfully in several problems of practical interest. This success has established multilayer perceptron – like machines as one of the principal models of neural networks currently in use (Gonzalez & Woods, 2002).

Artificial Neural Networks are innovative models of human cognitive function. The artificial equivalents of biological neurons are the nodes or units of a neural network. The human brain consists of an estimated 10^{11} (100 billion) nerve cells or neurons (Figure 2.20) (Russel & Norvig,. 2003). The term “network” is used to refer to any system of artificial neurons. A neuron is a cell in the brain whose main activity is the collection, processing and dissemination of electrical signals. The brain’s information-processing function is thought to emerge from networks of such neurons. Neurons communicate through electrical signals that are short-lived impulses or “spikes” in the voltage of the cell wall or membrane. The interneuron connections are mediated by electrochemical junctions called synapses that are located on branches of the cell referred to as dendrites.

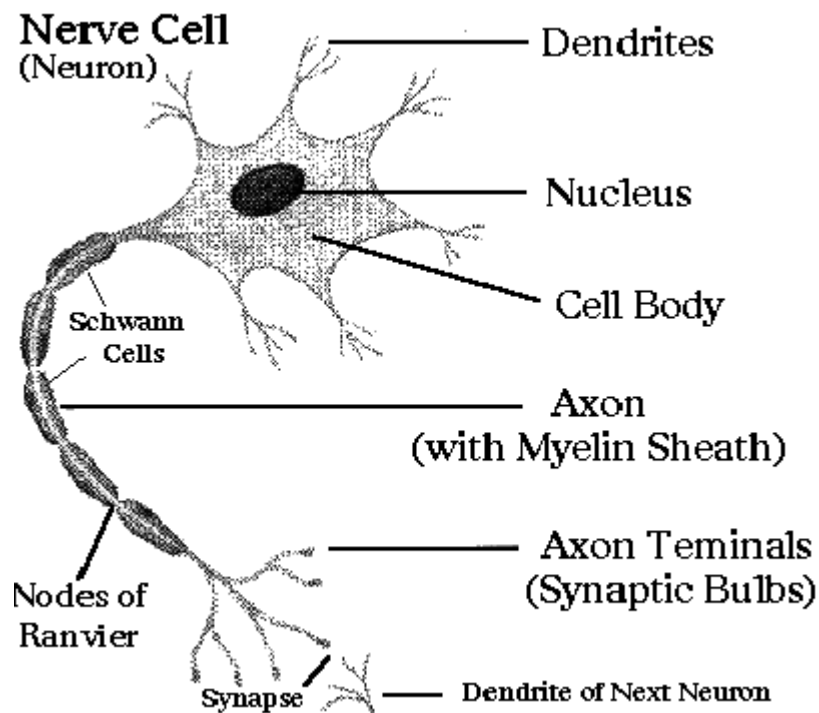


Figure 2.20: A neuron (The Nervous System,. 2011).

Each neuron is typically connected to many thousands of other neurons and continually receives a multitude of incoming signals, which reach the cell body. Signals are integrated or summed together and if the resulting signal exceeds some threshold, it causes the neuron to “fire” or generate a voltage signal in response. Then it transmits this signal to other neurons through a branching fibre known as the axon. It is this structure and method of processing that neural networks mimic. Each synapsis or connection has a weight, hence each input is multiplied by its weight before being sent to the equivalent of the cell body. The weighted signals are summed together by basic arithmetic addition to cause node activation. This may range from something as basic as a single node to a large set of nodes, each one connected to every other node in the net. They are nodes arranged in a layered structure in which each signal from an input passes to two nodes before reaching an output. In real neurons, the synaptic strengths may, under certain circumstances, be modified so that the behaviour of each neuron can change or adapt to its unique stimulus input. In artificial neurons, the equivalent of this is the connection weight value. The “knowledge” of a network is supposed to be stored in its weights (Gurney, 1997).

A neural network structure consists of nodes that represent a processing unit with relationships between the units that are indicated by the arcs shown in Figure 2.21. There is a number of processing units called hidden units. Each hidden unit must be used internally to be connected to the input and output units, using weighted connections and the network must be trained (Freeman & Skapura, 1992). Neural networks are an attempt to build machines that operate in the same way as the human brain.

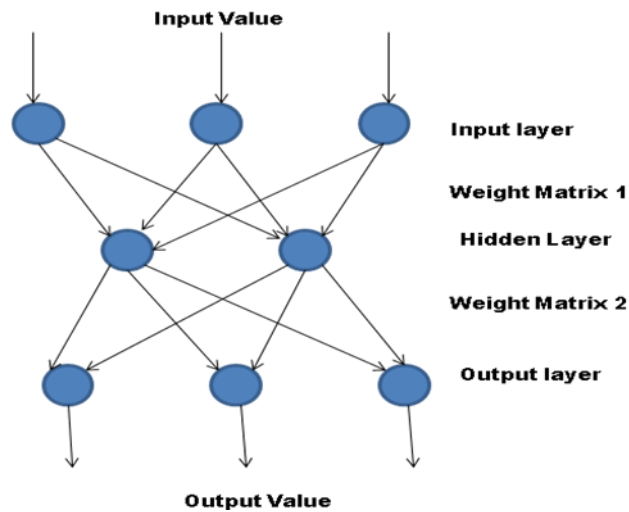


Figure 2.21: The architecture for a Neural Network

Neural network structures belong to two main categories, feed-forward networks and recurrent networks. A feed-forward network represents a function of its current input and weights. A recurrent network feeds its output back into its own input. This means that the activation levels of the network build a dynamical system that may influence a steady state or behaviour. Moreover, the response of the network given input depends on its initial state and previous inputs. Hence, recurrent networks can help short-term memory. This makes them more engrossing as models of the brain and difficult to understand. Feed-forward networks are arranged in layers so that each unit receives input units from the immediately preceding layer. Single-layer neural network or a perceptron network is a network with all the inputs connected directly to the outputs. Each output unit is independent of the others and each weight affects only one of the outputs (Russel & Norvig, 2003). The function of a neural network is to produce an output pattern when presented with a suitable input pattern (Picton, 1994). The “knowledge” in the network is distributed over the entire set of weights, rather than in a few memory locations as in a conventional computer. Developers are more concerned with the implementation of a result in digital

hardware, than with the efficiency and accuracy of specific techniques (Gurney, 1997).

Neural networks are used to solve problems in pattern classification, speech recognition, textual character recognition and fields of human knowledge such as medical diagnosis, geological survey of oil and financial market forecasts (Gurney, 1997). Pattern classification is the process of sorting patterns into one group or another in a large number of cases (Picton, 1994). In image classification the aim is to assign a set of objects to one of a set of classes. The objects are the pixels in an image, and the classes are the different categories such as foreground and background. Classification may also be thought of as a labeling problem where each object needs to be labeled with a class type. There are two basic categories of classification methods, supervised and unsupervised. In the supervised methods, the user “supervises” the process by initially selecting some pixels from each possible class. The classification algorithm attempts to determine the class of each of the remaining unassigned pixels. In the unsupervised methods, the classes are determined by the algorithm and the problem becomes one of cluster identification. A final step in the unsupervised case, which is performed by the user, is to determine the nature of the clusters. Methods of unsupervised classification try to find clusters in the distribution of the pixels in pixel space. Each pixel in the image is assigned to the class it is closer to. The mean of each class is recomputed as the average of the pixels assigned to it. However, supervised classification is not a fixed, automatic procedure, but an operation involving much interaction between the user and the computer system, and requiring many human decisions (Niblack, 1985). In feature extraction, the agent finds some small number of features in their input and passes them directly to their agent program, which can act reactively to the features, or combine them with some other information (Russel & Norvig, 2003).

2.5.4 Multilayer Perceptrons

The Multilayer Perceptron was introduced by M.Minsky and S.Papert in 1969. A Multilayer Perceptron is a feedforward artificial neural network model that maps a set of input data onto a set of appropriate output. A Multilayer Perceptron consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. A Multilayer Perceptron has either a simple threshold or sigmoidal activation function, which play a central role in the majority of training algorithms for multi-layered networks.

A Multilayer Perceptron provides a general framework for representing non-linear mappings from a set of input variables to a set of output variables. This is achieved by representing the non-linear function of many variables in terms of compositions of non-linear functions of a single variable, called activation functions. Each multivariate function can be represented in terms of a network diagram such that there is a one-to-one correspondence between the model components and the elements of the diagram (Bishop, 1995). Equally, any topology of a network diagram, provided it is feed-forward, can be translated into the corresponding mapping function. The units which are not treated as output units are called hidden units.

$$a_m = \sum_{i=1}^q E_{mh}^1 x_q + E_{m0}^1$$

Where E_{mh}^1 denotes the weight of the first layer, going from input h to hidden unit m , E_{m0}^1 denotes the bias of hidden unit and q is the number of nodes. The output of the network is obtained by transforming the activation of the hidden units using a second layer of processing elements. A network is feed-forward if it is possible to attach successive numbers to the inputs and to all the hidden and output units such

that each unit only receives connections from inputs or units having a smaller number (Bishop, 1995).

2.6 Heuristic Search

This thesis facilitated heuristic search methods for the detection of blast cells. In particular, three methods were used, namely HC, SA and a GA. Heuristics are generally used to find an approximate, “good” solution in large-size problem instances. In general, heuristics do not find the real best solution. However, they are designed to solve an actual problem well enough (Talibi, 2009). Heuristics are a popular approach to finding “acceptable” solutions for solving complex problems, in science and engineering. The word heuristic has its origin in the ancient Greek word “heuriskein” meaning the art of discovering new strategies or rules to solve problems (Talibi, 2009). Over the last decade, there is increasing interest in these kind of techniques, and a number of novel approaches have emerged (Reeves, 1993).

2.6.1 Heuristic Search Concepts

Nowadays, there is growing demand for solving real-world optimisation problems in medical image processing. Heuristics are a popular way of addressing difficult problems, in particular, because of their simplicity and speed (Falkenauer, 1998), and in cases of problems that do not fit the “standard” search type.

The range of possible solutions can be seen as a "landscape" where each solution is a "location". While looking for a better solution the search can exploit a so called fitness function, cost function or objective function. Depending on the problem

in hand, we might be looking for a global optimum (peak) or a global minimum (valey) in the landscape of solutions. Local search algorithms explore this landscape. A complete local search algorithm always finds a target if one exists; an optimal algorithm always finds a global minimum/maximum (Russel & Norvig, 2003).

Each specific problem is accompanied by a unique search space. Different "landscapes" are created by the different search operators used to search it. An important concept here is that of a neighbourhood, that is, solutions that do not differ significantly. For example, in genetic algorithms the solution is often represented by a binary vector, and a neighbourhood may be seen as consisting of all vectors differing in only one bit (Reeves, 1999).

The quality of the fitness function determines how well a program can solve a certain problem. Importantly, there are two main classes of problems, those where the fitness function does not change and those where the fitness function changes during the search process.

Heuristic search falls into several broad categories, which include greedy construction methods, local neighbourhood search, relaxation techniques, partial enumeration, decomposition and partition approaches. Local neighbourhood search is one of the most powerful approaches (Reeves, 1993). This research implemented local neighbourhood search methods in order to optimize the size of a circle capturing potential leukaemia cells.

The process of decision making, summarised in Figure 2.22, in the development of optimisation models is the following (Talibi., 2009):

- Formulate the problem

In the first step, the decision problem is identified. An initial statement of the problem is made, since formulation may be uncertain. In this project the problem is to find the size of the circle encompassing a cell.

- Model the problem

The next step is to build an abstract mathematical model for the problem. For our purposes eleven fitness functions were tested to select the best one.

- Optimise the problem

Once the problem has been modelled, the solving method is applied to generate possible solutions. Here, HC, SA and a GA were implemented.

- Testing a solution

The obtained result was practically tested by the decision maker and implemented if it is “acceptable”. If the result is unacceptable, the model or the optimisation algorithm has to be improved, and the decision-making process repeated.

- Implement the solution

After the process from model, then optimisation and testing finished. The final stage was implemented the model and optimisation to the real problem. In the thesis where defined the fitness function by using the HC, SA and GA and the testing for 20 images. After finding the “best” fitness function and the “best” optimisation then implement it the real data.

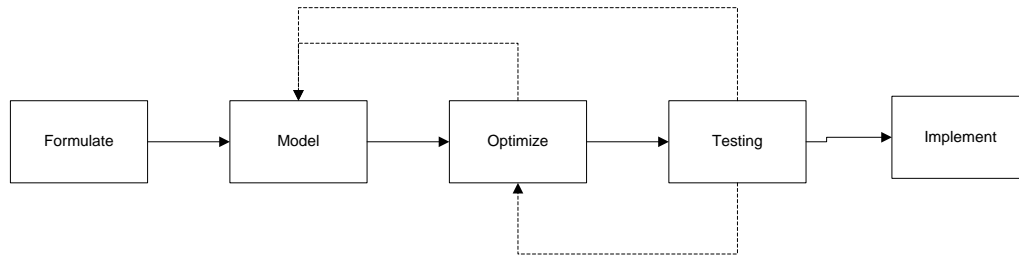


Figure 2.22: The classical process of decision making in an optimisation model.

As already discussed, three distinct methods were examined in this work, HC, SA and a GA in order to optimise the methodology of blast cell detection. Importantly, unlike HC and SA which work with a single solution search, GA works with population-based solutions.

An appropriate analogy to envisage a local search is that of a hiker trying to find his or her way to the top of a mountain, without visibility. The hiker feels the ground around and moves to higher ground. If a step takes him lower he goes back and tries another direction. Figure 2.23 shows a graphical representation of the search space (Thollesson, 2011).

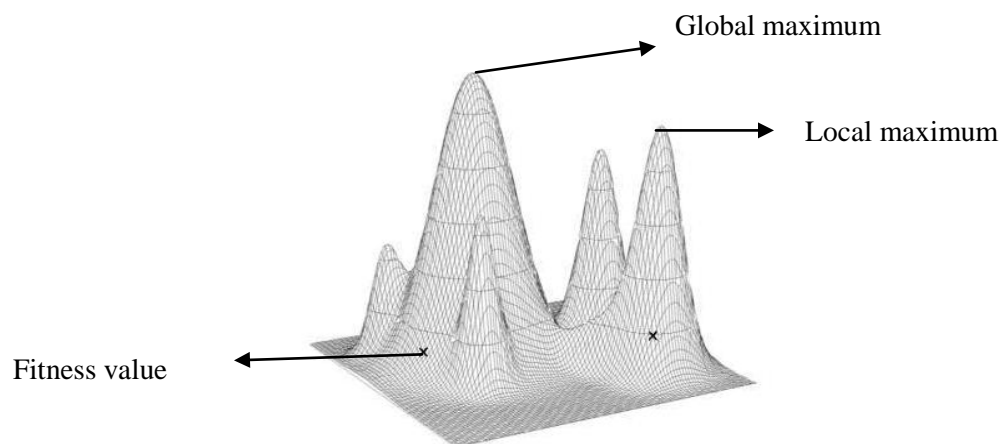


Figure 2.23: Search Space concept illustration (Thollesson, 2011)

In this thesis three heuristic search methods, Hill Climbing, Simulated Annealing and Genetic Algorithm were implemented. Table 2.9 identifies the differences of the three methods.

Table 2.9: Differences between Hill Climbing, Simulated Annealing and Genetic Algorithm

| Hill Climbing (HC) | Simulated Annealing (SA) | Genetic Algorithm (GA) |
|---|--|---|
| HC is a traditional method of local search. It often fails to find the best global solution and tends to get stuck in local optima, finding the best solution in the close neighbourhood. | In the SA algorithm, a newly selected point or solution is “accepted” with some probability p . This means that the rule of moving from the current point to the new neighbours is probabilistic. SA is also a popular traditional search method which performs better than the HC in finding a global optimal solution. | GA is a stochastic global search method that mimics the process of natural biological evolution. It implements selection, crossover and mutation operators. GA is suitable for global search. There are differences between GAs and the traditional methods (HC and SA) which are discussed in section 2.6.5. |

2.6.2 Hill Climbing (HC)

HC search algorithms, sometimes called greedy local search, continually move in the direction of increasing (or decreasing) fitness value, that is, uphill. It terminates when it reaches a “peak”. It grabs a close neighbour state without thinking where to go next. HC makes extremely rapid progress towards a solution. However, HC

algorithms often fail to find the global best solution because they can get stuck in local optima. Below are some of the main reasons causing HC algorithms to get stuck (Russel & Norvig, 2003):

- Local maxima
A local maximum is a peak that is higher than each of its neighbouring states, but lower than the global maximum.
- Ridges
A ridge results in a sequence of local maxima that is very difficult for greedy algorithms to navigate.
- Plateau
A plateau is an area of the state space landscape where the evaluation function is flat. It can be a flat local maximum, from which no uphill exit exists or a shoulder, from which it is possible to make progress. A HC search might be unable to find its way off the plateau.

HC is good for finding a local optimum which a solution that cannot be improved by considering a neighbouring configuration, but it is not guaranteed to find the best possible solution. In some problems, HC works better than advanced algorithms such as SA or Tabu Search. Figure 2.24 shows an example of a HC convergence graph. One can see that the algorithm converges after about 1700 iterations.

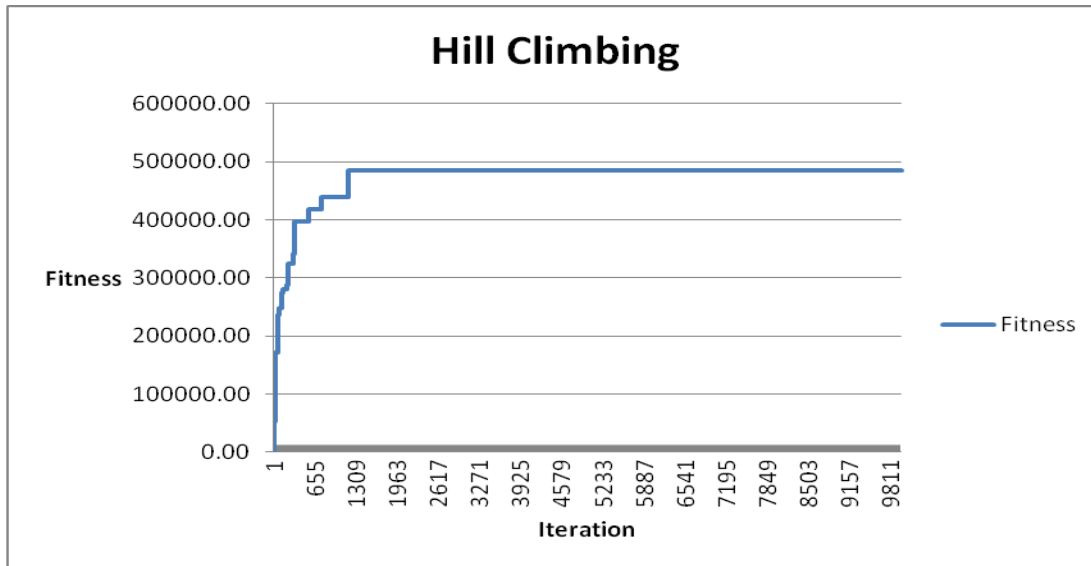


Figure 2.24: HC Convergence graph

Below is the pseudocode for the HC algorithm (Michalewicz & Fogel, 2004). HC algorithms differ mainly in the way a new solution is selected for comparison with the current solution. Initially, all possible neighbours of the current solution are considered, and the one V_n that returns that best value $eval(V_n)$ is selected to compete with the current point V_c . If $eval(V_n)$ is worse than $eval(V_c)$, then the new point, V_n becomes the current point. Otherwise, no local improvement is possible: the algorithm has reached a local or global optimum ($local = TRUE$). In such a case, the next iteration ($t \leftarrow t+1$) of the algorithm is executed with a new current point selected at random.

```

procedure Hill-Climbing
(1) begin
(2)  $t \leftarrow 0$ 
(3) initialise best
(4) repeat
(5)   local  $\leftarrow$  FALSE
(6)   select a current point  $v_c$  at random
(7)   evaluate  $v_c$ 
(8)   repeat
(9)     select all new points in the neighbourhood
        of  $v_c$ 
(10)    select the point  $v_n$  from the set of new
        points with the best value of evaluation
        function eval
(11)    if eval ( $v_n$ ) is better than eval ( $v_c$ )
(12)      then  $v_c \leftarrow v_n$ 
(13)    else local  $\leftarrow$  TRUE
(14)  until local
(15)  $t \leftarrow t + 1$ 
(16) if  $v_c$  is better than best
(17)  then best  $\leftarrow v_c$ 
(18) until  $t = MAX$ 
(19) end

```

Algorithm 2.1: HC algorithm

2.6.3 Simulated Annealing (SA)

SA is a probabilistic method proposed by Kirkpatrick, Gelett and Vecchi (1983) and Cerny (1985) for finding the global minimum or maximum of a cost function that may possess several local minima or maxima. (Dimistris & Tsitsiklis., 1993)

As mention in (Van Larrhoven & Aarts, 1987):-

“The SA algorithms are based on the analogy between the simulation of the annealing of solids and the problem of solving large combinatorial optimization problems. In theory of the physics, annealing denotes a physical process in which a

solid in a heat bath is heated up by increasing the temperature of the heat bath to a maximum value at which all particles of the solid randomly arrange themselves in the liquid phase, followed by cooling through slowly lowering the temperature of the heat bath. In this way, all particles arrange themselves in the low energy ground state of a corresponding lattice, provided the maximum temperature is sufficiently high and the cooling is carried out sufficiently slowly. However, it is well known that if the cooling is too rapid, i.e. if the solid is not allowed to reach thermal equilibrium for each temperature value, defects can be “frozen” into the solid and meta stable amorphous structure can be reached rather than low energy crystalline lattice structure. Furthermore, in a process known in condensed matter physics as quenching, the temperature of the heat bath is lowered instantaneously, which results again in freezing of the particles in the solid into one of the metastable amorphous structures”. (Van Larrhoven & Aarts, 1987 pg 10).

The usefulness of SA algorithm lies in the fact that it is able to escape local minima and maxima. Several parameters need to be included in an implementation of SA. These are summarized nicely by David and Harel (Davidson & Harel, 1996):

- a) The set of configurations the search space where there are random point will be chosen as starting points, or states of the system, including an initial configuration (which is often chosen at random).
- b) A general rule for new configurations, which is usually obtained by defining the neighbourhood of each configuration and choosing the next configuration randomly from the neighbourhood of the current one.
- c) The target, or cost, function, to be minimised over the configuration space (This is the analogue of the energy)

- d) The cooling schedule of the control parameter, including initial values and rules for when and how to change it. (This is the analogue of the temperature and its decreases).
- e) The termination condition, which is usually based on the time and the values of the cost function and/or the control parameter.

Figure 2.25 shows an example of the convergence of a SA algorithm.

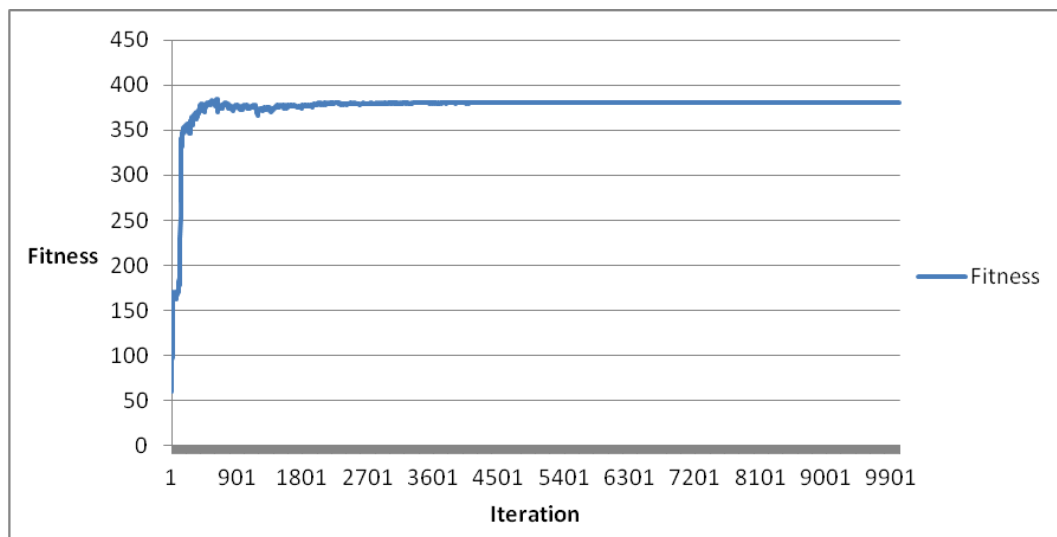


Figure 2.25: SA convergence graph

Below is the structure of the SA algorithm from (Michalewicz & Fogel, 2004). There are a number of distinguished features. First, the stochastic SA has only one loop. In the SA, there is no need to repeat iterations starting from different random points. Second, the newly selected point is “accepted” with some probability p . This means that the rule of moving from the current point V_c to the new neighbours, V_n , is probabilistic. It is possible for the new accepted point to be worse than the current point. However, the probability of acceptance depends on the difference in merit between these two competitors for example $eval(V_c) - eval(V_n)$, and

on the value of an additional parameter T . T remains constant during the execution of the algorithm.

```

procedure simulated annealing
  Input:  $T_0$    Starting temperature
        Iter   Number of iterations
         $\lambda$    The cooling rate
  (1) Set  $T \leftarrow T_0$ 
  (2) Let  $x \leftarrow$  a random solution
  (3) For  $i \leftarrow 0$  to Iter-1
  (4)   Let  $f \leftarrow$  Fitness of  $x$ 
  (5)   Make a small change to  $x$  to make  $x'$ 
  (6)   Let  $f' \leftarrow$  fitness of new point  $x'$ 
  (7)   If  $f'$  is worse than  $f$  then
  (8)     Let  $p \leftarrow PR(f', f, T_i)$ 
  (9)     If  $p < UR(0,1)$  then
  (10)      Reject change (keep  $x$  and  $f$ )
  (11)    Else
  (12)      Accept change (keep  $x'$  and  $f'$ )
  (13)    End If
  (14)  Else
  (15)    Let  $x \leftarrow x'$ 
  (16)  End If
  (17)  Let  $T_{i+1} \leftarrow \lambda T_i$ 
  (18) End For
Output: The solution  $x$ 

```

Algorithm 2.2: SA Algorithm

$$PR(f', f, T_i) = \exp\left(\frac{-\Delta f}{T_i}\right)$$

where $\Delta f = |f - f'|$

2.6.4 Genetic Algorithms (GA)

In biology, the gene is the basic unit of genetic storage (Purves et al., 1995). Living organisms are consummate problem solvers. Pragmatic researchers see evolution's remarkable power as something to be emulated rather than envied. Natural selection eliminates one of the greatest bundles in software design specifying in advance all the features of a problem and the action a program should take to deal with them. By harnessing the mechanisms of evolution, researchers may be able to "breed" programs that solve problems even when no one understands their structure. GA has already demonstrated the ability to make breakthroughs in the design of such complex systems as jet engines. GA make it possible to explore a far greater range of potential solutions to a problem than do conventional programs (Holland, 1992). The GA is a stochastic global search method that mimics the metaphor of natural biological evolution. (Chipperfield et al., 2011). This observation was first mathematically formulated by John Holland in 1975, in his paper "Adaptation in Natural and Artificial Systems". (Holland, 1975).

Furthermore, as researchers probe the natural selection of programs under controlled and well – understood conditions, the practical results they achieve may yield some insight into the details of how life and intelligence evolved in the natural world. Most organisms evolve by two primary processes: natural selection and sexual reproduction. The first determines which members of a population survive to reproduce, and the second ensures mixing and recombination between the genes of their offspring (Holland, 1992). The strings in candidate solutions to the search problem are referred to as chromosomes (Burke & Graham, 2005).

The standard GAs follows the method of haploid sexual reproduction. In GAs, a population of strings (called chromosomes or the genotype of the genome), which encode candidate solutions (called individuals, creatures or phenotypes) to an optimization problem, evolves toward better solutions as a shown in the Figure 2.26.

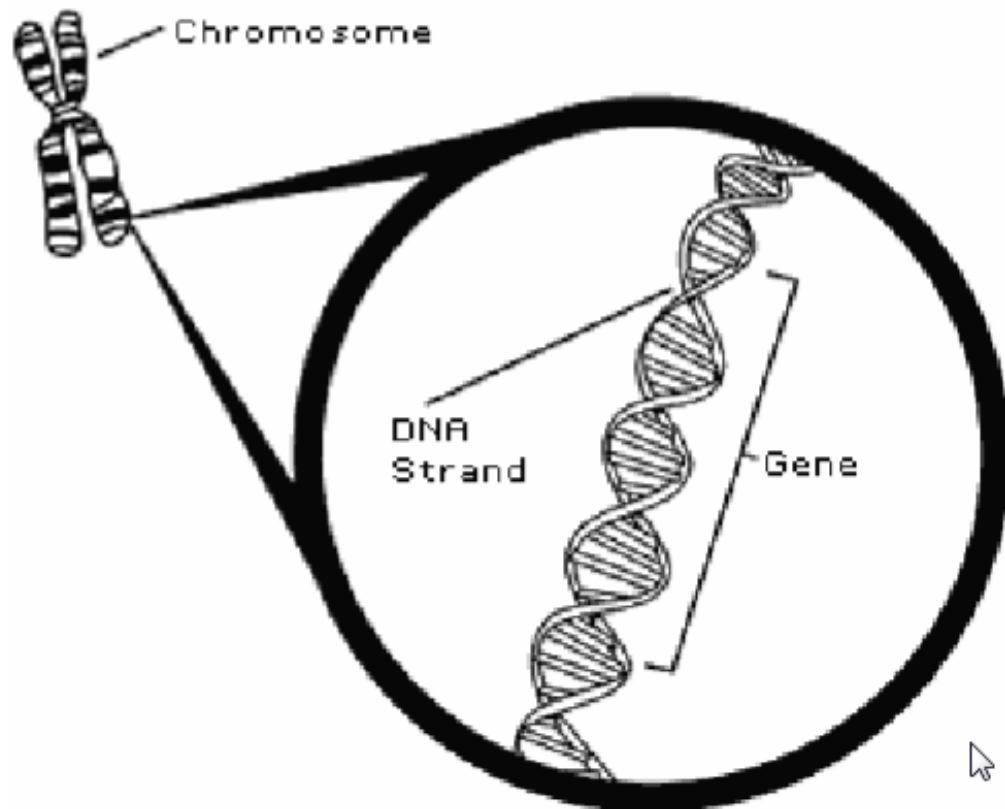


Figure 2.26: Example of a chromosome (Shasky, 2011)

In the standard GAs, the population is a set of individual binary integers such as 1001011. Each individual represents the chromosome of a life form. There is some function that determines how fit each individual is and another function that selects individuals from the population to reproduce. There are three main steps in a GAs (Mitchell, 1996):

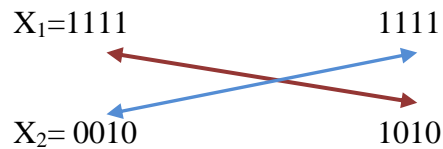
- a) Selection is the process of choosing a pair of organisms to reproduce. For example individuals can be encoded as strings, chromosomes composed over some alphabets, so that genotypes (Chromosome values) are uniquely mapped on the decision variable (phenotypic) domain. The most commonly used representations of GA is the binary alphabet $\{0,1\}$

$X_1 = 11111111$

$X_2 = 00101010$

- b) Crossover is a process of exchanging genes between the two individuals that are reproducing. There are several such processes, but we will consider one-point crossover, a process that is both standard and simple. A random integer i is selected uniformly between 1 and n , that is, the position in the chromosome at which, with probability p_c , crossover will occur. If crossover does occur, then the chunks up to i of the two chromosomes are swapped as shown in Figure 2.27.

$i = 4$



New chromosome

$X_1 = 0010 1111$

$X_2 = 1111 1010$.

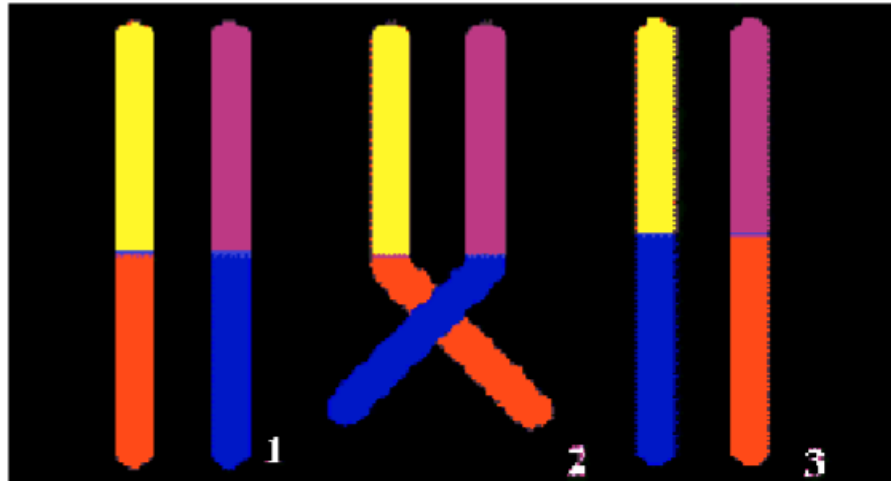


Figure 2.27: Example of crossover (Shasky,., 2011))

- c) Mutation is the process of randomly altering the chromosomes. Mutation also adds new information in a random way to the genetic search process and ultimately helps to avoid getting trapped in local optima. For example, in the case of a chromosome represented by a binary string the mutation operator may chose a gene (bit) and invert the bits (e.g. changing 1 to 0 and vice versa).

Figure 2.28 shows an example of a GA convergence graph. Here the initial fitness value is around 700000 and the algorithm converges at around 2600 iterations, reaching a fitness value of 1500000.

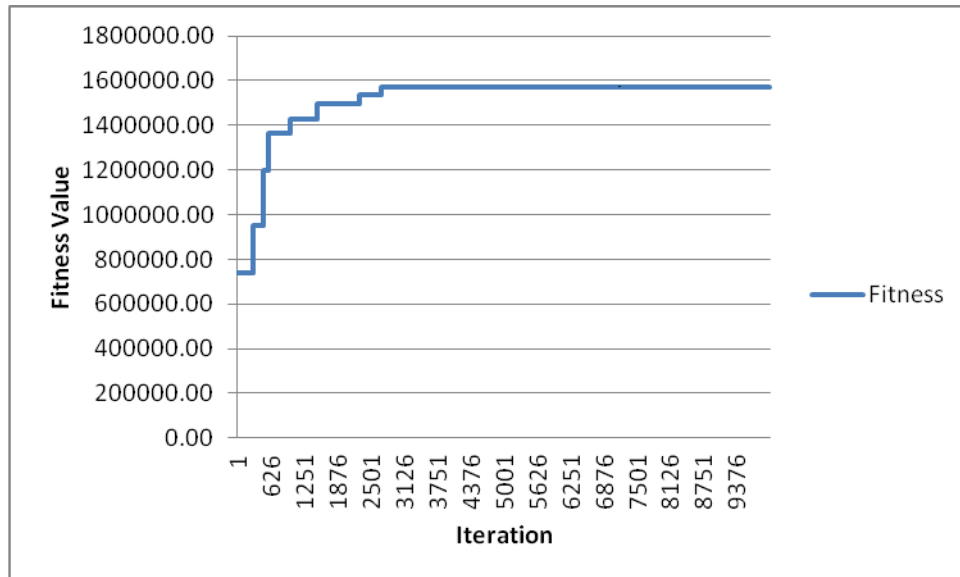


Figure 2.28: GA convergence graph

Below is the pseudocode of a GA algorithm (Michalewicz & Fogel, 2004). A GA maintains a population of individuals, $P(t) = \{x_1^t, \dots, x_n^t\}$ for iteration t . Each solution x_i^t is evaluated to give some measure of its “fitness”. A new population at iteration $t+1$ is formed by selecting the more fit individuals (the “select” step). The new population undergoes transformations (the “alter” step) by means of variation operators to form new solutions.


```
procedure genetic algorithm
(1) begin
(2)    $t \leftarrow 0$ 
(3)   initialise  $P(t)$ 
(4)   evaluate  $P(t)$ 
(5)   while (not termination-condition) do
(6)     begin
(7)        $t \leftarrow t+1$ 
(8)       select  $P(t)$  from  $P(t-1)$ 
(9)       alter  $P(t)$ 
(10)      evaluate  $P(t)$ 
(11)     end
(12) end
```

Algorithm 2.3: GA algorithm

2.6.5 Genetic Algorithms versus Traditional Methods

Genetic Algorithms differ substantially from traditional search and optimisation methods such as HC and SA. The four most significant difference are below (Chipperfield et al., 2011):

- a) GA search a population of points in parallel, not a single point.
- b) GA do not require derivative information or other secondary knowledge. Only the objective function and corresponding fitness levels influence the directions of search.
- c) GA use probabilistic transition rules rather than deterministic ones
- d) GA work on an encoding of parameter set rather than the parameter set itself.

2.7 Summary

Leukaemia is a form of blood cancer accompanied by severe health complications and painful episodes and sometimes untreatable. It was first officially diagnosed more than 100 years ago by John Hughes Bennett. Leukaemia cells are related to white blood cells. White blood cells have a highly defensive role in destroying invading organisms and assisting in the removal of dead or damaged tissue. Hence, one can imagine the abrupt deterioration of an organism when white blood cells do not fulfil their role. For example, patients are extremely susceptible to virus attacks.

There are four major types of leukaemia's, including ALL, AML, CLL and CML. The morphology of individual leukaemia cells is essential in classification, which is based on the WHO and FAB classification schemata. This thesis is focused on AML, characterised by the abnormal growth and development of early nongranular white blood cells. There are subtypes of AML, referred to as M0 to M7. Haematologists often face difficulties in classifying AML subtypes. However, correct diagnosis is important as M3 AML requires different treatment than the other subtypes.

The classification of leukaemia cells requires lab testing, including cytogenetic tests and immunophenotyping. The diagnostic process can be laborious and long, lasting three to five days. A faster diagnosis can reduce the time it takes patients to receive treatment. For the purposes of this thesis 322 images of blood and bone marrow smears were used, belonging to one of the following subtypes: M1, M2, M3 and M5. Only four subtypes were used due to image availability constraints. The images were provided by the Department of Haematology in the Universiti Sains Malaysia (USM).

This chapter discussed Digital Image processing, its importance and history since it first became relevant in the 1920s. There is extensive discussion of feature extraction given that the work presented here exploits feature extraction for the identification leukaemia cells.

The Otsu method and CA relevant to this work are also presented. The research has implemented the Otsu method to segment images into foreground and background and it will be further discussed in the following chapters. Furthermore, CA was implemented here in order to identify potential blast cells in the images in the analysed dataset. As previously mentioned CA was introduced by John Von Neumann, to provide a workable model for the behaviour of a complex and extended system. Stan Ulam also contributed to their development, proposing the idea of self-reproducing structures. CA was used in this thesis to find the potential cells and starting points for starting the heuristic search at the same time finding for radius.

In addition, this chapter provided an introduction to a number of other concepts relevant to this thesis. This includes discussion of distance metrics such as the Manhattan and Euclidean distances and of heuristic search methodologies, concentrating on HC, SA, and GA. Neural network were also presented as they are implemented in this work for the classification of the processed images. For this the popular WEKA application was facilitated, revealing that a Multilayer Perceptron was able to exhibit superior performance.

Chapter 3

RANDOM HEURISTIC SEARCH

3.1 Introduction

This chapter focuses on the process of choosing the fitness function, implemented for detection of blast cells in blood smear images. This is based on the use of Heuristic Search optimisation facilitating random starting coordinates.

To find the “best” fitness function eleven different fitness functions were tested. The methodology involved the implementation of the Otsu method, widely used in image processing and computer vision, involving the performance of shape-based histogram for image thresholding to transform a grey level image into a binary

image. The results show that a GA was able to reach higher fitness values than SA and HC.

This chapter is organised as follows: Section 3.2, presents an overview of the Otsu method and heuristic search, section 3.3 describes the work process in Random Heuristic Search, section 3.4 presents the findings of the optimisation, some useful notation and the implemented algorithms. Section 3.5 discusses the choice of the most appropriate fitness function. Finally, section 3.6 summarises this chapter.

3.2 Previous Work

This section provides some background of the Otsu method and Heuristic search algorithms.

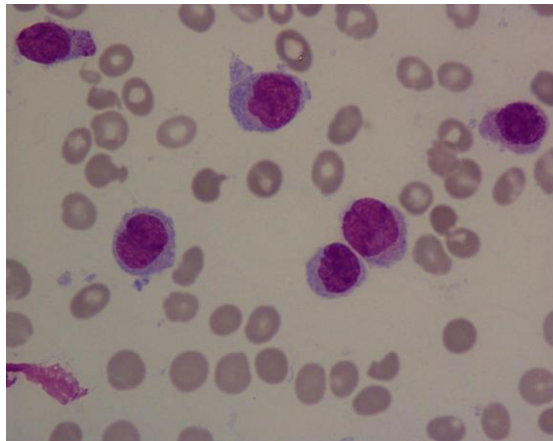
3.2.1 Otsu

The Otsu method has been implemented in video imaging. For example, after the application of a grey level thresholding method in the (R,G,B) color space to obtain a single threshold value for each domain, the three values are processed by an unsupervised clustering algorithm that is based on a between-class/within-class criterion suggested by Otsu's method (Du et al., 2004). In another publication (Zhang et al., 2010) the authors suggest a thresholding algorithm based on two neighbourhoods and Otsu's method. This method targets complex images such as rock images, and is capable of preserving more edges between the objects while removing noise within the objects.

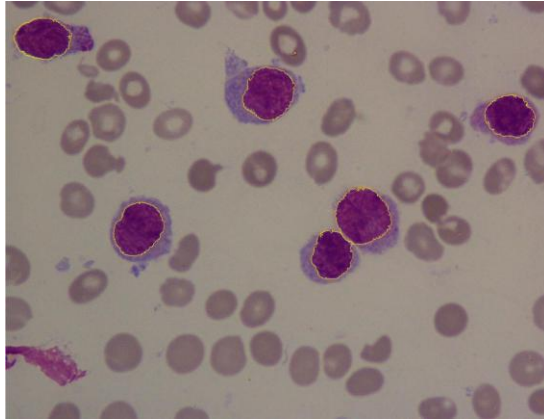
Otsu's method for image segmentation can be time-consuming because of the inefficient formula of the between-class variance. A faster version of Otsu's method has been proposed for improving the efficiency of computation for the optimal thresholds of an image (Liao et al., 2001). In (Wu et al., 2006) the authors use the Otsu approach for the analysis of white blood cell images, using a circular histogram to locate the cells.

Previous work by (Piuri & Scotti, 2004) has concentrated on using the nucleus for classification through image segmentation. The sample images are extracted from an accredited image repository, the Atlas of Blood Cells Differentiation. The data sets consist of 113 images that contain 134 expert-labelled leukocytes (white blood cells). (Zamani & Safabakhsh, 2006) used a colour gradient method to smooth the image before applying a GVF (Gradient Vector Flow) snake based method. Scale-space filtering and the watershed algorithm were applied to colour images for detecting nuclei (Jiang et al., 2003). This paper attempted to locate cell membranes; it required a few steps to implement edge detection, but the case studies are different from leukaemia cells because leukaemia cells have different features. Another approach using eigen cells for detecting white blood cells was introduced in (Yamprı et al., 2006) but with limited success in accurately classifying all white blood cells. Most of these approaches are based on colour images. However, in the case of overlapping cells, there can be some problems. In (Ritter & Cooper, 2007), a histogram of pixel counts focusing on the touching cells was created, and an edge cutting algorithm was then applied to separate the cells. This technique can be used for touching cells but not for overlapping cells. It is not applicable for the purposes of this thesis, since cutting the cells will create different morphological features, which might lead to incorrect classification.

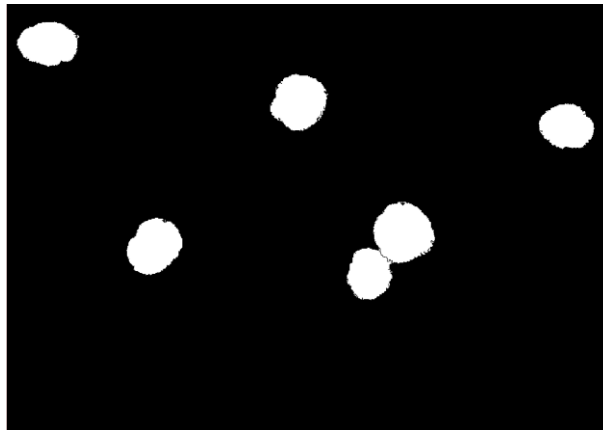
Topology is an important prior in segmentation tasks. (Zeng et al., 2008) describe a novel graph-based min-cut/max-flow algorithm that incorporates topology priors as global constraints. The key aspect of the algorithm is the organisation of the search for a maximum flow that allows consideration of topology constraints. The application of this algorithm was tested on ten randomly selected images, assuming that this is a reasonable number to allow us to examine results. These images are the same ones used in section 3.5.2. Targeting was successful in only four images (Figure 3.1), due to the fact that in the other images blast cells were overlapping. Then, the same images were subjected to processing by the methods proposed in this thesis.



(a) Real Image



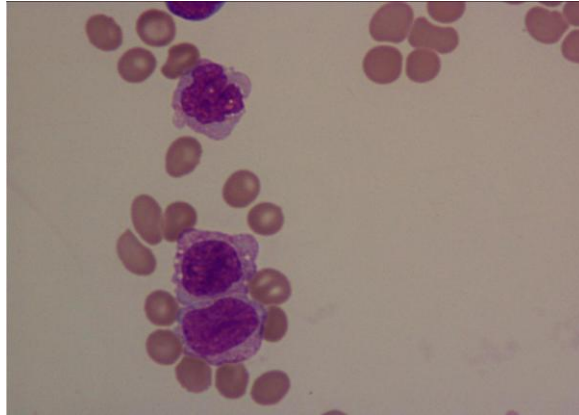
(b) Topology in colour images



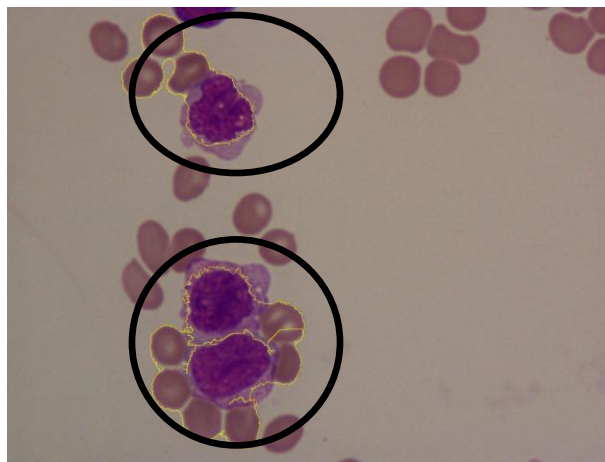
(c) Topology in black and white

Figure 3.1: Successful topology method for non-overlap blast cells (a) – (c)

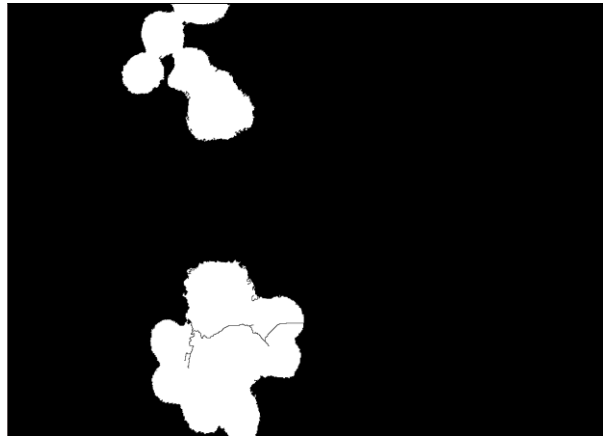
In three images the algorithm could not target potential blast cells efficiently, as shown in Figure 3.2. The topology segmentation is unsuitable in the blast cells environment because some of the cells are overlapping resulting in the algorithm targeting both cells. The result of the segmentation for a colour image is shown in Figures 3.2 (b) and (c). In the remaining three images the method did not work at all.



(a) Real Image



(b) Topology in colour images



(c) Topology in black and white

Figure 3.2: Unsuccessful topology method for overlapping blast cells (a) – (c)

3.2.2 Heuristic Search

Nowadays, there is growing demand for solving real-world optimisation problems in medical image processing. Heuristics are a relatively easy and appropriate choice of approach for addressing such difficult problems, because of their simplicity and fast performance (Falkenauer, 1998).

Heuristic search and optimisation techniques have been previously applied to microscope images for the identification of leukaemia cells. Circle detection has been used, facilitating GA and sobel's method in (Victor et al., 2006). The method is able to detect circle objects but not morphological features, which is important since blast cells have varying morphological features.

(Nilsson & Heyden, 2002) apply randomly chosen seed regions and heuristic search to find the “best” cell that can be grown around a seed assuming a maximum

cell size of $25\mu\text{m}$. This technique can be performed if cells do not overlap and are disjoint. Particle Swarm Optimisation combined with Neural Networks has also been used to escape from the local optimum in an application for detecting the colour nucleus in large-scale image data (Fang et al., 2005).

(Roth & Levine, 1994) have also implemented a GA based on a minimal subset representation of a geometric primitive to perform primitive extraction. They used three points to represent a unique circle, and five points for an ellipse.

3.3 Work Process

Figure 3.3 diagram depicts the work process followed here. The Otsu method is used for image segmentation into background and foreground. This is followed by random heuristic search, and in particular the application of HC, SA and GA to locate blast cells.

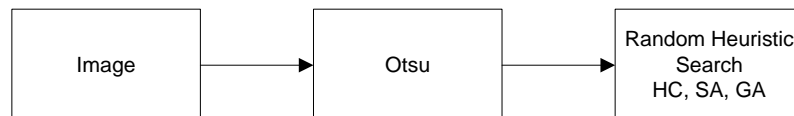


Figure 3.3: Work Process for Random Heuristic Search.

3.4 Methods

The following sections present the methods that have been used in this research, including Otsu's method and the three Heuristic Search approaches mentioned previously.

3.4.1 Otsu

The Otsu method facilitates a grey scale a threshold for separating an image into two objects, the background and the foreground (Otsu, 1979). The method was implemented using the Matlab image processing toolbox. In the resulting images the target cells are converted into black (foreground) while the background into white colour.

3.4.2 Determining the Fitness Function

A Fitness function is an objective function designed according to the problem in hand. Here the goal is to extract potential blast cells from blood smear images. This section describes the choice of the most suitable fitness function for identification of blast cells in the dataset.

In order to choose the most appropriate fitness function, 10 sets of coordinates were initially selected. These coordinates were chosen manually as shown in Figure 3.7(a). For example in Image 1 only the background is targeted, while in Image 2 there is a slight improvement with some foreground encapsulated by the targeting circle. The choice was based on visual inspection of each image, so that the starting

coordinates were ranked in terms of targeting actual blast cells. That means that some were completely outside any of the blast cells while the best one in the centre of a blast. Then, the processing method was applied for each pair of coordinates. The idea is to evaluate how reliable a potential circle is based on the number of valid points (white points from the black and white image) it covers. Then, direct correlation analysis was performed based on a scale of one to ten, from best to worst, using a number of possible circles (test cases that were simply ranked). The images were ranked by visual inspection and the correlation of the ranking to the fitness acquired for each image was established. The aim was to select a fitness function whose value agreed most with the ranking. Table 3.1 shows the coordinates x , y and the radius r of targeting circles in the ten images.

3.4.3 Fitness Function Notation

In the used notation, C is a circle, R is the radius of a circle, B is the number of black points, and W the number of white points within a circle, while \mathcal{G} is the number of pixels.

3.4.4 Fitness Function for Hill Climbing and Simulated Annealing

After choosing the “best” fitness function it is implemented in the HC and SA methods, to optimise the detection of blast cells.

3.4.4.1 Hill Climbing and Simulated Annealing Notation

Equation 3.1 presents the fitness function chosen from the results in Table 3.1, where C_i is the i th circle and $|C|$ as the number of circles within our list C . $R(i)$ is the radius of circle C_i , $B(i)$ is the number of black points within circle C_i , and $W(i)$ is the number of white points in C_i for a given image.

$$F(C) = \frac{R(C_i) \left(\sum_{i=1}^{|C|} B(C_i) + 1 \right)}{\sum_{i=1}^{|C|} W(C_i) + 1} \quad (3.1)$$

3.4.4.2 Algorithm for Hill Climbing

HC is a local search method which utilises the idea of searching the neighbourhood of a solution for a better one. This technique iterates a number of times, each time selecting a new solution from the neighbourhood of the current one. If the fitness value of the new solution is better, the new solution becomes the current solution. Otherwise, it is discarded and another one in the surrounding neighbourhood is randomly chosen and tested. The main disadvantage of using HC is that it may often get stuck in local optima (Michalewicz & Fogel, 2004). Below is the pseudocode of the HC algorithm. The number of circles N corresponds to the number of potential blast cells. At each iteration at step (6) of the algorithm the circle “moves” around the blast cells to improve detection.

```

Input: Image  $\kappa$  ( $\kappa$  - real image after converted into
Otsu)
(1)   Number of circles N
(2)   Number of iterations ITER
(3)   Create C = N circles (using Random)
(4)   Let Fit = F(C) applied to  $\kappa$  (equation 3.1)
(5)   For i = 1 to ITER
(6)       Create  $C_{new}$  from C using change generator
(7)       Let  $F_{new} = F(C_{new})$  applied to  $\kappa$ 
(8)       If  $F_{new} \geq Fit$ 
(9)           Fit =  $F_{new}$ , C =  $C_{new}$ 
(10)      End if
(11)  End For
Output: Highest Fitness F and circles C

```

Algorithm 3.1: Random HC Algorithm

3.4.4.3 Algorithm for Simulated Annealing

As discussed in chapter 2, SA is a heuristic search technique which aims at improving the problems inherent in HC. SA, incorporates the idea of a decreasing temperature, which serves to determine the probability of accepting a worse solution (Michalewicz & Fogel, 2004). In this way a SA may escape a local optimum and find a better solution in the search space. Algorithm 3.2 is based on Algorithm 3.1, the differences being the probability of accepting a worse solution at step (14).

```

Input: Image  $\kappa$  ( $\kappa$  - real image after converted into
Otsu)
(1)   Number of circles N
(2)   Number of iterations ITER
(3)   Start and end temperature TZero, TFinal
(4)   Define  $\lambda$  (equation 3.2)
(5)   Create C = N circles (using Random)
(6)   Let t = TZero, Let Fit = F(C)
(7)   For i = 1 to ITER
(8)       Create Cnew from C using change generator
(9)       Let Fnew = F(Cnew)
(10)      Diff = fnew - fit, P = exp(Diff/t)
(11)      If Fnew  $\geq$  fit
(12)          Fit = Fnew, C = Cnew
(13)      Else
(14)          If random value (0,1)  $\leq$  P
(15)              Fit = Fnew, C = Cnew
(16)          End If
(17)      End If
(18)      t =  $\lambda \times$  t
(19)   End For
Output: Highest Fitness F and circles C

```

Algorithm 3.2: Random SA Algorithm

The parameter λ is calculated based on (Swift S, et al., 2004) as shown in equation (3.2). The parameter settings are determined by the ITER setting, that is the number of iterations. TFinal is the final temperature from which the starting temperature is subtracted. A worse solution will be accepted with probability $0.368(e^{-1})$, which drops as the temperature cools down.

$$\lambda = \frac{\exp(\ln(\text{TFinal}) - \ln(\text{TZero}))}{\text{ITER}} \quad (3.2)$$

3.4.5 Circle Overlap Similarity Metric

The degree of overlap of two circles, in ratio is determined. Figure 3.4 exhibits the rationale behind this calculation which is used in the GA method. During crossover, circles might overlap, and the ratio of the overlap is used in the fitness function.

- a) 1st scenario - Circles overlap totally b) 2nd scenario - Circles do not overlap



- c) 3rd scenario – Internalised circles d) 4th scenario - Circles intersection

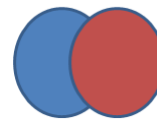
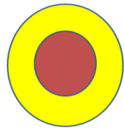


Figure 3.4: Overlap scenario

3.4.5.1 Circles Overlap Similarity Metric Notation

Here r represents the radius of a circle and (x, y) the coordinates of the centre. To investigate the scenarios discussed in the previous section, two circles are required, C_1 with $R(C_1), X(C_1), Y(C_1)$ and C_2 with $R(C_2), X(C_2), Y(C_2)$. If the circles appear as internalised circles as shown in Figure 3.5, or intersection as shown in Figure 3.6, Euclidean Distance is used to calculate the distance between points, (x_1, y_1) and (x_2, y_2)

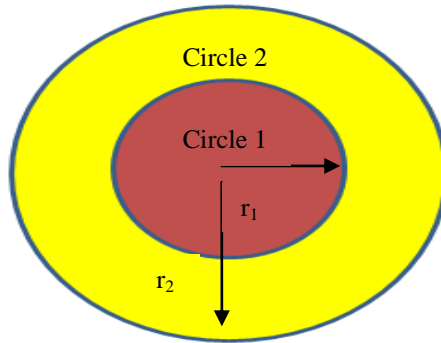


Figure 3.5: Internalised circles

$$\begin{aligned}
 A_1 &= \Pi r_1^2 \\
 A_2 &= \Pi r_2^2 \\
 \text{Ratio_inter} &= A_1 / A_2
 \end{aligned}
 \tag{3.3}$$

For the intersection equation

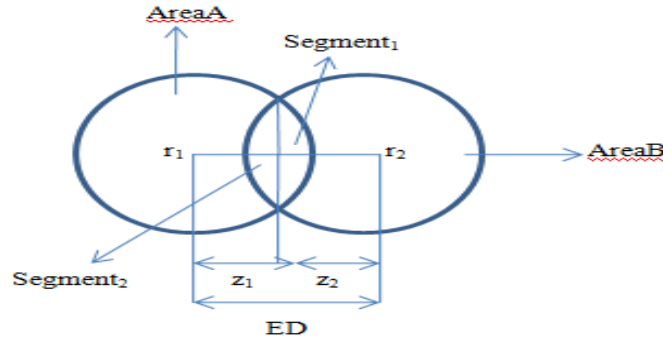


Figure 3.6: Intersection diagram

$$\begin{aligned}
 z_1 &= ((r_1^2 - r_2^2) + (ED^2)) / (2 * ED) \\
 z_2 &= ED - z_1 \\
 \text{Segment} &= 2r_1^2 (a \cos z_1 / 2r_1) - 0.5(z_2 \sqrt{4 * r_2^2 - z_2^2}) \\
 \text{Segment}_1 &= (r_1^2 a \cos(z_1 / r_1) - (z_1 \sqrt{r_1^2 - z_1^2})) \\
 \text{Segment}_2 &= (r_2^2 a \cos(d2 / r2) - (z_2 * \sqrt{r_2^2 - z_2^2})) \\
 \text{AreaA} &= (\Pi r_1^2) - (\text{Segment}_1 + \text{Segment}_2) \\
 \text{AreaB} &= (\Pi r_2^2) - (\text{Segment}_1 + \text{Segment}_2) \\
 \text{Ratio_inter} &= \text{Segment} / (\text{AreaA} + \text{AreaB} + \text{Segment})
 \end{aligned}
 \tag{3.4}$$

Here, ED is the Euclidean Distance, calculated according to Equation (2.2). In our conditional statement, below, if two circles overlap completely, S is equal to 1. If the circles are disjoint S is equal to 0. If the circles are in union S reveals the degree of union. If the circles intersection, S reveals the degree of intersection.

(45)

$$S(x_1, y_1, r_1, x_2, y_2, r_2) = \begin{cases} 1, & \text{if } x_1 = x_2, y_1 = y_2, r_1 = r_2 & \text{figure: 3.4(a)} \\ 0, & \text{if } ED(x_1, y_1, x_2, y_2) > r_1 + r_2 & \text{figure: 3.4(b)} \\ A_1 / A_2, & \text{if } ED(x_1, y_1, x_2, y_2) + r_1 < r_2 & \text{figure: 3.4(c)} \\ \text{Segment} / (\text{AreaA} + \text{AreaB} + \text{Segment}) & \text{Otherwise} & \text{figure: 3.4(d)} \end{cases}$$

where $r_1 \leq r_2$

3.4.5.2 Algorithm for Circle Overlap Similarity Metric

This section presents a method to check whether two circles overlap totally, do not overlap at all, appear in internalised circles or intersection. It is important to calculate the ratio of overlap and use this value in the GA fitness function calculation. The addition of 1 to the ratio is needed to ensure that whenever the actual ratio is 0 (disjoint circles) the fitness value does not become infinite. If the circles do not overlap, the fitness function in GA is same as in HC and SA.

3.4.6 Fitness Function for Genetic Algorithm

Measuring the ratio of overlap in the GA is useful to ensure that the genetic operators introduced at each generation does not generate any overlap. This calculation is computationally expensive and has an effect on the performance of the algorithm.

3.4.6.1 Genetic Algorithm Notation

Equation 3.6 represents the fitness function used by the GA. Here C_i is the i th circle and $|C|$ as the number of circles within list C . $R(i)$ is the radius of the i th circle, $B(i)$ is the number of black points and $W(i)$ the number of white points within circle C_i for a given image. The ratio of overlap is calculated according to the preceding section 3.4.5.

$$F(C) = \frac{R(C_i) \left(\sum_{i=1}^{|C|} B(C_i) + 1 \right)}{\sum_{i=1}^{|C|} W(C_i) + 1} / (1 + \text{ratio}) \quad (3.6)$$

3.4.6.2 Algorithm for Genetic Algorithms

GAs are global optimisation algorithms based on natural evolution, originally introduced by (Holland, 1992) and have gained wide spread popularity. GA genotypes can be defined as bit-vectors (bit-strings), that is string of ones and zeros, where a point mutation is defined as the switching of a particular bit, which may occur following a certain probability (Mitchell, 1996). However, in this chapter we describe the use of a real valued GA. Recombination combines the parts of two or more parental solutions to create a new (potentially better) set of solutions. The GA used here implements one point crossover. The mutation operator is similar to the small change operator used by the HC and SA methods. Given that the GA algorithm is based on the same pseudocode as algorithm 2.3. Its depiction here has been omitted.

3.5 Results

This section presents the results produced by the discussed algorithms and comments on their comparative performance. Each method was executed 10 times for 10000 iterations using 10 randomly selected images. The distribution of images of different subtypes was as follows; 2 M1, 2 M2, 3 M3 and 3 M5 AML images. The distribution of the subtypes did not impact the results because the images were converted with the Otsu method into black and white images.

3.5.1 Choice of Fitness function

A number of possible fitness functions were initially tested based on the ten randomly selected images, in order to choose the one best suited for the purposes, that is, the one presented in equation (3.1). The aim was to select a fitness function whose value agreed most with the discussed ranking of 1 to 10. On Figure 3.7 the circles were drawn manually based on a 1 to 10 ranking scale, where “1” represents the worst and “10” the best targeting of blast cells. As image 1, on Figure 3.7(a) shows, the circle is not targeting the blast cells at all. In images 2 to 9 (Figure 3.7 (a),(b),(c)), the detection of blast cells shows an improvement, while in image 10 observe the best performance.

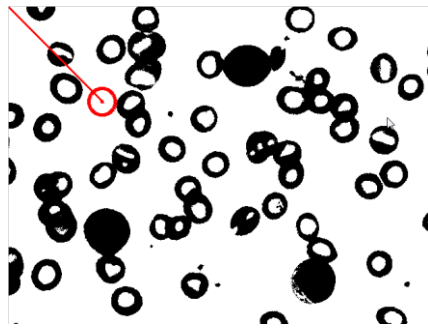


Image 1
($x = 288, y = 285, r = 40$)

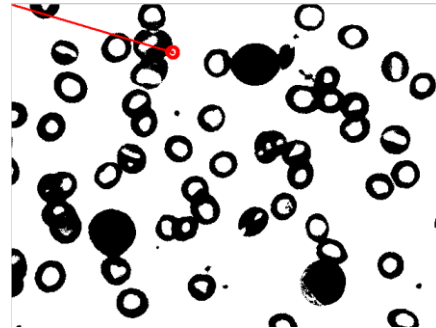


Image 2
($x = 143, y = 480, r = 16$)

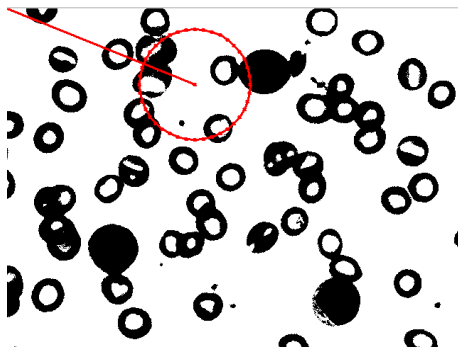


Image 3
($x = 215, y = 530, r = 156$)

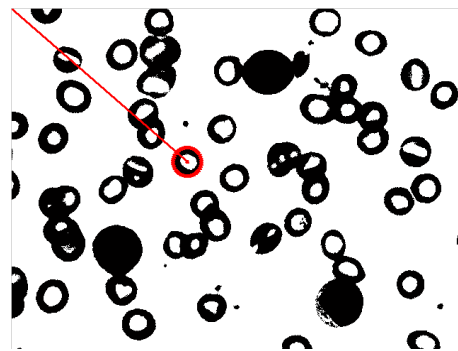


Image 4
($x = 431, y = 498, r = 40$)

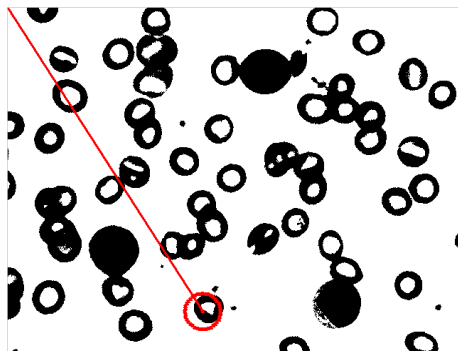


Image 5
($x = 850, y = 550, r = 50$)

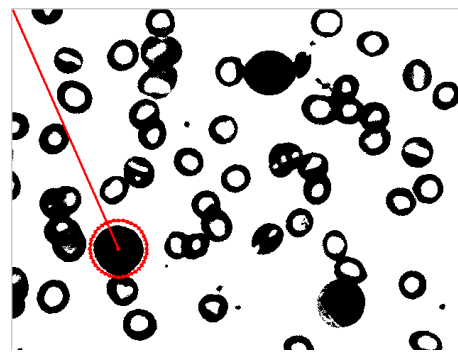


Image 6
($x = 675, y = 300, r = 80$)

Figure 3.7(a): Coordinates for x, y, r for ten test cases (Image 1 – Image 6)

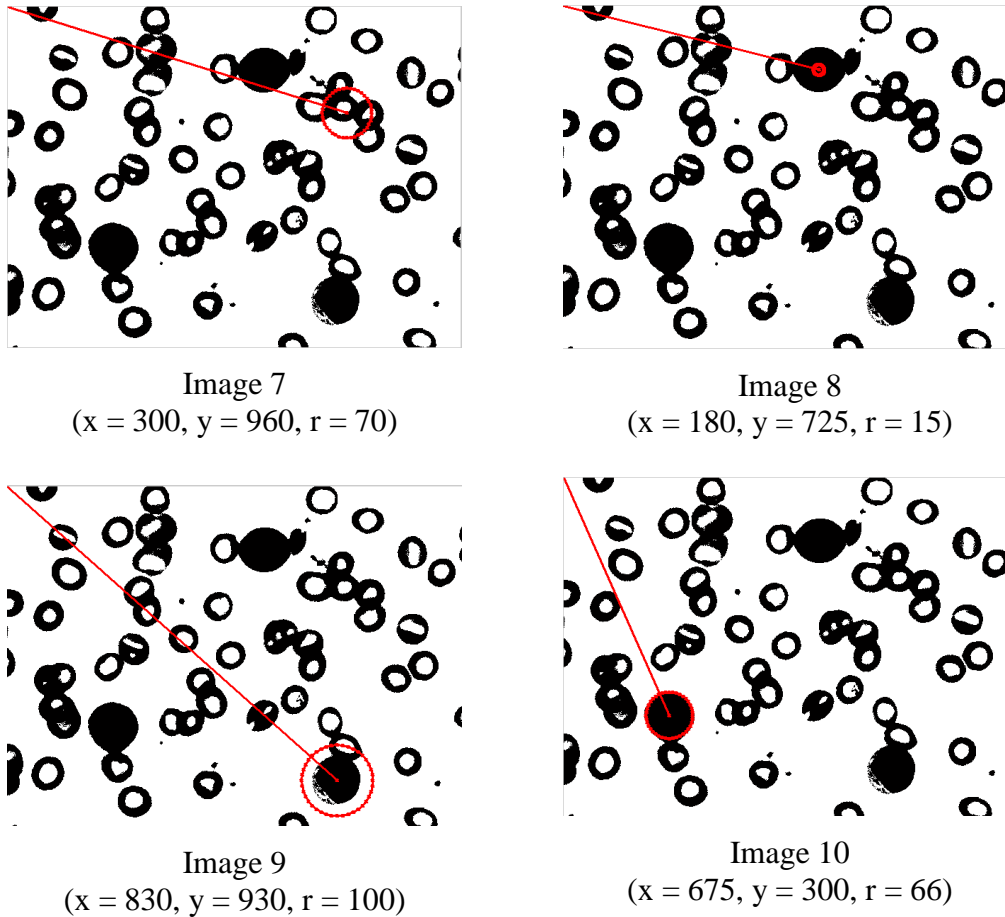


Figure 3.7(b): Coordinates for x, y, r for ten test cases (Image 7 – Image 10)

Table 3.1 shows the result of eleven test fitness functions from images on Figure 3.7. The coordinates of cells were defined based on visual inspection. The findings show fitness function (6) exhibits the highest correlation coefficient to the manual ranking.

Table 3.1: Test Method for ten test cases

| Method Fitness | Worst image | | | | | | | | | Best image | |
|---|----------------|-------------|--------------|--------------|--------------|---------------|--------------|--------------|--------------|---------------|----------------------------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Correlation Coefficient |
| 1. $R \times (B-W) / \rho$ | -40 | -8.73 | -83.96 | 15.79 | -2.72 | 43.29 | 9.77 | 15 | -2.03 | 65.41 | 0.70 |
| 2. $(B-W) / \rho$ | -1 | -0.54 | -0.54 | 0.39 | -0.05 | 0.54 | 0.14 | 1 | -0.02 | 1.00 | 0.78 |
| 3. $(B \times R) / \rho$ | 0 | 3.63 | 36.02 | 27.90 | 23.64 | 61.65 | 39.88 | 15 | 48.99 | 65.71 | 0.72 |
| 4. $(B \times R) / W$ | 0 | 4.70 | 46.83 | 92.15 | 44.84 | 268.7 | 92.70 | Inf | 96.03 | 14728 | 0.88 |
| 5. B / ρ | 0 | 0.23 | 0.23 | 0.70 | 0.47 | 0.77 | 0.57 | 1 | 0.49 | 1.00 | 0.78 |
| 6. $R \times (B+1) / (W+1)$ | 0.01 | 4.72 | 46.83 | 92.12 | 44.84 | 268.66 | 92.70 | 10650 | 96.03 | 14491 | 0.90 |
| 7. $B - W$ | -5025 | -435 | -41135 | 1983 | -427 | 10867 | 2145 | 709 | -637 | 13551 | 0.59 |
| 8. B | 0 | 181 | 17645 | 3504 | 3709 | 15474 | 8759 | 709 | 15390 | 13612 | 0.43 |
| 9. W | 5025 | 616 | 58780 | 1521 | 4136 | 4607 | 6614 | 0 | 16027 | 61 | -0.21 |
| 10. R | 40 | 16 | 156 | 40 | 50 | 80 | 70 | 15 | 100 | 66 | 0.24 |
| 11. $R \times (B) / (W)$ | 0 | 4.70 | 46.83 | 92.15 | 44.84 | 268.7 | 92.70 | Inf | 96.03 | 14728 | 0.88 |

Table 3.2 summarises the data on Table 3.1 the highest correlation revealed the most appropriate fitness function (6) which was implemented in the HC, SA and GA methods.

Table 3.2: Summary of Fitness Functions Evaluation

| Number | Method | Correlation |
|----------|--|-------------|
| 1 | $F = \sum R(i)((B(i) - W(i)) / (B(i) + W(i)))$ | 0.70 |
| 2 | $F = \sum (B(i) - W(i)) / (B(i) + W(i))$ | 0.78 |
| 3 | $F = \sum (R(i)B(i)) / (B(i) + W(i))$ | 0.72 |
| 4 | $F = \sum (R(i)B(i)) / W(i)$ | 0.88 |
| 5 | $F = \sum B(i) / (B(i) + W(i))$ | 0.78 |
| 6 | $F = \sum R(i)(B(i) + 1) / (W(i) + 1)$ | 0.90 |
| 7 | $F = \sum B(i) - W(i)$ | 0.59 |
| 8 | $F = \sum B(i)$ | 0.43 |
| 9 | $F = \sum W(i)$ | 0.21 |
| 10 | $F = \sum R(i)$ | 0.24 |
| 11 | $F = \sum R(i)B(i) / W(i)$ | 0.88 |

3.5.2 Comparison between Random Hill Climbing, Simulated Annealing and Genetic Algorithm

This section compares the performance of the random HC, SA and GA methods for ten images. Hence, the starting circle coordinates (x, y, r) for the subsequent search were generated randomly. Table 3.3 summarises the reached fitness

values by the three methods for ten randomly chosen images from data set, in ten runs of each algorithm. SA reached the highest fitness value in most cases.

Table 3.3: Comparison between HC, SA and GA Random

| Image | HC | SA | GA | Max. |
|---------|------------------|-------------------|-------------------|-------------------|
| 1 | 229796.20 | 94805.21 | 1393758.30 | 229796.20 |
| 2 | 227858.70 | 829226.10 | 812467.90 | 829226.10 |
| 3 | 141609.90 | 1334923.60 | 1720899 | 1334923.60 |
| 4 | 951630.35 | 5260004.69 | 2184407.20 | 5260004.69 |
| 5 | 784762.14 | 75808.42 | 2508479.50 | 2508479.50 |
| 6 | 809407.93 | 4378871.82 | 2475291.10 | 4378871.82 |
| 7 | 465109.89 | 3158457.79 | 1102489.52 | 3158457.79 |
| 8 | 71543.46 | 41776.95 | 82162.15 | 82162.15 |
| 9 | 44192.84 | 189514.99 | 2371489.50 | 2371489.50 |
| 10 | 515361.48 | 905159.28 | 1550242.70 | 1550242.70 |
| Average | 424127 | 1626855 | 1620169 | |

Figure 3.8 show the images for Hill Climbing random heuristic search which targeting most of the blast cells was incorrect. Figure 3.9 show the images of SA method although was able to produce the highest fitness values in the majority of cases, the targeting of blast cells was incorrect. Figure 3.10 shows that the GA performs better than the HC and SA in detecting blast cells.

In conclusion, the results of the analysis presented in this chapter show that the GA is an acceptable method for random starting points. Although in some

cases a certain method produces the highest fitness value visual examination of the image reveals that the detection of blast cells is incorrect. For example image 2 on Figure 3.8, reveals that despite the SA reaching the highest fitness the circles targeting the blast cells are too big, also encompassing red blood cells. On the other hand, image 9 in the same Figure shows that the GA has been quite successful in detecting blast cells (three out of four).

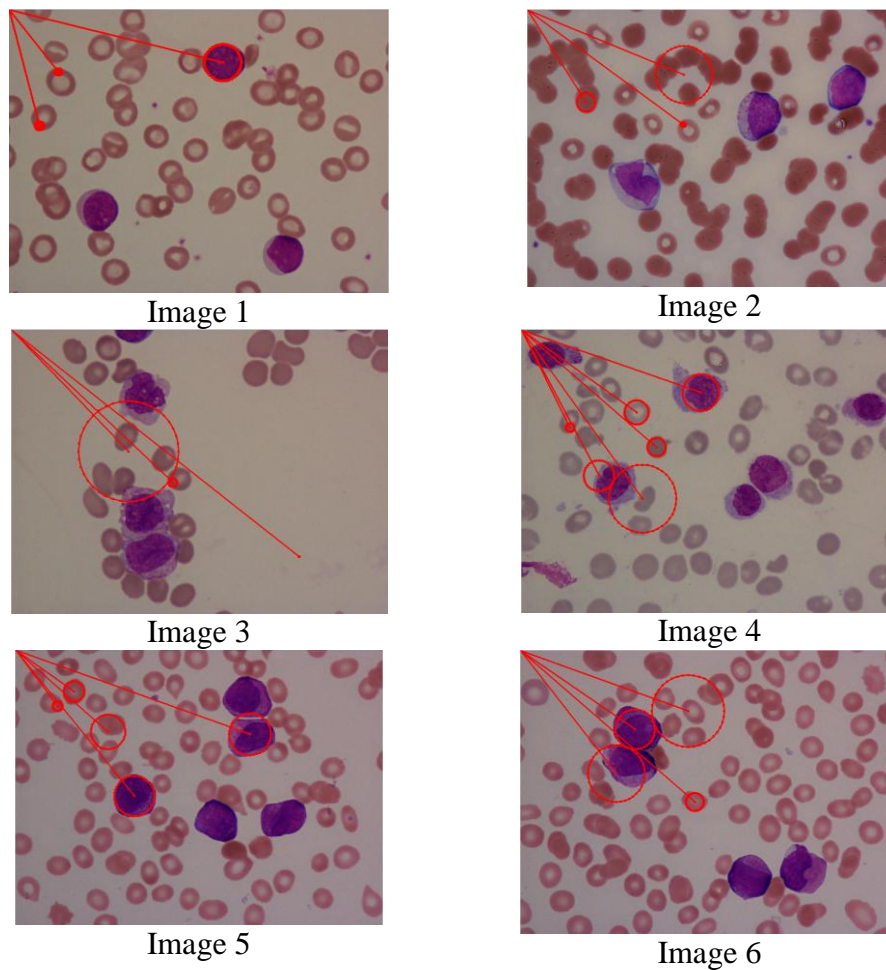


Figure 3.8 (a) : Hill Climbing Random Heuristic Search (Image 1 – Image 6)

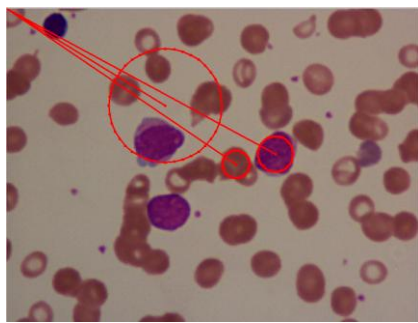


Image 7

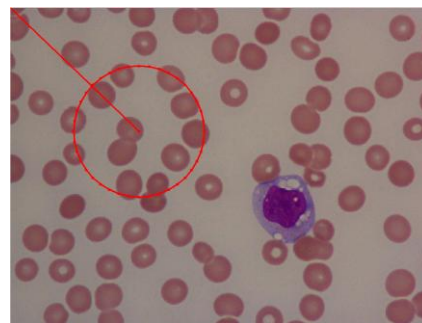


Image 8

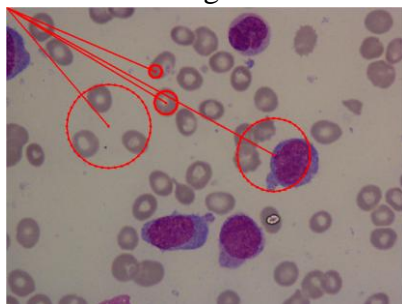


Image 9

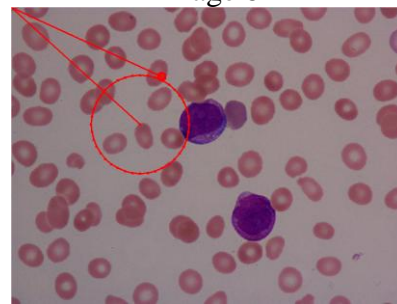


Image 10

Figure 3.8(b): Hill Climbing Random Heuristic Search (Image 7 – Image 10)

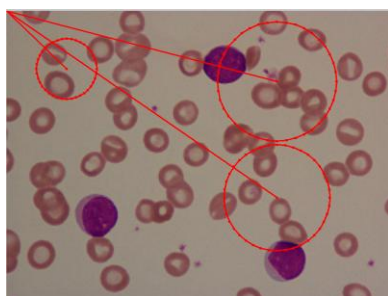


Image 1

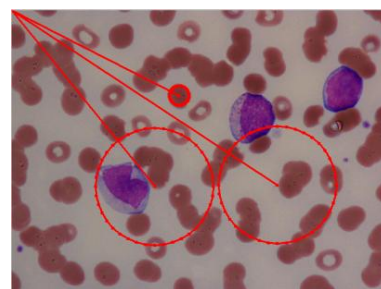


Image 2

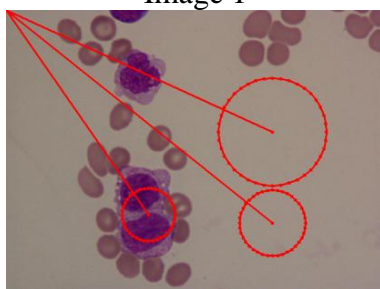


Image 3

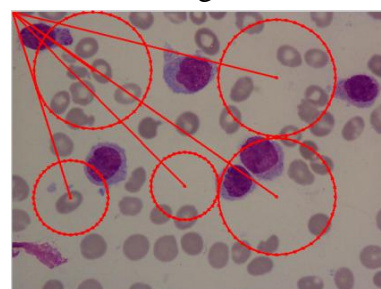


Image 4

Figure 3.9(a): Simulated Annealing Random Heuristic Search (Image 1 – Image 4)

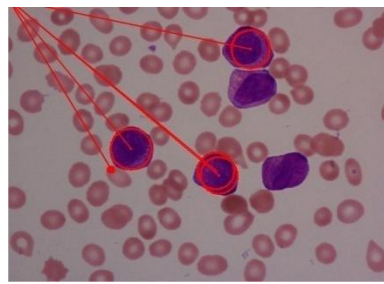


Image 5

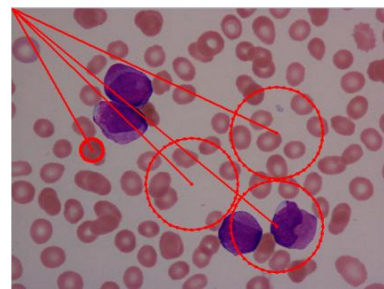


Image 6

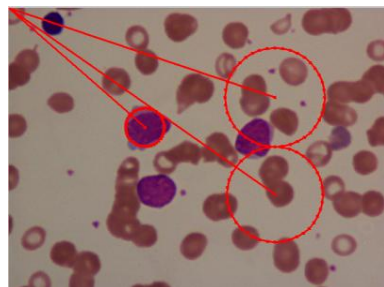


Image 7

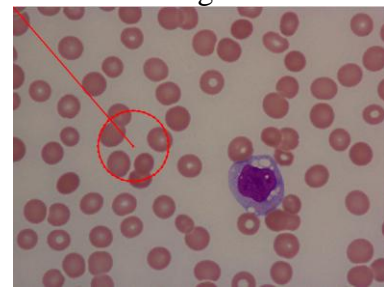


Image 8

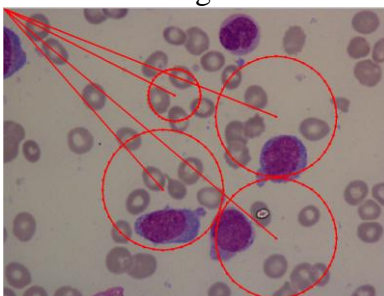


Image 9

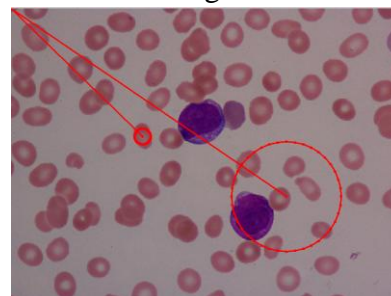


Image 10

Figure 3.9(a): Simulated Annealing Random Heuristic Search (Image 5 – Image 10)

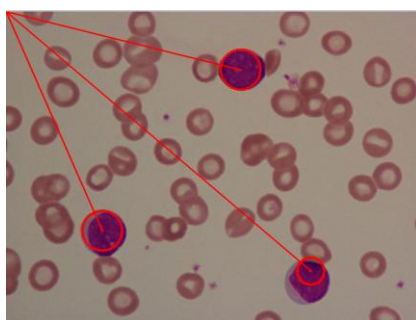


Image 1

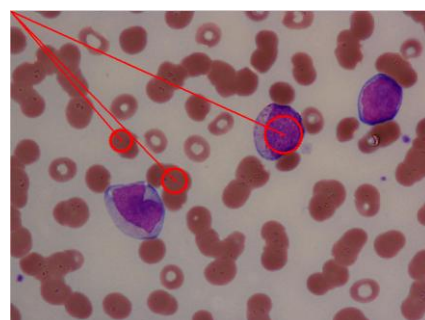


Image 2

Figure 3.10(a): Genetic Algorithm Random Heuristic Search (Image 1 – Image 2)

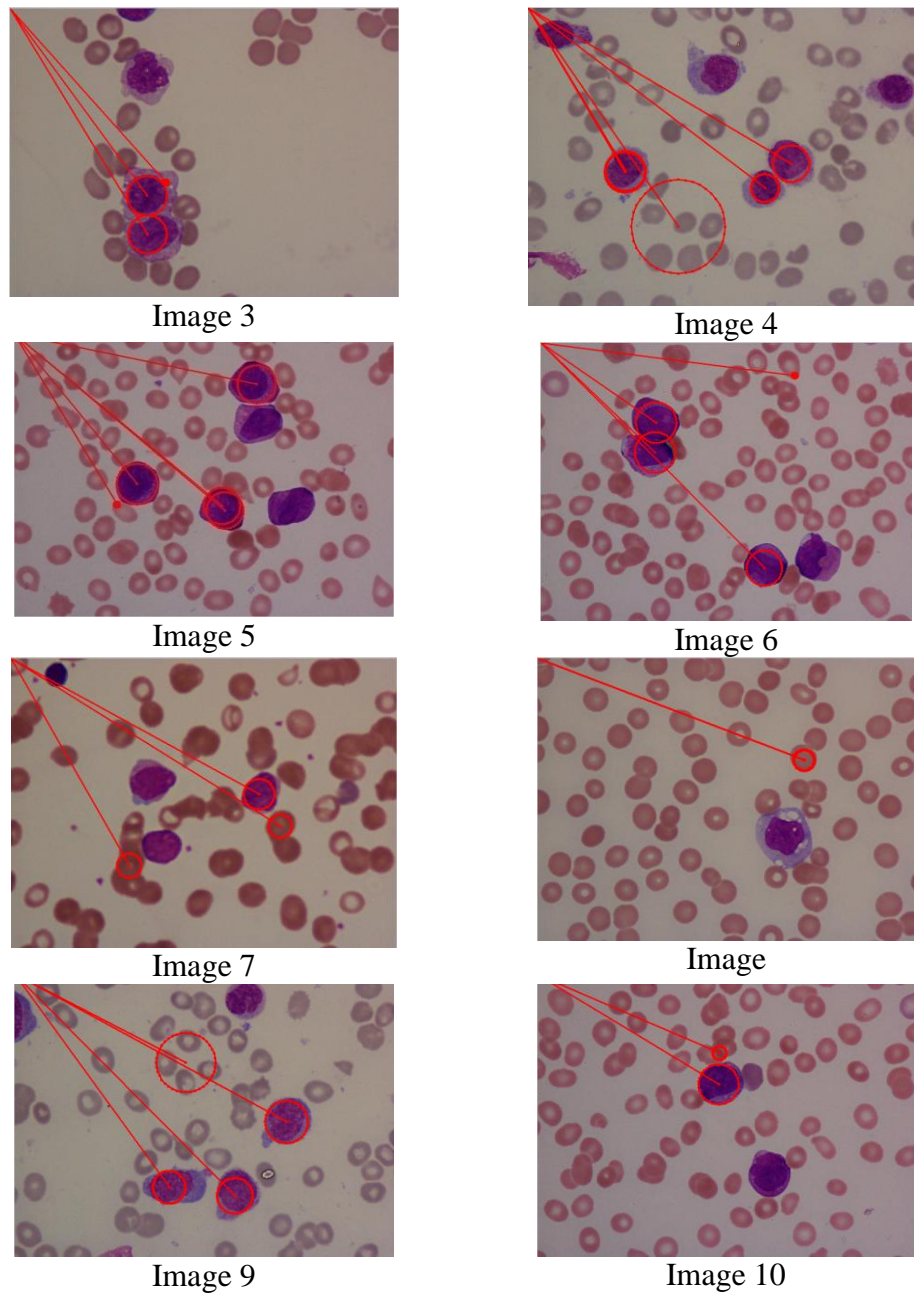


Figure 3.10(b): Genetic Algorithm Random Heuristic Search (Image 3 – Image 10)

Figure 3.11 provide an example of the convergence of the three algorithms for image 10 for 10000 iterations. The GA used a population of 250 individuals. Although the GA's fitness function is different from the HC and SA one it has no impact on the results. The ratio represents the degree of overlap and when this

occurs the GA can never have fitness value higher than HC and SA. However, whenever the circles do not overlap, the GA has the same fitness function with HC and SA.

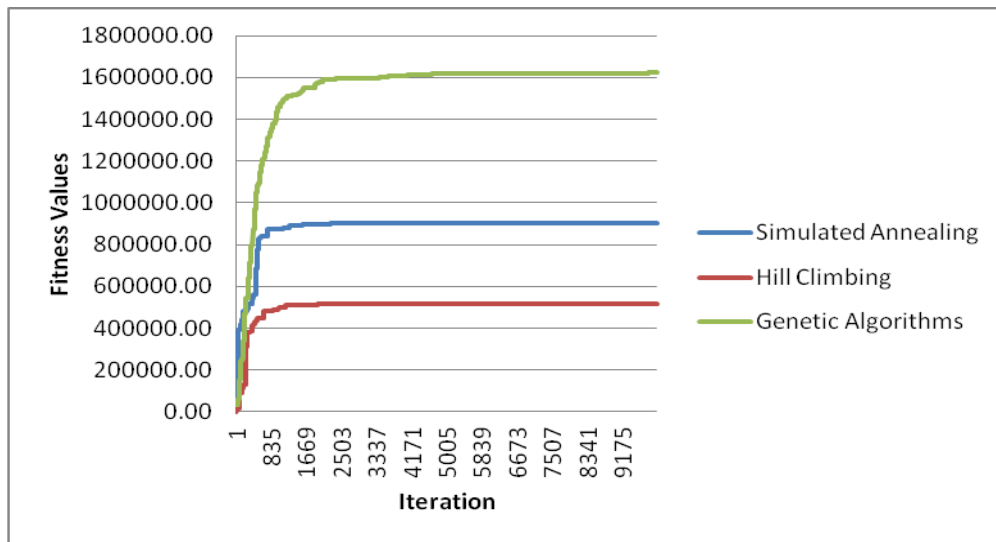


Figure 3.11 : Convergence graph for HC, SA and GA

3.6 Summary

This chapter discussed the choice of an appropriate fitness function based on 11 alternative ones. The selection process was based on randomly chosen images, visual examination of the results and correlation of the findings to the actual fitness values produced. The three heuristic search methods, HC, SA, and GA implemented in this work were also discussed. HC is a traditional local search method worth considering, which however tends to get stuck in local optima. SA is a way to target this issue and improve the search by allowing acceptance of worse solutions with a certain probability. The implementation of GA was also discussed. This is a popular evolutionary algorithm subject to mutation and crossover operators.

The presented results show that SA produces the highest fitness values. However, visual examination of the images reveals that the GA performs better in actually detecting blast cells. The results were deemed promising but unsatisfactory, leading to the decision to conduct further research and in particular to implement a “seeded” approach to define the starting point of a search, as discussed in the next chapter.

Chapter 4

CELLULAR AUTOMATA FILTERING

4.1 Introduction

The results in chapter 3 showed that the proposed methods did not detect all blast cells correctly and thus were deemed unsatisfactory. One plausible explanation is the random choice of starting points for the search, which may often lead to inappropriate choices. A way to deal with this issue is the use of CA to choose better starting points, which is the subject of this chapter. CA was used to identify the bigger cells in an image which are likely to correspond to blast cells. At the same time, the use of CA, allows to estimate the size of potential blast cells and remove noise from the images.

There are a number of papers dealing with the use of CA methods in medical image processing. In the previous chapter the implementation of the Otsu method in extracting objects from their background was discussed. Otsu's method allows using a grey scale threshold for the separation of objects that are in the

foreground and background of an image. CA are used to convert a black and white cell image to a matrix of the same size as the input image, where each point in this matrix represents the shortest distance, each point in the image is away from the background. This CA matrix can be then used to locate suitable candidate regions that correspond to the largest objects in the image. The hypothesis is that these will correspond to blast cells. Following that, CA were also used to perform image filtering based on the size of white blood cells in order to identify their location on the image.

This chapter is organised as follows: in section 4.2 previous work on CA is discussed, section 4.3 describes the work flow and methodology for blast cell detection. Section 4.4 discusses the implemented CA approach in some detail, including notation, rules and the used algorithms. Section 4.5 presents the obtained results. Finally, section 4.6 presents a summary of the contents of the chapter.

4.2 Previous Work

There are a few successful applications of CA that can be used in image processing and medical research. One of the CA applications in image processing is for removing noise in pictures and modelling the evolution of tumours. CA models can develop in three dimensions to describe tumour growth. A number of studies involving CA and cellular image processing have been conducted in the area of tumour growth forecast via 3D simulation (Moreira & Deutsch, 2002 and Bankhead & Heckendorn, 2007). CA can also reduce the noise in cellular images (Guieb & Samaneigo, 2007). In a publication discussing the modelling of the Immune System (De Boer et al., 1992), the authors use asynchronous CA in understanding pattern formation in immune network models and show the randomness involved. The CA model is used to study the evolution of human

immunodeficiency virus (HIV) infection and the onset of acquired immunodeficiency syndrome (AIDS). The model reproduces the pattern observed in T cell and virus counts of infected patients (Santos & Continho, 2001). CA algorithms are used to enhance local difference in grey level values and in noise removal for binary and grey scale images, and in spots detection for breast cancer diagnosis, helping physician and doctors in their work (Wongthanavase & Tangvorphonkchai, 2007). CA can simulate brain tumour growth and the effect of treatment on cellular divisions, mutations and treatment –induced deaths probabilistically. Using CA (Schmitz et al., 2002) were able to produce survival time data, which is vital in observations of tumour morphology.

In the work presented here CA are used to remove noise such as plasma and red blood cells from images. At the same time, they allow us to obtain a representation of the shortest distance that each point on the image is away from the background. The motivation behind the use of CA is to find the “best” starting points before proceeding with a heuristic search, to detect blast cells.

4.3 Work Process

This section provides a brief overview of the experimental steps followed to analyse colour images. First each image is converted into black and white with the Otsu method, segmenting it into foreground (cells) and background. CA is used to find the distance of each pixel from the background and detect potential blast cells. Figure 4.1 depicts the work flow.

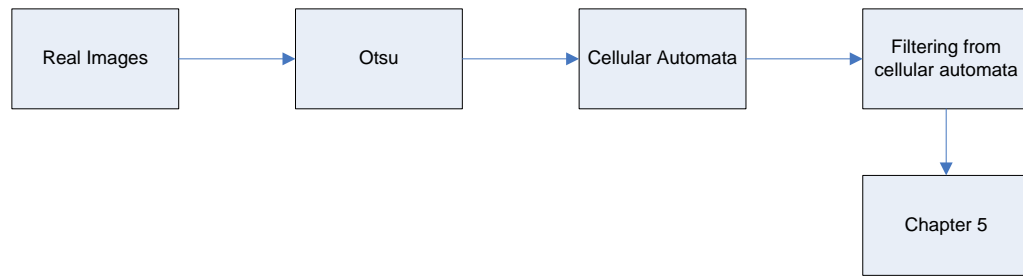


Figure 4.1: Work Process – chapter 4

4.4 Methods

The following sections present the methods used in this research, including CA and CA Filtering.

4.4.1 Cellular Automata

Cellular Automata (CA) are a type of complex system with only few parameters and are a development of Conway's game of life (Levy,. 1993). CA are used to convert an image into a matrix (of the same size as the input image) where each point in the matrix represents the shortest distance each point in the image is away from the background. The background is represented as white (or “dead”) and the cells' body (as identified by Otsu) as black (or “alive”).

4.4.1.1 Cellular Automata Notation

This section provides the notation used in the CA method. Conversion of a colour image $I(r,g,b)$ is performed with the Otsu method using a threshold $\kappa(x,y)$. Then the CA method is applied to produce a “distance” matrix. The movement of CA start at the four angle of the images which are top left, top right, bottom left and

bottom right. Let $\kappa_{(1)}$ be the cellular automata applied to the input image κ , i.e. $\kappa_1 = \Lambda(\kappa)$, then we define a series of matrices $\kappa_{(i)}$ such that $\kappa_{(i+1)} = \Lambda(\kappa_{(i)})$, i.e. the cellular automata applied to the results of the previous application of the cellular automata. The process terminates when all of the pixels in $\kappa_{(i)}$ are zero/dead. The CA works by looking at each pixel x, y and if the pixel is already “dead” it stays dead, otherwise its immediate adjoining neighbours (of City Block distance of one) are examined. If any neighbours are “dead”, the point itself becomes “dead” otherwise it remains “alive”. The neighbours of each point were examined, thus a corner point will only have eight neighbours to examine. This process is repeated until all the points within $\kappa_{(g)}$ are “dead” or white. Throughout the procedure another matrix Λ (the same size as the input matrix) is maintained, where each x, y element represents the CA iteration at which the point “died”, i.e. turned from black to white. For example, if after 5 iterations of the CA, which has created matrices $\kappa(1), \kappa(2), \dots, \kappa(5)$, point 100,200 “dies” (it was “alive” in $\kappa(4)$), then the element 100,200 in matrix Λ is set to 5. The values in the Λ matrix represent the shortest path between an “alive” and a “dead” point. Therefore, points within the matrix with high values represent dense areas of the input image and thus good starting points for locating the largest cells within the input image. Once the CA image (the visual representation of matrix Λ) is produced, points of high value which are located next to each other, e.g. dense spots and areas, are identified. The CA remove a the noise and unnecessary objects such as red blood cells. The pixels having a distance less than or equal to some value r from (x,y) are the points contained in a disk of radius r , centred at (x,y) as shown in Figure 4.2.

Image matrix after conversion with Otsu method =

$$\mathbf{K}_{(0)}, \mathbf{K}_{(1)}, \mathbf{K}_{(2)}, \dots, \mathbf{K}_{(g)}$$

$$Sum(\kappa_{(g)}) = 0 ; Sum(\kappa_{(g-1)}) \neq 0$$

$$\kappa_{(g)} = \prod \prod \prod \Lambda(x, y)$$

$$\Lambda(x, y) = \left\{ 1, \text{if } x < 1 \text{ or } a > x_{\max} \text{ or } y < 1 \text{ or } b > y_{\max} \right\}$$

$$\kappa_{(g)}, \text{Otherwise}$$

Refer to Appendix B for mathematical proof for CA

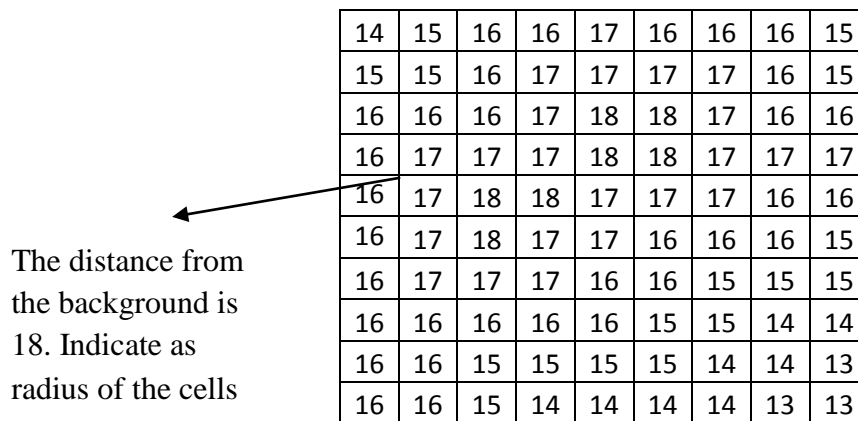


Figure 4.2: Distance using Manhattan Cellular Automata

4.4.1.2 Algorithm for Cellular Automata

After application of the Otsu method for separating the foreground and background a black and white image is produced. The black and white cell image is converted into a matrix. The matrix now consists of “alive” (black) and “dead” (white) cells. The algorithm based is on certain rules which examine the neighbourhood of each cell on the matrix. If a cell is white (or “dead”), then the cell will remain dead at the next iteration. The algorithm will continue until all the neighbourhood is white (or “dead”). The algorithm looks into each cell and examines how many of its eight neighbours are alive. If at least eight of the surrounding cells are alive, then the cell under examination will remain alive. If there are less than a eight than the cell dies. Another matrix will record, when

each cell dead. This matrix is the output of the algorithms where each of the points in the matrix represents the shortest distance each point in the image is away from the background. The resulting text file reveals the radius of potential blast cells.


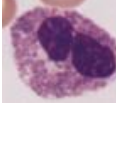

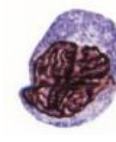

4.4.2 Filtering of Cellular Automata

CA filtering is used to find the location of the cells. The process is based on the expected size (radius) of the five white blood cells (Neutrophil, Eosinophil, Basophil, Monocyte and Lymphocyte) shown on Table 4.1. From the literature and on consulting expert opinion it was decided that the minimum diameter of a white blood cell (Lymphocyte) is 8 micrometres. Each image possibly being under varying magnification and assuming that the maximum value in the Λ matrix corresponds to the size of the largest white cell, filtering is applied accordingly. In particular, by transforming the matrix Λ to a binary image based on this computed threshold. The concepts of filtering are based on the following method:

$$\text{Real Image } (I) \rightarrow \text{CA } (\Lambda) \rightarrow \text{CA filtered image } (G)$$

A system Λ is considered a black box with an input, I and output G . I is the original image, represented in one dimension for simplicity, G the filtered output image, and Λ the filtering operation.

Table 4.1: Mature White Blood Cells and their sizes

| Image | Type |
|---|--|
|  | Neutrophil Size: 12-15 μm in diameter (Tagliasacchi & Carboni, 1997) |
|  | Eosinophil Size: 12 to 15 μm in diameter (The Encyclopedia of Science, 2011) |
|  | Basophil Size: 9-10 μm in diameter (Tagliasacchi & Carboni, 1997) |
|  | Monocyte (Bell & Sallah., 2005). Size: 16-20 μm in diameter (Tagliasacchi & Carboni, 1997) |
|  | Lymphocyte Size: 8-10 μm in diameter (Tagliasacchi & Carboni, 1997) |

4.4.2.1 Algorithm for Filtering of Cellular Automata

Each of the points in the matrix represents the shortest distance each point in the image is away from the background. The calculation is based on the five white blood cells where the minimum value is 8, and the difference between the smallest and biggest of white blood cells is equal to 13. The highest value in the matrix is used to perform the calculation (equation (4.2)). For example, if the last point is 40 we get $(40 \times 8) / 13 = 25$. All the points that have a value less than 25 are converted into black. The output is the processed image which will be used for detection of blast cells. Once we have filtered the image based on the sizes of the blast cells, what remains in the image are regions that represent the largest cellular

bodies. The largest value from the CA change matrix within each of these regions is used as the starting radius for searching potential blast cells. This value represents the shortest distance between a point and the background and thus can be used as an approximate radius. A proof of this can be found in Appendix B.

$$f_i = \left\lceil \frac{8}{13} \max(\Lambda_i) \right\rceil \quad (4.2)$$

4.5 Results

Figure 4.3 shows samples of real images of blood smears corresponding to AML of type M1, M2, M3 and M5 provided by the Departmental Heamatology, USM, Malaysia.

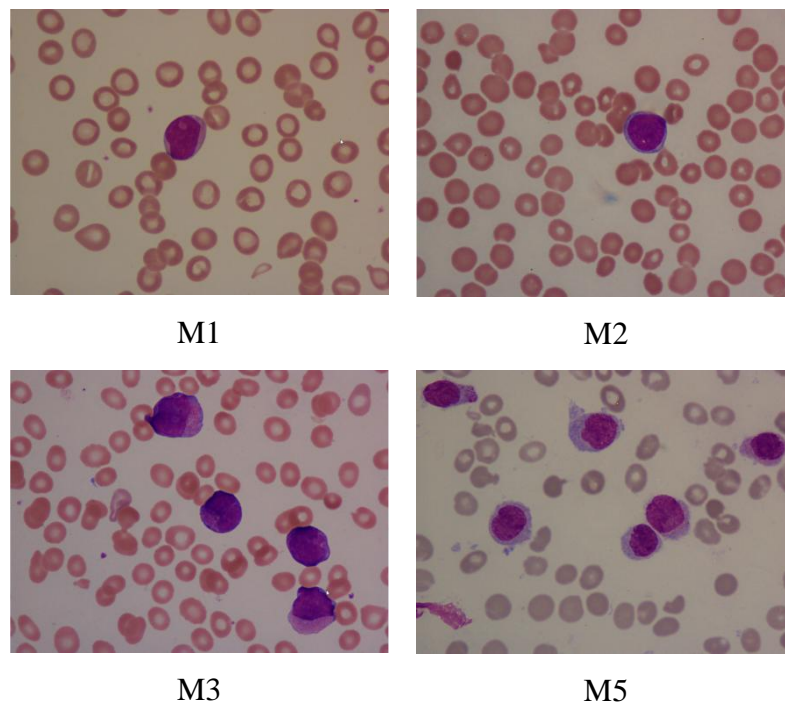


Figure 4.3: AML Images

4.5.1 Otsu

The dataset consists of 47 images of M1 leukaemia subtype, 129 images of M2 leukaemia subtype, 92 of M3 subtype, and 54 of M5 AML subtype. Thus a total of 322 images. The images were subjected to processing by the Otsu method with a 100% success rate. Figure 4.4 shows a few examples of the obtained results.

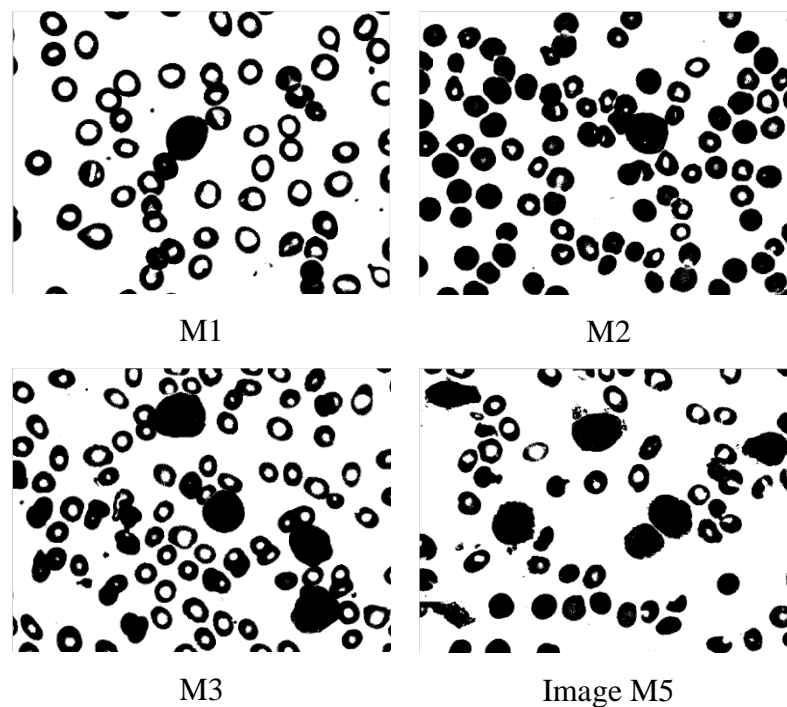


Figure 4.4: Examples of images processed by the Otsu Method

4.5.2 Cellular Automata

Following application of the Otsu method, images are subjected to processing by CA, according to the rules discussed in section 4.4.1. All images were successfully converted by the CA method. Figure 4.5 shows a few examples of the images produced following conversion. CA are used to locate potential blast cells which occupy the largest areas in the image. At the same time, they are used

to determine the radius of the cells. Following conversion, the images were checked manually.

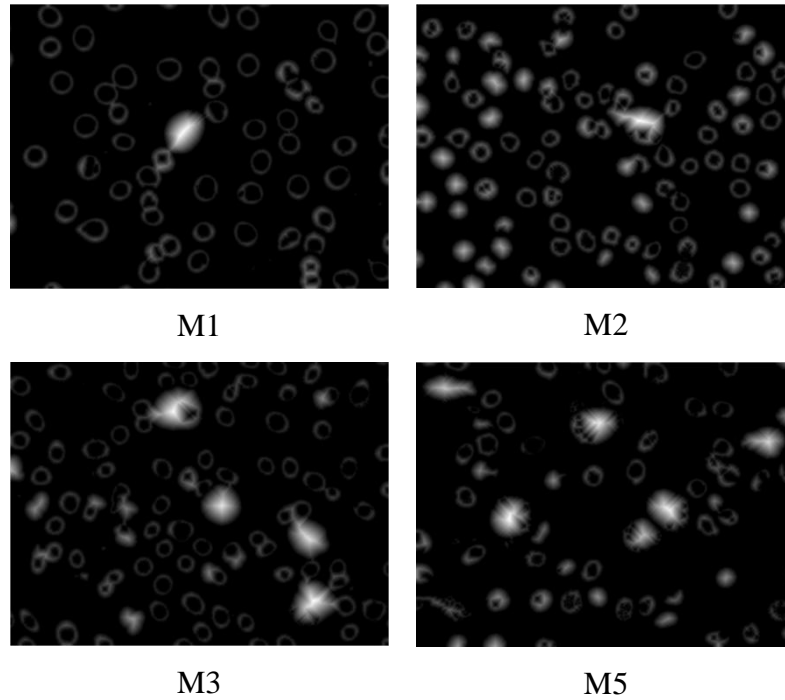


Figure 4.5: Examples of images processed by CA

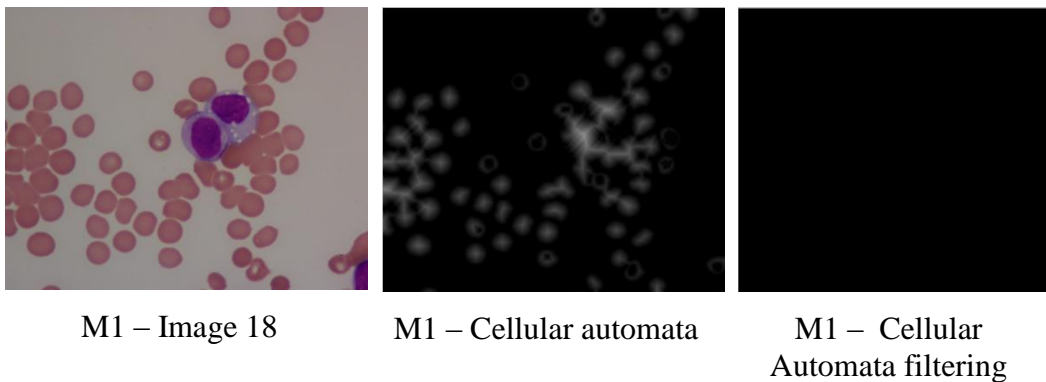
4.5.3 Cellular Automata after filtering

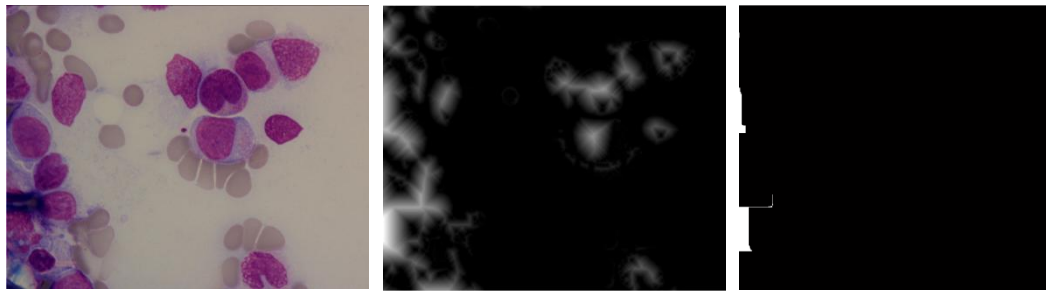
The final step in detecting potential blast cells consists of CA filtering. A total of 320, hence 98.76% of the 322 images were successfully converted. Table 4.2 summarise the success of CA conversion per AML subtype. Again, the images were manually checked following conversion.

Table 4.2: Percentage of successful converted images using CA Filtering method

| Subtypes of leukaemia | Real Images | Image not working | Cellular Automata after filtering method | Target |
|-----------------------|-------------|-------------------|--|--------|
| M1 | 47 | 1 | 46 | 97.87% |
| M2 | 129 | 0 | 129 | 100% |
| M3 | 92 | 0 | 92 | 100% |
| M5 | 54 | 1 | 53 | 98.15% |

The CA filtering method did not work for two images. As mentioned in the algorithms section 4.4.2.1 all points of value less than the one obtained by applying equation (4.2) are converted to black. For the two images, M1-18 and M5-34 the method did not work as they are blurred (Figure 4.6). The two images where CA filtering did not work can be subjected to an enhancement technique to make them darker.

**Figure 4.6(a):** Images where CA filtering was unsuccessful (M1 – Image 18)

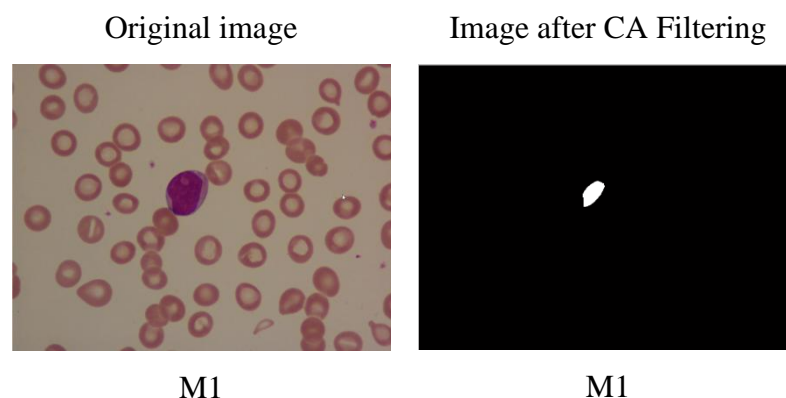


M5 – Image 34

M5 – Cellular Automata

M5 – Cellular
Automata filtering**Figure 4.6(b):** Images where CA filtering was unsuccessful (M5 – Image 34)

Figure 4.7 shows a few examples of successful CA filtering. The white spots on the black and white images correspond to blast cells and constitute successful starting points for the seeded search. The application of CA filtering has also removed noise from the images. The small dots in the M2 image are not noise but a result of where the condition in equation 4.2 has been accepted. As will be seen in the following chapter, all of the remaining parts of the image will be checked to see if they correspond to a purple part of the original slide image (blast cells) or pink (red blood cells). If the small dots are pink (red blood cells) the coordinates will not be accepted for the next stage (heuristic search).

**Figure 4.7(a):** Examples of successful application of CA filtering (M1)

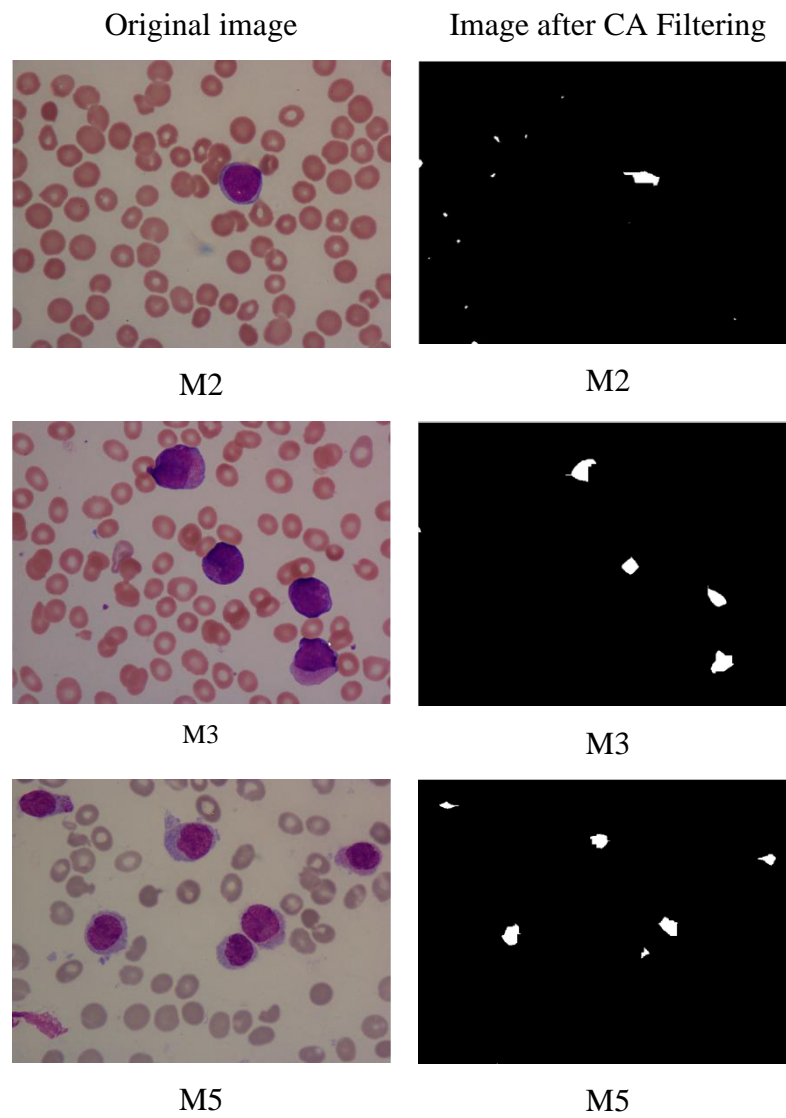


Figure 4.7(b): Examples of successful application of CA filtering (M2, M3, M5)

4.6 Summary

This chapter, expanded on a methodology to detect the coordinates of potential blast cells in colour blood smear images, to serve as starting points for a subsequent “seeded” heuristic search.

The Otsu and CA methods, constituting the first two steps in the process, were successful in conversion of all images in the dataset, which was confirmed by visual inspection. CA filtering appeared successful in 98.76% of instances, failing in two cases. The results regarding these initial steps in the process of blood cell detection are very promising.

Chapter 5

COORDINATE DETECTION AND COLOUR IMAGE CLASSIFICATION

5.1 Introduction

The previous chapter expanded on the Otsu method, CA and CA filtering. In order to identify the coordinates of potential blast cells in blood smear images, which serve as starting points for the heuristic search methods.

The work described in this chapter consists of three main steps, detecting the starting coordinates, colour image classification and lastly a method of merging circles, targeting blast cells, when they overlap. In essence the chapter is centered on what is referred to as the “seeded” step. This is based on the concept of seeded region growing (Rolf & Bischof, 1994). The technique aims to segment an image into areas containing objects of interest. This method uses the locations of one or more pixels as seeds or starting points for processing by the algorithm described in (O'Neill, 2005). The method of coordinate detection aims to locate

blast cells present in the image. The next step is the application of colour image classification to determine if the detected cell is a blast (purple) or a red (pink) blood cell. If coordinates of circles overlap, then both coordinates are merged into one circle. The method is called to duplicate coordinates.

This chapter is organised as follows: section 5.2 provides a discussion of previous work in the field, section 5.3 describes the work process and methods used in this chapter. Section 5.4 presents the method and algorithm used to locate the coordinates of cells, image clustering and duplicate to coordinates. Section 5.5 presents the produced result. Lastly, section 5.6 presents a summary of this chapter.

5.2 Previous Work

This section provides an overview and discussion of previous works conducted in the field of image clustering.

5.2.1 Image Classification

Classification is a supervised method where the categories/classes, to which images may belong, have been determined by the user prior to processing by the used algorithm. For example, here there are two categories, which are pink (red blood cells) and purple (blast cells) and the algorithm uses the Euclidean Distance metric to perform the classification step. This metric was chosen since it is a popular, straightforward method for estimating the shortest distance between two points. (Bosch et al., 2007), research in finding few objects in the large images. Three steps were used which are target the shape and appearance of the objects in

the image. Secondly, automated selection of region to perform training data and lastly, performed multi forrest for classifier.

In (Malinga et al., 2006) the authors use a technique to partition the image into $M \times N$ local blocks, then extract the features for each block and calculate the pattern similarity measures that are used in the applied variation of the K-Means clustering technique. In the K-Means algorithm, they use Euclidean Distance as a similarity metric to cluster the images. In computer graphics, colour image quantization is a process that reduces the number of different colours present in an image. It is an essential process of representing true-colour images using a small number of colours in a colourmap. The objective of colour image quantization is to quantize a true-colour image into one with fewer colours. In (Sirisathitkul et al., 2004) the authors propose an algorithm that employs the squared Euclidean distance of adjacent colour points along the highest colour variance axis. Once the colourmap is constructed, each colour pixel can be replaced by finding the nearest neighbour of a colour in the colourmap.

This chapter presents the use of Euclidean distance matrix to examine if each pixel in the image is closer to purple or pink, based on its RGB values. Image histogram is a graphical representation of colour distribution in digital images. It plots the number of pixels for each tonal value. This means that images clustered or retrieved by using the global colour histogram may not be semantically related even though they might share similar colour distributions (Malinga et al., 2011).

5.2.2 Duplicate coordinates

After finding the coordinates of circles, Venn diagrams are used to check if the circles overlap, and if so to calculate their internalise and intersection. If duplicate

coordinates are found, the circles need to be separated by using a midpoint. Otherwise, the next step (Heuristic Search) cannot be performed, since the presence of overlaps would entail a large amount of computational overhead.

Venn diagrams were invented by John Venn in 1880 as a way of picturing relationships between different groups of objects. Venn diagrams can be used to represent both class relationships and logical relationships (Staple, 2011). A Venn diagram is a schematic diagram used in logic theory to depict a collection of sets and represent their relationships. The Venn diagrams of two and three sets are illustrated in Figure 5.1. The 2-set diagram consists of two intersecting circles, producing a total of four regions A , B , $A \cap B$ denoting the intersection of sets A and B and the empty set. The order-three diagram consists of three symmetrically placed, mutually intersecting circles, comprising a total of eight regions. The regions labelled A , B and C consist of members which are only in one set and no others, the three regions labelled $A \cap B$, $A \cap C$, $B \cap C$ consist of members which are simultaneously in two sets but not in the third. The $A \cap B \cap C$ region consists of members that are in all three sets (Weisstein, 2011).

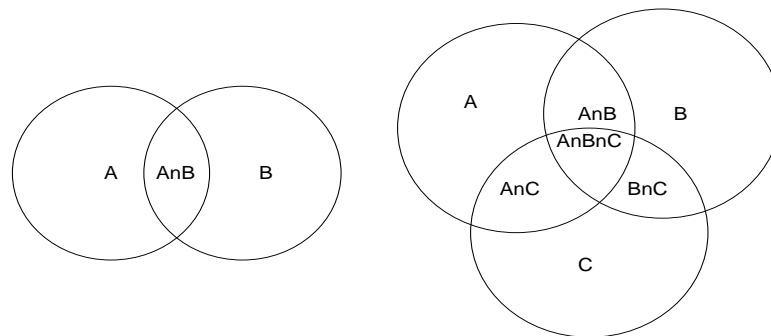


Figure 5.1: Traditional Venn diagrams

If the circles overlap, a midpoint is required. A midpoint is a point that denotes the middle of any given line segment. The midpoint theorem says the x coordinate of the midpoint is the average of the x coordinates of the endpoints and the y coordinate is the average of the y coordinates of the endpoints.

5.3 Work Process

This section presents the work process described in this chapter. First, the starting point coordinates are identified, which is followed by colour image classification and processing by the duplicate coordinates method. The steps are depicted in Figure 5.2. Further analytical steps are the subject of subsequent chapters.

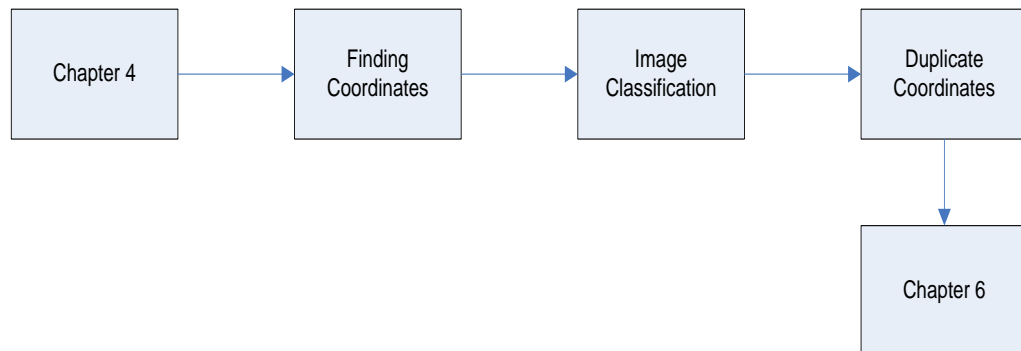


Figure 5.2: Work Process for chapter 5

5.4 Methods

The following sections, comment on the methodologies of starting coordinates detection, colour image classification and treatment of occurrences of duplicate coordinates.

5.4.1 Finding Starting Coordinates

This section deals with the process of finding the locations of potential blast cells, to serve as starting points for the subsequent search. The choice of radius was the subject of the CA method, defined as the shortest distance each point in the image is away from the background.

5.4.1.1 Finding Starting Coordinates Notation

Firstly, let's assume a box encompassing a potential blast cell as shown in Figure 5.3 acquired by drawing a square box outside the potential blast cells. By finding the middle coordinate in the blast cells, then it required x_1, y_1 are the coordinates top left of potential blast cells located and x_2, y_2 as the coordinates bottom right of the potential blast cells located. Then both coordinates will add and divide by 2 to find center of the blast and radius (r) from CA distance

$x_1 =$ Finding the x_1 in starting the cells location
 $y_1 =$ Finding the y_1 in starting the cells location
 $x_2 = x_1 + r$
 $y_2 = y_1 + r$
 $x = (x_1 + x_2) / 2$
 $y = (y_1 + y_2) / 2$
 $r =$ Radius from cellular automata

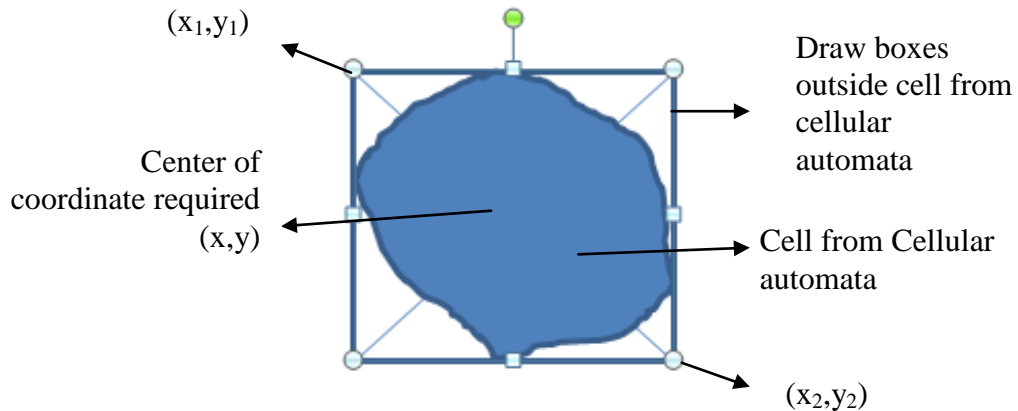


Figure 5.3: Finding the starting coordinate

5.4.1.2 Algorithm for finding starting coordinates

The input of the algorithm for detecting the starting coordinates is an image processed with the CA filtering method. The algorithm draws a square box around

potential cells. The output will be coordinates x and y of the centre of a potential blast cell.

5.4.2 Image Classification

After selecting the coordinates it is checked if a given cell is a blast or a red blood cell. For this the radius of blast cells calculated as described in chapter 4 is required.

5.4.2.1 Image Classification Notation

Having acquired coordinates x and y and the radius r , a circle is drawn at the potential blast cell's location. Table 5.1 shows the colour values corresponding to Figure 5.4. Colours were manually determined based on eye judgment from twenty randomly selected images. Colour image clustering is based on the use of basic partitions for Red (R), Green (G) and Blue (B). Here blue is not used, due to its minimal presence.

Table 5.1: RGB values for purple and pink for image classification

| | Red | Green | Blue |
|---------------|------------|--------------|-------------|
| Purple | 103 | 15 | 81 |
| Pink | 132 | 81 | 79 |

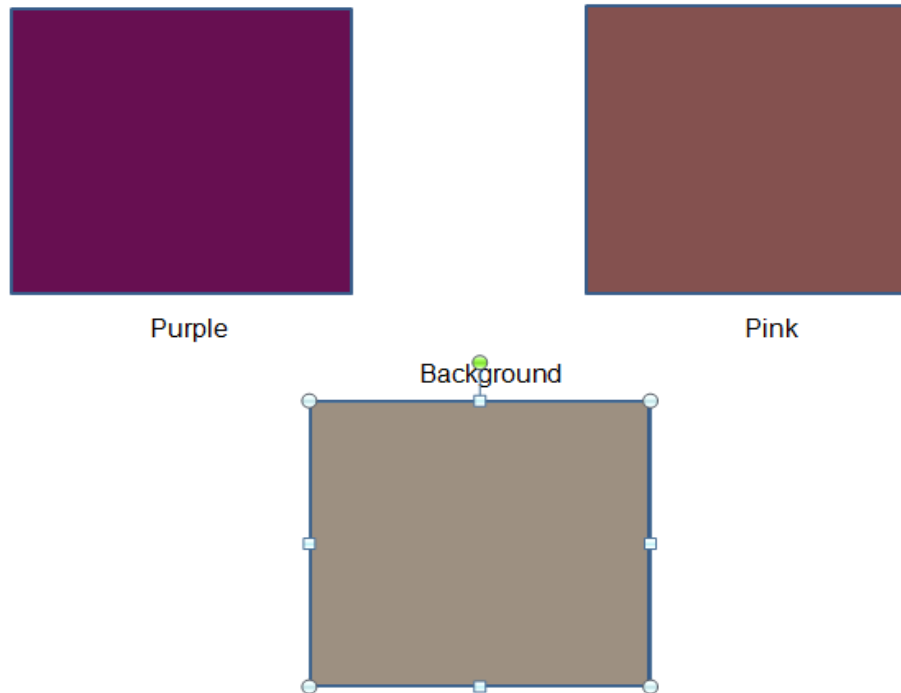


Figure 5.4: Colours corresponding to red, blast cells and background

Below is the calculation of purple, corresponding to blast cells (pink corresponds to red blood cells). Colours were cross-referenced with Microsoft Power point colours to identify their *RGB* values. In future work, HSL (Hue, Saturation, Lightness) values can also be utilised along with RGB values.

5.4.2.2 Algorithm for Image Classification

The input of the classification algorithm consists of an AML image, a list of x and y coordinates of potential blast cells and their radius r . The circles are expected to encapsulate blood cells. Once large circular objects have been located, each pixel within that circle needs to be checked to see if it is “pink” (meaning that it corresponds to a red blood cell) or purple (corresponding to the dyed colour of the blast cell). This is done by comparing the RGB scales of each pixel to see whether

they are closer to a typical “pink” or “purple” vector. Once all of the pixels within a circle have been classified, the overall type of a cell is determined by which count is the largest.

5.4.3 Duplicate Coordinates

After identifying the coordinates of the potential blast cells, the next step is to check if some circles are found at the same location, and if so calculate their internalised and intersection as shown in Figure 3.4. If two circles are located at different locations, both coordinates are accepted. Otherwise the duplicate coordinates method is applied. This method will identify the middle of two circles as shown in Figure 5.5 to estimate a pair of new coordinates. If the coordinates cannot be separated, the next step, that is the heuristic search, cannot be performed. This is due to large amount of computational overhead that the optimisation of blast cell detection would entail.

5.4.3.1 Duplicate Coordinates Notation

Again, in the used notation r represents the radius, and (x) and (y) the coordinates of the centre of a circle. Figure 5.5. shows an example of two overlapping circles.

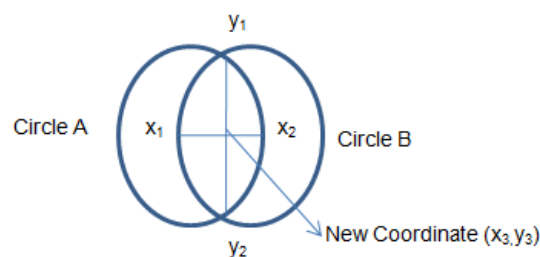


Figure 5.5: New Coordinate Diagram

5.4.3.2 Algorithms for Duplicate Coordinates

The algorithm for duplicate coordinates aims to find the center of (x_1) and (x_2) (coordinate x) by adding the distance of both coordinates and dividing it by two. The same equation is used to calculate the value of coordinate y .

5.5 Results

This section presents the results of applying the methods described in the preceding sections of the chapter.

5.5.1 Finding the starting coordinates

As previously discussed, the dataset consists of a total of 320 images, corresponding to four AML leukaemia subtypes, with 2 images unsuccessfully processed by the CA Filtering method. The identification of the coordinates of potential blast cells, to serve as starting points for the heuristic search methods, was successful in all instances. The results were confirmed manually.

Figure 5.6 shows a few examples of images, processed with CA filtering and the resulting identification of the x and y coordinates of cells on the images.

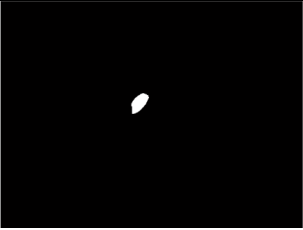
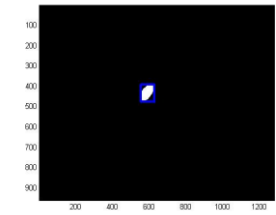
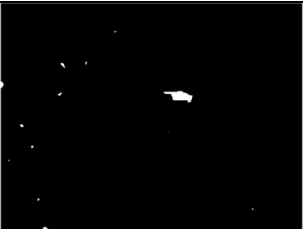
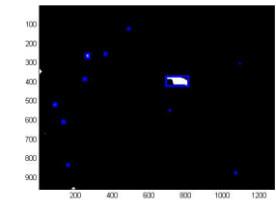

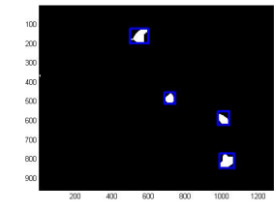
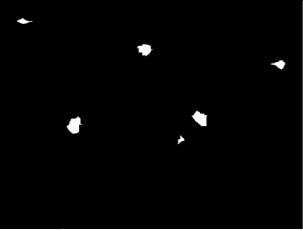
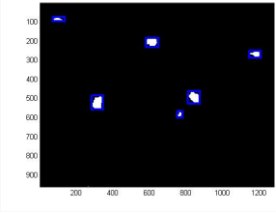
| Image from CA (a) | Detecting the potential blast cells (b) |
|---|--|
|  <p data-bbox="539 680 587 712">M1</p> |  <p data-bbox="1070 689 1118 721">M1</p> |
|  <p data-bbox="539 956 587 987">M2</p> |  <p data-bbox="1070 956 1118 987">M2</p> |
|  <p data-bbox="539 1218 587 1249">M3</p> |  <p data-bbox="1070 1218 1118 1249">M3</p> |
|  <p data-bbox="539 1487 587 1518">M5</p> |  <p data-bbox="1070 1487 1118 1518">M5</p> |

Figure 5.6: Examples of blast cells coordinate detection

5.5.2 Image Classification

Table 5.2 show the results of the classification process. In both M1 and M3 AML cases all images were correctly processed and cells assigned to the right type. Here the term classification refers to the process of correctly identifying cells

either as red or blast cells. For three images in (1 in M2 and 2 in M5) the classification did not work. It means that all identify cell where classification into red blood cells. However, in images belonging to the M2 subtype, one was classified incorrectly while in the group of M5 images this occurred in two cases. Thus the success was 99.22% and 96.22% respectively, with overall success of 98.86% for the entire dataset. Please refer to Appendix C for full results of image classification.

Table 5.2: Summary of Image Classification efficiency overall

| Subtypes of leukaemia | Number of Images | Wrong Classification | Correct Classification | Success rate |
|-----------------------|------------------|----------------------|------------------------|--------------|
| M1 | 46 | 0 | 46 | 100% |
| M2 | 129 | 1 | 128 | 99.22% |
| M3 | 92 | 0 | 92 | 100% |
| M5 | 53 | 2 | 51 | 96.22% |

Figure 5.7 displays the wrongly classified images. Although there are areas targeting potential blast cells, the targeting circles contain more pink than purple pixels, in image 113 (M2 subtypes), image 1 and image 2 (M5 subtypes). This is due to overlapping between blast and red blood cells.

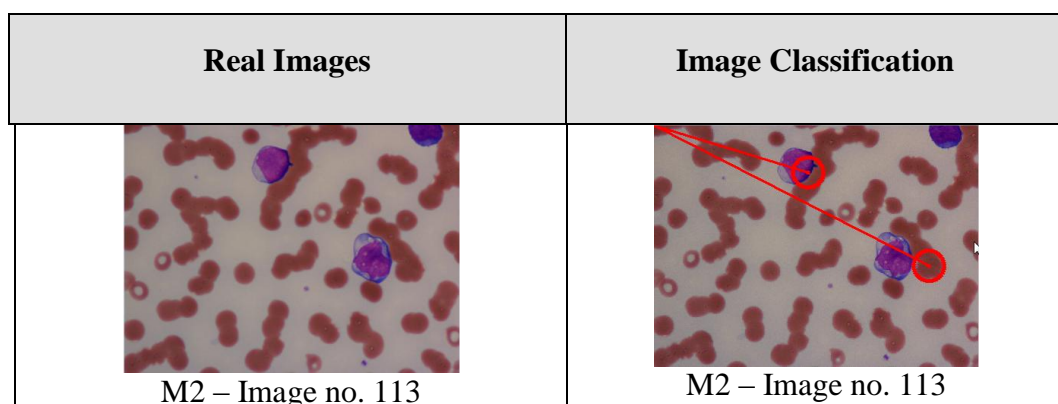


Figure 5.7(a): Wrong classification of blast cells (M2)

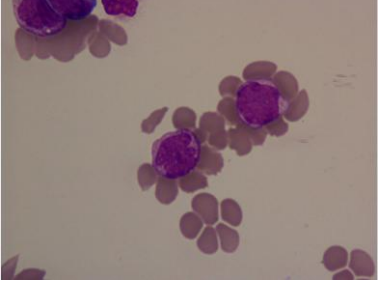
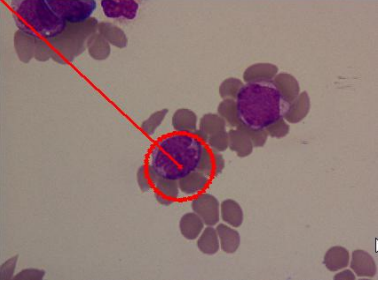
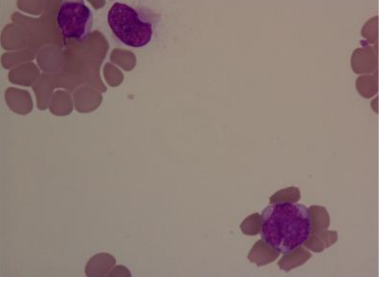

| Real Images | Image Classification |
|--|--|
|  <p data-bbox="467 745 703 779">M5 – Image no. 1</p> |  <p data-bbox="999 745 1235 779">M5 – Image no. 1</p> |
|  <p data-bbox="467 1068 703 1102">M5 – Image no. 2</p> |  <p data-bbox="999 1068 1235 1102">M5 – Image no. 2</p> |

Figure 5.7(b): Wrong classification of blast cells (M5)

Figure 5.8 shows some examples of images where clustering has been successful. The circles correctly surround blast cells.

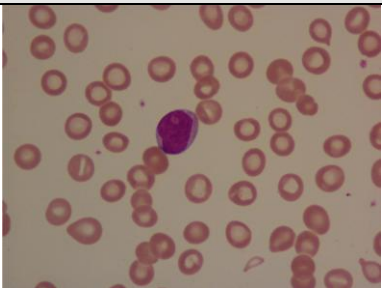
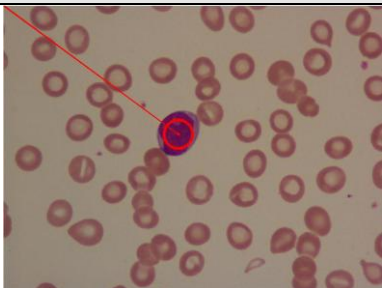
| Real Image | Image Classification |
|---|--|
|  <p data-bbox="568 1776 616 1809">M1</p> |  <p data-bbox="1086 1776 1134 1809">M1</p> |

Figure 5.8(a): Examples of successful Image Classification (M1)

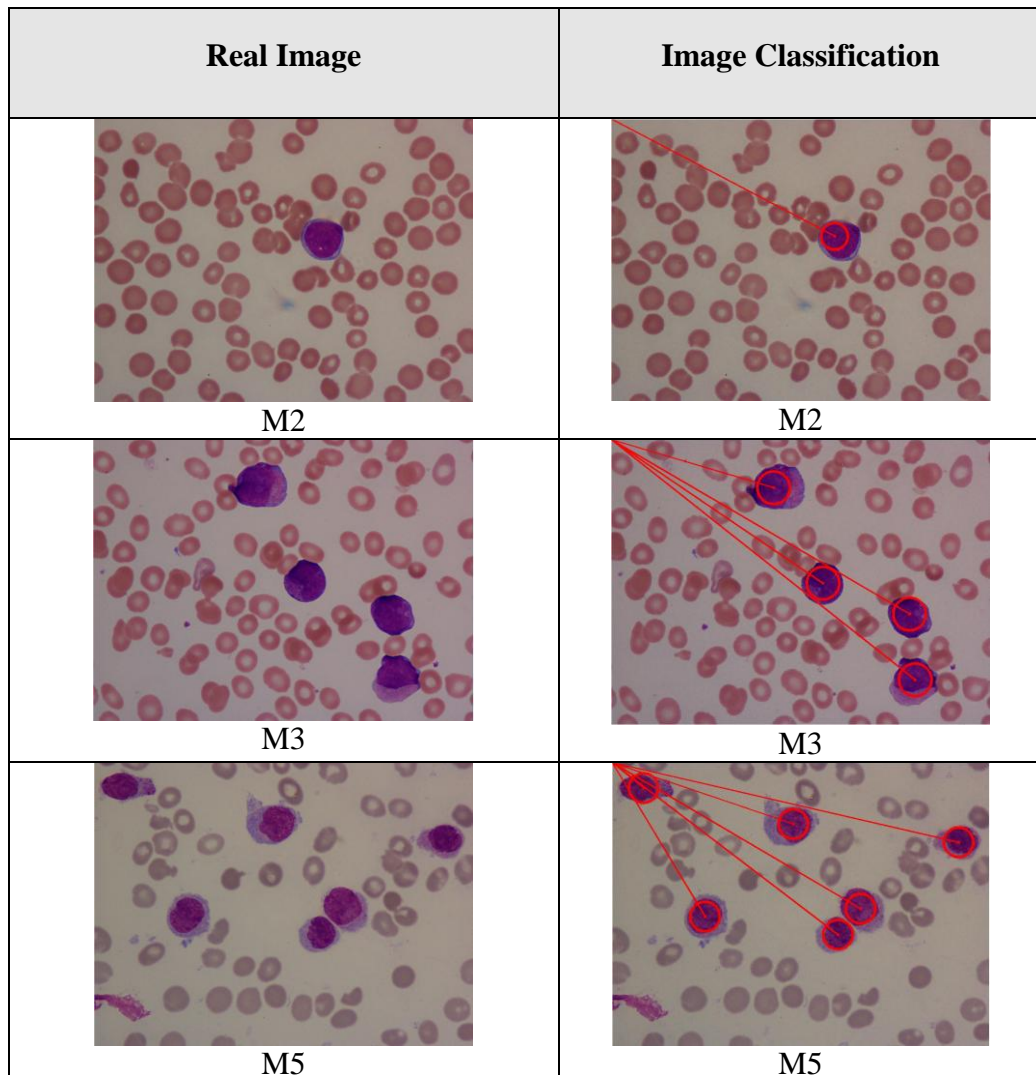


Figure 5.8(b): Examples of successful Image Clustering (M2, M3, M5)

Table 5.3 and Figure 5.9 show a summary of the image classification step efficiency. From the result shown, it is clear that it is best in the case of M3 images (96.27%). This is because they are all blood smear images of good quality. The worse results correspond to images of M5 subtype AML, taken from bone marrow. In the diagnosis of leukaemia of M5 subtype, blood from bone marrow was required. The images are blurred, contain a lot of noise and a large number of overlapping blast cells. In the case of M1 and M2 images, some are taken from

bone marrow and others from blood smears. A number of images are blurred and noisy and some cells overlap, but not as many as in the case of M5 AML.

Table 5.3: Summary of Image Classification efficiency

| Sub Type | Number of cells | Detected Cells | Undetected Cells | Wrong Classification | Percentage % |
|----------|-----------------|----------------|------------------|----------------------|--------------|
| M1 | 96 | 87 | 9 | 2 | 90.63 |
| M2 | 276 | 234 | 42 | 6 | 84.78 |
| M3 | 161 | 155 | 6 | 0 | 96.27 |
| M5 | 237 | 187 | 40 | 3 | 79.90 |

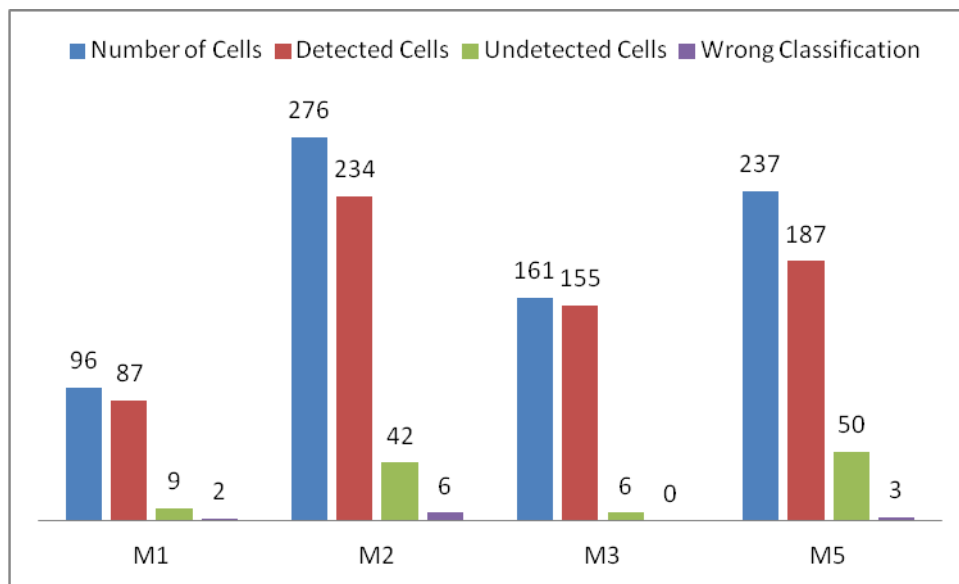


Figure 5.9: Summary of Image Classification Efficiency

Figure 5.10 shows another example where one blast cell has been classified incorrectly. In some cases the circles that target blood cells may contain too many background pixels which can create problems for the classification step.

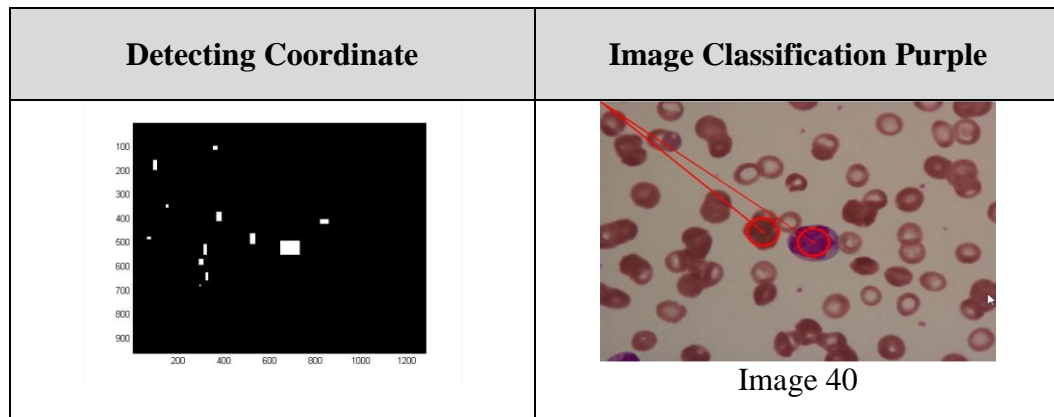


Figure 5.10: Wrong classification of blast cells

Figure 5.11 here the starting coordinates did not target the potential blast cells. The coordinates detection method was unsuccessful, and the targeting areas in the images contain mostly pink pixels.

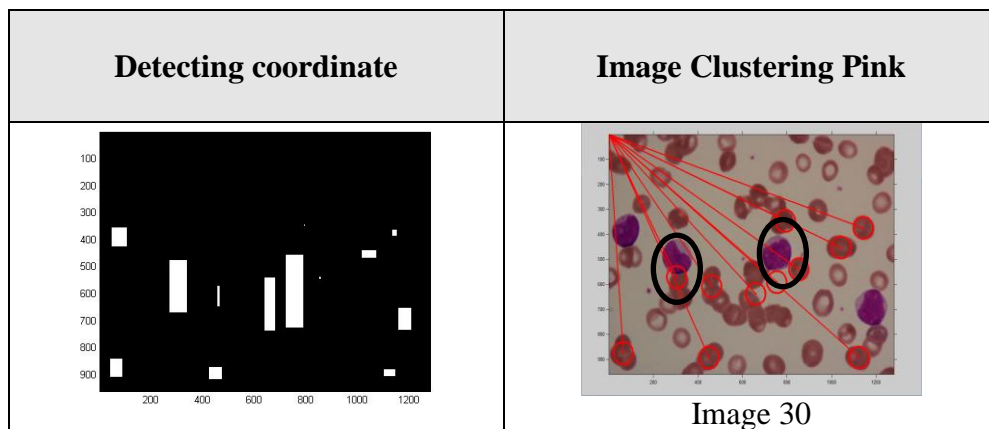


Figure 5.11: Wrong classification of blast cells from detecting coordinates

Figure 5.12 shows an example of one potential blast cell that has been classified as red blood cell. As the image shows, the circle contains some background and red blood cells (black circle).

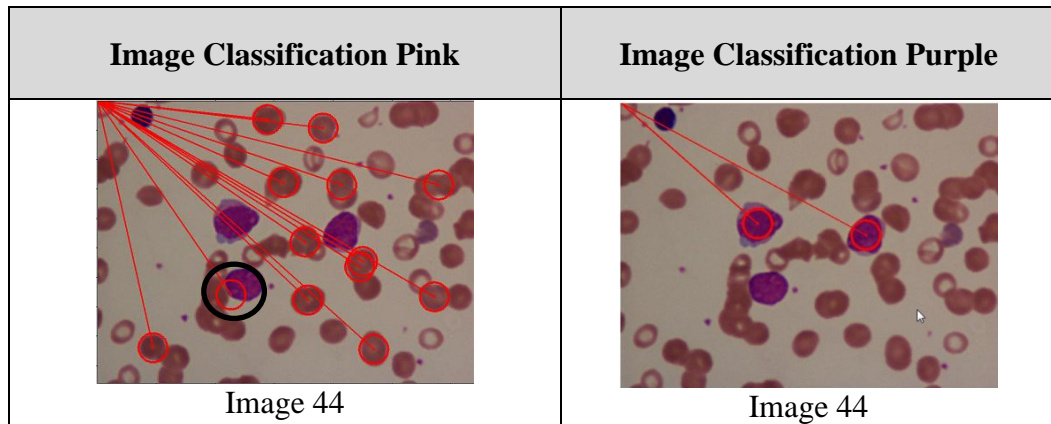


Figure 5.12: Potential blast cells classified to pink

A figure 5.13 shows an example of overlapping blast cells. As a result some blast cells have been missed by the analysis.

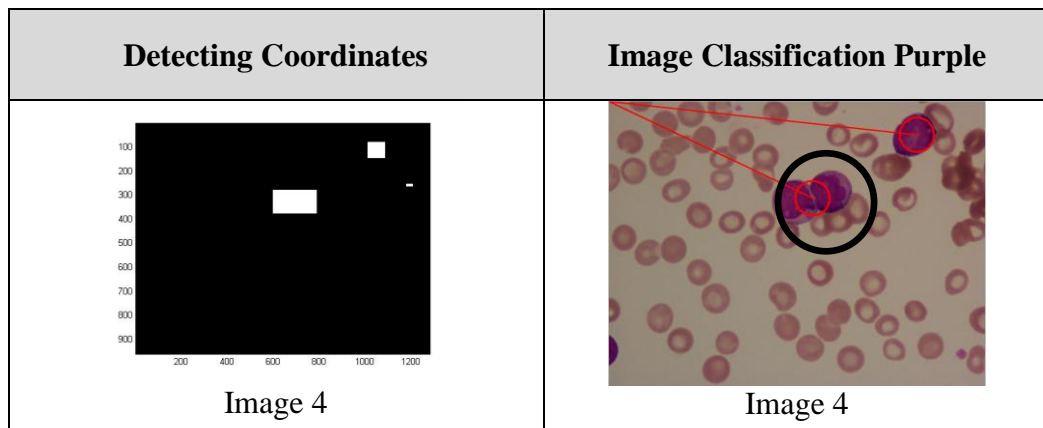


Figure 5.13: Overlapping blast cells

The black circle indicates the blast cells, showing that some have not been detected. Figures 5.14 shows examples where blast cells have been completely missed during the CA filtering stage. This may happen during the CA filtering where the condition of the minimum and maximum of white blood cells is not met as shown in the section 4.4.2.

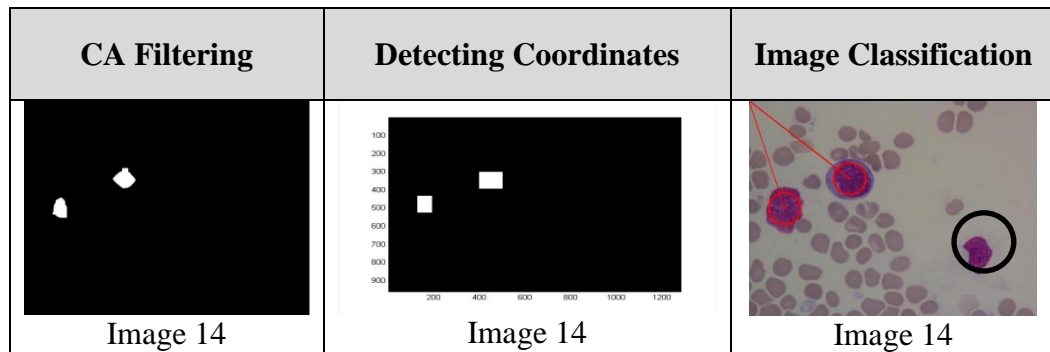


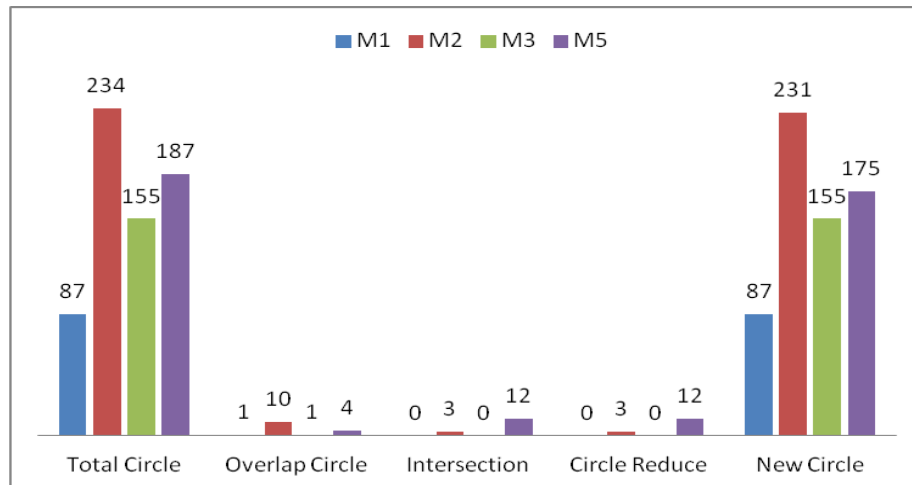
Figure 5.14: Blast cells undetected

5.5.3 Duplicate Coordinates

Table 5.4 summarises the results of the duplicate coordinates method for all AML images, of subtype M1, M2, M3 and M5. Overall, targeting of overlapping circles was 98.74% efficient. The duplicate coordinates method is relevant when circles are located in the same area and overlap. The table indicates the internalised and intersection of circles. The circle reduced field displays how many circles have been reduced to one common circle by the method. The new circles field indicates how many circles are identified following the merging process. The images were checked manually. Figure 5.15 displays a graphical summary of the method's performance. The total number of new circles, detecting the blast cells, is 648. The full images are contained in Appendix D.

Table 5.4: Summary of Duplicate Coordinate

| Sub Types | Total Circles | Overlap Circles | Inter section | Circle Reduced | New Circles | Accuracy |
|-----------|---------------|-----------------|---------------|----------------|-------------|----------|
| M1 | 87 | 1 | 0 | 0 | 87 | 100% |
| M2 | 234 | 10 | 3 | 3 | 231 | 98.72% |
| M3 | 155 | 1 | 0 | 0 | 155 | 100% |
| M5 | 187 | 4 | 12 | 12 | 175 | 96.22% |

**Figure 5.15:** Summary of Duplicate Coordinate

In Figure 5.16, image 43 (a) two circles internalised. After processing, as shown in image 43 (b), the two circles are merged into one.

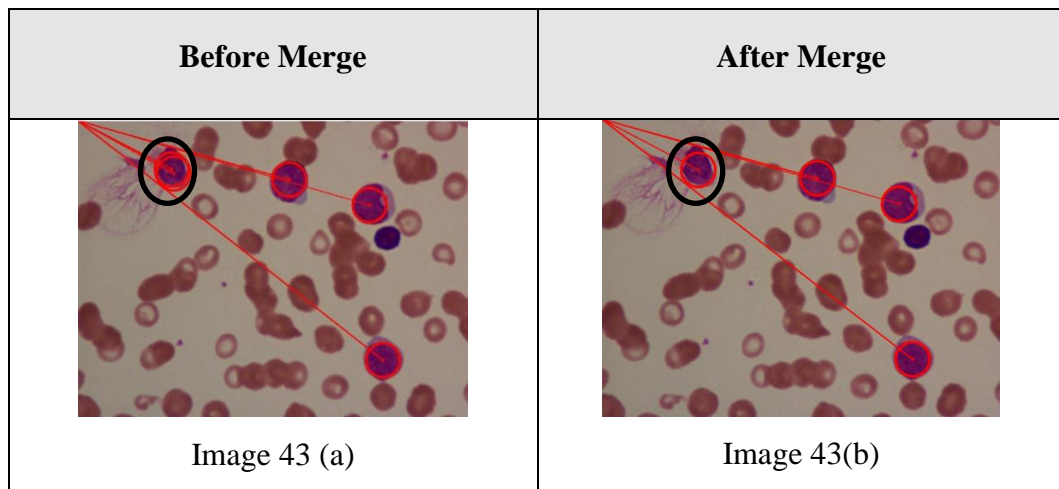


Figure 5.16: Internalised method applied

Figures 5.17 show examples of intersecting circles, where the number of identified cells is reduced, as only one pair of new coordinates is defined. For example, in image 1, two circles are reduced to one in the area highlighted by the bigger black circle.

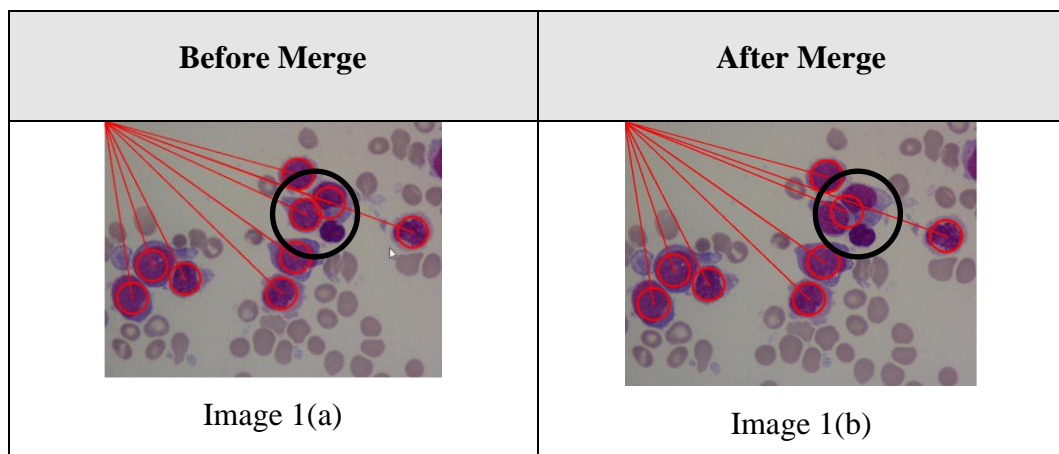


Figure 5.17: Intersection circles duplicate

Figure 5.18 shows M3 subtypes of correct merging of circles targeting the same blast cell.

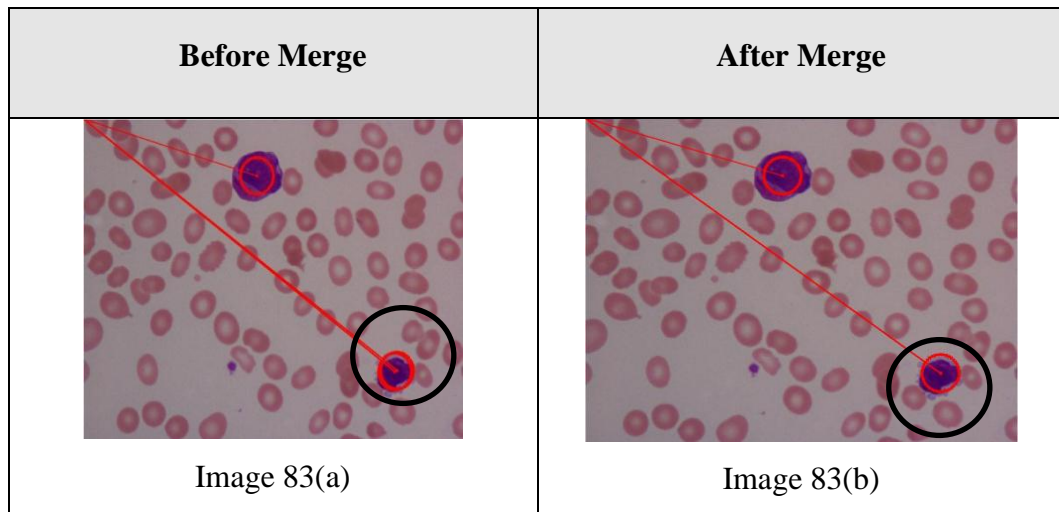


Figure 5.18: Same location duplicate coordinates method

5.6 Summary

In brief, three methods to process the AML images were discussed. Firstly, the CA method for coordinate detection, to extract the x and y coordinates of the centre of a circle, corresponding to a potential blast cell.

Second, the colour image classification method, using the Euclidean Distance metric to classify cells into blasts and red blood cells, based on pixel colour was discussed. 317 images were classified correctly. In M1 images 89.58% of blast cells were targeted correctly, in M2 subtypes 85.14% while in M3 and M5 96.27% and 79.77% respectively. There are several reasons for incorrect detection of blast cells. One of them is the overlapping of blast cells. Another reason is the failed capturing of potential blast cells during application of the CA filtering method. In some cases conditions for capturing a cell are not met. Additionally, in many cases the targeting circle includes a large number of pixels belonging to red blood cells.

Finally, the duplicate coordinates method was discussed. The aim of this step is to remove overlapping circles targeting the same cell. In many cases the merging process was successful, leading to detection of the correct number of blast cells, especially in M1 and M3 AML images. In other cases, the merging resulted in detection of less blast cell than the ones actually present on the slide.

In conclusion the results are quite good and satisfactory, indicating that the methodology is useful and worth developing further. The next analytical step, namely the application of heuristic search, is discussed in the next chapter.

Chapter 6

SEEDED HEURISTIC SEARCH

6.1 Introduction

The work process described here extends the work in chapters 4 and 5 regarding finding the coordinates of starting points for the application of Seeded Heuristic Search. The aim of this chapter is to optimise the targeting circle for potential blast cells before proceeding with the classification step. The results show the “best” heuristic search method for detecting blast cells seems to be the SA rather than the HC and GA methods. Colour images are used for experimentation, to identify blood cells and divide them into blast (purple) and red blood cells (pink). Furthermore, the chapter presents further work on verifying the detection process of overlapping cells, to continue with the optimisation of blast cells detection with different methods. The classification step that follows the work presented in this chapter is the subject of chapter 7.

This chapter is organised as follows: section 6.2, presents previous work on heuristic search and section 6.3 describes the work process followed here. Section 6.4 comments on the fitness function, the notation and algorithms used, including the HC, SA and GA. Section 6.5 presents the results and the choice of the most appropriate method to analyse the actual data. Lastly, 6.6 present the conclusions of this chapter.

6.2 Previous Work

Chapter 3 showed that a search starting from random points does not produce satisfactory results. The use of HC and SA methods is presented in (Waidah et al., 2010). Here, the methodology is extended to a Seeded Heuristic Search where the coordinates of potential blast cells are detected to serve as initial points in the search, producing good results. As previously discussed a HC, SA and a GA are facilitated for the analysis. In (Waidah et al., 2010) the use of GA Random and Seeded process is discussed in some detail. Both papers mentioned here present a comparison of the different heuristic search techniques. This chapter is an extension of previous research by (Waidah et al., 2010) using real AML images.

A publication by (Lochanambal & Karnan, 2010) discusses a number of methods of heuristic search, including Simulated Annealing (SA), Tabu Search (TS), Genetic Algorithms (GA) and Ant Colony System (ACS). The paper also discusses the use of hybrids, that is, combinations of the four methods, used to maximise the mammogram segmentation. The paper shows that a combination of ACS and TS produces better results. This thesis also presents a combination of methodologies, namely the use of CA and Heuristic search. (Jie-Sheng et al., 2009) use a novel approach based on Harmony search (HS) and SA to successfully detect image edges automatically. The effectiveness of the proposed method was verified by simulation.

6.3 Work Process

This section presents two work processes. The first one (Figure 6.1) continues from the previous chapters, where the Otsu method for extracting objects from their background was described. Otsu's method uses a grey scale threshold for the separation of objects in the foreground and background of an image. This was followed by the application of CA to transform the black and white cell image to a matrix (the same size as the input image) where each point in this matrix represents the shortest distance each point in the image is away from the background. This CA matrix can be then used to locate suitable candidate regions that correspond to the largest objects in the image. The hypothesis is that these will correspond to the blast cells which are trying to locate. Then it is determined if the candidate cells are blasts or red blood cells, based on their colour. The heuristic search described here is termed Seeded because it uses predefined starting points (seeds). The methods are executed 10 times, in order to examine the consistency of the end results. The ratio of the overlap between the detecting circle, estimated at each run, was used for that purpose. However, in seeded heuristic search, the coordinates have been defined as described in Chapter 5.

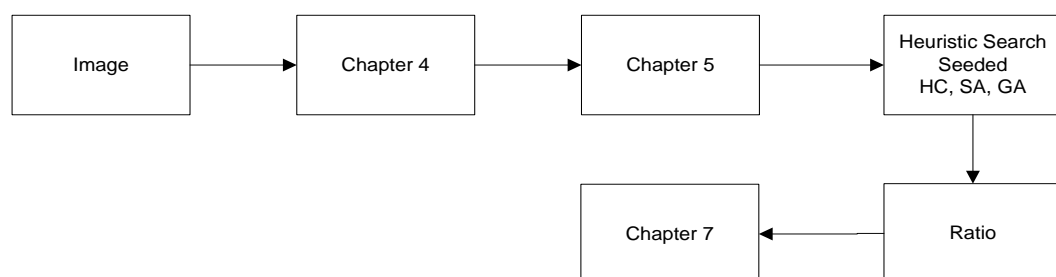


Figure 6.1 Work Process for Seeded Heuristic Search.

The process is described in Figure 6.2 for heuristic search, using colour fitness function and colour images. There are two analytical processes, a random

search discussed in chapter 3 and a seeded search, where starting coordinates have been defined.

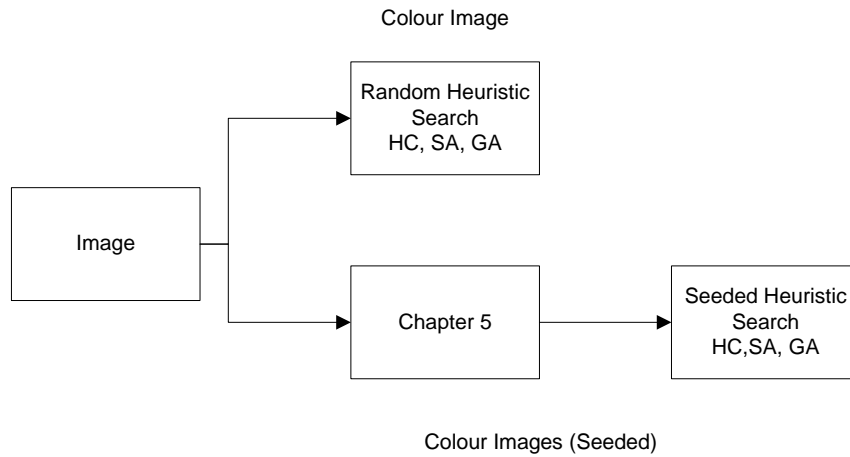


Figure 6.2: Work Process for colour image for Random and Seeded Heuristic Search

6.4 Fitness Function

The fitness function used by the GA is a slightly modified version of the one used in the HC and SA method if the circles overlapping.

6.4.1 Fitness Function for Hill Climbing, Simulated Annealing and Genetic Algorithm

The fitness function used by the HC and SA was presented in section 3.4.4 and the process of coordinates detection was defined in Chapter 5. The pseudo codes of the HC and SA algorithms were also presented in sections 3.4.4.2 and 3.4.4.3 respectively. The fitness function of the GA was discussed in section 3.4.6 as it differs slightly from the ones used by the HC and SA methods, in that it incorporates circles overlap similarity metric parameter discussed in section 3.4.5.

6.4.2 Colour Images Fitness Function for Hill Climbing, Simulated Annealing and Genetic Algorithm

This section expands on the colour images fitness function for HC, SA and GA. As mentioned above the GA uses a slightly modified version of the fitness function implemented in the HC and SA methods.

6.4.2.1 Colour Fitness Function Notation

The fitness equation for HC and SA same as Equation 3.1. The GA same with Equation 3.6, but the differences by changing the form $B(i)$ is the number of black points pixel to $P(i)$ is the number of purple points and $W(i)$ is the number of white points to $K(i)$ is the number of pink points of circle C_i , for a given image.

6.5 Results

This section presents the results of application of the seeded heuristic search, using the HC, SA and GA methods. Nine methods were used to find the “best” search approach which is then used to process 317 images and the resulting 648 features extraction, following segmentation, to detect potential blast cells before proceeding with the classification step, discussed in Chapter 7. Given the stochastic nature of the algorithms, each method was executed 10 times for 10000 iterations to observe their average performance. The seeding step, that is, the detection of the coordinates of potential cells, to serve as starting points, for the search was performed as discussed in chapters 3, 4 and 5.

6.5.1 Comparison between Hill Climbing, Simulated Annealing and Genetic Algorithm Seeded

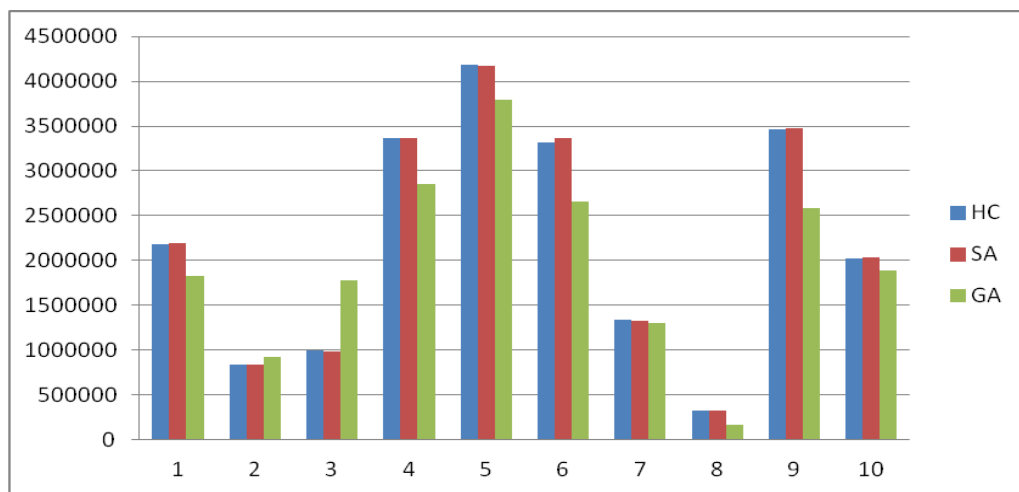
This section focuses on the comparison between HC, SA and GA for seeded search based on the analysis of ten images, randomly selected from the dataset. Unlike the random heuristic search approach discussed in chapter 3 here the starting points coordinates have been defined. Table 6.1 and Figure 6.3 display the fitness values reached by each method revealing that SA is able to outperform the other methods and reach higher fitness values. Overall, results on Table 6.1 are quite satisfactory.

Table 6.1(a): Comparison between Seeded HC, SA and GA (Image 1 – Image 9)

| Image | HC | SA | GA | Max. |
|-------|----------------|----------------|---------------|----------------|
| 1 | 2182311 | 2190947 | 1822904 | 2190947 |
| 2 | 834347 | 833635 | 920395 | 920395 |
| 3 | 993554 | 982490 | 1775628 | 993554 |
| 4 | 3363227 | 3363630 | 2857114 | 3363630 |
| 5 | 4178647 | 4168576 | 3792456 | 4178647 |
| 6 | 3311760 | 3364078 | 2655160 | 3364078 |
| 7 | 1332994 | 1322776 | 1299860 | 1332994 |
| 8 | 319436 | 318624 | 160262 | 319436 |
| 9 | 3458144 | 3480803 | 2585096 | 3480803 |

Table 6.1(b): Comparison between Seeded HC, SA and GA (Image 10)

| Image | HC | SA | GA | Max. |
|---------|----------------|-----------------------|----------------|----------------|
| 10 | 2016221 | 2028946 | 1891587 | 2028946 |
| Average | <i>2199064</i> | <i>2205451</i> | <i>1976046</i> | |

**Figure 6.3:** Comparison between seeded HC, SA and GA

Figures 6.4 correspond to the highest fitness reached by the seeded heuristic search, showing that the blast cells were targeted successfully. In image 2, the GA reached the highest fitness value. Visual inspection reveals that two circles overlap. This occurred during the crossover process. For images 1, 4, 6, 9 (62) and 10 the results show that SA is more efficient than the HC and GA. In some cases the HC search was able to reach a higher fitness, reflecting the stochastic nature of the algorithms. Figure 6.5 show all the results for HC. Figure 6.6 show all the results for SA and Figure 6.7 show all the results for GA.

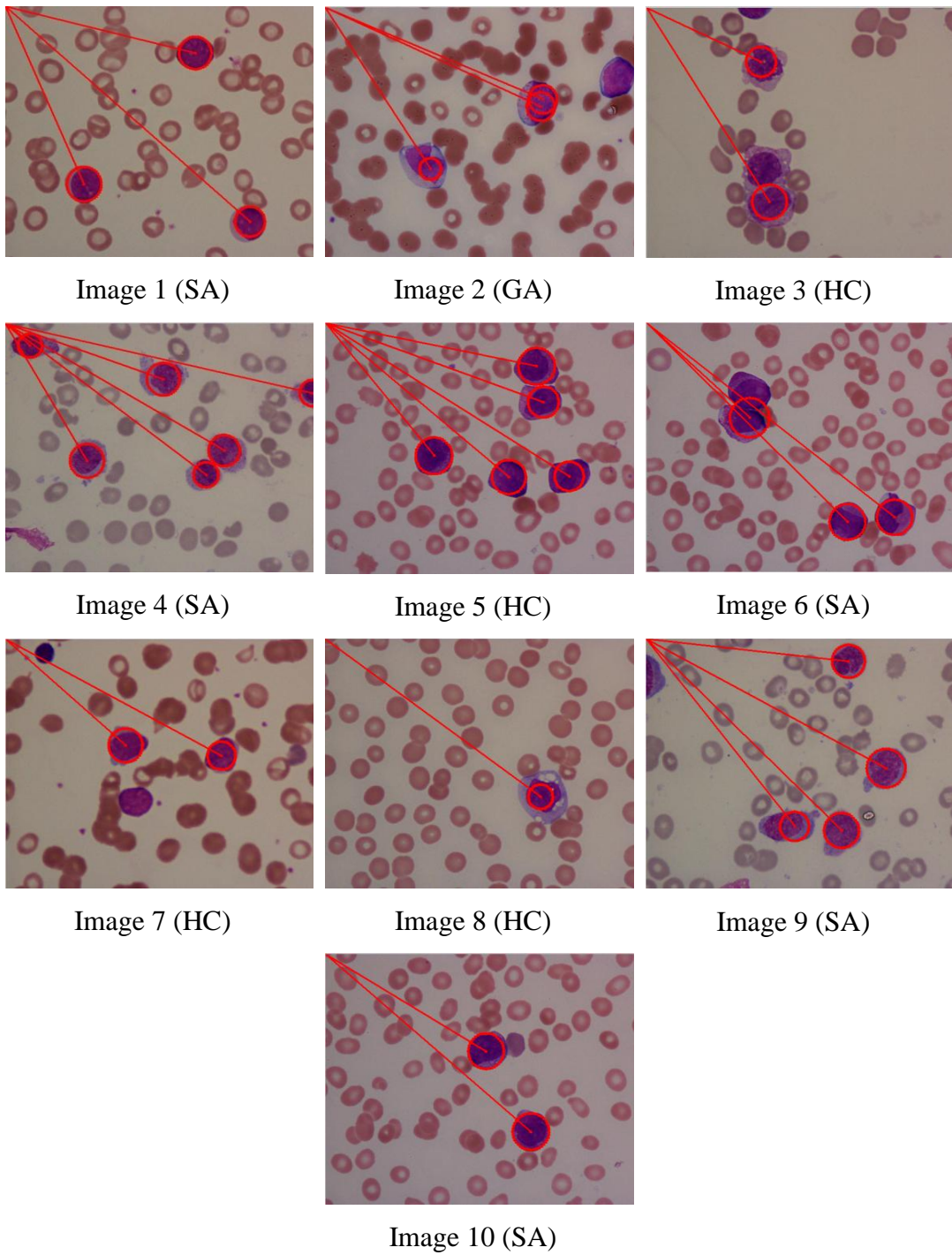


Figure 6.4: Images corresponding to the highest fitness value

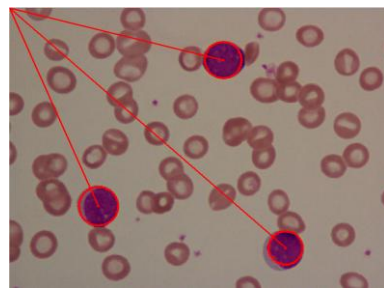


Image 1

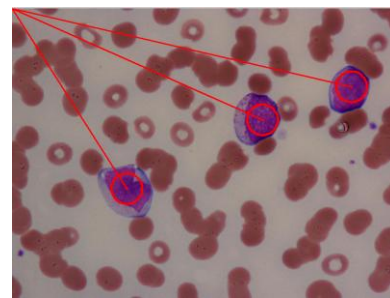


Image 2

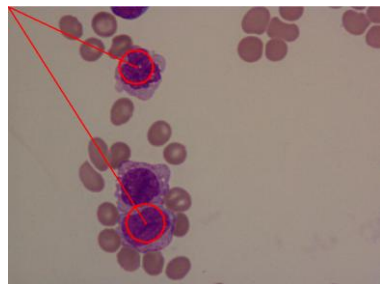


Image 3

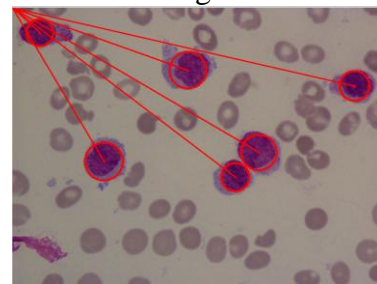


Image 4

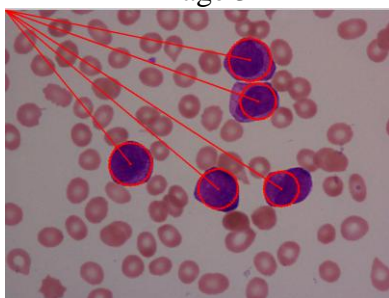


Image 5

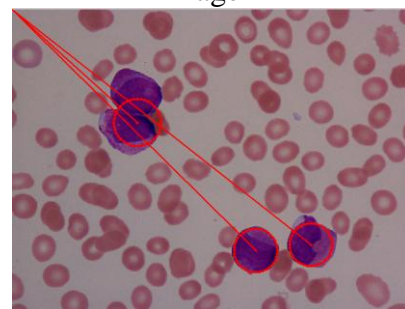


Image 6

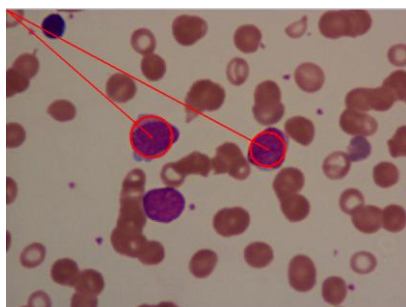


Image 7

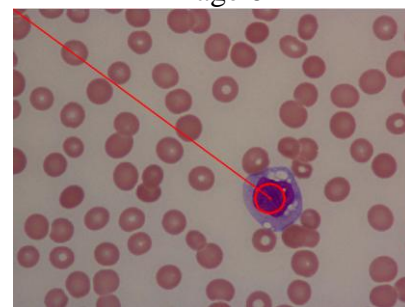


Image 8

Figure 6.5(a): Hill Climbing Seeded Heuristic Search (Image 1 – Image 8)

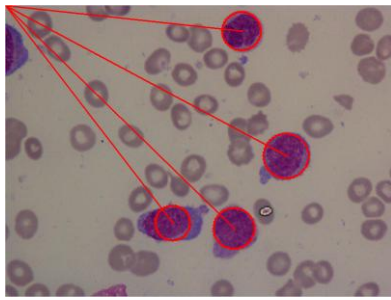


Image 9

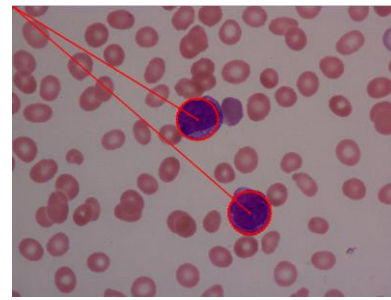


Image 10

Figure 6.5(a): Hill Climbing Seeded Heuristic Search (Image 9 – Image 10)

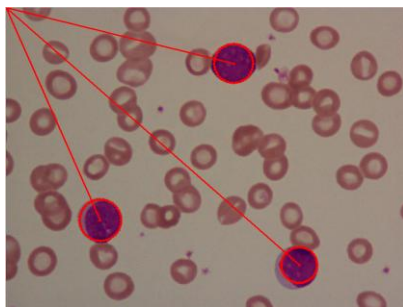


Image 1

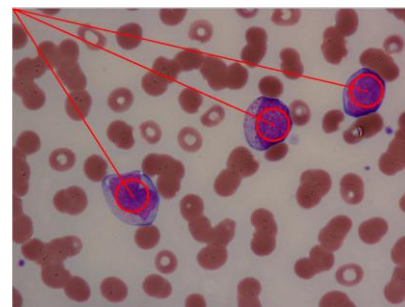


Image 2

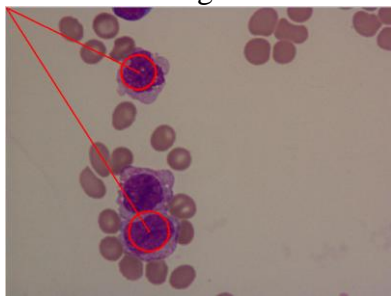


Image 3

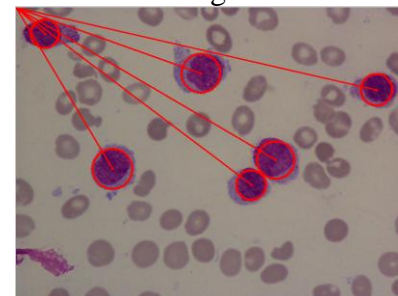


Image 4

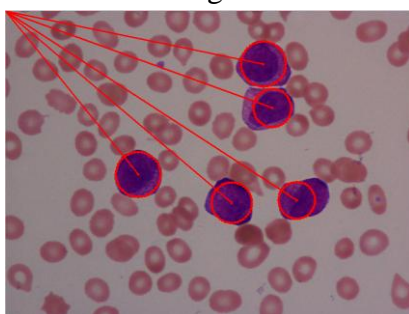


Image 5

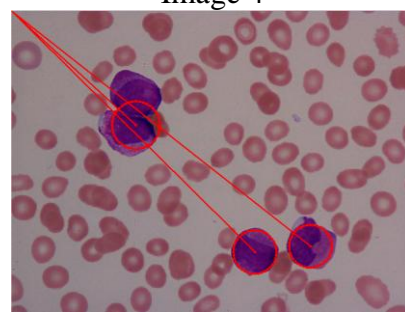


Image 6

Figure 6.6(a): Simulated Annealing Seeded Heuristic Search (Image 1 – Image 8)

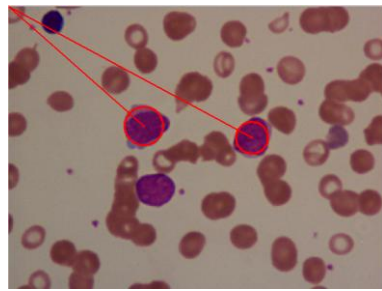


Image 7

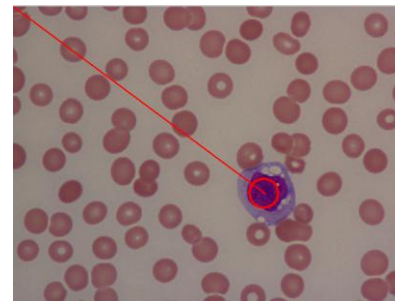


Image 8

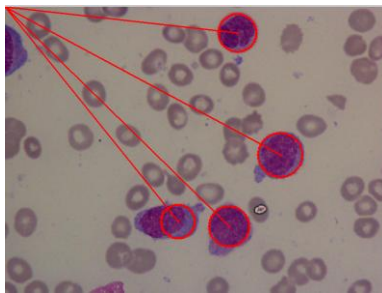


Image 9

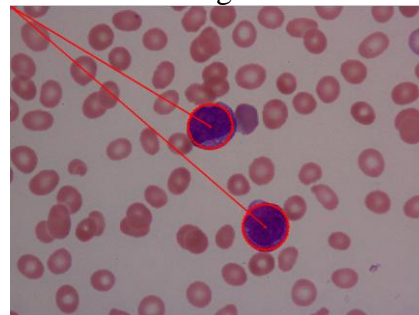


Image 10

Figure 6.6(b): Simulated Annealing Seeded Heuristic Search (Image 7 – Image 10)

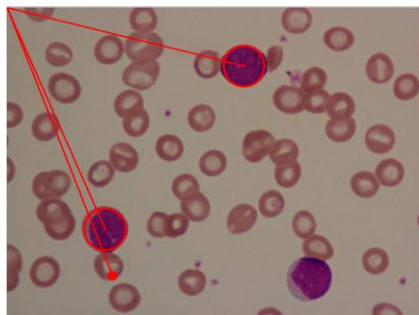


Image 1

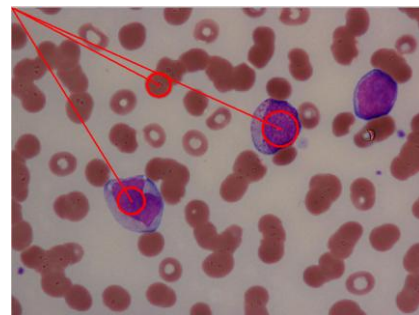


Image 2

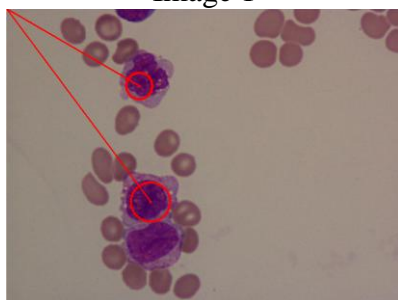


Image 3

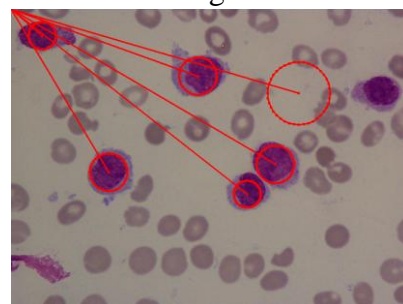


Image 4

Figure 6.7(a): Genetic Algorithm Seeded Heuristic Search (Image 1 – Image 4)

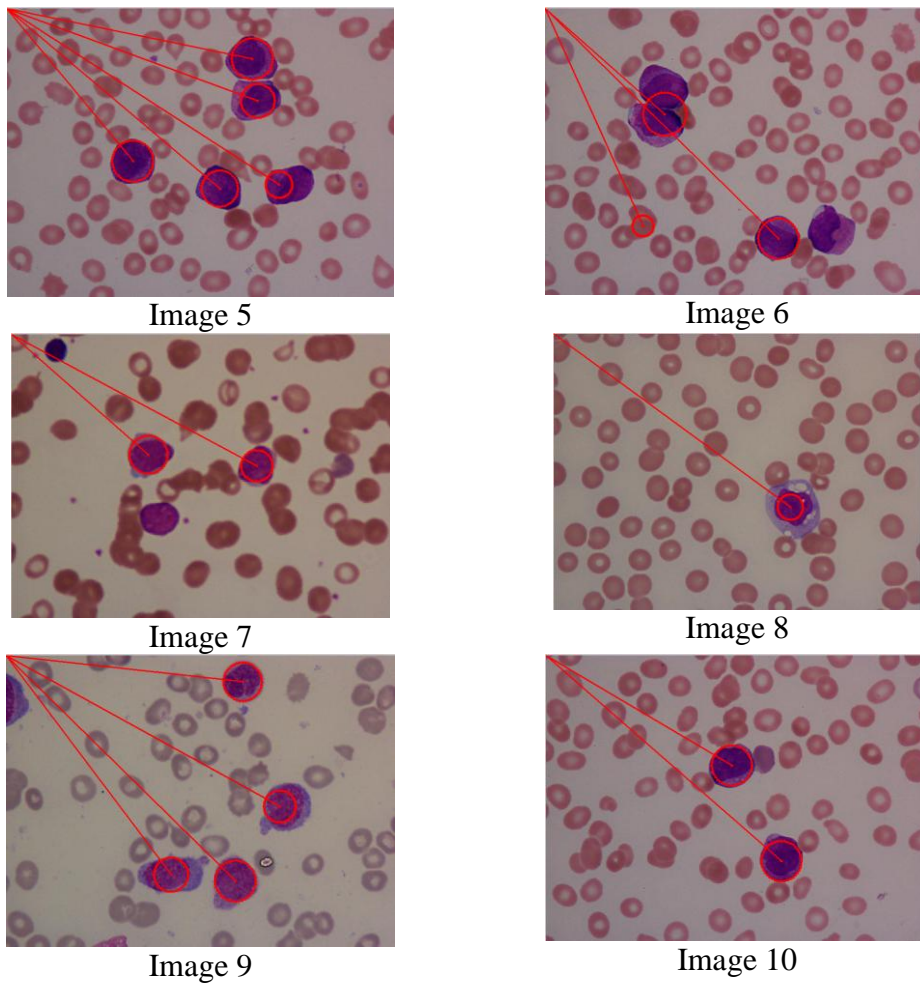


Figure 6.7(b): Genetic Algorithms Seeded Heuristic Search (Image 5 – Image 10)

Table 6.2 displays a comparison between the Random and Seeded search. The result reveals that the seeded search shows superior performance. Given that the starting points, expected to correspond to cells, have been defined, it is only natural for the seeded search to capture blast cells more efficiently. The highest fitness values are highlighted in bold.

Table 6.2: Comparison between Seeded and Random HC, SA and GA methods

| Image | HC Random | HC Seeded | SA Random | SA Seeded | GA Random | GA Seeded |
|---------|--------------|--------------|-------------------|----------------|--------------|----------------|
| 1 | 229796.20 | 2182311 | 94805.21 | 2190947 | 1393758.30 | 1822904 |
| 2 | 227858.70 | 834347 | 829226.10 | 833635 | 812467.90 | 920395 |
| 3 | 141609.90 | 993554 | 1334923.60 | 982490 | 1720899 | 1775628 |
| 4 | 951630.35 | 3363227 | 5260004.69 | 3363630 | 2184407.20 | 2857114 |
| 5 | 784762.14 | 4178647 | 75808.42 | 4168576 | 2508479.50 | 3792456 |
| 6 | 809407.93 | 3311760 | 4378871.82 | 3364078 | 2475291.10 | 2655160 |
| 7 | 465109.89 | 1332994 | 3158457.79 | 1322776 | 1102489.52 | 1299860 |
| 8 | 71543.46 | 319436 | 41776.95 | 318624 | 82162.15 | 160262 |
| 9 | 44192.84 | 3458144 | 189514.99 | 3480803 | 2371489.50 | 2585096 |
| 10 | 515361.48 | 2016221 | 905159.28 | 2028946 | 1550242.70 | 1891587 |
| Average | 424127 | 2199064 | 1626855 | 2205451 | 1620169 | 1976046 |

6.5.2 Processing using colour images

Following the analysis of images processed by the Otsu method, colour images were facilitated for the same purpose. Again the discussed HC, SA and GA approaches were implemented. Table 6.3 shows that the GA outperformed the HC and SA methods. First, the performance of random heuristic search described in chapter 3 was tested. For the testing, here, only one image was facilitated (image 1).

Table 6.3: Comparison between HC, SA and GA for Colour Images in random

| Image | HC | SA | GA | Max. |
|-------|-------|---------|----------------|----------------|
| 1 | 11.01 | 9550.86 | 1089960 | 1089960 |

Figure 6.8 shows that the GA has been more efficient in detecting blast cells than the HC and SA. HC did not capture any of the blast cells, while SA was only able to capture one blast cell. As a conclusion, the GA appears to be the most suitable method in the case of random search for both colour and black and white images.

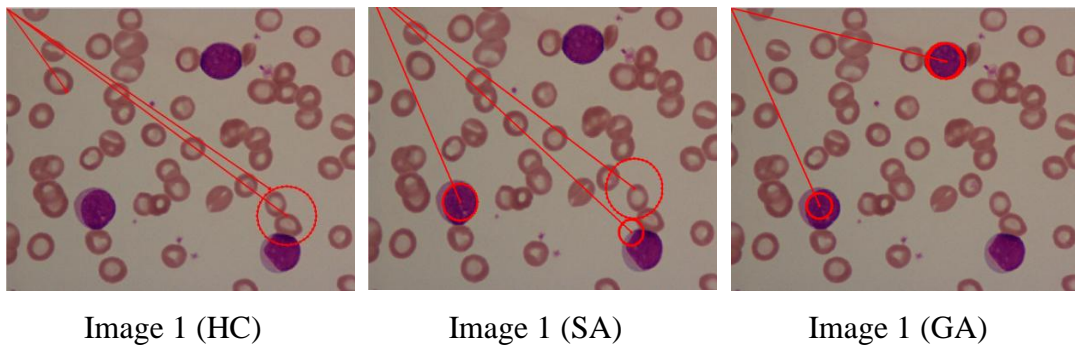
**Figure 6.8:** Random HC, SA and GA for colour image

Table 6.4 shows the results of applying the same analysis to colour images, this time using the seeded approach. The SA performed best in targeting blast cells. Hence, SA seems to be the most suitable method for detection of blast cells in both colour and black and white images, when the seeded approach is implemented.

Table 6.4: Comparison between HC, SA and GA for Colour Images in Seeded

| Image | HC | SA | GA | Max. |
|-------|----------|------------------|--------|------------------|
| 1 | 963731.7 | 1351918.7 | 881698 | 1351918.7 |

Figure 6.9 shows the SA has been able to capture blast cells more efficiently than the HC and GA. Although, the HC search has also detected the blast cells the SA appears more accurate. The GA performed worse than the three approaches.

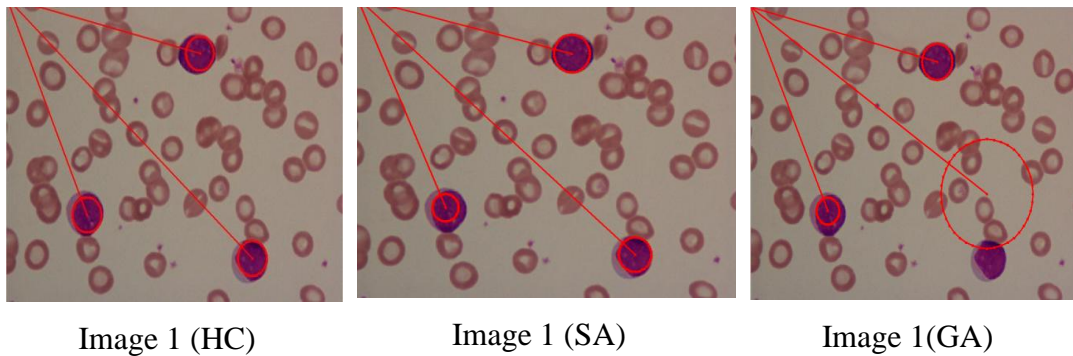


Figure 6.9: Seeded HC, SA and GA for colour image

6.5.3 Comparison between Hill Climbing, Simulated Annealing, Genetic Algorithm for seeded with twenty images

The comparison between random and seeded HC, SA and GA search approaches, for detection of blast cells in colour and black and white images, showed that the seeded method performed better. The comparison was not limited to the fitness values obtained but also included visual inspection of the location of blast cells in the images and the quality of detection. By the observation that higher fitness does not absolutely guarantee better targeting of blast cells, especially in the case of random heuristic search. Here further experimentation is presented, based on the use of 20 images and the application of the seeded approach. The number of images is increased to ensure the consistency of the final results and that SA is indeed the “best” method for blast cells detection.

The results displayed on Table 6.5 and Figure 6.10 show that SA performed best in targeting blast cells, followed by the HC method. This is

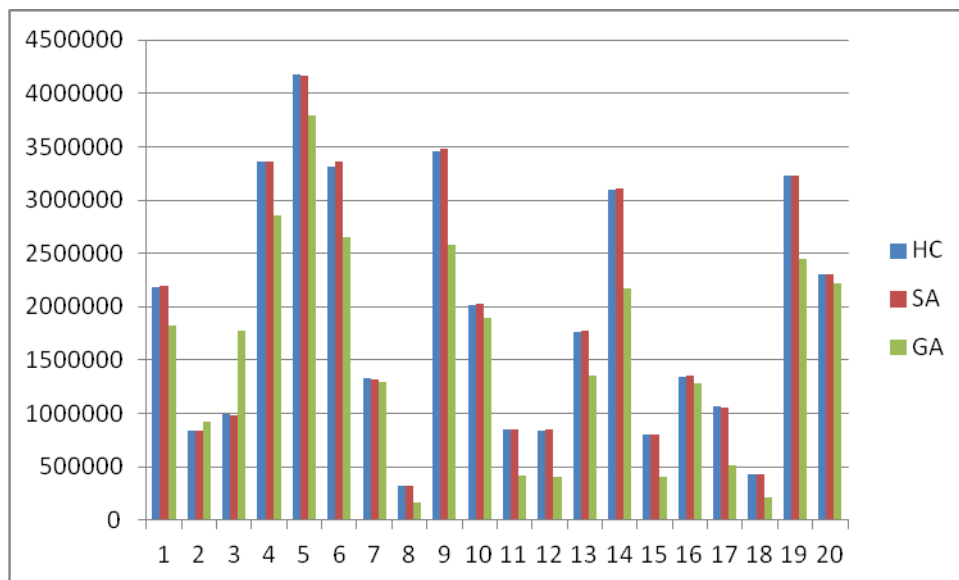
reflected on the highest fitness values and good capturing of potential blast cells. For example, in the Table 6.5, image 10, the SA has reached the highest fitness value. Visual inspection of image 10, in Figure 6.9 reveals that the circle has fully captured the blasts.

Table 6.5(a): Fitness values for HC, SA and GA (Image 1 - Images 17)

| Images | HC | SA | GA | Max. |
|---------------|-------------------|-------------------|------------------|-------------------|
| 1 | 2182310.7 | 2190947.17 | 1822904.22 | 2190947.17 |
| 2 | 834346.57 | 833634.98 | 920394.62 | 920394.62 |
| 3 | 993554.37 | 982490.29 | 1775627.76 | 993554.37 |
| 4 | 3363227.26 | 3363630.14 | 2857113.60 | 3363630.14 |
| 5 | 4178646.75 | 4168576.48 | 3792455.80 | 4178647.75 |
| 6 | 3311759.97 | 3364077.86 | 2655159.88 | 3364078.86 |
| 7 | 1332993.91 | 1322775.84 | 1299860.28 | 1332994.91 |
| 8 | 319435.77 | 318624.02 | 160261.66 | 3194360.77 |
| 9 | 3458143.61 | 3480803.11 | 2585096.40 | 3480803.11 |
| 10 | 2016220.54 | 2028946.07 | 1891586.68 | 2028946.07 |
| 11 | 849665.07 | 851916.46 | 420161.02 | 851916.46 |
| 12 | 841061.70 | 850810.39 | 409722.38 | 850810.39 |
| 13 | 1757974.03 | 1777696.46 | 1360590.92 | 1777696.46 |
| 14 | 3100250.51 | 3110146.46 | 2175044.20 | 3110149.46 |
| 15 | 799335.62 | 800866.26 | 408710.64 | 800866.26 |
| 16 | 1346332.59 | 1352445.36 | 1281975.44 | 1352445.36 |
| 17 | 1063237.92 | 1060222.42 | 513705.44 | 1063237.92 |

Table 6.5(a): Fitness values for HC, SA and GA (Image 18 - Images 20)

| Image | HC | SA | GA | Max. |
|---------|-------------------|-------------------|------------|-------------------|
| 18 | 430769.28 | 430494.21 | 218171.64 | 430769.28 |
| 19 | 3232761.84 | 3232675.90 | 2450877.40 | 3232761.84 |
| 20 | 2305509.44 | 2305714.78 | 2221493.70 | 2305714.78 |
| Average | 1885876.873 | 1891374.73 | 1561045.70 | 1891374.73 |

**Figure 6.10:** Fitness values for HC, SA and GA (20 images)

Figures 6.11 correspond to the highest fitness values on Tables 6.5(a) to (b). Based on the images (1, 4, 6, 9, 10, 11, 12, 13, 14, 15, 16 and 20) SA appears to have targeted blast cells more efficiently than the other methods. SA performs better than HC and GA because it can escape local optimum. As Table 6.5 shows, the highest fitness values are mostly reached by the SA search. Images 2, 3, 5, 7 and 8, are displayed in Figure 6.6. Figure 6.12 shows the results produced by the SA for images 11 and 12.

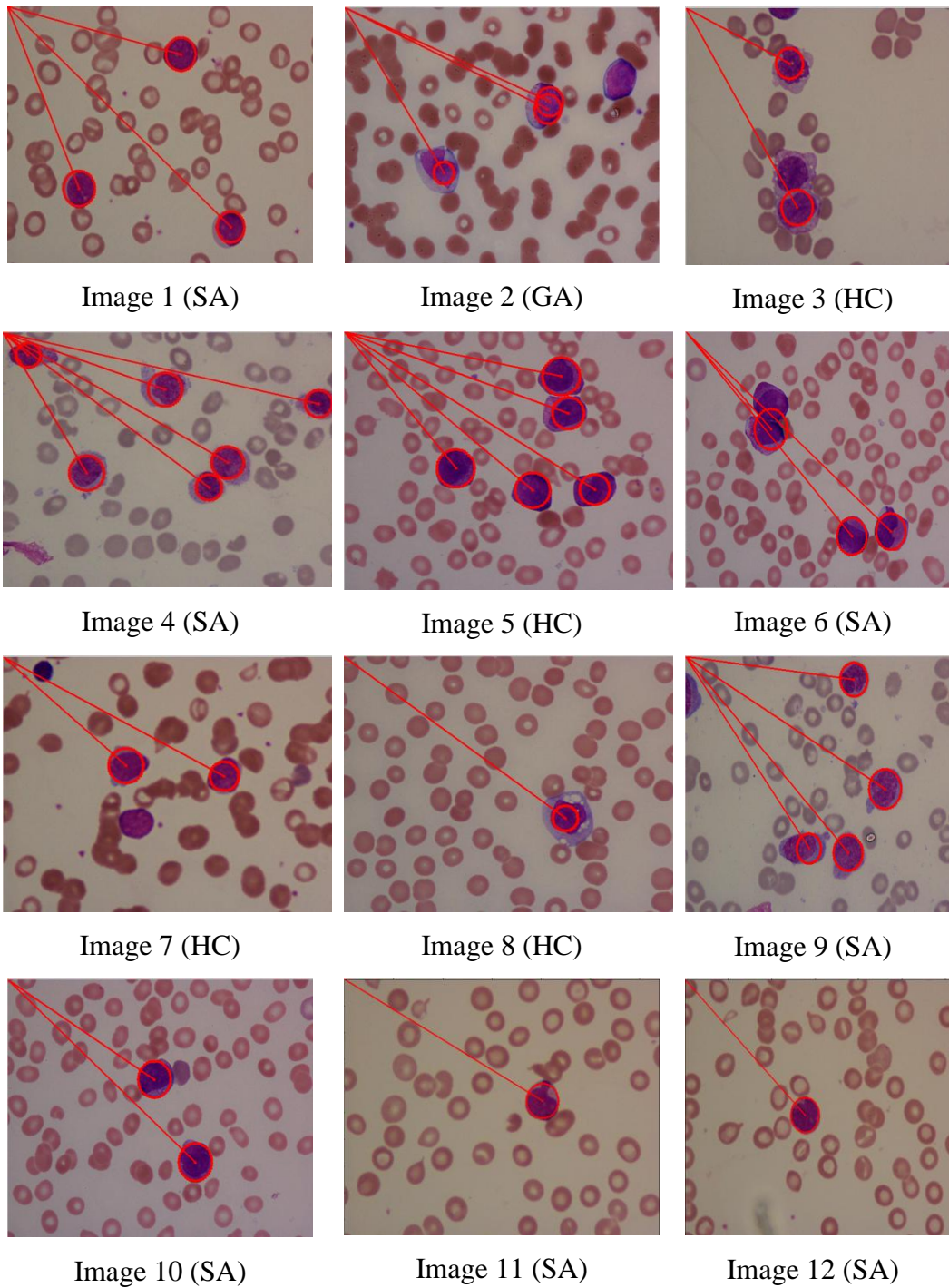


Figure 6.11(a): Seeded HC, SA and GA (Images 1 – Image 12)

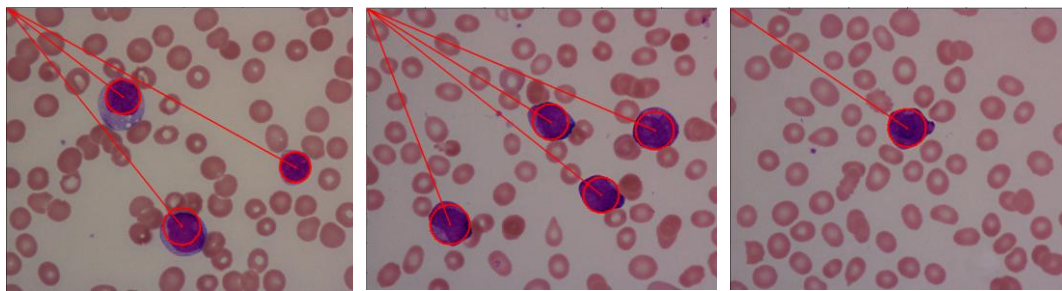


Image 13 (SA)

Image 14 (SA)

Image 15 (SA)

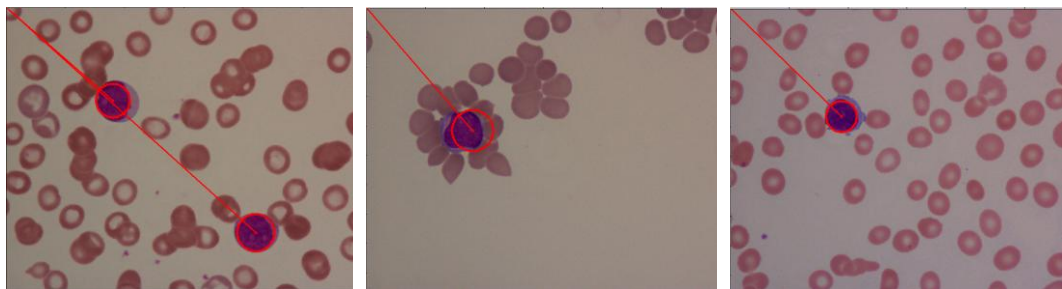


Image 16 (SA)

Image 17 (HC)

Image 18 (HC)

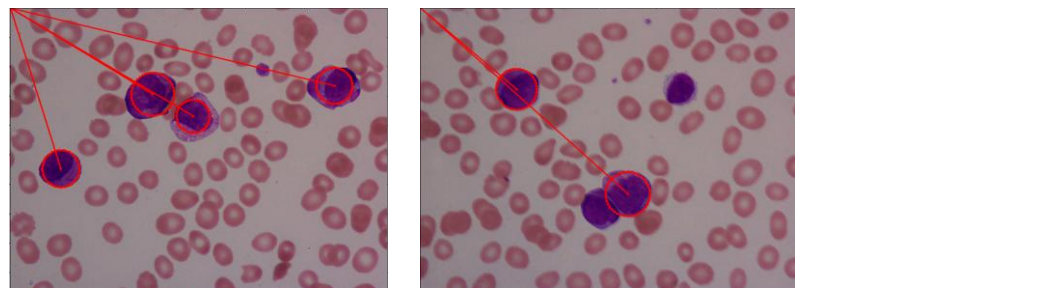


Image 19 (HC)

Image 20 (SA)

Figure 6.11(b): Seeded HC, SA and GA (Image 13 – Image 20)

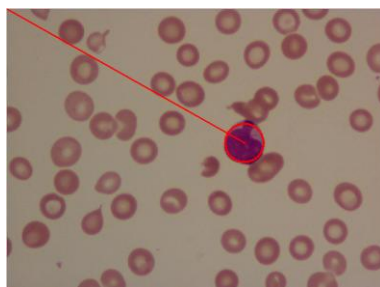


Image 11

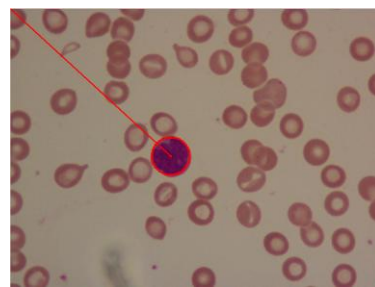


Image 12

Figure 6.12(a): Seeded SA (Image 11– Image 12)

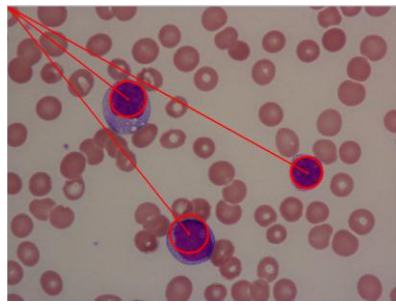


Image 13

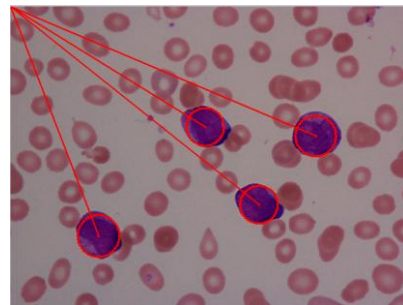


Image 14

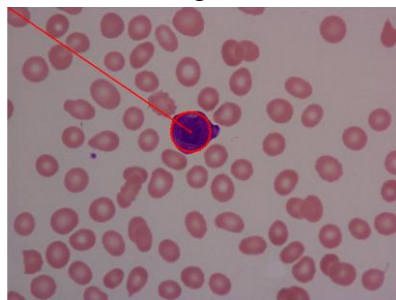


Image 15

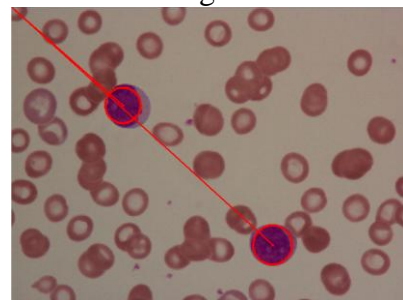


Image 16

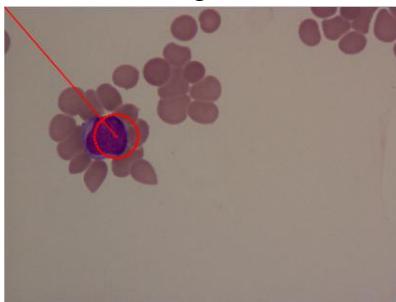


Image 17

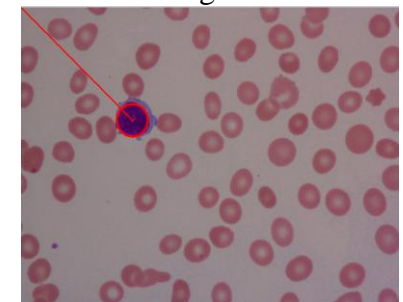


Image 18

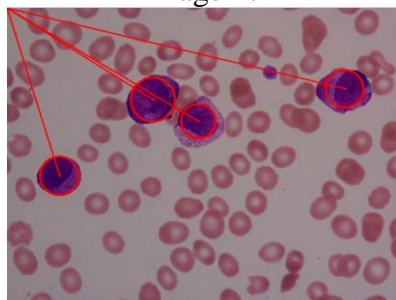


Image 19

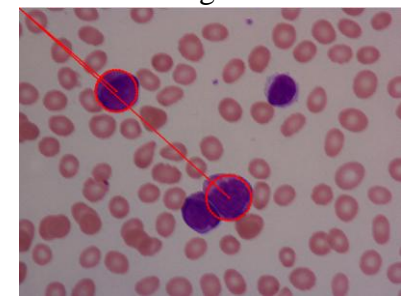


Image 20

Figure 6.12(a): Seeded SA (Image 13– Image 20)

6.5.4 Circles Overlap Similarity Metric

The previous section concluded that the seeded heuristic search approach shows superior performance. As already mentioned the comparison was not limited to observing the fitness values reached by each method but also included visual inspection of the images to judge the quality of the targeting of blast cells. It is observed that high fitness values may not guarantee good detection of blast cells, especially in the case of random search. Further experiments were conducted, using 20 images, to examine the consistency of the produced results and it was concluded that SA is the best choice of method for the detection of blast cells. Table 6.6 shows that the average and standard deviation for SA are better than in the HC case. The utilised equation is the one presented in section 3.4.5.

Table.6.6(a): Result circles overlap similarity metric for SA and HC (Image 1 – Image 9)

| | SA | | HC | |
|-------|--------------|--------------------|--------------|--------------------|
| Image | Average | Standard Deviation | Average | Standard Deviation |
| 1 | 0.984 | 0.019 | 0.980 | 0.020 |
| 2 | 0.881 | 0.122 | 0.889 | 0.092 |
| 3 | 0.895 | 0.122 | 0.896 | 0.145 |
| 4 | 0.972 | 0.029 | 0.970 | 0.030 |
| 5 | 0.954 | 0.084 | 0.968 | 0.028 |
| 6 | 0.971 | 0.031 | 0.971 | 0.033 |
| 7 | 0.978 | 0.017 | 0.983 | 0.014 |
| 8 | 0.968 | 0.021 | 0.966 | 0.015 |
| 9 | 0.963 | 0.046 | 0.954 | 0.060 |

Table.6.6(b): Result ratio for SA and HC (Image 10 – Image 20)

| Image | SA | | HC | |
|----------------|--------------|--------------------|--------------|--------------------|
| | Average | Standard Deviation | Average | Standard Deviation |
| 10 | 0.963 | 0.043 | 0.959 | 0.048 |
| 11 | 0.966 | 0.021 | 0.963 | 0.025 |
| 12 | 0.986 | 0.014 | 0.979 | 0.013 |
| 13 | 0.970 | 0.019 | 0.968 | 0.032 |
| 14 | 0.971 | 0.034 | 0.969 | 0.039 |
| 15 | 0.988 | 0.013 | 0.991 | 0.010 |
| 16 | 0.948 | 0.059 | 0.955 | 0.060 |
| 17 | 0.975 | 0.016 | 0.971 | 0.019 |
| 18 | 0.984 | 0.017 | 0.984 | 0.017 |
| 19 | 0.966 | 0.027 | 0.962 | 0.031 |
| 20 | 0.984 | 0.018 | 0.982 | 0.027 |
| Average | 0.964 | 0.772 | 0.963 | 0.758 |

6.5.5 Simulated Annealing Real Images

The previous section presented a number of experiments, concluding that the seeded SA is the best choice of method for the detection of blast cells. Hence, this method is applied to all 317 images (in 2 images CA Filtering was unsuccessful and in 3 colour image clustering failed). Tables 6.7 and 6.8 summarise the result of targeting blast cells before and after application of the heuristic search

approach. The percentages are calculated manually through visual inspection. Appendix E shows the results of M1, M2, M3 and M5 subtypes.

Table 6.7 highlights the percentage of circles targeting red blood cells (pink) and blast cells (purple). The In between field indicates cases of circles where both blast and red blood cells are captured and cases of overlapping circles.

Table.6.7: Result before performing application seeded SA

| Sub-types | Images | Before Heuristic | | |
|-----------|--------|------------------|-------|------------------------|
| | | Purple | Pink | In between blast cells |
| M1 | 46 | 91.95% | 1.15% | 6.90% |
| M2 | 129 | 85.14.% | 0.36% | 14.50% |
| M3 | 92 | 96.12% | 0.65% | 3.23% |
| M5 | 53 | 88% | 2.29% | 9.71% |

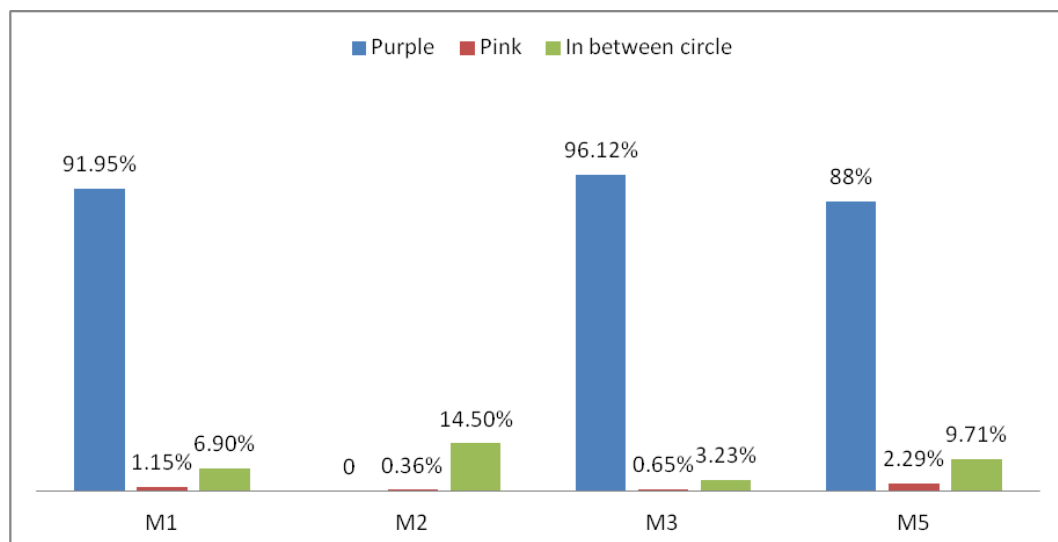


Figure.6.13: Result before performing seeded SA

Table 6.8 displays the result after performing Seeded SA which show a clear improvement.

Table.6.8: Results after performing seeded SA

| Sub-types | Images | After Heuristic | | | |
|-----------|--------|-----------------|-------|------------|----------------------------------|
| | | Purple | Pink | In between | Wrong target from Purple to Pink |
| M1 | 46 | 95.40% | 1.15% | 3.45% | 0% |
| M2 | 129 | 92.21% | 1.08% | 4.55% | 2.16% |
| M3 | 92 | 100% | 0 | 0% | 0% |
| M5 | 53 | 91.43% | 1.17% | 5.57% | 1.83% |

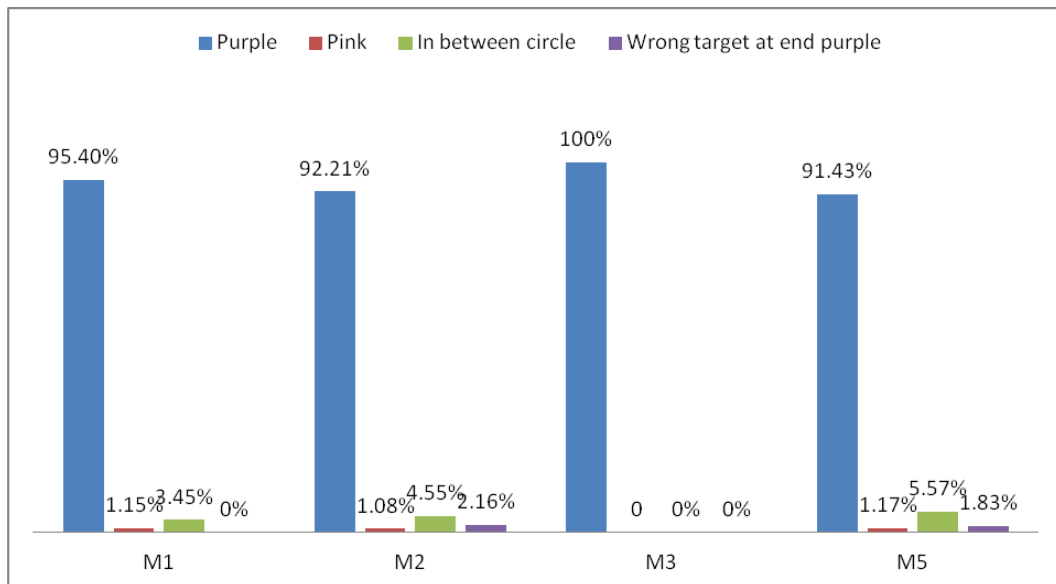


Figure .6.14: Result after performing seeded SA

Figure 6.15 shows an increase of the percentage of detected cells following the application of Heuristic Search, where the blast cells are captured more efficiently.

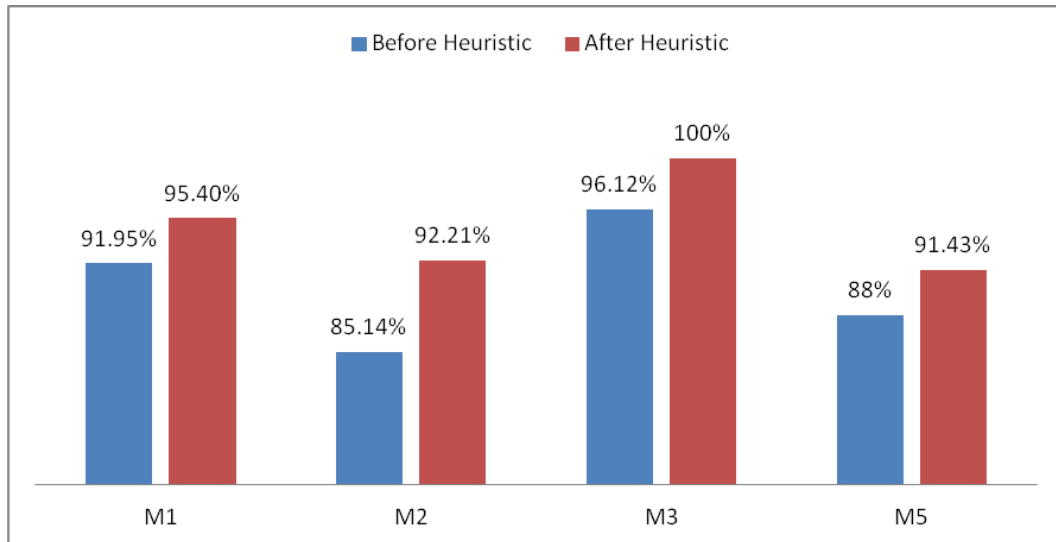


Figure .6.15: Result from Seeded Heuristic Search before and after

Figure 6.16 shows image 4, an example of M1 subtype, and the difference between blast cell detection before (the circle targets between the blast cells) and after performing the seeded SA search, where the blast cell is detected. Correct detection is necessary for classification of potential blast cells, which is the subject of the next chapter.

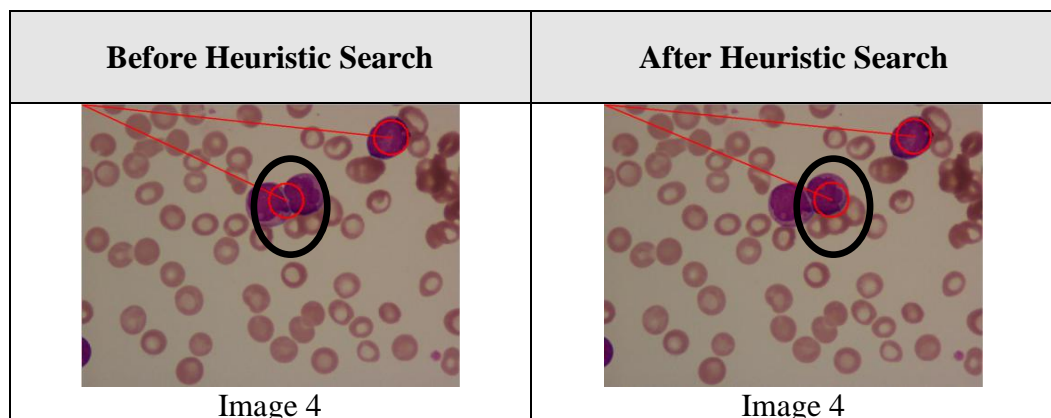


Figure 6.16: Targeting in between blast cells before and targeting full blast cells after performing heuristic search

Figure 6.17 displays the result of targeting of blast cells, following the application of heuristic search. Image 5 shows an example of M2 subtype, where the blast and red blood cell overlap. Application of the search has resulted in the circle targeting a red blood cell (pink). Overall, the application of heuristic search tends to improve results.

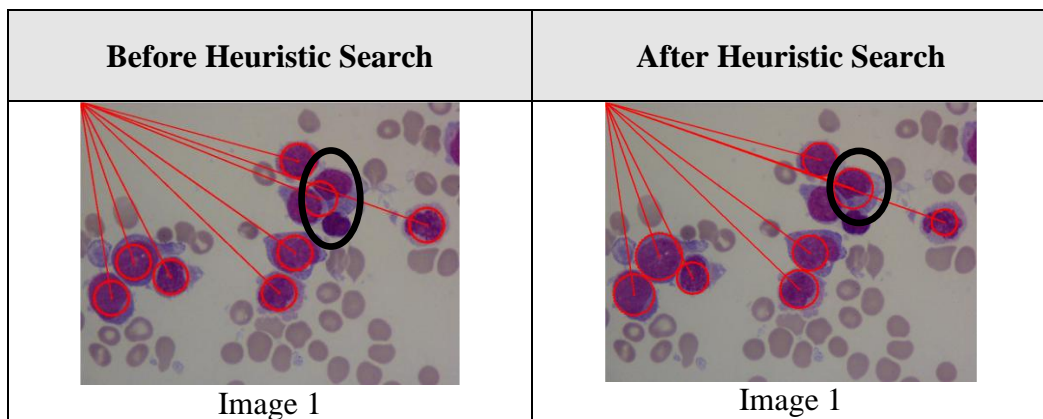


Figure 6.17: Example of M2 subtype before and after Heuristic Search

Figure 6.18 displays one example where before application of heuristic search the circle is targeting mostly purple pixels, but after heuristic search it is targeting pink. This is because the blast cell overlaps with red blood cells.

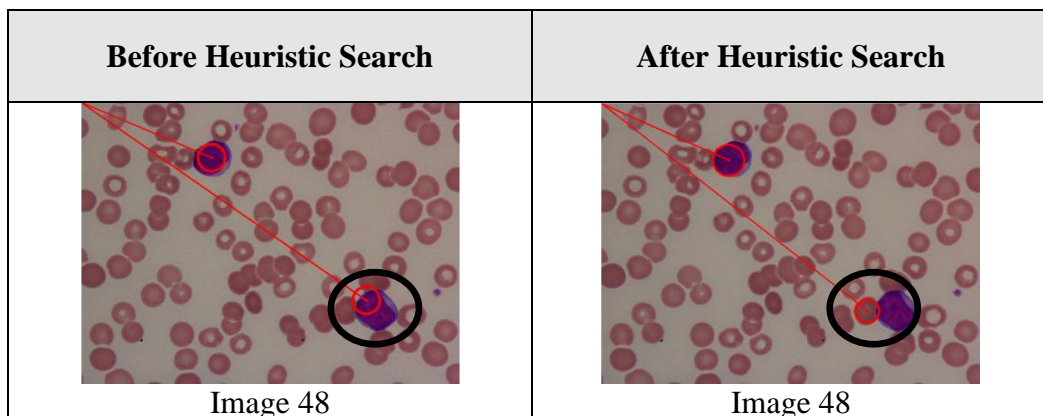


Figure 6.18: Before heuristic search targeting blast cells but after heuristic search targeting red blood cells

The result is also satisfactory for the image in Figure 6.19, an example of M3 AML subtype, before and after the application of the seeded SA method.

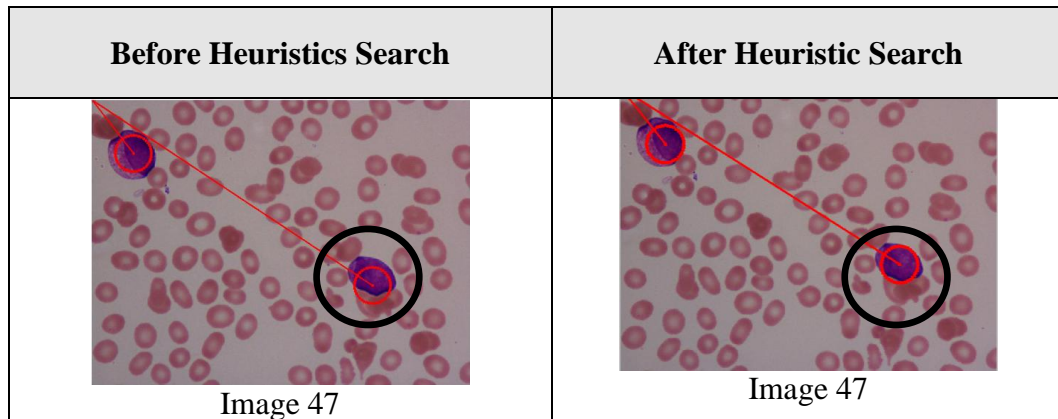


Figure 6.19: Before heuristic search the circle is targeting blast cell and red blood cells, but after heuristic search it is only targeting the blast cell

In Figures 6.20 there are more images showing the result of applying seeded SA for detection of blast cells. In image 5, showing an example of M5 subtype, after performing the search the circle is targeting some red blood cells. This is because the fitness calculation is based on black and white images. Since, the blast cell is overlapping with red blood cells, in the black and white images it appears as one large blast cell.

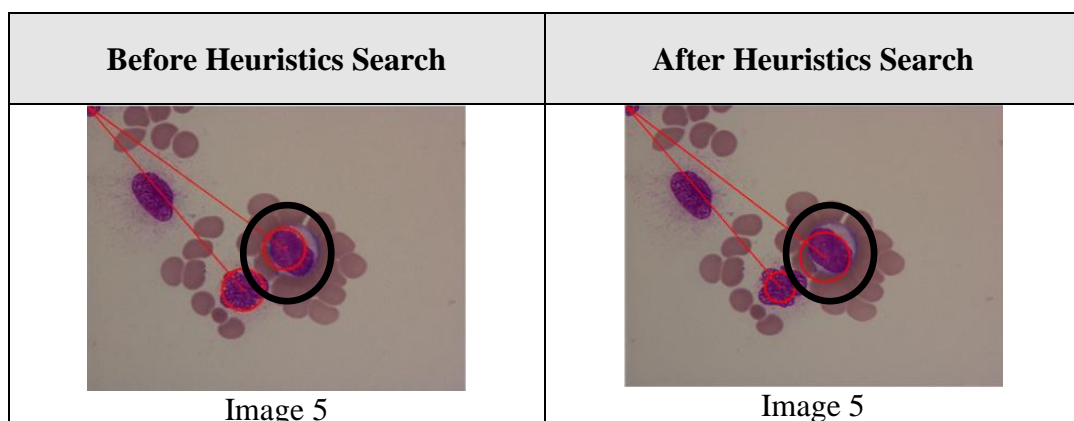


Figure 6.20: Before heuristic search the circle is targeting blast cells, but after heuristic search it is targeting blast and red blood cells

In Figure 6.21, image 10 shows an example of M5 subtype. Before heuristic search the circle is targeting blast cells but after applying heuristic search it is targeting both blast and red blood cells.

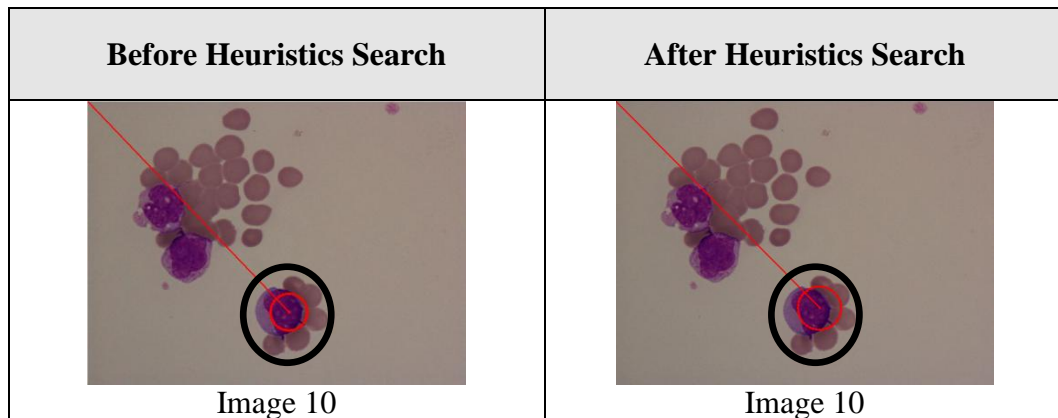


Figure 6.21: Example of M5 subtype before and after Heuristic Search

6.6 Computation Complexity

The main objective of computational complexity is to derive an asymptotic limit on the growth rate of an algorithm in terms of the input size (McConnell, 2008). This gives an indication of how well an algorithm scales on different size problems and can be used as a measure of runtime.

In this section a brief discussion of the computational complexity of the main components of the proposed methodology is presented, specifically an analysis of the performance of Otsu, the Seeding Cellular Automata and the Heuristic Search methods is discussed. The calculations apply to a single image of size X_{max} by Y_{max} .

6.6.1 Otsu

From the literature (Chen et. al, 2009) it can be seen that Otsu's method is $O(NL^2)$ where N is the number of pixels in the image and L is the number of greyscales.

6.6.2 Cellular Automata

Without loss of generality, it can be assumed that the number of iterations the CA will run for is when there is a single dead pixel and it is located in one of the corners of the image. In a rectangle, the furthest two points away from each other is when the two points are in opposite corners. Any more than one dead pixel will result in one of the dead pixels being closer to the corner and thus one less iteration. The City-Block distance between the two corners, which corresponds to the number of CA iterations, will be $K = X_{max} + Y_{max}$ where X_{max} is the number of pixels in the x-axis and Y_{max} is the number of pixels in the y-axis. Without using any special data structures, the time complexity of running the CA one iteration is $O(X_{max}Y_{max})$, i.e. processing each pixel each iteration. Thus the final time complexity for the CA is:

$$O(X_{max}Y_{max}K) = O(NK)$$

6.6.3 Heuristic Search

Deriving exact measures of the computational complexity for Heuristic Search methods is very complicated and is dependent on the many parameters these methods need in order to be implemented. It is well know that for complex problems 99% of the computation of the Heuristic Search methods is in the fitness function. Thus these calculations for this part of the complexity of the methods will be concentrated in the Circle fitness function which is assumed to be called for a fixed number of times, i .

The fitness function depends on the number of circles being fitted and the number of pixels in each circle. An absolute upper bound on this measure is N , i.e. there can never be more circles than the area of the image. The computational complexity of the heuristic search methods is therefore $O(iN)$.

6.6.4 Overall Complexity

The overall complexity is as follows:

$$\begin{aligned} &O(\text{Otsu}) + O(\text{CA}) + O(\text{Heuristic Search}) \\ &O(NL^2) + O(NK) + O(IN) = O(N(L^2+K+i)) \end{aligned}$$

The computational complexity of the classification party of the thesis has been omitted since there are a large number of classifiers being used which all have differing complexities.

6.7 Summary

This chapter presented a work process that is a combination of methods described in previous chapters. It described the application of seeded heuristic search for the detection of blast cells. In the seeded process, discussed in chapter 5, the starting coordinates for the search are defined prior to application of the method.

The goal here is the optimisation of heuristic search for the detection of blast cells in images of blood and bone marrow smear. The comparison between three heuristic methods showed that a seeded SA is the best choice for the purposes of optimising the detection of blast cells. A number of real images were facilitated to compare the HC, SA and GA methods starting from random points

to the seeded approach. The comparison between Random Heuristic Search and Seeded Heuristic Search that Seeded Heuristic Search targeting better than Random Heuristic Search.

As a conclusion, a seeded SA is the “best” optimisation method for detection of blast cells. It was applied to 317 real images with 648 feature extractions of potential blast cells, of M1, M2, M3 and M5 AML subtypes. The results are very satisfactory with 95.40% of blast cells targeted in M1 images. In the subset of M2 images 92.21% of blast cells were correctly targeted. Interestingly, 100% of all blast cells were targeted in M3 images. The targeting was 91.43% efficient in M5 images.

Chapter 7

MULTILAYER PERCEPTRON FOR THE CLASSIFICATION OF LEUKAEMIA CELLS

7.1 Introduction

Classification is the last step in the experimental methodology presented in this thesis. Nowadays, classification is widely used in pattern recognition, which includes a number of information processing problems from speech recognition and the classification of handwritten characters to fault detection in technology and medical diagnosis. In this thesis the WEKA application was used for classification of leukaemia cells. WEKA is a data analysis software tool which implements a set of machine learning algorithms for data mining tasks (Hall et al., 2009). The choice of the “best” classifier required the performance of a number of experiments from the Artificial Intelligence Networks (ANN) methods.

The classification was aimed at distinguishing between M1, M2, M3 and M5 AML cases. Firstly, the “best” classifier in WEKA needed to be identified. The findings show that Multilayer Perceptron with 10-fold cross validation is best suited for our analysis. Additionally, the analysis investigated the choice of a heuristic search method, concluding that SA is the most appropriate search approach for detection of blast cells. Hierarchy classification was performed, to first distinguish between M3 and all other AML subtypes. Then M5 and the remaining subtypes (M1 and M2) and lastly M1 and M2 were classified. The final results show that fifteen attributes were most appropriate for successful classification. A Multilayer Perceptron proved to be the best classifier. The results show that images were classified into either M3 or other subtypes with 97.22% accuracy, into M5 and the remaining subtypes (M1 and M2) with 90.69% accuracy and into M1 or M2 with 93.71% accuracy. The reason for incorrect classification is that in some cases the detected circles are located in-between blast cells or encapsulate red blood cells.

This chapter is organised as follows: Section 7.2, expands on previous work in the field of classification and Machine Learning, section 7.3 describes the work process presented in this chapter. Section 7.4 discusses the choice of the “best” classifier, the “best” heuristic method and the “best” method of detection of leukaemia cells. Some useful notation, the algorithms and testing results are presented. Section 7.5 discusses classification results and section 7.6 concludes the chapter.

7.2 Previous Work

One of the first steps in image analysis and pattern recognition is image segmentation. The simplest case in image segmentation is to have two regions, an object region and a background region. For example, here blast cells are the object

region and red blood cells the background region. The segmentation process is concerned with the process of partitioning the image but not with what the regions represent (Niblack, 1985). Therefore, feature extraction from colour images is required in many tasks of computer vision, e.g., object recognition, image retrieval and image matching (Kobayashi & Otsu, 2009). Object detection is the process of identifying and locating objects of interest in an image. Traditional approaches to object detection glance over the image with a smaller window, iterating over each pixel location and performing a classification subtask on each cut-out, identifying it as object or background. This is much more computationally expensive than the general image classification problem as it involves not only properly classifying a window as object or background, but also correctly locating the object inside the larger image. It also results in a large imbalance between objects and background, since there are always many times more non-object locations than object locations (Liddle et al., 2010).

WEKA is used to find the “best”, most promising results in models of Artificial Intelligent Networks (ANNs). When ANNs are trained, the cardinality of the input configurations can become an issue, in particular when a large number of hidden units is required to process information where multilayer networks are trained with the Back propagation learning rule. The experimental result, obtained using 10-fold cross-validation, was impressive (Bandini et al., 2009). In this thesis, WEKA was used to find the “best” classification of AML images into M1, M2, M3 and M5 AML subtypes.

There are a few papers on detecting white blood cells but not leukaemia cells using image processing. Previous work in (Piuri & Scott, 2004) used feed-forward neural network for classification and only concentrate on the nucleus. (Zamani & Safabakhsh, 2006; Jiang & S.Y.A, 2003) used a GVF (Gradient Vector Flow) snake and scale-space filtering with watershed algorithm on colour images for detecting nuclei. Another method for detection of blood cells using

eigen cells was introduced by (Yampri et al., 2006). In (Sarojini et al., 2010) a histogram of pixel counts focusing on the touching cells was created and an edge cutting algorithm was then applied to separate the cells. (Paya et al., 2006) used multilayer perceptron to develop a software tool to help urologists in obtaining an automatic diagnosis for complex multi-variable systems and to avoid painful and costly medical treatments. The experimental results showed high percentage of certainty of about 85%. In (Maree et al., 2007) the authors used a large set of randomly extracted image subwindows (or patches), describing each one by high-dimensional feature vectors composed of raw pixel values. Then, they used a method called extremely randomised decision trees to build a subwindow classification model. To predict the class of a new image, the method aggregates subwindow class predictions given by the decision trees and it uses majority voting to assign a class to the image. The paper evaluates the potential of the proposed image classification method in cell biology, by evaluating its performance on four datasets of images, related to protein distributions or subcellular locations and red-blood cells (Maree et al., 2007).

7.3 Work Process

This chapter focuses on feature extraction for classification of AML subtypes. Figure 7.1 presents the work-process. Machine Learning techniques were applied using WEKA (Hall et al., 2009). In chapter 6, blast cells were identified through the use of heuristic search. Here, to proceed with the classification step, feature extraction is performed, where the attributes of the extraction need to be defined.



Figure 7.1: Work Process: sub images - classification

7.4 Method

In the following sections the methods used in this chapter are presented, including feature extraction and classification of blast cells into M1, M2, M3 and M5 subtypes. Firstly, 20 images submitted to a SA seeded heuristic search were used to identify the “best” classification method, using WEKA (Hall et al., 2009) as shown in Figure 7.2. The twenty images, presented in section 6.5.3, were chosen randomly in order to find the “best” classifier. In particular, 5 M1, 3 M2, 8 M3 and 4 M5 images were utilised. Only twenty images were chosen because it is only test data and it is easier to evaluate with twenty images. The selection of greater number of M3 images was due to the fact that M3 AML requires a distinct chemotherapy approach with addition of the ATRA drug to the mixture.

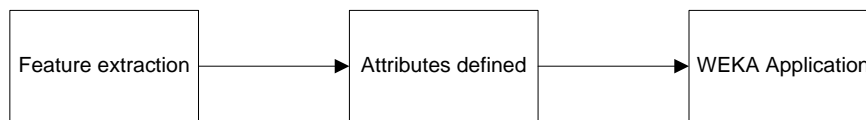


Figure 7.2: Finding the “best” method for classification

Then classification of 20 images was performed with the “best” classifier, using images submitted to HC, SA and GA as shown in Figure 7.3. First, classification is performed to separate images into those corresponding to M3 and those to all other AML subtypes. The reason for this is that identifying M3 subtype is of great importance because patients suffering from it require different treatment than patients suffering from the other subtypes. In particular, in M3 cases All-Trans-Retinoic-Acid (ATRA) is added to the initial chemotherapy. Then, the remaining images for subtypes M1, M2 and M5 were classified into M5 and the remaining two subtypes (M1 and M2). Lastly, classification into M1 and M2 subtypes was performed.

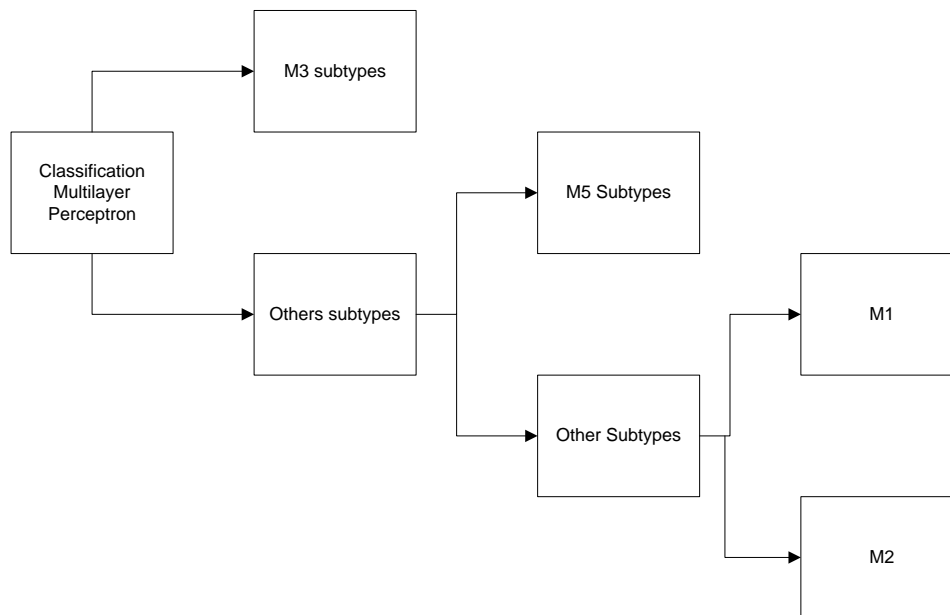


Figure 7.3: Classification for M3 Subtypes with other subtypes using multi-binary classification model

7.4.1 Sub Images

Sub images were performed on images of blast cells processed by the heuristic search methods described previously. The input files are the real images (Figure 7.4) and the coordinates produced by the heuristic search, that is coordinates x , y and radius (r). The extraction of blast cells from the images is then performed. Some example results of the feature extraction step are shown in the Figure 7.5, for circle shape. The 15 features extraction was used Mean, Median, Variance, Low and High of the pixels in each of the three RGB channels based on the simple statistical method. Other features extraction will not be considered because it will not do any correlated between the features. In the 15 feature extraction will look at the individual pixels to ensure the features of the subtypes.

7.4.2 Finding the “best” classifier

Once the location of blast cells is identified, feature extraction and classification is performed, using the WEKA application (Hall et al., 2009). The experimental result, obtained using ten fold cross-validation, was impressive. This study used 20 images with 15 attributes, including mean, median, variance, high and low of RGB levels for each pixel from SA image segmentation results. In order to find the “best” classifier, 20 images with 51 sub-images of individual blasts were used as training data.

The main factor affecting classification accuracy is not the classification method but the quality of the training data. The data for a class should not include other classes, yet must include a representative spread of pixels from the class. Getting a good set of training data is a necessary but difficult task. Cross-validation is another technique that reduces overfitting. The basic idea is to estimate how well each hypothesis will predict unseen data. This is done by setting aside some fraction of the known data and using it to test the prediction performance of a hypothesis induced from the remaining data (Russel & Norvig, 2003). Cross-validation was performed in this study. Some of the data was removed from the training dataset. Once the training was completed, the removed data was used to test the quality of the results.

The standard practice is to use ten-fold cross-validation, where the data is divided randomly into ten parts in which the class is represented in approximately the same proportion as in the full dataset. Each part is held out in turn and the learning scheme trained on the remaining nine-tenths; then its error rate is calculated on the holdout set. Thus, the learning procedure is executed a total of ten times on different training sets. Finally, the ten error estimates are averaged to yield an overall error estimates. Ten-fold cross validation is a good choice because extensive tests on numerous datasets, with different learning techniques,

have shown that ten is about the right number of folds to get the best estimate of error. Thus, the standard evaluation technique in situations where only limited data is available is stratified ten fold cross validation (Witten & Frank,. 2005).

7.4.3 Finding the “best” methods

Following the choice of the most efficient classifier, the next step is to use it to identify the “best” method between HC, SA and GA for processing of the images. It involves first classifying images into M3 and all other subtypes, then the remaining images into M5 and all other subtypes and finally into M1 and M2 subtypes. This part explains the feature extraction step for a circle shape and presents the used notation and algorithm. In this section a mean value of RGB levels is calculated which is a standard measure of mean, the median is the middle value when all values are sorted and to estimate the variance, each difference between a value and the mean is calculated, then squared and then the mean result calculated

Calculation of low and high values of RGB is based on range between 0 and 255, as shown in equations (7.1) and (7.2).

$$H(\underline{x}) = \sum_{i=1}^{|\underline{x}|} h(x_i) \quad (7.1)$$

$$L(\underline{x}) = \sum_{i=1}^{|\underline{x}|} (1 - h(x_i)) = |\underline{x}| - H(\underline{x}) \quad (7.2)$$

$$h(a) = \begin{cases} 0 & , \text{if } a < 127 \\ 1 & , \text{otherwise} \end{cases}$$

The input files are the attributes shown in the Table 7.1.

Table 7.1: Attributes for classifier

| Attributes | Red | Green | Blue |
|-------------------|------------|--------------|-------------|
| Mean | √ | √ | √ |
| Median | √ | √ | √ |
| Variance | √ | √ | √ |
| Sum of (7.1) | √ | √ | √ |
| Sum of (7.2) | √ | √ | √ |

7.4.4 Real data

This section presents the application of the analysis to real images in the dataset. The data used 648 blast cells resulting from image segmentation of 317 blood smear and bone marrow images. These are the final stages of testing for the classification stage. The classification hierarchy is presented in Figure 7.3. In this section 15 attributes are used, including the mean, median, variance, and middle of high and low values as shown in equations (7.1) and (7.2). The multilayer perceptron was the choice of WEKA classifier and SA the method for detecting the blast cells.

7.5 Results

This section presents some experimental results produced during the process of identifying the most appropriate attributes and classification method, using WEKA.

7.5.1 Sub Images

Figure 7.4 shows an example of an AML M3 image.

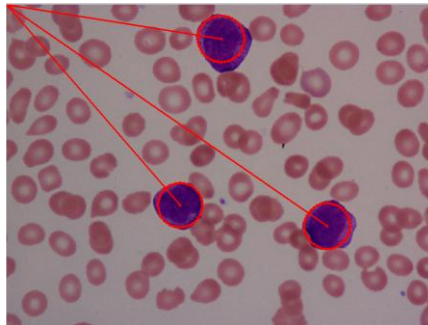


Figure 7.4: Real Image of M3 subtype

Figure 7.5 shows an example of the feature extraction for circle shape in image segmentation. The target area is the blast cell only. All the unnecessary features have been removed (black pixels). When performing the classification, only the target area is included.

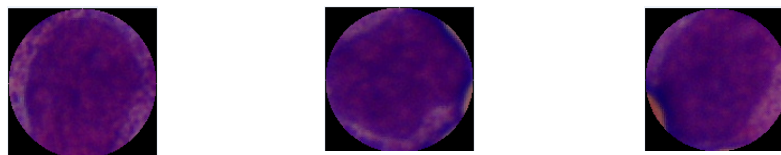


Figure 7.5: Sub-images of M3 subtype

7.5.2 Selecting the “best” classifier

To identify the “best” classifier, images processed with SA were used. As discussed, SA produces better results than the HC and GA methods. WEKA includes a number of classifiers, termed Classifier Bayes, Classifier Function, Classifier Lazy, Classifier Meta and so on. Each of the classifiers uses different methods. For example, in Classifier Function a Multilayer Perceptron and Support Vector Machines are implemented. All the methods were applied with the same parameter settings, that is, the WEKA defaults, using 10 folds cross validation (appendix H). Further research will include testing other settings. 20 images with 51 sub-images of blasts were used.

In this thesis, all WEKA classifiers were tested as shown in Table 7.2. Please refer to the appendix F for all results.

Table 7.2(a): Summary of the test result from WEKA (Classification Bayes, Function and Lazy)

| Method | Percentage Correct | Percentage Incorrect |
|------------------------------|--------------------|----------------------|
| Classification Bayes | | |
| NaiveBayes | 70.59% | 29.41% |
| Classifier Function | | |
| Logistic | 76.47% | 23.53% |
| <i>Multilayer Perceptron</i> | 92.16% | 7.84% |
| SMO (Support Vector Machine) | 74.50% | 25.50% |
| Classifier Lazy | | |
| <i>IB1</i> | 94.12% | 5.88% |
| <i>IBK</i> | 94.12% | 5.88% |

Table 7.2(b): Summary of the test result from WEKA (Classification Meta, Rules, Tree)

| Method | Percentage Correct | Percentage Incorrect |
|-----------------------------|--------------------|----------------------|
| Classifier Meta | | |
| ClassificationViaClustering | 54.90% | 45.10% |
| ClassificationViaRegression | 86.27% | 13.73% |
| Classifier Rules | | |
| ConjunctiveRule | 58.82% | 41.18% |
| Decision Table | 62.75% | 37.25% |
| Classifier Tree | | |
| <i>Functional Tree</i> | 92.16% | 7.84% |
| RandomForest | 78.43% | 21.57% |
| Random Tree | 76.47% | 23.53% |

7.5.3 Further testing for finding the “best” classifier

The results shown on Table 7.3, correspond to the four methods performing correct classification in more than 90% of instances. These are the Multilayer Perceptron and Functional Tree (FT) (92.16%), the Instance-based-learner (IB1) and the Instance-based-K-neighbourhood (IBK) method (94.12%). In conclusion, Multilayer Perceptron is the “best” method for classification.

In addition, the analysis was applied to all 648 images, resulting from segmentation of 317 images. The distribution of the images into the different AML types is shown on Table 7.3. The figures show that the M2 subtype has the highest number of sub-images. These percentages can be used to rate a random or majority classifier, as the results show, the accuracy using WEKA is significantly better.

Table 7.3: Summary of the distribution of images

| Images | Images | Sub-images | Percentage |
|---------------|---------------|-------------------|-------------------|
| M1 | 46 | 87 | 13.45% |
| M2 | 128 | 231 | 35.65% |
| M3 | 92 | 155 | 23.92% |
| M5 | 51 | 175 | 27.01% |

The result in Table 7.4 shows that the Multilayer Perceptron has correctly classified instances in 85.80% of all cases. The Functional Tree has correctly classified 82.41% of all instances. The Instance based Learner (IB1) was correct in 81.02% of cases. Finally, the Instance-based-K-neighbourhood (IBk) method classified correctly 81.02% of all instances.

Table 7.4: Summary of further testing for 317 images

| Method | Percentage Correct | Percentage Incorrect |
|--------------------------------------|---------------------------|-----------------------------|
| Multilayer Perceptron | 85.80% | 14.20% |
| Functional Tree | 82.41% | 17.60% |
| Instance-based Learner | 81.02% | 18.98% |
| Instance-based K-neighbourhood (IBK) | 81.02% | 18.98% |

7.5.4 Finding “best” method for HC, SA and GA

To identify the most efficient classification method, multi-binary classification was applied as described in Figure 7.3. The testing was based on 20 images containing 51 blast cells. In particular, there are 9 sub-mages of M1 blasts, 7 of M2, 22 of M3 and 13 of M5. We performed 10 fold cross-validations which is more robust than leave one out method to preserve results consistency between HC, SA and GA. This section discusses the choice of the best heuristic search method between HC, SA and GA, in order to prove the findings presented in chapter 6.

The results proved that SA is indeed the “best” method. Table 7.5 shows that the HC was overall correct in 92.45%, while the SA in 94.75% of instances. Although in chapter 6, using the average and standard deviation estimation it was shown that the HC performed better than the SA. The SA and HC differ in classification of M5 and remaining AML subtypes (second step). As far as the GA is concerned, overall correct classification is lower at 88.78%.

Table 7.5: Summary of HC, SA and GA for twenty images

| Heuristic Method | Classify | Multilayer Perceptron | | | | | | Average |
|------------------|----------------------------------|------------------------|--------|------------------------|--------|--------------------|--------|---------------|
| | | M3 with Other Subtypes | | M5 with Other Subtypes | | M1 and M2 Subtypes | | |
| HC | Correctly Classified Instances | 50 | 98.04% | 23 | 79.31% | 16 | 100% | 92.45%, |
| | Incorrectly Classifier Instances | 1 | 1.96% | 6 | 20.69% | 0 | 0.00% | 7.55% |
| SA | Correctly Classified Instances | 50 | 98.04% | 25 | 86.21% | 16 | 100% | 94.75% |
| | Incorrectly Classifier Instances | 1 | 1.96% | 4 | 13.79% | 0 | 0.00% | 5.25% |
| GA | Correctly Classified Instances | 49 | 96.08% | 24 | 82.76% | 14 | 87.50% | 88.78% |
| | Incorrectly Classifier Instances | 2 | 3.92% | 5 | 17.24% | 2 | 12.50% | 11.22% |

7.5.5 Real data

This section discusses the classification of 648 images of blast cells, resulting from the segmentation of the 317 images in the data set. The SA method was chosen for the detection of blast cells and Multilayer Perceptron for classification.

As mentioned before the classification approach is based on Figure 7.3, where the first step is the classification of all images into those corresponding to M3, labelled with 3, and those corresponding to all remaining AML subtypes, labelled with 0. Table 7.7 displays the results for this step. There are 630 blast cells that were correctly classified (97.22%). 18 images were incorrectly classified (2.77%). 10 images were wrongly classified as M3 and 8 images of M3 were wrongly classified as other types. As Table 7.7 shows the classification between M3 and other subtypes produced an extremely satisfactory result. The correct classification here is very important because the treatment for M3 is different than the one in the other subtypes.

Then all classified M3 images were removed and classification performed for M5, labelled with 5, and the remaining subtypes (M1 and M2) labelled with 0. There are 493 images in total corresponding to M1, M2 and M5 AML. 90.67% of images were correctly classified (447 images), while 9.33% were wrongly classified (46 images). 291 images in total were classified as M5 type and 156 as belonging to M1 and M2 types. The reason the classification was only correct in 90.67% of cases lies in the fact that in some instances the leukaemia cells were not correctly and fully captured as shown in the Figure 7.6.

The last step was the classification into M1 and M2 subtypes (Table 7.6), following remove of all classified M3 and M5 images. Here 93.71% (298 images) were classified correctly and 6.29% (twenty images) misclassified. In particular, 223 M2 images were classified correctly and 8 wrongly, while for M1 the same is true for 75 and 12 images respectively. In conclusion, the classification accuracy for all 4 subtypes (M1, M2, M3 and M5) was 93.86%. The utilised method is a multi-binary classification model, with classification of M3 *vs* (M1+M2+M5), M5 *vs* (M1+M2), and finally M1 *vs* M2.

The results of the KAPPA statistic show an agreement of 0.92 between the real and produced classification of images into M3 and other subtypes. For M5 and other subtypes and M1 and other subtypes the respective KAPPA values are 0.80 and 0.84. According to Table 2.8 agreement is almost perfect for all subtypes.

The utilised confusion matrix consists of TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative) values, as show on Table 7.6. Sensitivity is the calculation of the actual correctly identified instances, while Specificity the calculation of correctly identified negatives.

Table 7.6: Table for the Confusion matrix

| | | Prediction | |
|--------|---------------------|---------------------|---------------------|
| | | True Positive (TP) | True Negative (TN) |
| Actual | True Positive (TP) | | |
| | False Positive (FP) | | |
| | | False Positive (FP) | False Negative (FN) |

$$\text{Sensitivity} = \frac{\text{Number of true positive}}{\text{Number of true positive} + \text{Number of false negative}}$$

$$\text{Specificity} = \frac{\text{Number of true negative}}{\text{Number of true negative} + \text{Number of false positive}}$$

Table 7.7: Summary of using SA on the Figure 7.3

| Classify | Multilayer Perceptron | | | | | |
|----------------------------------|------------------------|--------|------------------------|--------|--------------------|--------|
| | M3 with Other Subtypes | | M5 with Other Subtypes | | M1 and M2 Subtypes | |
| Correctly Classified Instances | 630 | 97.22% | 447 | 90.69% | 298 | 93.71% |
| Incorrectly Classifier Instances | 18 | 2.78% | 46 | 9.33% | 20 | 6.29% |
| Kappa Statistic | 0.92 | | 0.80 | | 0.84 | |
| Total Number of Instances | 648 | | 493 | | 318 | |
| Confusion matrix | Other subtypes | M3 | Other Subtypes | M5 | M2 | M1 |
| | 485 | 8 | 291 | 27 | 223 | 8 |
| | 10 | 145 | 19 | 156 | 12 | 75 |

$$\text{Sensitivity for M3 with others} = \left(\frac{485}{485 + 145} \right) \times 100\% = 77\%$$

$$\text{Specificity for M3 with others} = \left(\frac{8}{8 + 10} \right) \times 100\% = 44\%$$

$$\text{Sensitivity for M5 with others} = \left(\frac{291}{291 + 156} \right) \times 100\% = 65\%$$

$$\text{Specificity for M3 with others} = \left(\frac{27}{27 + 19} \right) \times 100\% = 59\%$$

$$\text{Sensitivity for M1 and M2} = \left(\frac{223}{223 + 75} \right) \times 100\% = 75\%$$

$$\text{Specificity for M1 and M2} = \left(\frac{8}{8 + 12} \right) \times 100\% = 40\%$$

This mean that the test recognised 77% of sensitivity was highlighted as M3 subtype were correctly identified and 44% is specificity was wrongly identified. The second targeting result for the sensitivity for M1 and M2 subtypes were 75% and specificity were 40%. The results produced by the KAPPA metric and the estimated specificity and sensitivity are quite good.

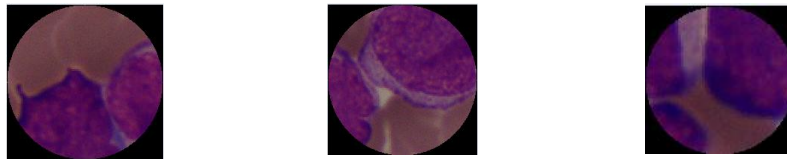


Figure 7.6: Example of wrong classification of M5 subtype

7.6. Summary

In this chapter the final steps and implemented methods for classification of blast cell images, using the WEKA application, a data analysis software tool, were presented.

The main purpose of the experiments presented here was to choose the “best” classifier. For this purpose, 317 images with ten-fold cross validation were used as training data, and a number of methods implemented using WEKA with 15 attributes including Mean, Median, Variance, High and Low of the colour

attributes (255). It was revealed that the Multilayer Perceptron exhibited the best performance.

Hierarchical classification was performed according to the diagram on Figure 7.3. Images were first classified into M3 and all remaining AML subtypes. The remaining images were classified into M5 and M1 and M2. Finally, the images of M1 and M2 AML were classified.

One of the main contributions of this study is that it proposes a methodology that can assist the haematologist's diagnosis of AML cases, by providing an automatic method of image processing for classification of blast cells.

Chapter 8

CONCLUSION AND FUTURE WORK

8.1 Conclusion

This thesis discussed the difficulties faced by haematologists in identifying the subtypes of Acute Myeloid Leukaemia (AML) manually, using a microscope. Given that morphological features of AML subtypes differ, the objectives of the research presented here was to automatically detect blast cells in blood and bone marrow images and classify them. The dataset used consisted of 322 images of four AML subtypes, namely M1, M2, M3 and M5. The images were collected from the Department of Haematology in the Hospital University Sains Malaysia (USM).

Leukaemia is a form of cancer where white blood cells fail to mature. White blood cells have a highly protective role in destroying invading organisms and assisting in the removal of dead or damaged tissue. Acute Myeloid

Leukaemia (AML) was selected as a case study because it is a serious illness caused by the abnormal growth and development of early nongranular white blood cells. AML can belong to a number of subtypes, M0 to M7. Each of the subtypes requires particular treatment and use of medication. The most important step in the classification process is to differentiate between M3 and all other AML subtypes, because M3 treatment requires the addition of All-Trans-Retinoic-Acid (ATRA) to the initial chemotherapy. In a medical environment classification of leukaemia cells requires lab testing such as cytogenetics and immunophenotyping for confirmation and diagnosis, which takes between three and five days.

This thesis presented a method for automated detection of blast cells and explored automated detection of blast cells, using random and seeded heuristic search methods. A comparison between Hill Climbing, Simulated Annealing and a Genetic Algorithm, in the first instance, showed that SA produced the higher fitness values but the GA targeted most of the blast cells. When using random search, the starting points are chosen randomly, and thus not located at the same place each time the algorithm is run. The obtained results were not consistent.

In the seeded heuristic search the starting coordinates, corresponding to potential blast cells, were defined. The analytical steps involved the application of the Otsu method, CA, CA Filtering, coordinate extraction, colour image clustering and heuristic search. The CA Filtering did not work in two images (98.76% success rate). In the image clustering step, three images were not processed successfully, with circles targeting pink areas. But almost 86.69% of blast cells were targeted correctly. In some cases the targeting circles overlap and the heuristic search cannot complete. On average 96.18% of blast cells were targeted correctly.

When performing the heuristic search step, it needs to be ensured that there are no overlapping circles. The starting points are consistently located at the

same positions each time the processing is applied, because their coordinates have been defined. In the case of seeded search, the SA method outperformed the rest.

The WEKA application was used for classification of AML images of four subtypes (M1, M2, M3 and M5). To choose the “best” classifier, a number of different ones were tested using ten-fold cross validation. The results show that a Multilayer Perceptron is the “best” classifier. The findings showed that fifteen attributes were the most appropriate for our purposes (Mean, Median, Variance, High and Low of the colour attributes (255)). In addition having reached this stage performance of the examined heuristic search methods, namely HC, SA and a GA can be validated. Upon completion of the classification, it was observed that SA produced the best results, based on 20 images.

Finally, images were processed with SA (to identify the most appropriate method), using the selected fifteen attributes, and multi-binary classification (the “best” classification). The accuracy was of 97.22% for the classification between M3 and other AML subtypes. For M5 with other subtypes, the accuracy drops to 90.66%. Finally between M1 and M2 subtypes the obtained accuracy was equal to 93.71%. A summary of the processing steps in the case of seeded heuristic search is shown in Figure 8.1(a). Figure 8.1(b) depicts the feature extraction step of Figure 8.1(a).

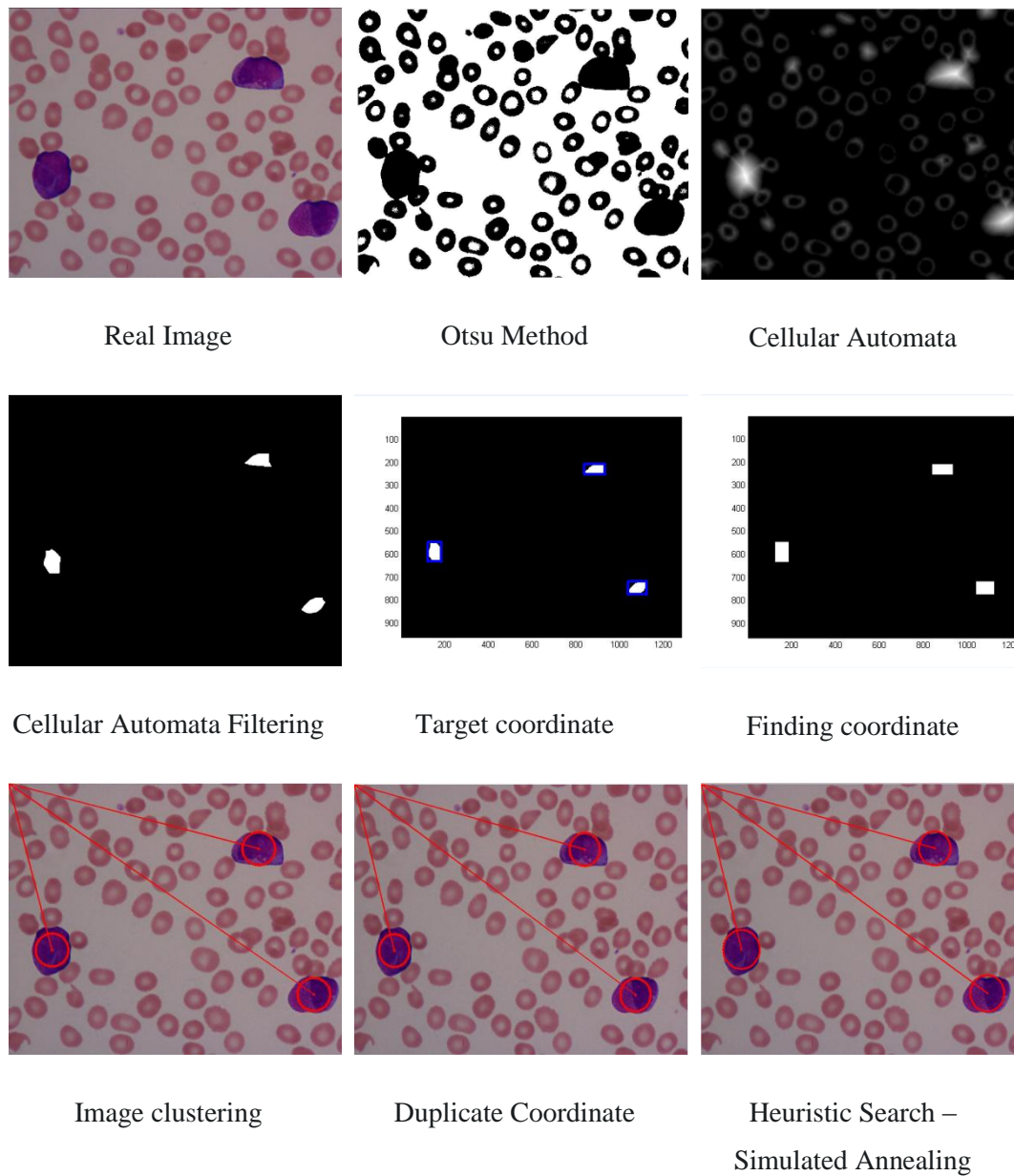


Figure 8.1(a): Example of processing steps in Seeded Heuristic Search

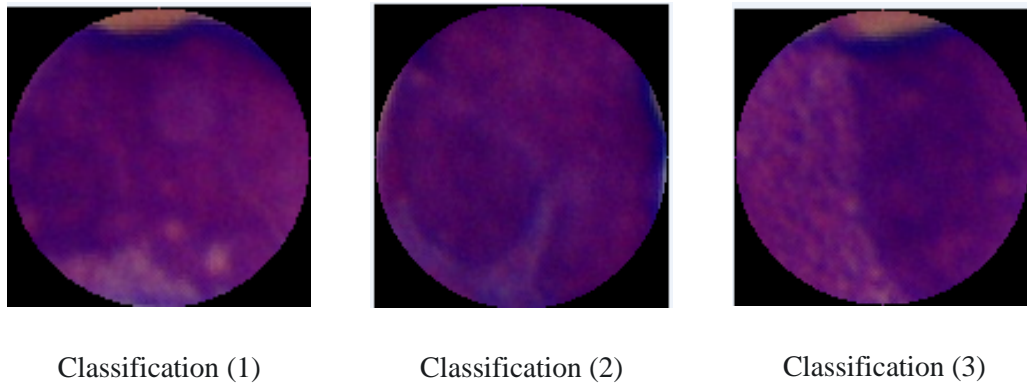


Figure 8.1(b): Example of processing steps in Seeded Heuristic Search - Classification

8.2 Achievements

This section highlights the achievements of the thesis, which are the following :-

- a) Produced an automated method of analysis of AML blast cells images using the Cellular Automata method, Cellular Automata Filtering, Finding Coordinate, Image Classification, Duplicate Coordinate and heuristic search techniques to help haematologists detect blast cells.
- b) A Multilayer Perceptron was used for classification into M3 and other AML subtypes. The reason for concentrating on M3 is that its treatment differs from the treatment of the rest, requiring All-Trans-Retinoic-Acid (ATRA) to be added to the initial chemotherapy.
- c) Produced a prototype of image processing software to help Haematologists diagnose AML more effectively and efficiently (Appendix I). There are a few amendments that need to be included in future work, such as linking MATLAB with WEKA.

8.3 Contribution to knowledge

The key contributions of this thesis are below:

- a) Applied heuristic search algorithms, including, Hill climbing, Simulated Annealing and a Genetic Algorithm, for processing of images of leukaemia cells and the detection of abnormal cells, known as blast cells.
- b) Determined the “best” amongst a number of alternative fitness functions for the automatic detection of blasts. The fitness functions were discussed in the Chapter 3.
- c) CA filtering can remove all the unwanted cells such as red blood cells using a new method of filtering, based on five types of white blood cells. The purpose of using a CA is to convert a black and white cell image to a matrix (the same size as the input image) where each point in the matrix represents the shortest distance each point in the image is away from the background. The method is also used to determine the radius of a blast cell (Chapter 4).
- d) Image classification was performed to identify red blood and leukaemia cells. The calculation of purple (blast cells) and pink (red blood cells) was performed manually based on eye judgment and twenty randomly selected real images. Colours were cross-referenced with the colours in Microsoft Power point to find RGB values (Chapter 5).
- e) Implemented a combination of Otsu’s method, CA and Heuristic Search for detection of leukaemia cells. Used Seeded Heuristic Search. Where the coordinates of the blast cells were been defined. The results show Simulated Annealing proved to be the “best” out of the three distinct search approaches (Chapter 6).

8.4 Limitation

The limitations of the thesis are the following:

- a) The utilised images were produced with the same brightness and by a single microscope. It would be useful to test the methodology with images of varying brightness levels coming from different sources.
- b) Figure 4.6 shows two images where CA filtering was unsuccessful. Both images can be subjected to enhancement.
- c) For the image classification step, besides the RGB values, additional features, such as HSL (Hue, Saturation and Lightness), can be utilised.

8.5 Future Research

This section comments on future research resulting from this thesis. Some directions are discussed below:

- a) Reduce Time
As a standard procedure, the haematologist needs to diagnose Acute Myeloid Leukaemia (AML) based on cytogenetic testing. The process lasts from three to five days. The time required for automatic classification of Acute Myeloid Leukaemia (AML), for the twenty images, is shown in Appendix G, lasting from minutes to up to 2 hours at most. Thus, the methodology can help the haematologist to diagnose patients more efficiency. In (Niblack,. 1985) the author mentions as the main disadvantage of digital image processing its speed and cost. However, these factors have been significantly reduced by recent developments in computer technology and the associated lower cost. Many useful digital image processing operations are now available on personal computers and desktop workstations. Graphic Processing Units (GPU) bring together an unprecedented combination of high performance at low cost. (Hartley et

al., 2008) used cooperative parallelization in implementing a large-scale, biomedical image analysis application, which involves a number of diverse kernels including typical streaming operators, concurrence matrices, convolutions and histograms. The test results show that linear speed-up is achieved when all coexisting methods in Central Processing Unit (CPU)s and Graphic Processing Unit (GPU)s are combined. This will significantly reduce the runtime of the procedure.

b) Feature Annotation

Developing effective methods for automated annotation of digital pictures continues to inspire computer scientists. The capability of annotating pictures by computers can lead to breakthroughs in a wide range of applications such as web image search, online picture sharing and others. The Automatic Linguistic Indexing of Pictures – Real Time (ALIPR) application has been developed for fully automated and high-speed annotation of online pictures (Li & Wang, 2007). Currently, the haematologist is expected to diagnose leukaemia into subtypes. By performing automatic classification the haematologists can see if the diagnosis is correct. In future research, the application can provide real-time annotation with exceptional accuracy.

c) Counting the blast cells

The presence of 20% or more blasts in bone marrow or blood smear images based on World Health Organization (WHO) classification, confirms AML diagnosis. (Golchin & Paliwal, 2003) used classification of small blocks named quadtree. Currently haematologists calculate the number of blast cells manually. The counting will be performed automatically by the system to aid haematologists in diagnosis.

8.6 Process learn of the PhD

Most of the important things that I learned are supervising team and the network. In the supervising team, choose the right person to be my supervisor which he is expert in the area. The area of research must really interesting because I will study about the area for 4 years. In the networking areas which are widen my focus and accept other people idea by attending conferences. I also developed a number of skills and learned how to be committed to my research, how to work efficiently and manage my time appropriately, given that my government allowed me limited time to complete my PhD. I was able to learn in multi-tasking work such as writing conference paper and at the same time executing the experience. Last minutes work and time constraint really helps me in developing manageable skill. Each of the problems encountered needed to be solved and this was achieved through discussion and help from my supervisor. PhD is the contribution to knowledge which I have developed prototyping software which can help medical doctors in diagnosis the leukaemia cells more effectively and efficiency.

This thesis is only a preliminary study of automated detection and classification of leukaemia cells. The results are very promising. The prototype has been developed and is available for use. Although this is a preliminarily study, the results are very promising and further research can be beneficial.

REFERENCES

- Able Software Corp, (2011) *Modelling and Raster to Vector conversion Leading Software Developer for 3D Imaging*. Retrieved from :- <http://www.ablesw.com/>. (21th January 2011).
- Adelson, E. H., Anderson, C. H., Bergen, J.R, Burt, P.J and Ogden, J.M. (1984). Pyramid methods in image processing. *RCA Engineer*: 33 - 41.
- Aguilera, D G. and Lahoz, J. G., (2008). Learning from Educational Software in 3d Cartography. *British Journal of Educational Technology* 39.4. 726 - 731.
- Antonie, M-Luiza, Osmar R Z., and Alexandry, C.(2001). Application of Data Mining Techniques for Medical Image Classification. *Second International Workshop on Multimedia Data Mining (MDM/KDD'2001) in conjunction with ACM SIGKDD Conference. San Francisco, USA*. 94 - 101
- Bandini, S., Vanneschi, L., Wuensche, A. and Shehata, A. B., (2009). Cellular Automata Pattern Recognition and Rule Evolution Through a Neuro-Genetic Approach. *Journal of Cellular Automata* 4. 171-181.
- Bankhead, A. and Heckendorn, R. B., (2007). Using evolvable genetic cellular automata to model breast cancer. *Genet Program Evolvable March*(8). 381 – 393.
- Basic Morphology of Blood Film., (2011). Retrieved from :- <http://apple.objectivepathology.com/moodle/mod/resource/view.php?id=441>. (22nd February 2011)
- Bell, A. and Sallah, S., (2005). *The Morphology of Human Blood Cells*. Seventh Edition ed: Abbott Ltd, USA.

- Bennett, J.M., Catusky, D., Marie-Theregsa Danieal, Flandrin, G., Galton, D.A.G., Gralnick, H. R. and Sultan, C., (1976). Proposal for the Classification of the Acute Leukaemias French-American-British (FAB) Co-Operative Group. *British Journal of Heamatology*. Volume 33, Issues 4. 451-458
- Bishop, C. M., (1995). *Neural Network for Pattern Recognition*. Clarendon Press, Oxford. ISBN 0 19 853849 9.
- Blood – CH19, (2011) Retrieve from :-
http://www.utdallas.edu/dept/abp/PDF_Files/AP_Folder/Blood.pdf. (24th March 2011).
- Bosch, A., Zisserman, A and Munoz, X., (2007). Image Classification using Random Forests and Ferns. *Computer Vision 2007 ICCV*. Rio de Janeiro, Brazil, IEEE 11th International Conference.
- Brunetti, A. and Haraldseth, O., (2007). *Medical Imaging for Improved Patient Care*. Retrieve from :-
http://www.esf.org/fileadmin/links/EMRC/ESF_POLICY28_V09_HD.pdf
 . (12th December 2010)
- Burke, E. K. and Kendall, G., (2005). *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*. Springer Science+Business Media. Inc. ISBN 0-387-23460-8
- Chang, T and C. C Jay Kuo. (1993) Texture Analysis and Classification with Tree-Structured Wavelet. *IEEE Transactions on Image Processing* 2.4. 429 - 441.
- Chen, J., Kim, M., Wong, Y. and Ji. Q., (2009). Switching Gaussian process dynamic models for simultaneous composite motion tracking and recognition. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami Florida. 2655-2662.
- Cheng, H. D., Jiang, X. H., Sun, Y and Wang, J., (2001). Colour image segmentation: advances and prospects. *Pattern Recognition* 32. 2259 – 2281.

- Chipperfield, A., Fleming, P., Pohlheim, H. and Fonseca, C. (2011). *Genetic Algorithm Toolbox - for Use with Matlab*. Department of Automatic Control and System Engineering - University Sheffield.
- Chronic Lymphocytic Leukemia/Small Cell Lymphoma (CLL/SLL). (2010)."
Retreive from:-
[http://www.clinicalflow.com/cases/case_list/mature_b-cell_neoplasms_\(general\)/chronic_lymphocytic_leukemia%2F%2Fsmall_cell_lymphoma_\(cll%2F%2Fsll\)](http://www.clinicalflow.com/cases/case_list/mature_b-cell_neoplasms_(general)/chronic_lymphocytic_leukemia%2F%2Fsmall_cell_lymphoma_(cll%2F%2Fsll)). (10th October 2010)
- Davidson, R. and Harel, D., (1996). Drawing Graphs Nicely Using Simulated Annealing. *ACM Transactions on Graphics. Vol 15. No. 4.* 301 – 331.
- De Boer, R. J. Hogeweg, P., and Perelson, A. S., (1992). Growth and Recruitment in the Immune Network. *Springer-Verlag Berlin Heidelberg 66.* 223-247.
- Dimistris, B and Tsitsiklis, J., (1993). Simulated Annealing. *Statistical Science 8.1.* 10-15.
- Dugdale, D. C., (2010). *Bone Marrow Aspiration*. Retrieved from:-
<http://health.allrefer.com/pictures-images/bone-marrow-aspiration.html>.
(2nd March 2010).
- Du, Y., Chang, C., Thouin, P. D., (2004). Unsupervised approach to color video thresholding. *Optical Engineering 43(2).* 282 – 289.
- Golchin F and Paliwal, K. K., (2003). Quadtree-based classification in subband image coding. *Elsevier - Digital Signal Processing 13.* 656-668.
- Falkenauer., E. (1998). *Genetic Algorithms and Grouping Problems*. John Wiley & Sons Ltd. ISBN 0-471-97150-2.
- Fang, Y., Pan C., Liu, L. and Fang, L., (2005.). Fast Training of SVM via Morphological Clustering for Color Image Segmentation. *Part I, LNCS, © Springer-Verlag Berlin Heidelberg 2005.* 135 – 138.
- Fang, Y., Zheng, C., Pan, C. and Liu, L., (2005). White Blood Cell Image Segmentation Using On-line Trained Neural Network. *Engineering in Medicine and Biology 27th Annual Conference, Shanghai, China.* 263 – 271.

- Freeman, J. A. and Skapura D. M., (1992). *Neural Networks: Algorithms, Applications, and Programming Techniques*. Addison-Wesley Publishing Company. ISBN 0-201-51376-5
- Frost, Britt-Marie., (2003). *Chemotherapy in Childhood Acute Lymphoblastic Leukemia: In Vitro Cellular Drug Resistance and Pharmacokinetics*. (Doctoral Dissertation). Retrieve from :-
<http://uu.diva-portal.org/smash/get/diva2:161996/FULLTEXT01>. (24th March 2011)
- Ganguly, N., Sikdar, B. K., Duetsch, A., Canright. G. and Chaudhuri, P. P., (2003). "A Survey on Cellular Automata." Retrieved from :-
<http://www.cs.unibo.it/bison/publications/CAsurvey.pdf>. (11th November 2011).
- Gonzalez, R. C., and Woods, R. E., (2002). *Digital Image Processing*. 2nd Edition Prentice hall. ISBN - 0-130-94650-8.
- Griffeath, D. and Moore, C. E., (2003). *New Constructions in Cellular Automata. Cellular Automata for Imaging, Art and Video*. New York, Oxford University Press. 285 – 292. ISBN 0-19-513717-5
- Guieb, E. C., and Samaneigo. J. M., (2007). Image Noise Reduction Using Cellular Automata. *CMSC 190 Special Problem, Institute Of Computer Science*.
- Gurney, K., (1997). *An Introduction to Neural Networks*. UCL Press Limited. ISBN 1-85728-673-1.
- Haken, D., (2010). *Immunology*. Retrieved from :-
<http://users.csc.calpoly.edu/~zwood/teaching/csc471/finalw10/dhaken/>. (2nd February 2011).
- Hassan, R. (1996). *Diagnosis and outcome of patients with Acute Leukemia*. Hematology department. Malaysia, Universiti Sains Malaysia. Degree of master of Medicine.
- Hassan, R. (2008). Molecular Diagnosis in Acute Leukaemia : Hospital University's experience. *International Joint Symposium Frontier In Biomedical Sciences: From Genes to Applications, Faculty of Medicine, Gadjah Mada University, Yogyakarta, Indonesia*.

- Hanbury, A (2002). The taming of the hue, saturation and brightness colour space. In *Proceedings of the 7th Computer Vision Winter Workshop, Bad Aussee, Austria*. 234 – 243.
- Hartley, T. D. R., Catalyurek, U and Ruiz, A,. (2008). Biomedical Image Analysis on a Cooperative Cluster of GPUs and Multicores. *22nd ACM International Conference on Supercomputing, Island of Kos - Aegean Sea - Greece, ACM*. 15 – 25.
- Harmon, P and King, D,. (1985). *Artificial Intelligence in Business*. New York: John Wiley. ISBN 0-47-180824-5.
- Hoffbrand, A.V., Pettit, J.E., and Moss, P.A.H., (2001) *Essential Haematology*, Fourth Edition. Fourth ed: Blackwell Science. ISBN 0-63-205153-1.
- Holland, J. H., (1975) *Adaptation in Natural and Artificial System*. The MIT Press. ISBN 0-26-208213-6.
- Holland, J. H., (1992). *Genetic Algorithms*. Scientific American.
- Huang, Zhi-Kai, and Kwok-Wing Chau., (2008). A New Image Thresholding Method Based on Gaussian Mixture. *Applied Mathematics and Computation*. 899 – 907.
- Image Processing and Vision: Introduction., (2011). Retrieved from: <http://www.heppenstall.ca/academics/doc/472/CIS472.Lectures01.Fundamentals.pdf>. (20th January 2011).
- Jiang, K., Liao, Q. G. and Dai, S. Y. (2003). A Novel White Blood Cell Segmentation Scheme Using Scale-Space Filtering And Watershed Clustering. *Second International Conference on Machine Learning and Cybernetics, Xi'an, Proceedings of the Second International Conference on Machine Learning and Cybernetics, Xi'an. China*. 2820 – 2825.
- Sarojini, K. Thangavel, K and Devakumari, D., (2010). Supervised Feature Subset Selection based on Modified Fuzzy Relative Information Measure for classifier Cart. *International Journal of Engineering Science and Technology (Vol 2(5))*. 2456-2465.
- Krause, E. F., (1987). *Taxicab Geometry*. ISBN 0-486-25202-7.
- Krap E. J (2007). *Acute Myelogenous Leukemia*. Human Press. ISBN 15-882936-210.

- King, R. J. B. and Robins, M. W., (2006). *Cancer Biology*. (3rd ed.). Pearson Prentice Hall. ISBN 0-13-129454-7.
- Liao, P.-S., Chen, T.-S. and Chung, P.-C., (2001). A Fast Algorithm for Multilevel Thresholding. *Journal of Information Science and Engineering* 17. 713-727.
- Liddle, T., Johnston, M and Zhang, M., (2010). Multi-Objective Genetic Programming for object detection. *Evolutionary Computation (CEC), Shanghai, China. 18 – 23 July 2010*. 1-8.
- Li, J. and Wang, J. Z., (2007). Real-Time Computerized Annotation of Pictures. *IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol 30. Issues 6*. 970 – 984.
- Liu, Z. and Gibbon, D. (2010) Lecture 1. Introduction to Digital Image Processing. Retrieved from :- http://www.ece.tufts.edu/en/74/01_introduction.pdf. (11th November 2011).
- Leukaemia statistics – Key Facts,. (2010). Retrieved from :- <http://info.cancerresearchuk.org/cancerstats/types/leukaemia/uk-leukaemia-statistics>. (25th November 2010).
- Leukaemia survival statistics,. (2010). Retrieved from :- <http://info.cancerresearchuk.org/cancerstats/types/leukaemia/survival/>. (25th November 2010).
- Levy, S., (1993). *Artificial Life*. Vintage Press. ISBN 0-679-74389-8.
- Lochanambal, K.P., and Karnan., M (2010) Hybrid Heuristics for Mamogram Segmentation. *Computational Intelligence and Computing Research (ICCIC), IEEE International Conference*. 710 – 714.
- Malinga, B., Raicu, D. and Furst, J., (2006). *Local vs. Global Histogram-Based Color Image Clustering*. Retrieved from:- <http://www.mendeley.com/research/local-vs-global-histogrambased-color-image-clustering/>. (11th January 2011).
- Maree, R., Geurts, P and Wehenkel, L., (2007). Random Subwindows and extremely randomized tree for image classification in cell biology. *BMC Cell Biology*(8). 1 – 12.

- Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann and Ian H. Witten,. (2009); *The WEKA Data Mining Software: An Update; SIGKDD Explorations*, Volume 11, Issue 1.
- Mc Connell J. Jeffrey (2008). *Analysis of Algorithms: An Active Learning Approach*. 2nd Edition. Jones and Bartlett Publishing.
- Mcgaufflin, S., Munger, J and Nelson, R,. (2005). *Chronic Myelogenous Leukemia*. Retrieved from :-
<http://rebeccanelson.com/leukemia/cml.html>. (13th March 2010).
- Michalewicz, Z. and Fogel, D. B., (2004). *How to Solve It: Modern Heuristics*. Springer-Verlag Berlin Heidelberg 2000,2004. ISBN 3-540-22494-7.
- Mitchell, M., (1996). *Computation in Cellular Automata*. Retrieved from :-
<http://web.cecs.pdx.edu/~mm/ca-review.pdf>. 1 – 41.(20th January 2010).
- Mitchell, M., (1996). *An Introduction to Genetic Algorithms*. The MIT Press. ISBN 0-262-13316-4.
- Moreira, J. and Deutsch, A., (2002). Cellular Automaton Models Of Tumor Development: A Critical Review. *Advances in Complex System 5 (Nos. 2 & 3)*: 247 - 267.
- Morse, B. S., (2000). *Lecturer 4: Thresholding*. Retrieved from :-
http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MORSE/threshold.pdf. (1st April 2010).
- Natural Biosciences : Autologous Stem Cell Generated Molecular Therapy. (2011) Retrieved from :-
<http://www.natural-biosciences.com/therapy.html>. (3rd March 2011).
- Niblack, W., (1985). *An Introduction to Digital Image Processing*. Prentice-Hall International (UK) Ltd. ISBN 0-13-48074-3.
- Nilsson, B. and Heyden. A., (2002). Model-based Segmentation of Leukocytes Clusters. *Proceedings of the 16th International Conference on Pattern Recognition (ICPR02)*. Quebec City, Canada. 10727 – 10731.
- Nipon, T. U. and Gader, P., (2002). System-level training of neural networks for counting white blood cells. *IEEE Trans. SMS-C, Vol. 32(1)*. 48-53.

- Teri, N. D and McCoyca, C. C., (2011). *Acute Lymphoblastic Leukemia (ALL) - General Information*. Retrieve from :-
[http://www.clinicalflow.com/Cases/Case_List/Acute_Lymphoblastic_Leukemia_\(ALL\)_-_General_Information](http://www.clinicalflow.com/Cases/Case_List/Acute_Lymphoblastic_Leukemia_(ALL)_-_General_Information). (10th January 2011).
- O'Neill, Paul. (2005). *Improved Analysis of Microarray Images*. School of Information System and Computing. Brunel University. (Doctoral Dissertation).
- Otsu, N., (1979). A Threshold Selection Method from Gray-Level Histograms." *IEEE Transactions On Systems, Man, And Cybernetics SMC-9.No. 1*. 62 - 66.
- Orazi, A, O'Malley, P. D and Arber, A. D., (2006). *Illustrated Pathology of Bone Marrow*. Cambridge University Press. ISBN 0-511-22602-0.
- Paya, A. S., Fernandex, R. D., Mendex, D. G. and Hernandex, C. A. M., (2006). Development of an artificial neural network for helping to diagnose diseases in urology. *Proceeding of the 1st International Conference on Bio inspired models of network, information and computing systems (BIONETICS 06)*. Article No.9.
- Picton, P., (1994). *Neural Network*. Palgrave. ISBN 0-333-80287-X.
- Piuri, V. and Scotti, F., (2004). Morphological Classification of Blood Leucocytes by Microscope Image. *CIMSA 2004 - IEEE International Conference on Computational Intelligence for Memremment Systems and Applications*. Boston, MA. USA, IEEE. 103-108.
- Plesa, A., Cuipera, G., Louvet, V., Pujo-Menjouet, L., Genieys, S., Dumontet, C., Thomas, X. and Volpert, V., (2008). Dignostics of the Aml with Immunophenotypical Data. *Math Model. Nat. Phenom 2.1*. 104-23.
- Poulsen, R. S. and Pedron, I., (1995). Region of Interest Finding in Reduced Resolution Colour Imagery - Application To Cancer Cell Detection in Cell Overlaps and Clusters. *Engineering in Medicine and Biology and Biology Society 1995. IEEE 17th Annual conference. Montreal. Canada*. 499-500.
- Puetter, R.C., Gosnell T. R, and Yahil, A., (2005) Digital Image Reconstruction: Deblurring and Denoising. *Annu. Rev. Astron. Astrophys 43* : 139 - 94.

- Purves, W., Orians, G., and Heller, C., (1995) *Life, the Science of Biology*. Freeman. ISBN 0-71-677893-9.
- Rapson, D and Matthews, J (2011). *Intrepreting Blood Count*. Retreived From:- <http://meds.queensu.ca/medicine/deptmed/hemonc/lectures/cbc/cbc.pdf> (13th May 2012)
- Ritter, N. and Cooper, J., (2007). Segmentation and Border Identification of Cells in Images of Peripheral Blood Smear Slides. *Thirtieth Australasian Computer Science Conference (ACSC2007), Victoria, Australia*. 161-169.
- Reeves, C. R., (1993). *Modern Heuristic Techniques for Combinatorial Problems*. John Wiley & Sons. ISBN:0-470-22079-1.
- Reeves, C. R., (1999). Landscapes, Operators and Heuristic Search. *Annals of Operations Research 86*: 473 - 90.
- Robison M.D. , Chiu S.J., Lo J.Y., Toth C.A., Izatt J.A., Forsiu S., (2010). *Novel Applications of Super-Resolution in Medical Imaging* Book Chapter in Super-Resolution Imaging, Peyman Milanfar(Editor). CRC Press. Page 383-412.
- Rolf, A. and Bischof, L., (1994). Seeded Region Growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence 16.No. 6*. 641 – 647.
- Roth, G. and Levine. M.D., (1994). Geometric Primitive Extraction Using a Genetic Algorithm. *IEEE Transactions On Pattern AnalysisS And Machine Intelligence 16.No. 9 (1994)*. 901-05.
- Russel, S. and Norvig, P., (2003). *Artificial Intelligence: A Modern Approach*. ISBN 0-13-080302-2.
- Santos, R. M. Z. d. and Continho, S., (2001). Dynamics of HIV Infection: A Cellular Automata Approach. *Physical Review Letters 87(16)*. 102-104.
- Schmitz, J. E., Kansal, A. R. and Torquato, S., (2002). A Cellular Automaton Model of Brain Tumor Treatment and Resistance. *Journal of Theoretical Medicine 4(4)*. 223-239.
- Scholl, I, Aach, T., Deserno, T. M. and Kuhlen, T., (2010) Challenges of Medical Image Processing. *Computer Sci Res Dev*. 5-13.

- Scotti, F., (2006). Robust Segmentation and Measurements Techniques of White Cells in Blood Microscope Images. *IMTC 2006 - Instrumentation and Measurement Technology Conference. Sorrento, Italy: IEEE.* 43 – 48.
- Shasky, M. T., (2011). *Basic of Genetic Algorithms*. Retrieved from :- <http://dc317.4shared.com/doc/K2z9fxPP/preview.html>.
- Sirisathitkul, Y., Auwatanamongkol, S. and Uyyanonvara, B., (2004). Fast Color Image Quantization using Squared Euclidean Distance of Adjacent Color Points along the Highest Color Variance Axis. *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04). Cambridge, UK.* 656-659.
- Staple, E., (2011). *Venn Diagrams*. Retrieved from <http://www.purplemath.com/modules/venndiag.htm>. (23th February 2011).
- Stock, W and Hoffman, R (2000). White Blood Cells 1: Non-Malignant Disorders. *The Lancet. Vol 355.* 1351 – 1357.
- Stasi, Roberto, et al. Aml-Mo: A Review of Laboratory Features and Proposal of New Diagnostic Criteria. *Blood Cells, Molecules and Diseases (1999)* 25.8.
- Swift, S., Tucker, A., Vinciotti, V., Martin, N., Orengo, C., Liu, X. and Kellam, P., (2004) Consensus clustering and functional interpretation of gene-expression data. *Genome Biology 2004*, BioMed Central Ltd.
- Kobayashi, T and Otsu, N., (2009) Color Image Feature Extraction Using Color Index Local Auto-Correlations. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009), Taipei, Taiwan.* 1057 – 1060.
- Tagliasacchi. D and Carboni. G., (1997). *Blood cells*. Retrieved from :- http://www.funsci.com/fun3_en/blood/blood.htm. (20th January 2010).
- Talibi, E.-G., (2009). *Metaheuristics From Design to Implementation*. John Wiley & Sons. ISBN 978-0-470-27858-1.
- The Blood., (2011). Retrieved from :- <http://greenfield.fortunecity.com/rattle/46/intro.htm>. (23rd March 2011).

- The Encyclopedia of Science., (2011). Retrieved from :-
<http://www.daviddarling.info/encyclopedia/E/eosinophil.html>. (14th
 February 2011).
- The Jackson Laboratory., (2011). *Normal Karyotype*. Retrieved from
<http://www.jax.org/cyto/chromo.html>. (15th February 2011).
- The Nervous System., (2011). Retrieved from :-
http://www.naturalhealthschool.com/9_2.htm. (14th March 2011).
- Thollesson, M., (2011). *Phylogenetic Inference*. Retrieved from:-
http://artedi.ebc.uu.se/course/Embo01/Phylogeny/phylogeny_readme.html.
 (13th January 2011).
- Toffoli, T and Margolus, N., (1987). *Cellular Automata Machines : A New
 Environment for Modeling*. MIT Press Series in Scientific Computation.
 ISBN 0-262-20060-0.
- Uthman, E., (2008). *Blood Cells and the CBC*. Retrieved from :-
http://web2.airmail.net/uthman/blood_cells.html. (30th January 2010).
- Van Larrhoven, P.J.M and Aarts, E.H.L., (1987). *Simulated Annealing: Theory
 and Applications*. Kluwer Academic Publishers. ISBN 9-02-772513-6.
- Victor, A-R., Carlos, G-C. H., Arturo, P-G. and Raul, S-Y., E., (2006). Circle
 detection on images using genetic algorithms. *Pattern Recognition Letters*
 27: 652–657.
- Viera, A. J., and Garrett. J.M., (2005) Understanding Interobserver Agreement:
 The Kappa Statistic. *Research Series Vol. 37. No. 5*. 360-363.
- Waidah, I, Rosline, H and Swift, S, (2010). Detecting Leukaemia (AML) blood
 cells using Genetic Algorithms. *2nd Student Conference Operational
 Research (SCOR 2010), University Nottingham, UK*.
- Waidah, I, Rosline, H and Swift, S., (2010) Detecting Leukaemia (AML) blood
 cells using Cellular Automata and Heuristic Search. *Intelligent Data
 Analysis (IDA 2010) University of Arizona, USA*. 54 – 66.
- Wiernik, H. P, Goldman, M. J, Dutcher, P.J and Kyle. A.R., (2003). Chronic
 Leukemias and Related Disorders. *Cambridge University Press*.

- Witten, I. H. and Frank, E., (2005). *Data Mining : Practical Machine Learning Tools and Techniques*. Morgan Kaufmann. ISBN 978-0-12-088407-0.
- Weinberg, R., (1998). *One Renegade Cell, The Quest for the Origins of Cancer*. London Weidenfeld & Nicolson. ISBN 0297816454
- Weisstein, W., (2011). *Distance*. From MathWorld--A Wolfram Web Resource. Retrieved from :- <http://mathworld.wolfram.com/Distance.html>. (3rd April 2011).
- Weisstein, E. (2011). *Venn Diagrams*. From MathWorld--A Wolfram Web Resource. Retrieved from :- <http://mathworld.wolfram.com/VennDiagram.html>. (30th May 2011).
- Wongthanavase, S. and Tangvorphonkchai, V., (2007). Cellular Automata-Based Algorithm and Its Application in Medical Image Processing. *International Conference on Image Processing, San Antonio, TX, IEEE*. 41- 44.
- Wu, J., Zeng, P, Zhou, Y and Olivier, C., (2006). A Novel Color Image Segmentation Method and Its Application to White Blood Cell Image Analysis. *8th International Conference on Signal Processing, ISCP 2006, Guilin, CHINA, IEEE*.
- Yampri, P., Pintavirooj, C., Daochai, S., and Teartulakarn, S., (2006). White Blood Cell Classification based on the combination of Eigen Cell and Parametric Feature Detection. *Industrial Electronics and Applications, Singapore, IEEE*. 1-4.
- Zamani, F & Safabakhsh, R., (2006). An unsupervised GVF Snake Approach for White Blood Cell Segmentation based on Nucleus. *8th International Conference on Signal Processing, ICSP Guilin, China*.
- Zeng, Y., Chen, W., Saramas, D. and Peng, Q., (2008). Topology Cuts: A Novel Min-Cut/Max-Flow Algorithm for Topology Preserving Segmentation in N-D Images. *Journal Computer Vision and Image Understanding 112(1)*. 81-90.
- Zhang, G. Y., Liu, G. Z., Zhu, H. and Qiu, B., (2010). Ore image thresholding using bi-neighbourhood Otsu's approach. *Electronics Letters Vol. 46. (No. 25)*. 1666 – 1668.

Zhang, Y. and Le Beau M. M., (2011) *Cytogenetics in Acute Myeloid Leukemia*.
Retreived from <http://www.uptodate.com/contents/cytogenetics-in-acute-myeloid-leukemia>.

Appendix A – Ethical Letter

School of Information Systems, Computing and Mathematics
David Gilbert, Head of School, Professor of Computing
Jens Kujala, Head of Information Systems and Computing, Professor of Computing
Julia Knapton, Head of Mathematical Science, Professor of Applied Mathematics

Brunel
UNIVERSITY
WEST LONDON

Brunel University, Uxbridge,
Middlesex UB8 3PH, UK
Telephone: +44(0) 1895 274000
Fax: +44(0) 1895 201000
Email: Academic.Privacy@brunel.ac.uk
Law@brunel.ac.uk
Ethics@brunel.ac.uk

Date: 06th February 2009

STATEMENT OF ETHICS APPROVAL

Proposer: Waideh Jemal

Title: Detecting, Counting and Prediction of The White Blood Cells in Leukaemia

The school's research ethics committee has considered the proposal recently submitted by you. Acting under delegated authority, the committee is satisfied that there is no objection on ethical grounds to the proposed study. Approval is given on the understanding that you will adhere to the terms agreed with participants and to inform the committee of any change of plans in relation to the information provided in the application form.

Yours sincerely,



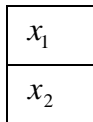
**Dr. Annette Payne Chair of the Research Ethics Committee
SISCM**

Appendix B – Proof for Cellular Automata

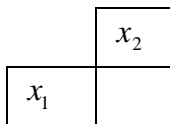
Below, the positions in the image is horizontal



The above position is not limited, but it can also rearrange as below

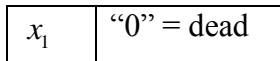


or

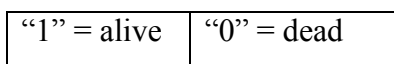


Assume $x_1 \geq x_2$ without loss of generality, since an image can be reused.

If $x_1 > x_2$ where



then point with x_1 change after point with x_2 iteration survived to where

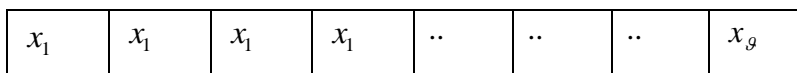


The state therefore $x_1 = x_2 + 1$ since it will change next iteration.

Any two points

| | |
|-------|-------|
| x_1 | x_2 |
|-------|-------|

Either to have $x_1 = x_2$ or $x_1 = x_2 + 1$



The shortest distance will be when no x_i equals any other $x_1 \neq x_2 \neq x_3 \neq x_4 \neq \dots 0$

∴ if they are not equal

$$x_1 = x_2 + 1$$

$$x_2 = x_3 + 1$$

$$x_3 = x_4 + 1$$

...

...

...

$$x_g = 0$$

∴ $x_1 = g$ (Number of pixel as radius)

If κ_0 all ones, algorithm near to end, none = “died”

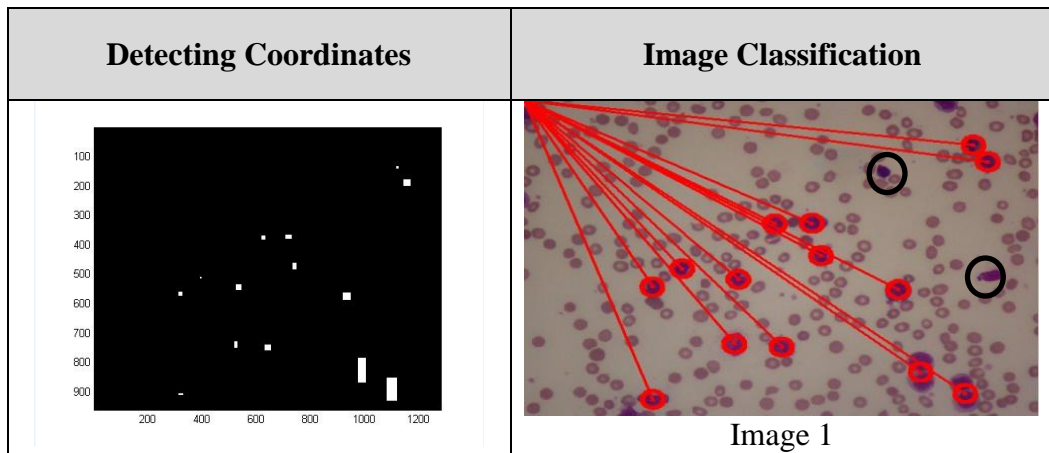
If κ_1 all zero, algorithm terminate with $\mathfrak{h} = 0$

Worst case scenario is when only one pixel is Dead in κ , worst Run time for g iterations.

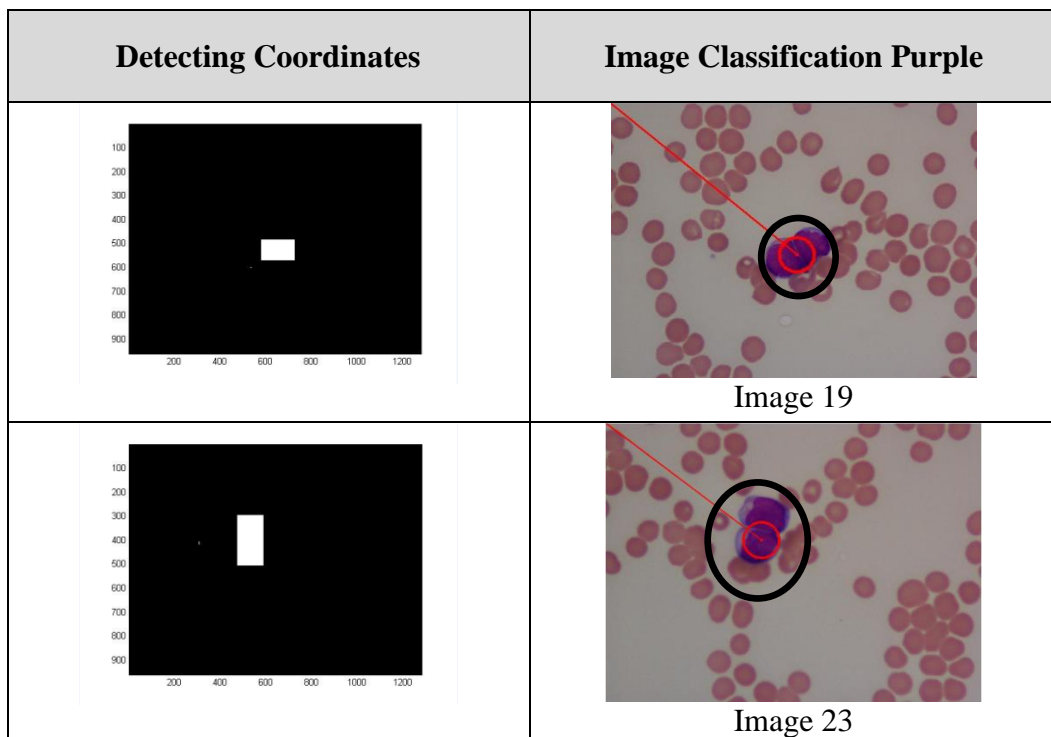
Appendix C – Full results in Detecting Coordinates

M1 Subtypes


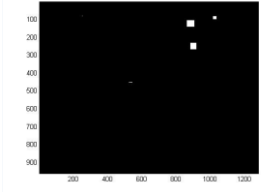
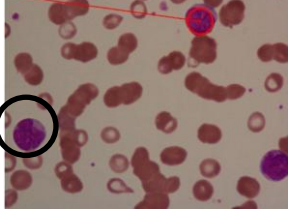
a) Missing target blast cells



b) Overlapping blast cells

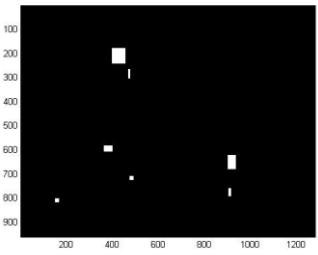
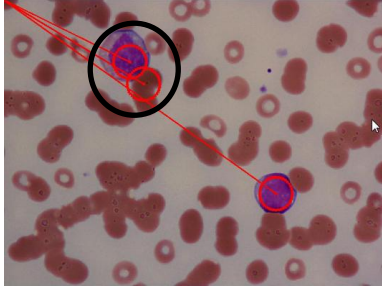


c) Do not capture during the detecting coordinates

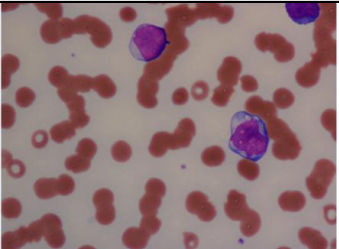
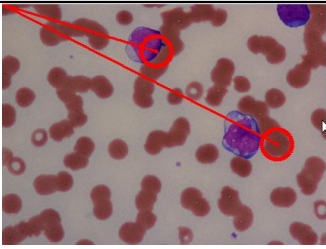
| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|---|
|  |  |  <p data-bbox="1129 685 1254 714">Image 47</p> |

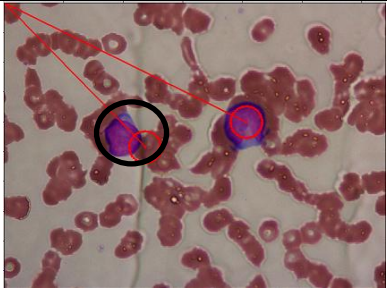
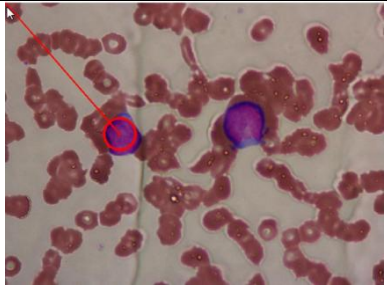
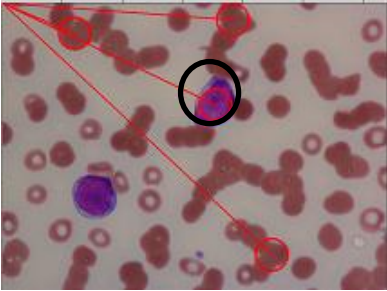
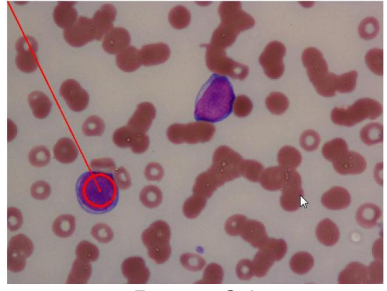
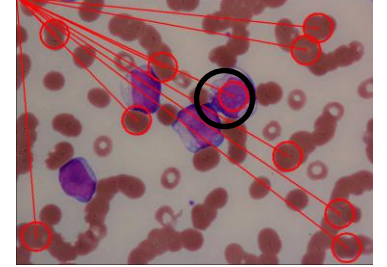
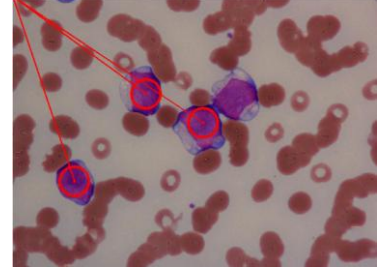
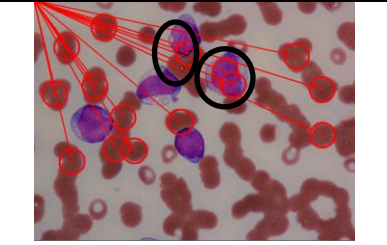
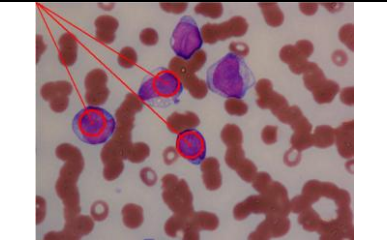
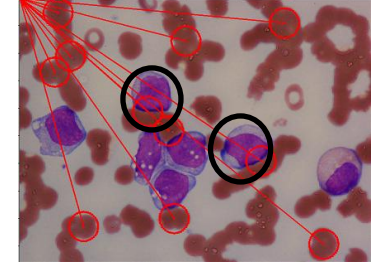
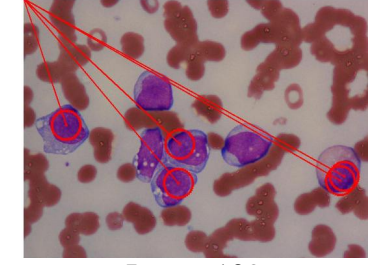
M2 Subtypes

a) Incorrect Classification to Pink from detecting coordinates

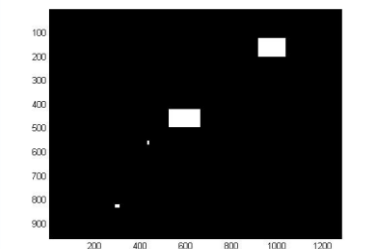
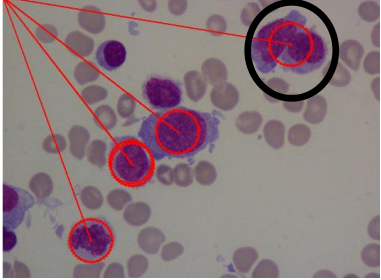
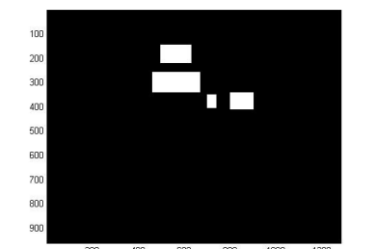
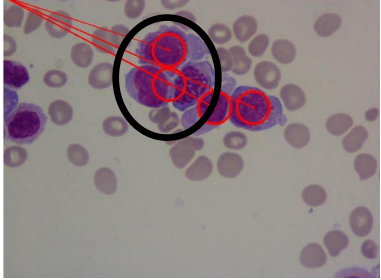
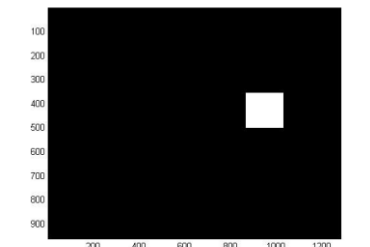
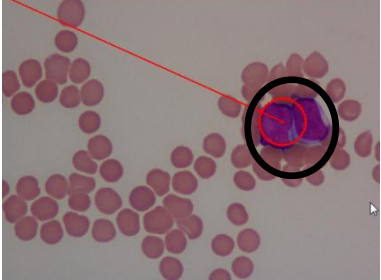
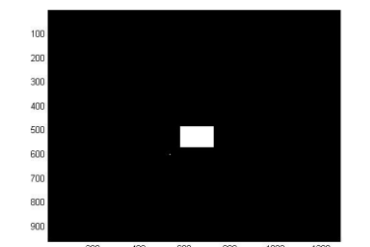
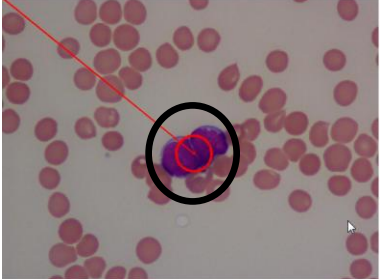
| Detecting Coordinates | Image Classification Purple |
|---|---|
|  |  <p data-bbox="1027 1317 1168 1350">Image 107</p> |

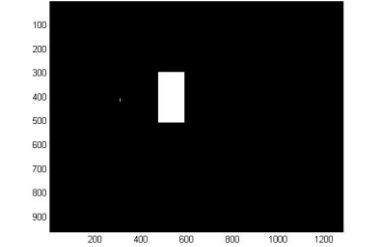
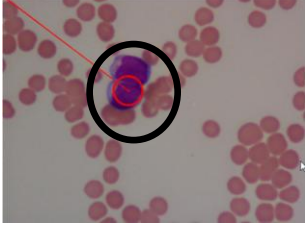
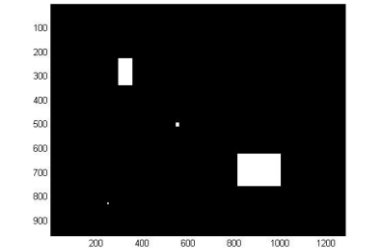
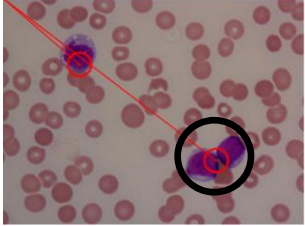
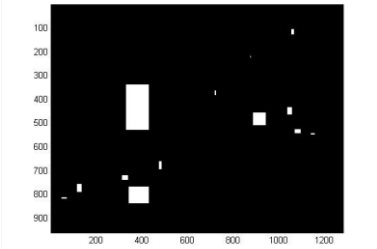
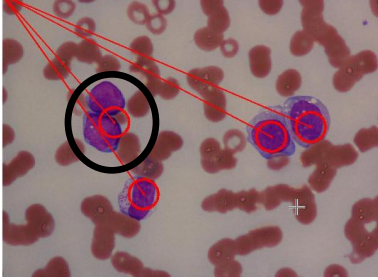

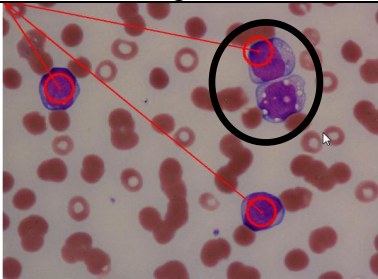
b) Wrong Classified to Pink

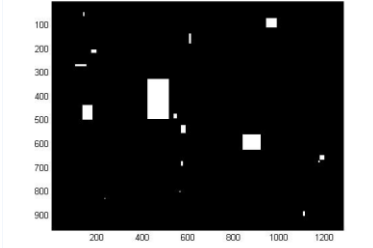
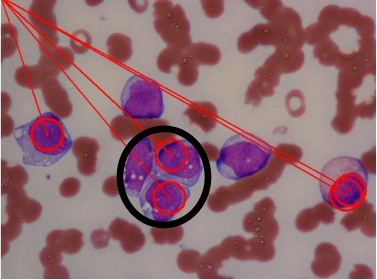
| Real Images | Image Classification |
|---|---|
|  <p data-bbox="453 1821 719 1854">M2 – Image no. 113</p> |  <p data-bbox="979 1821 1246 1854">M2 – Image no. 113</p> |

| Image Classification Pink | Image Classification Purple |
|---|---|
|  <p data-bbox="512 680 639 719">Image 80</p> |  <p data-bbox="1038 680 1166 719">Image 80</p> |
|  |  <p data-bbox="1038 999 1166 1043">Image 85</p> |
|  |  <p data-bbox="1038 1308 1166 1346">Image 97</p> |
|  |  <p data-bbox="1038 1581 1166 1626">Image 108</p> |
|  |  <p data-bbox="1038 1890 1166 1919">Image 109</p> |


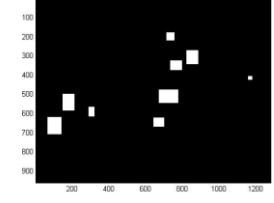
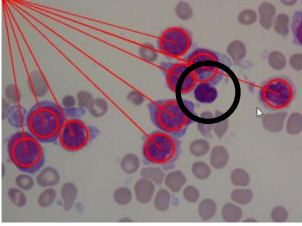

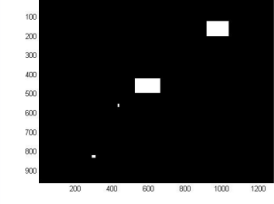
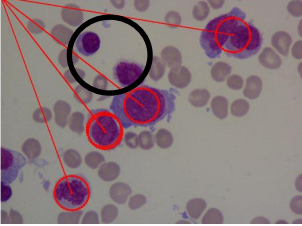

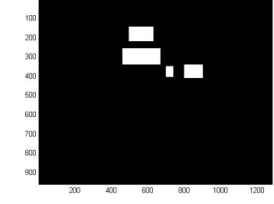
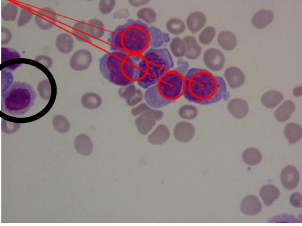
c) Overlapping blast cells

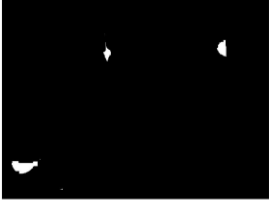
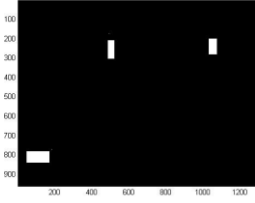
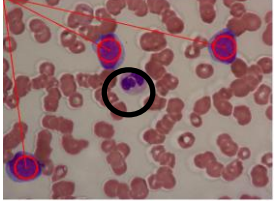

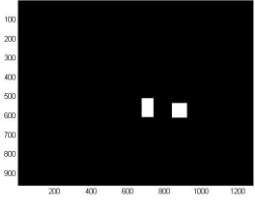
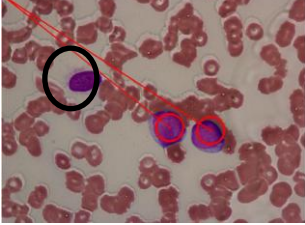

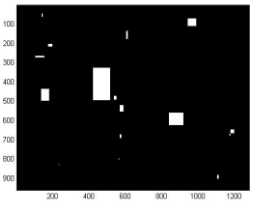
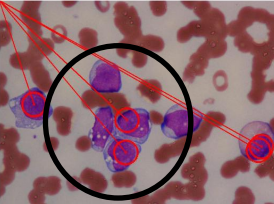

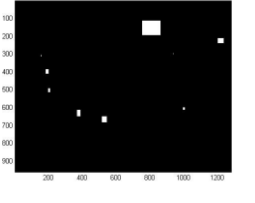
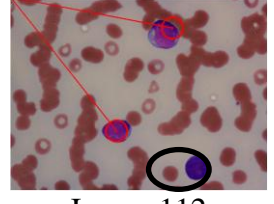

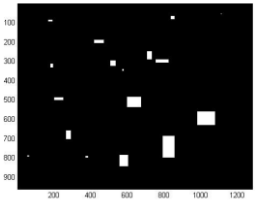
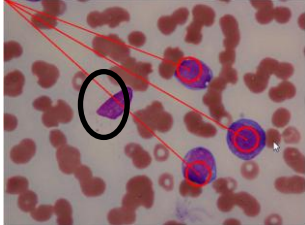
| Detecting Coordinates | Image Classification |
|---|--|
|  <p>Image 2</p> |  <p>Image 2</p> |
|  <p>Image 4</p> |  <p>Image 4</p> |
|  <p>Image 8</p> |  <p>Image 8</p> |
|  <p>Image 10</p> |  <p>Image 10</p> |


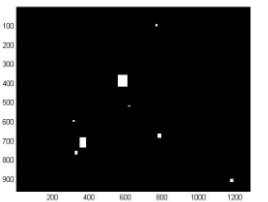
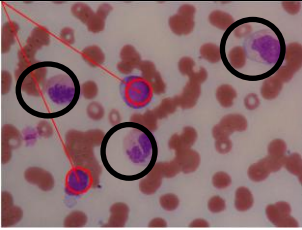

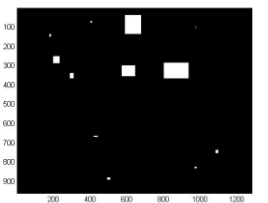
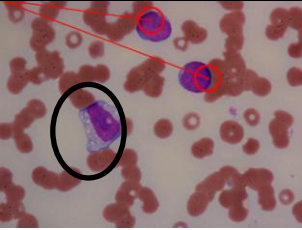
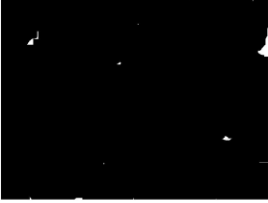
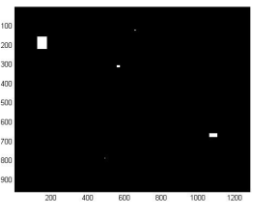
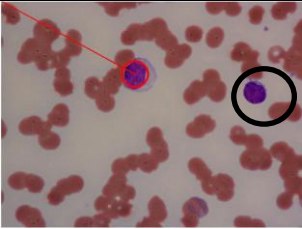

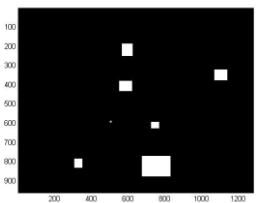
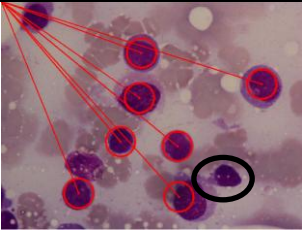

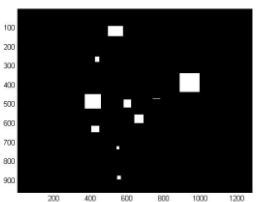

| Detecting Coordinates | Image Classification |
|---|---|
|  |  <p data-bbox="1023 712 1149 750">Image 15</p> |
|  |  <p data-bbox="1023 996 1149 1034">Image 17</p> |
|  |  <p data-bbox="1013 1335 1158 1373">Image 103</p> |
|  |  <p data-bbox="1013 1646 1158 1684">Image 105</p> |


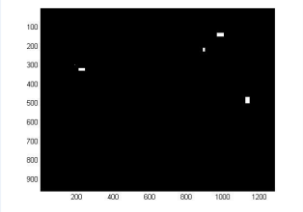
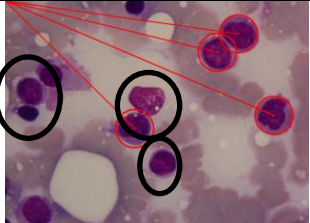
| Detecting Coordinates | Image Classification |
|---|---|
|  |  <p data-bbox="1021 721 1165 757">Image 109</p> |

d) Do not capture during the detecting coordinates

| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|--|
|  |  |  <p data-bbox="1141 1227 1252 1258">Image 1</p> |
|  |  |  <p data-bbox="1141 1491 1252 1523">Image 2</p> |
|  |  |  <p data-bbox="1141 1756 1252 1789">Image 4</p> |

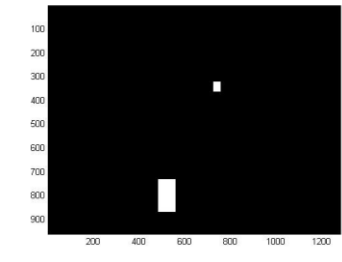
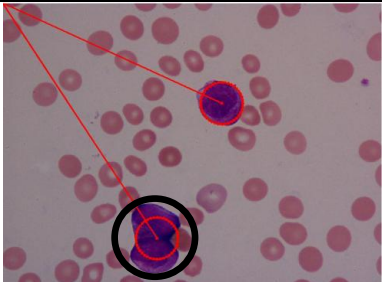
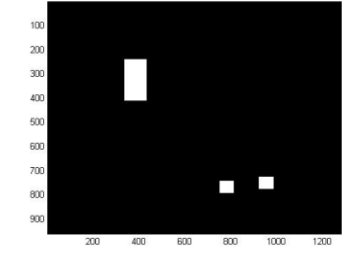
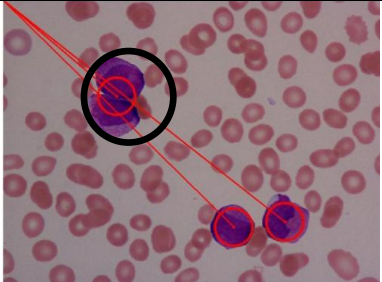
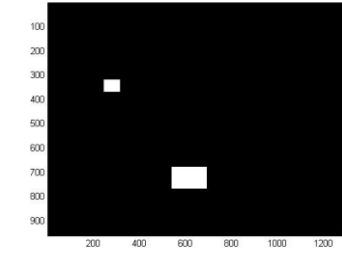
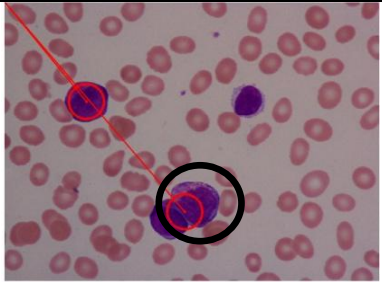
| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|--|
|  |  |  Image 73 |
|  |  |  Image 78 |
|  |  |  Image 109 |
|  |  |  Image 112 |
|  |  |  Image 115 |

| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|--|
|  |  |  Image 118 |
|  |  |  Image 119 |
|  |  |  Image 122 |
|  |  |  Image 127 |
|  |  |  Image 128 |


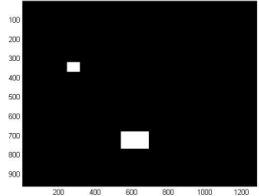
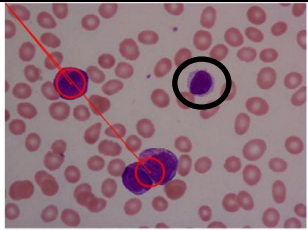
| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|--|
|  |  |  <p data-bbox="1109 622 1252 656">Image 129</p> |

M3 Subtypes

a) Overlapping blast cells

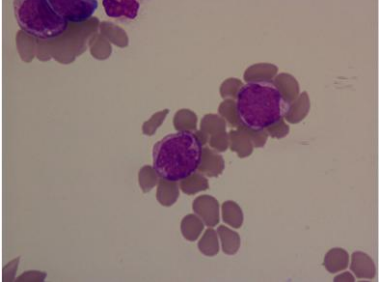
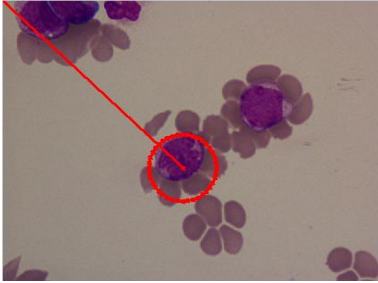
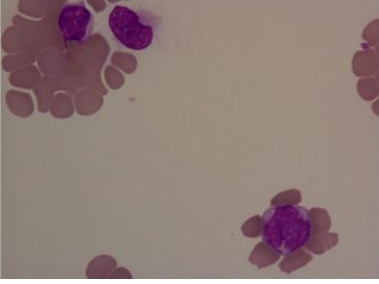
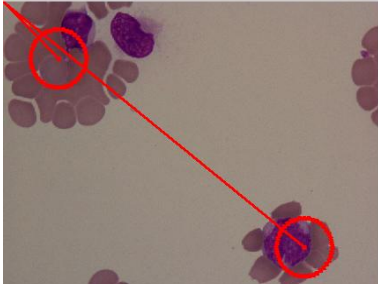
| Detecting coordinates | Image Classification |
|---|--|
|  |  <p data-bbox="1029 1227 1157 1261">Image 21</p> |
|  |  <p data-bbox="1029 1545 1157 1579">Image 75</p> |
|  |  <p data-bbox="1029 1868 1157 1901">Image 88</p> |

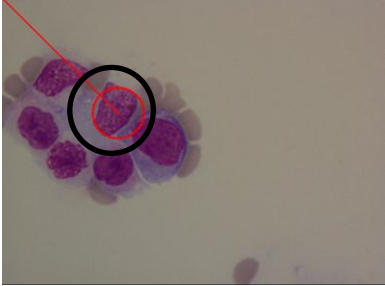
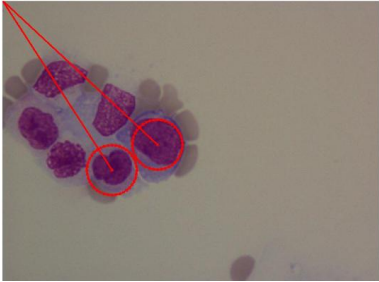
b) Do not capture during the detecting coordinates

| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|---|
|  |  |  <p data-bbox="1129 703 1254 736">Image 88</p> |

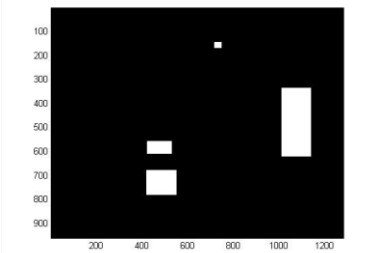
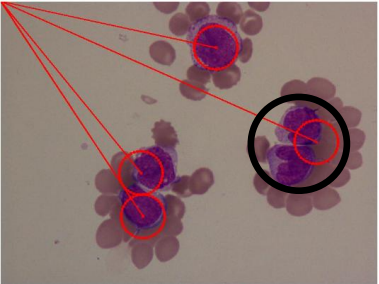
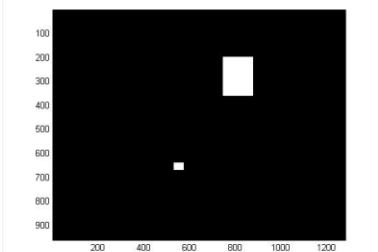
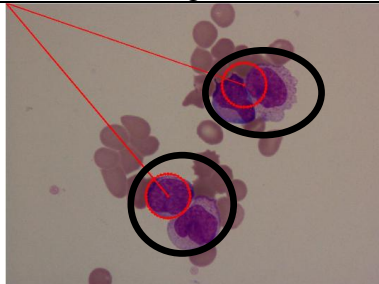
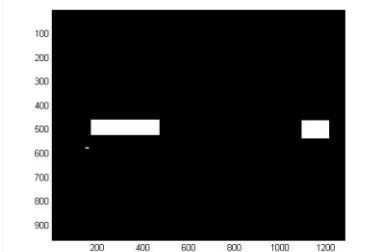
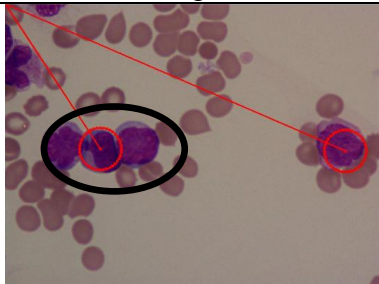
M5 Subtypes

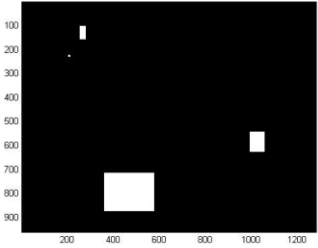
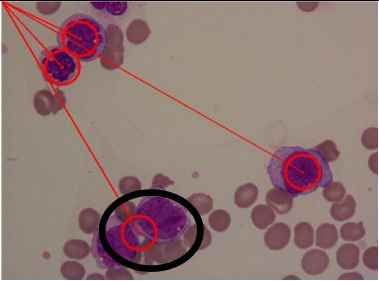
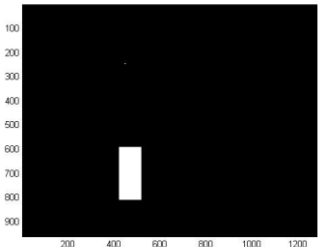
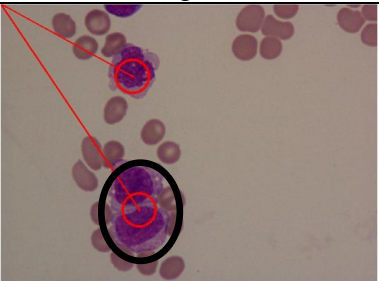
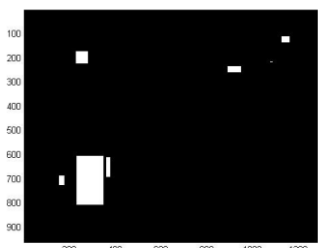
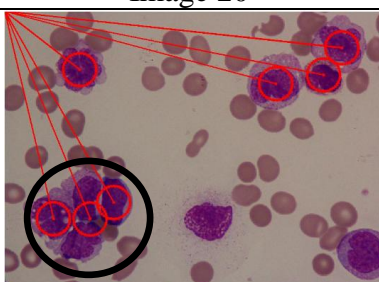
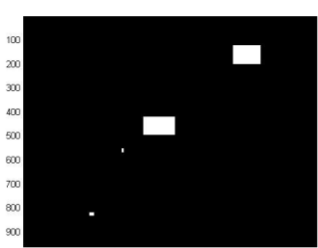
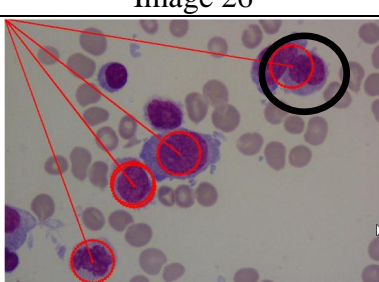
a) Wrong classification to Pink

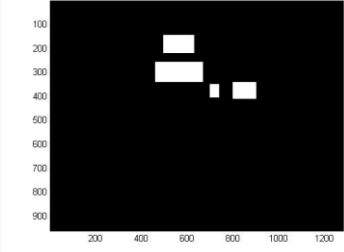
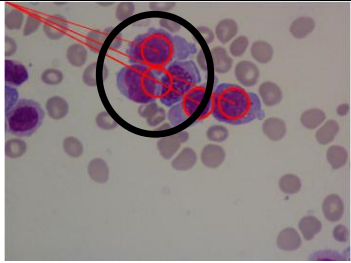
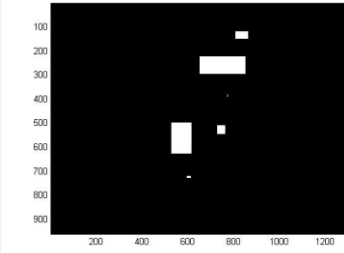
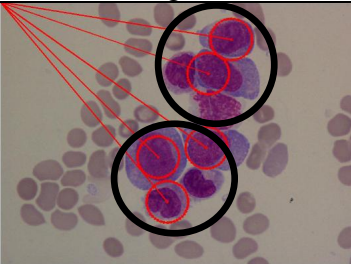
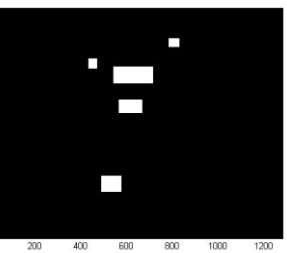
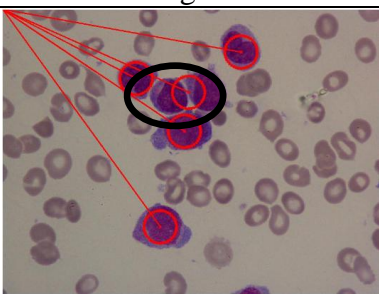
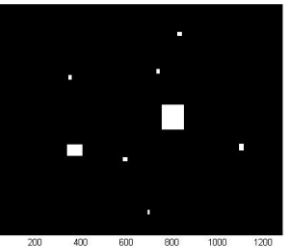
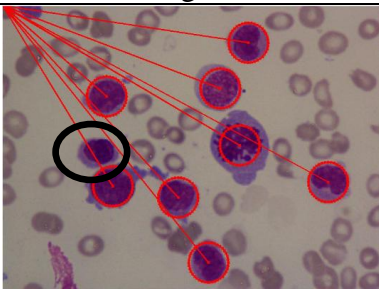
| Real Images | Image Classification |
|---|---|
|  <p data-bbox="464 1359 699 1395">M5 – Image no. 1</p> |  <p data-bbox="997 1359 1232 1395">M5 – Image no. 1</p> |
|  <p data-bbox="464 1682 699 1718">M5 – Image no. 2</p> |  <p data-bbox="997 1682 1232 1718">M5 – Image no. 2</p> |

| Image Classification Pink | Image Classification Purple |
|---|--|
|  |  <p data-bbox="1050 712 1174 743">Image 35</p> |

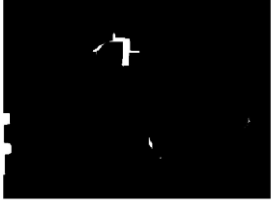
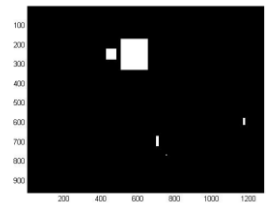
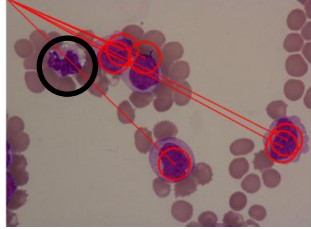
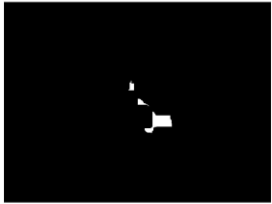
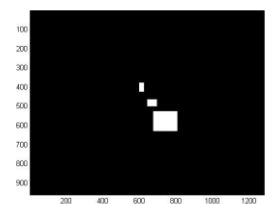
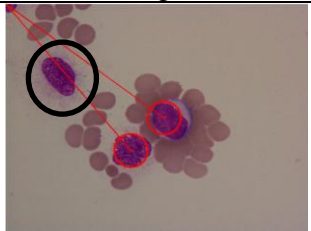
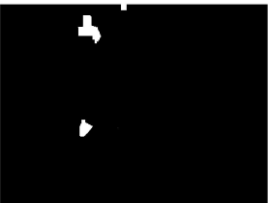
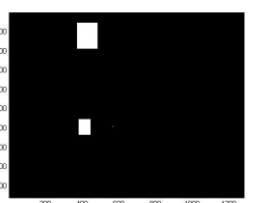
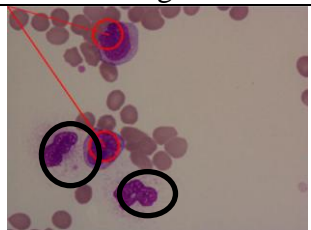
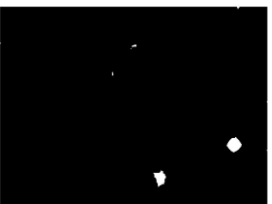
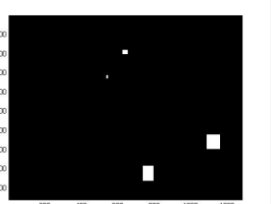
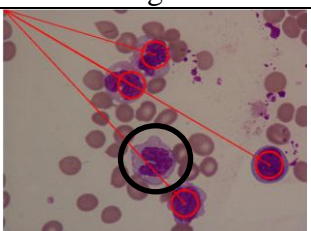

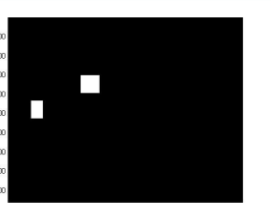
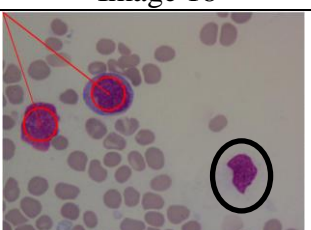
b) Overlapping circles

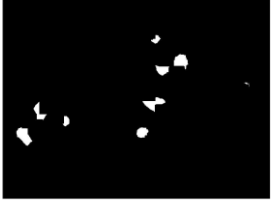
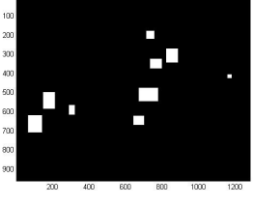
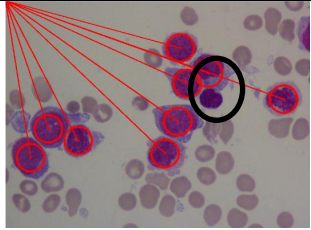
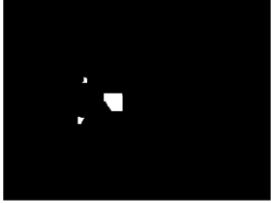
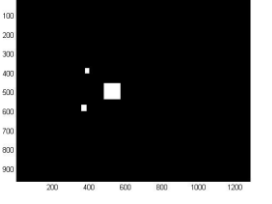
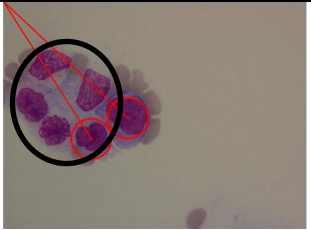

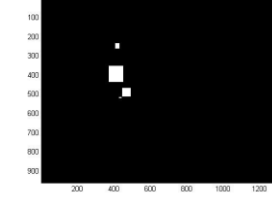
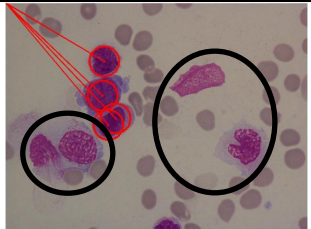

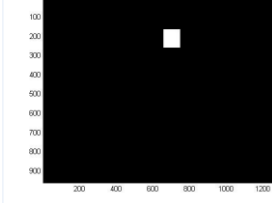
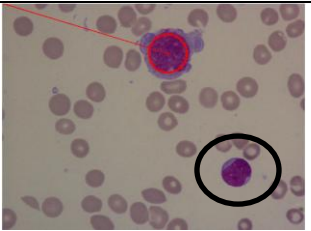
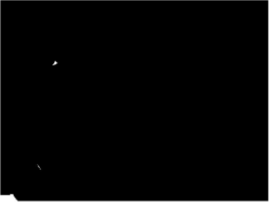
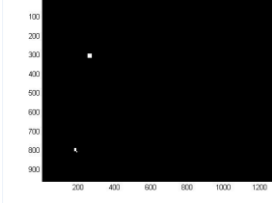
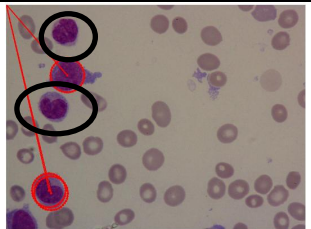
| Detecting coordinates | Image Classification |
|---|--|
|  |  <p data-bbox="1034 1236 1145 1267">Image 3</p> |
|  |  <p data-bbox="1034 1554 1145 1585">Image 6</p> |
|  |  <p data-bbox="1034 1872 1145 1904">Image 14</p> |


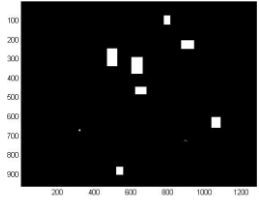
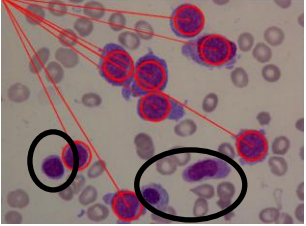

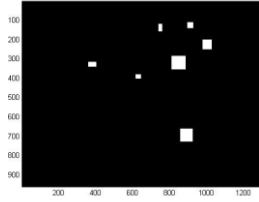
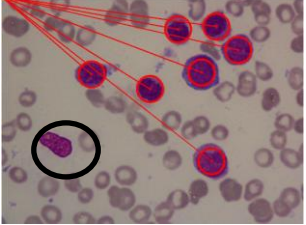
| Detecting coordinates | Image Classification |
|--|---|
|  <p>A binary mask on a black background with white shapes. The y-axis is labeled from 100 to 900, and the x-axis from 200 to 1200. There are three distinct white shapes: a small square at approximately (250, 150), a larger square at (450, 750), and another small square at (1050, 650).</p> |  <p>Image 22: A microscopic image of cells. A large black circle highlights a cell in the lower-left quadrant. Three red circles highlight other cells in the upper-left and middle-right areas. Red lines connect the corners of the black circle to the corners of the red circles, indicating a bounding box relationship.</p> <p>Image 22</p> |
|  <p>A binary mask on a black background with a single white vertical rectangle. The y-axis is labeled from 100 to 900, and the x-axis from 200 to 1200. The rectangle is centered vertically at approximately x=500.</p> |  <p>Image 20: A microscopic image of cells. A large black circle highlights a cell in the lower-left quadrant. One red circle highlights a cell in the upper-left area. Red lines connect the corners of the black circle to the corners of the red circle.</p> <p>Image 20</p> |
|  <p>A binary mask on a black background with several white shapes. The y-axis is labeled from 100 to 900, and the x-axis from 200 to 1200. There are several small white shapes scattered across the image, including a cluster of three shapes in the lower-left and several individual shapes in the upper-left and middle-right.</p> |  <p>Image 26: A microscopic image of cells. A large black circle highlights a cell in the lower-left quadrant. Five red circles highlight other cells in the upper-left and middle-right areas. Red lines connect the corners of the black circle to the corners of the red circles.</p> <p>Image 26</p> |
|  <p>A binary mask on a black background with three white shapes. The y-axis is labeled from 100 to 900, and the x-axis from 200 to 1200. There is a small square at (1050, 200), a larger square at (550, 550), and another small square at (1050, 800).</p> |  <p>Image 32: A microscopic image of cells. A large black circle highlights a cell in the upper-right quadrant. Three red circles highlight other cells in the upper-left and middle-left areas. Red lines connect the corners of the black circle to the corners of the red circles.</p> <p>Image 32</p> |

| Detecting coordinates | Image Classification |
|---|--|
|  <p>A coordinate detection plot for Image 33. The x-axis ranges from 0 to 1200 and the y-axis from 0 to 900. White rectangular regions are highlighted on a black background, indicating detected coordinates.</p> |  <p>Image 33: A microscopic image of cells with purple nuclei. Two cells are circled in black, and red lines connect them to a common point at the top left of the image.</p> <p>Image 33</p> |
|  <p>A coordinate detection plot for Image 36. The x-axis ranges from 0 to 1200 and the y-axis from 0 to 900. White rectangular regions are highlighted on a black background, indicating detected coordinates.</p> |  <p>Image 36: A microscopic image of cells with purple nuclei. Two groups of cells are circled in black, and red lines connect them to a common point at the top left of the image.</p> <p>Image 36</p> |
|  <p>A coordinate detection plot for Image 43. The x-axis ranges from 0 to 1200 and the y-axis from 0 to 900. White rectangular regions are highlighted on a black background, indicating detected coordinates.</p> |  <p>Image 43: A microscopic image of cells with purple nuclei. Two cells are circled in black, and red lines connect them to a common point at the top left of the image.</p> <p>Image 43</p> |
|  <p>A coordinate detection plot for Image 51. The x-axis ranges from 0 to 1200 and the y-axis from 0 to 900. White rectangular regions are highlighted on a black background, indicating detected coordinates.</p> |  <p>Image 51: A microscopic image of cells with purple nuclei. One cell is circled in black, and red lines connect several other cells to a common point at the top left of the image.</p> <p>Image 51</p> |

c) Do not capture during the detecting coordinates

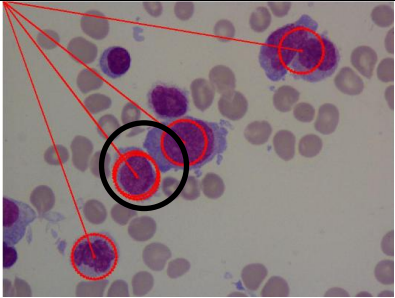
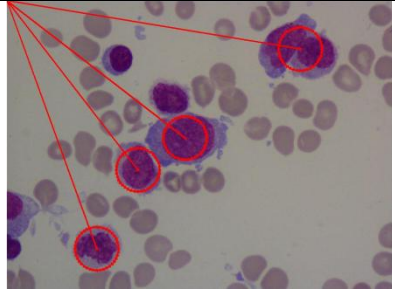
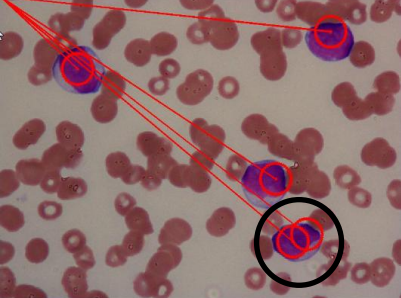
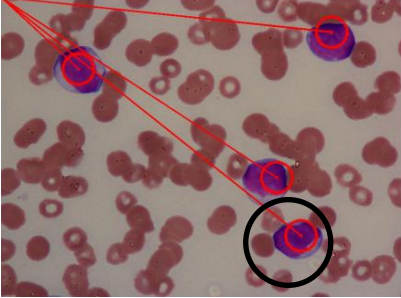
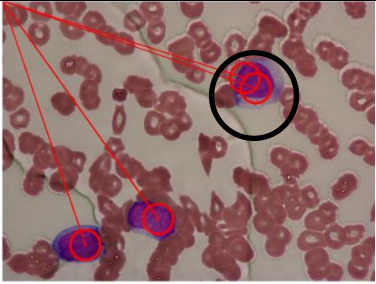
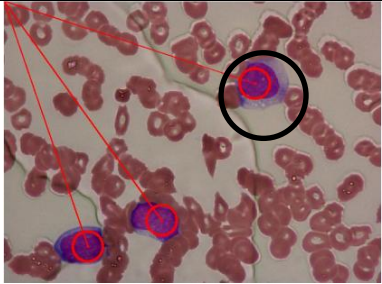
| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|---|
|  |  |  <p data-bbox="1142 745 1246 779">Image 4</p> |
|  |  |  <p data-bbox="1142 1008 1246 1041">Image 5</p> |
|  |  |  <p data-bbox="1142 1270 1246 1303">Image 11</p> |
|  |  |  <p data-bbox="1142 1532 1246 1565">Image 18</p> |
|  |  |  <p data-bbox="1142 1794 1246 1827">Image 30</p> |

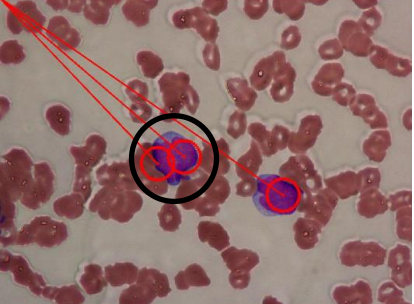
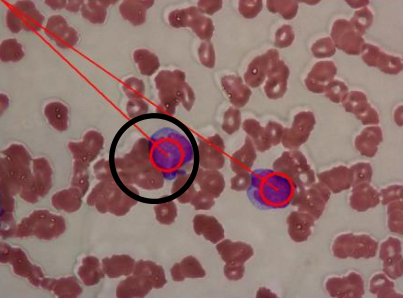
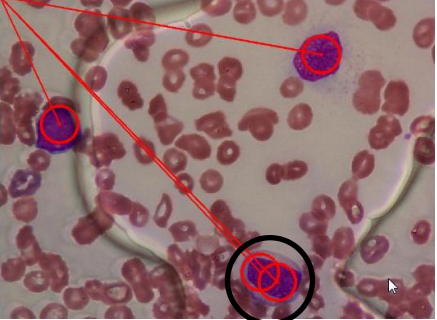
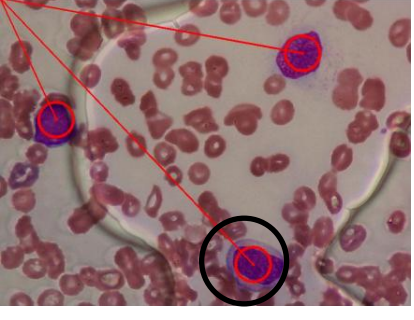
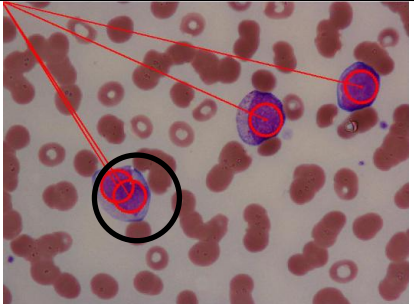
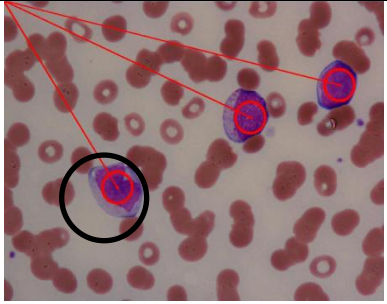
| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|---|
|  |  |  <p data-bbox="1129 712 1257 748">Image 31</p> |
|  |  |  <p data-bbox="1129 976 1257 1012">Image 35</p> |
|  |  |  <p data-bbox="1129 1240 1257 1276">Image 37</p> |
|  |  |  <p data-bbox="1129 1505 1257 1541">Image 41</p> |
|  |  |  <p data-bbox="1129 1769 1257 1805">Image 42</p> |

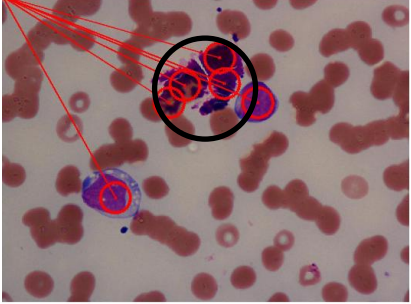
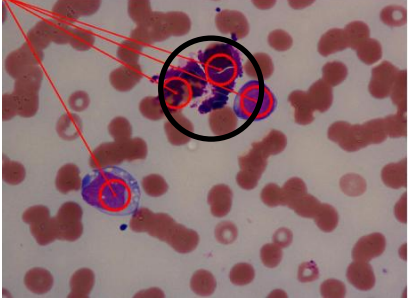
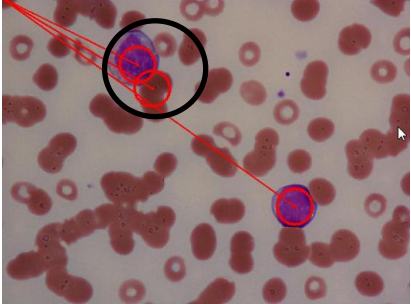
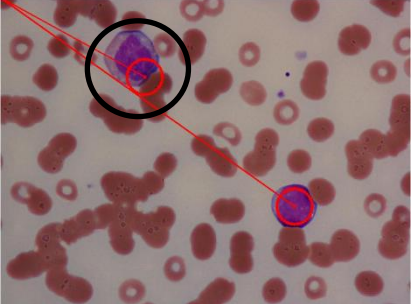
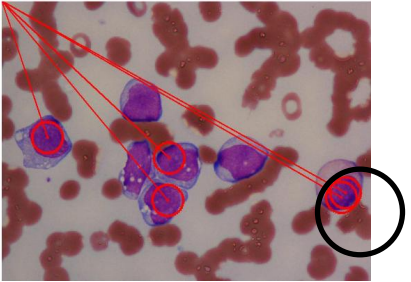
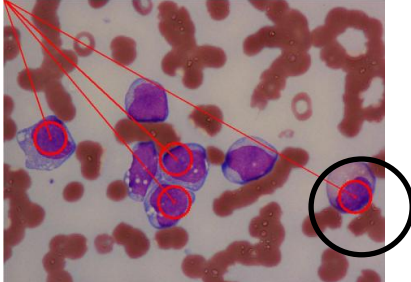
| CA Filtering | Detecting Coordinates | Image Classification |
|---|---|--|
|  |  |  <p data-bbox="1134 712 1257 748">Image 50</p> |
|  |  |  <p data-bbox="1134 976 1257 1012">Image 52</p> |

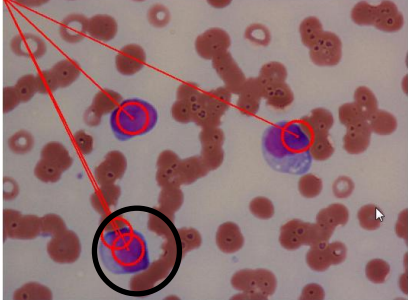
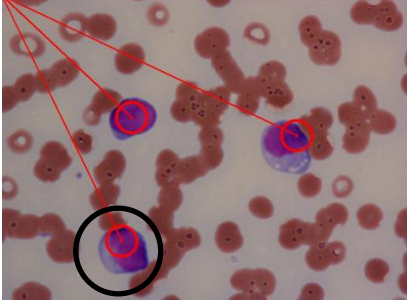
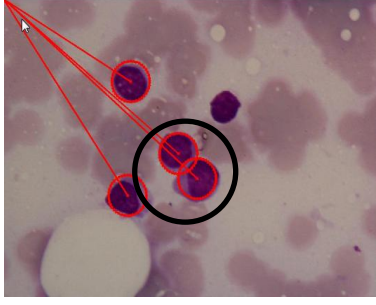
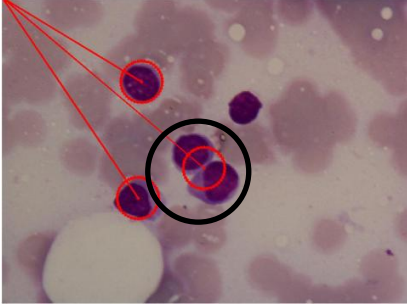
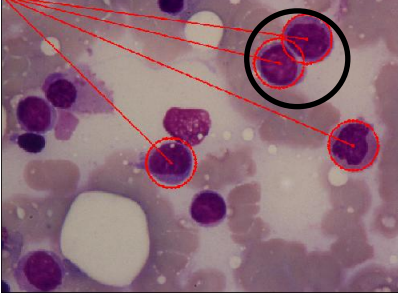
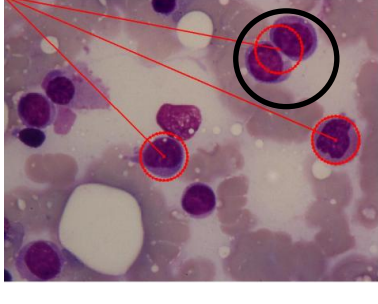
Appendix D – Full results in Duplicate Coordinates

M2 Subtypes

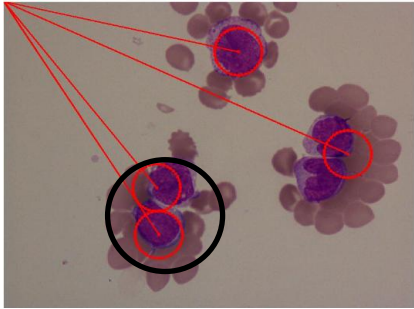
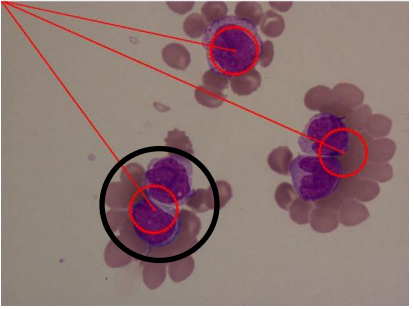
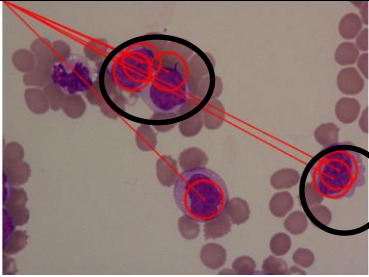
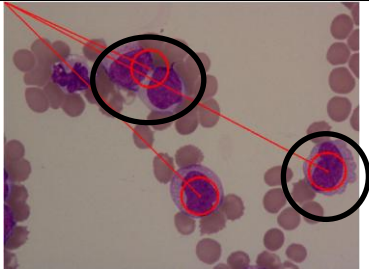
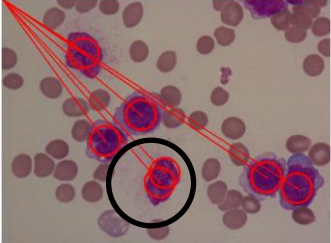
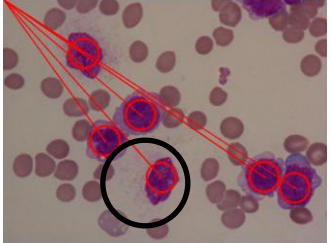
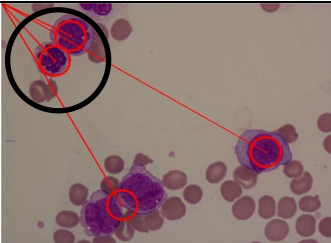
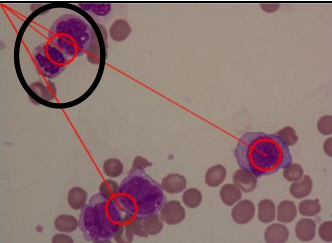
| Before Merge | After Merge |
|--|---|
|  <p data-bbox="496 898 643 936">Image 2(a)</p> |  <p data-bbox="1026 898 1173 936">Image 2(b)</p> |
|  <p data-bbox="488 1279 651 1317">Image 69(a)</p> |  <p data-bbox="1018 1279 1181 1317">Image 69(b)</p> |
|  <p data-bbox="488 1637 651 1675">Image 77(a)</p> |  <p data-bbox="1018 1637 1181 1675">Image 77(b)</p> |

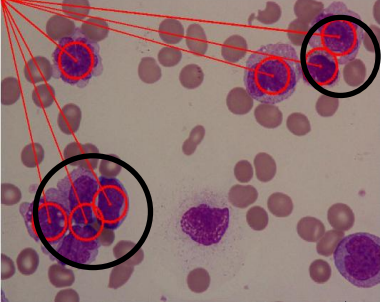
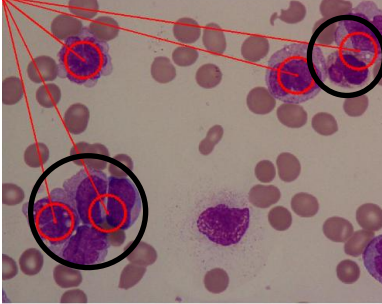
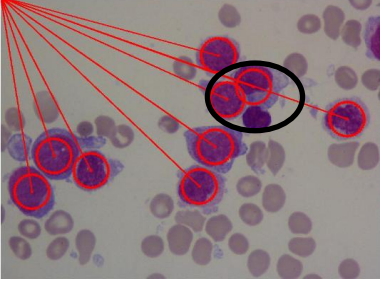
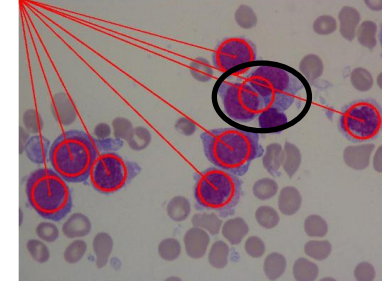

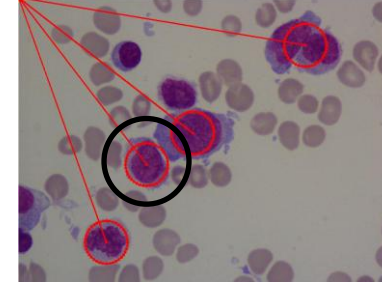
| Before Merge | After Merge |
|--|---|
|  <p data-bbox="491 786 651 819">Image 81(a)</p> |  <p data-bbox="1027 786 1187 819">Image 81(b)</p> |
|  <p data-bbox="491 1182 651 1216">Image 84(a)</p> |  <p data-bbox="1027 1171 1187 1205">Image 84(b)</p> |
|  <p data-bbox="491 1559 651 1592">Image 94(a)</p> |  <p data-bbox="1027 1559 1187 1592">Image 94(b)</p> |

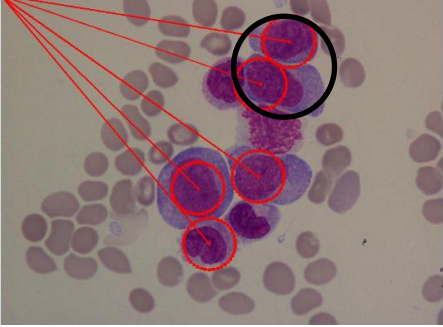
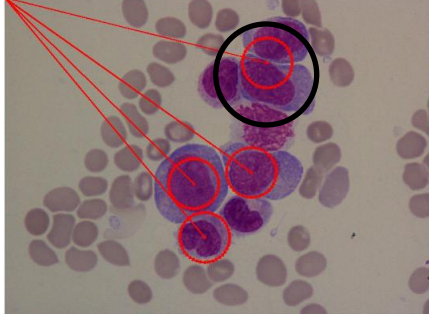
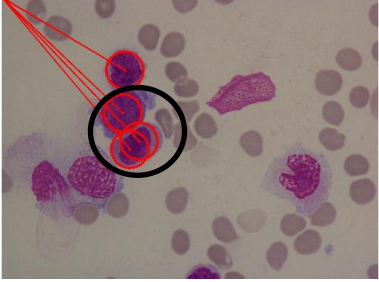
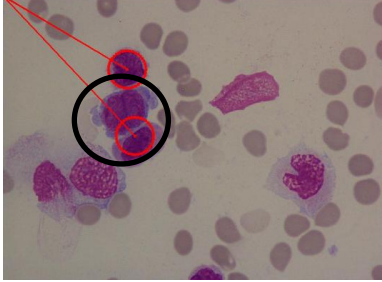
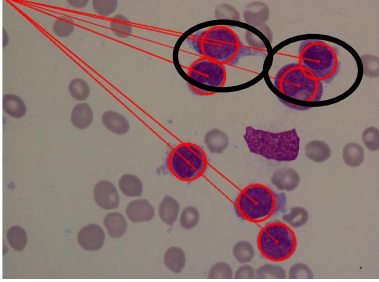
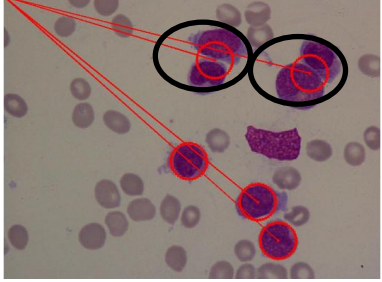
| Before Merge | After Merge |
|---|--|
|  <p data-bbox="480 786 655 819">Image 102(a)</p> |  <p data-bbox="1011 786 1187 819">Image 102(b)</p> |
|  <p data-bbox="480 1167 655 1200">Image 107(a)</p> |  <p data-bbox="1011 1167 1187 1200">Image 107(b)</p> |
|  <p data-bbox="480 1525 655 1559">Image 109(a)</p> |  <p data-bbox="1011 1525 1187 1559">Image 109(b)</p> |

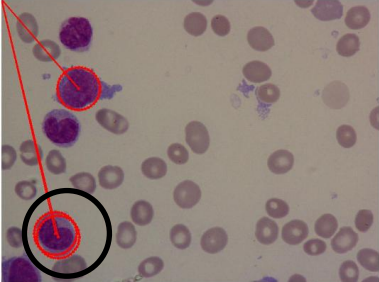
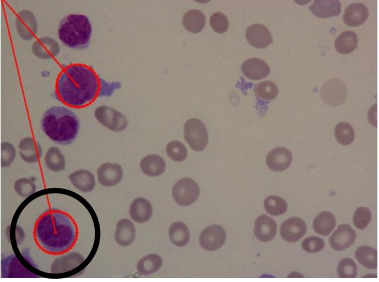
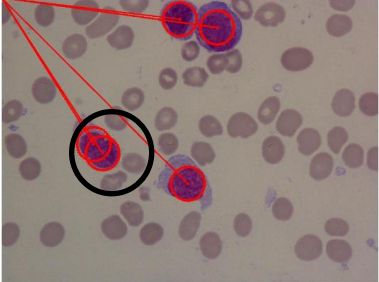
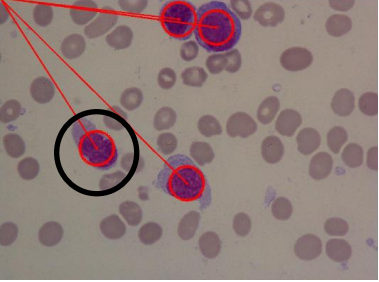
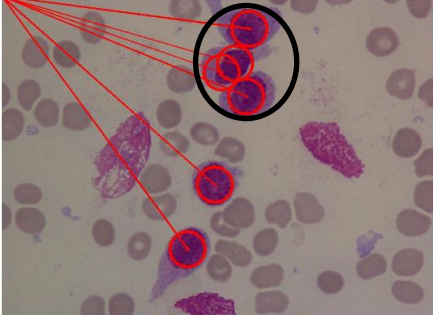
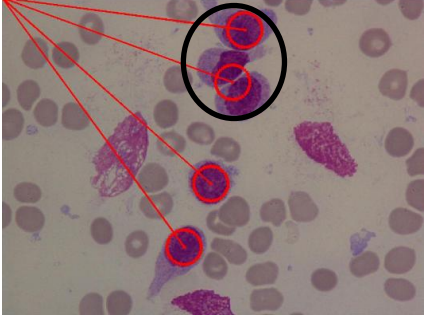
| Before Merge | After Merge |
|---|--|
|  <p data-bbox="475 779 655 817">Image 117(a)</p> |  <p data-bbox="1007 779 1187 817">Image 117(b)</p> |
|  <p data-bbox="475 1155 655 1193">Image 128(a)</p> |  <p data-bbox="1007 1155 1187 1193">Image 128(b)</p> |
|  <p data-bbox="475 1536 655 1574">Image 129(a)</p> |  <p data-bbox="1007 1536 1187 1574">Image 129(b)</p> |

M5 Subtypes

| Before Merge | After Merge |
|--|---|
|  <p>Image 3(a)</p> |  <p>Image 3(b)</p> |
|  <p>Image 4(a)</p> |  <p>Image 4(b)</p> |
|  <p>Image 13(a)</p> |  <p>Image 13(b)</p> |
|  <p>Image 22(a)</p> |  <p>Image 22(b)</p> |

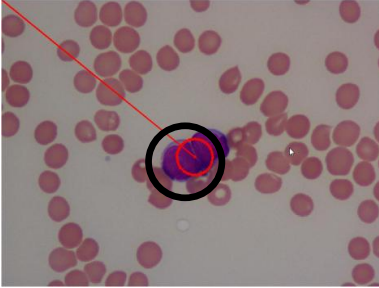
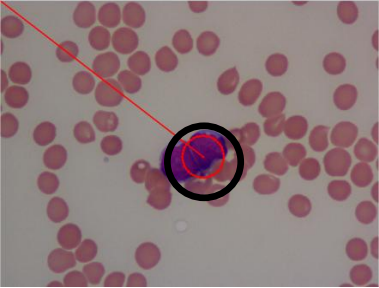
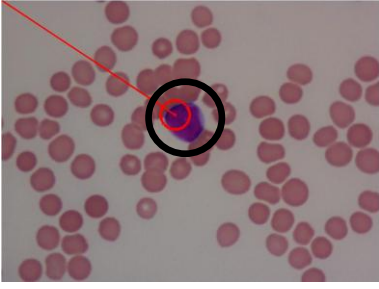
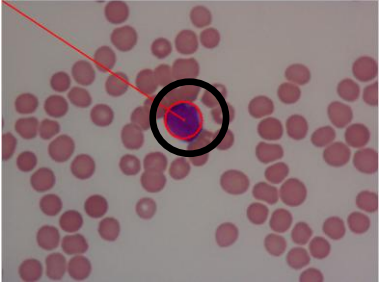
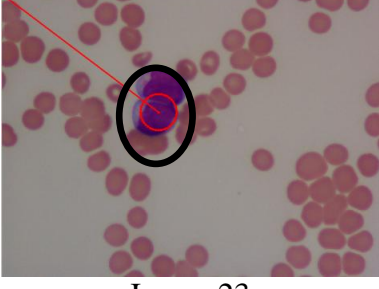
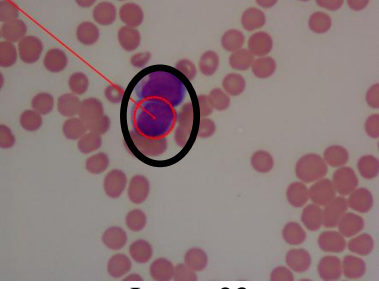
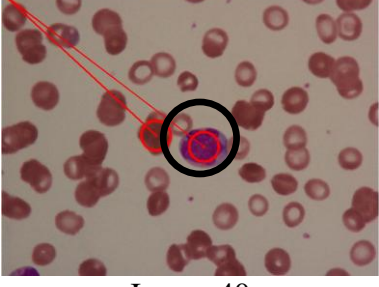
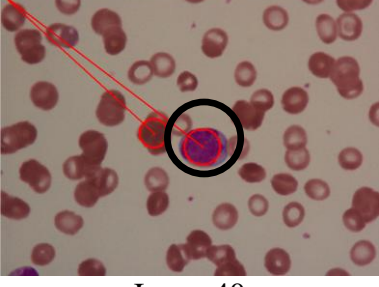
| Before Merge | After Merge |
|--|---|
|  <p data-bbox="491 779 651 817">Image 26(a)</p> |  <p data-bbox="1018 779 1177 817">Image 26(b)</p> |
|  <p data-bbox="491 1137 651 1176">Image 31(a)</p> |  <p data-bbox="1018 1137 1177 1176">Image 31(b)</p> |
|  <p data-bbox="491 1507 651 1545">Image 32(a)</p> |  <p data-bbox="1018 1507 1177 1545">Image 32(b)</p> |

| Before Merge | After Merge |
|--|---|
|  <p data-bbox="488 801 651 842">Image 36(a)</p> |  <p data-bbox="1018 792 1181 833">Image 36(b)</p> |
|  <p data-bbox="488 1160 651 1200">Image 37(a)</p> |  <p data-bbox="1018 1160 1181 1200">Image 37(b)</p> |
|  <p data-bbox="488 1518 651 1559">Image 40(a)</p> |  <p data-bbox="1018 1518 1181 1559">Image 40(b)</p> |

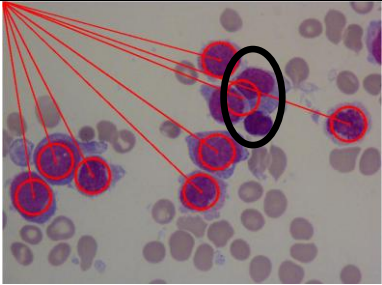
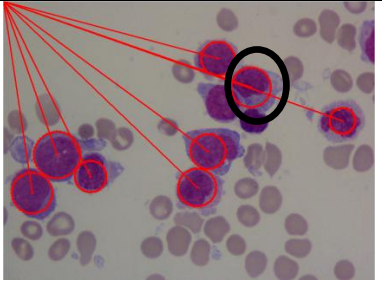
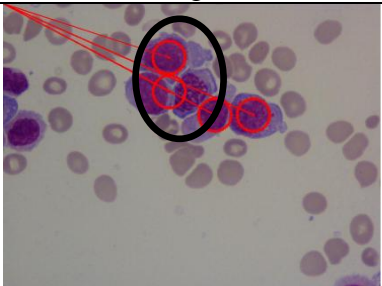
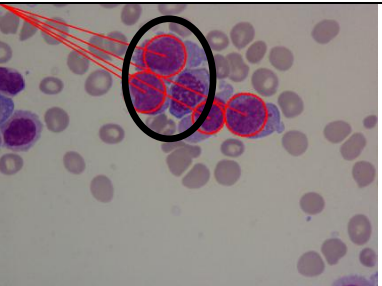
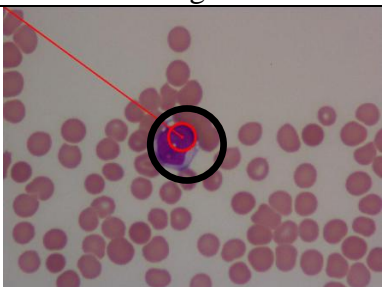
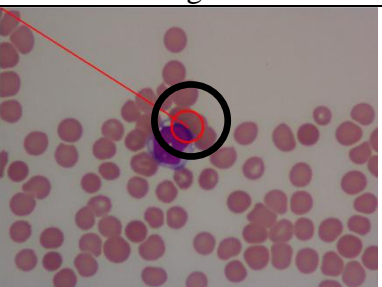
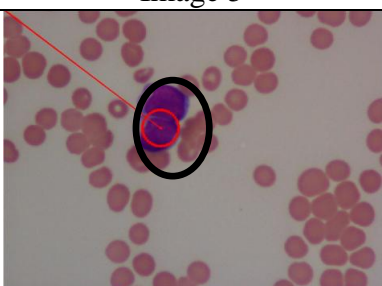
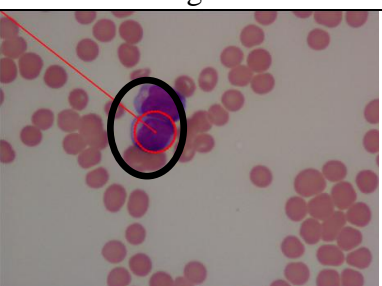
| Before Merge | After Merge |
|--|---|
|  <p data-bbox="491 763 647 797">Image 42(a)</p> |  <p data-bbox="1023 763 1179 797">Image 42(b)</p> |
|  <p data-bbox="491 1122 647 1155">Image 46(a)</p> |  <p data-bbox="1023 1122 1179 1155">Image 46(b)</p> |
|  <p data-bbox="491 1514 647 1547">Image 47(a)</p> |  <p data-bbox="1023 1514 1179 1547">Image 47(b)</p> |

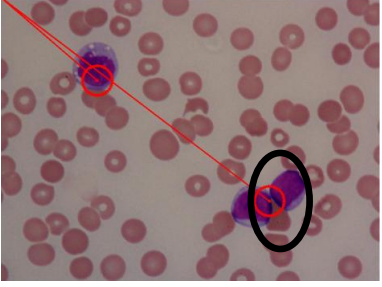
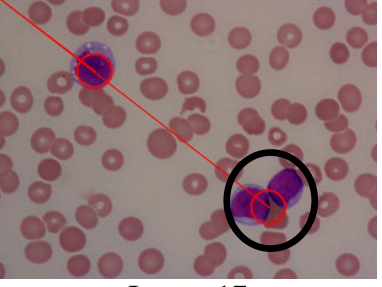
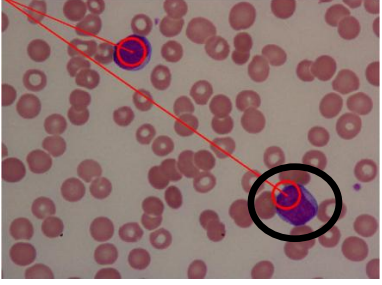
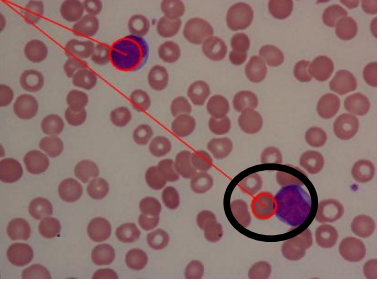
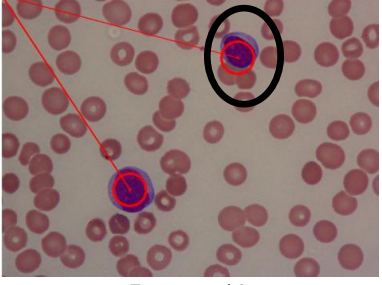
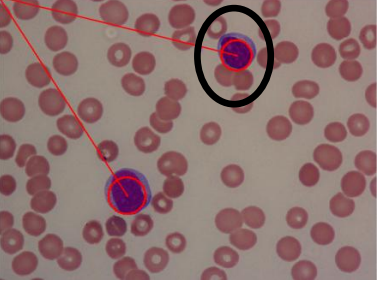
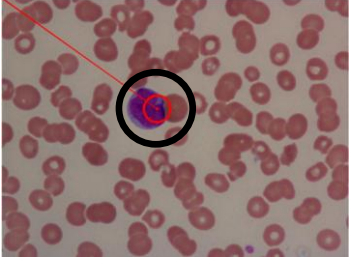
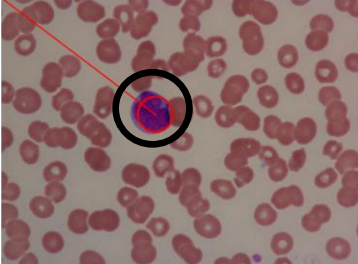
Appendix E – Full results in Heuristic Search

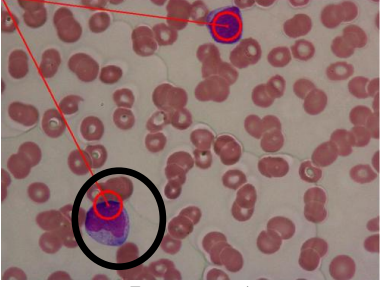
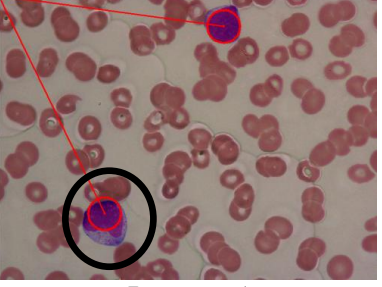
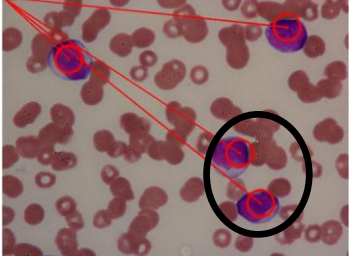
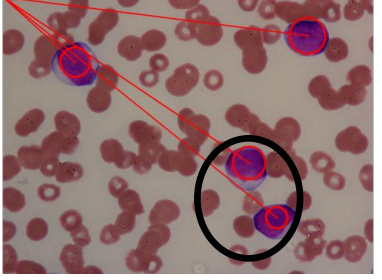
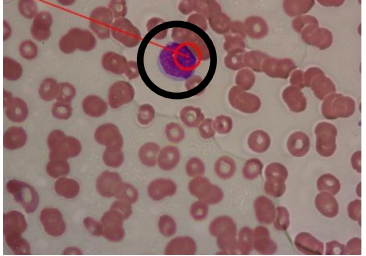

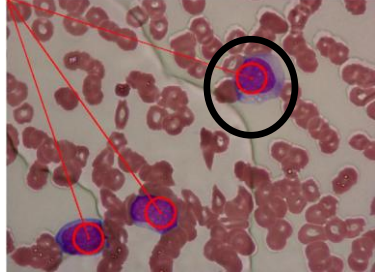
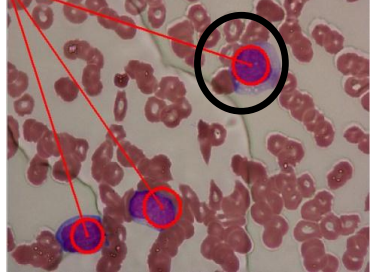
M1 Subtype

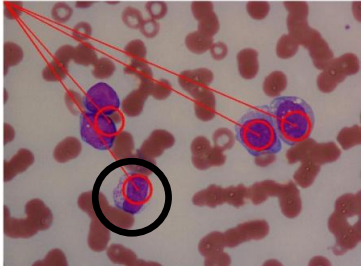
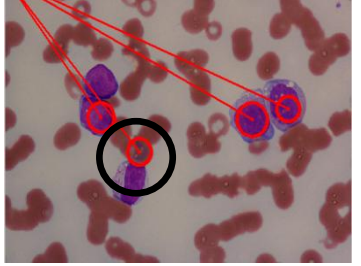
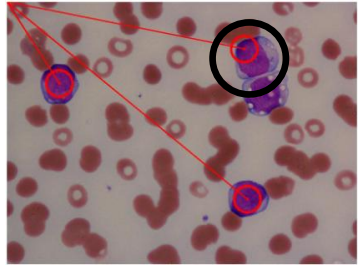
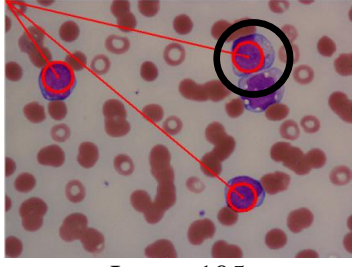
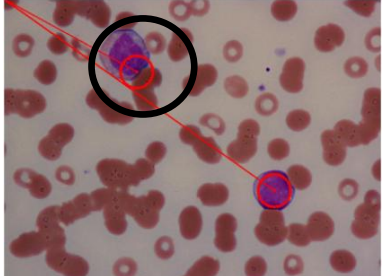
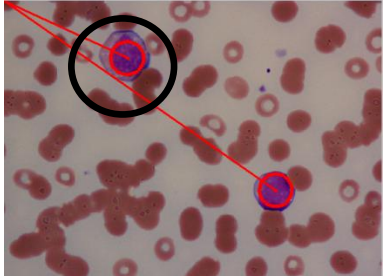
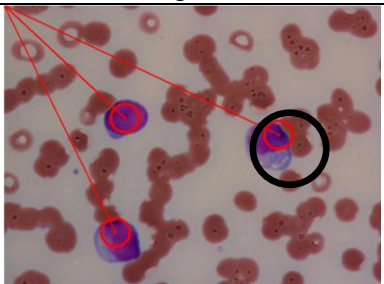
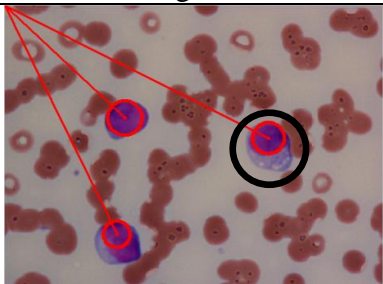
| Before Heuristic Search | After Heuristic Search |
|---|--|
|  <p data-bbox="504 857 628 887">Image 19</p> |  <p data-bbox="1031 857 1155 887">Image 19</p> |
|  <p data-bbox="504 1182 628 1211">Image 21</p> |  <p data-bbox="1031 1182 1155 1211">Image 21</p> |
|  <p data-bbox="504 1507 628 1536">Image 23</p> |  <p data-bbox="1031 1507 1155 1536">Image 23</p> |
|  <p data-bbox="504 1832 628 1861">Image 40</p> |  <p data-bbox="1031 1832 1155 1861">Image40</p> |

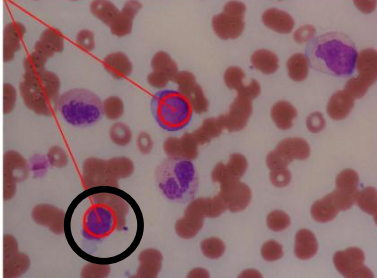

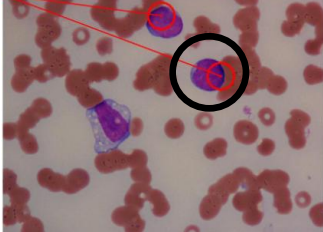
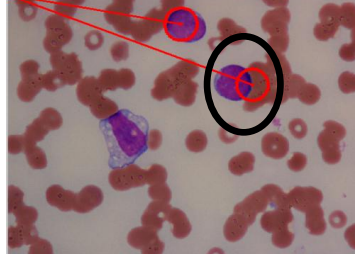
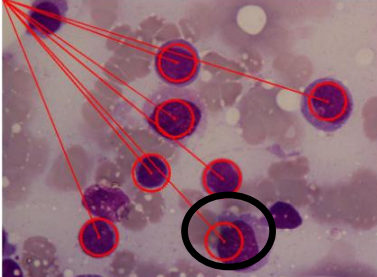
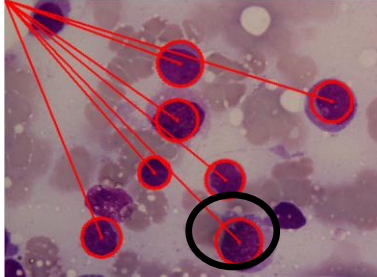
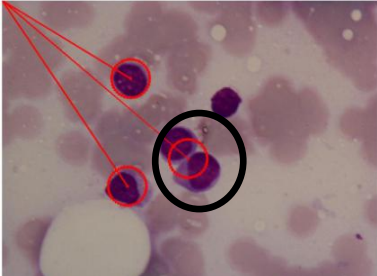
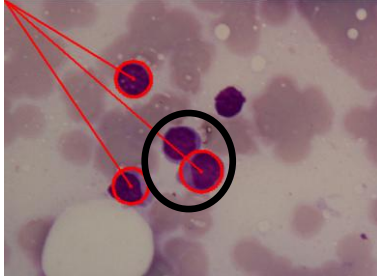
M2 Subtype

| Before Heuristic Search | After Heuristic Search |
|---|--|
|  <p>Image 1</p> |  <p>Image 1</p> |
|  <p>Image 4</p> |  <p>Image 4</p> |
|  <p>Image 5</p> |  <p>Image 5</p> |
|  <p>Image 15</p> |  <p>Image 15</p> |

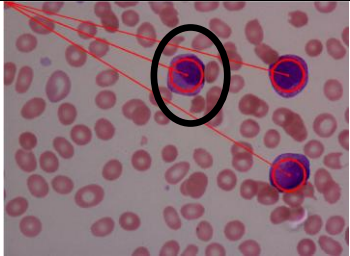

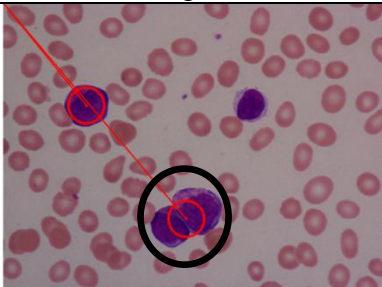
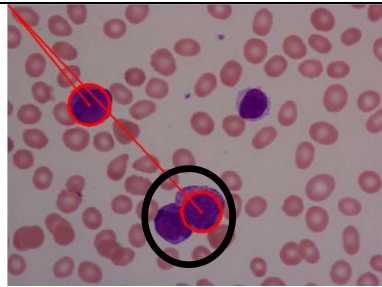
| Before Heuristic Search | After Heuristic Search |
|---|--|
|  <p data-bbox="518 683 646 712">Image 17</p> |  <p data-bbox="1045 683 1173 712">Image 17</p> |
|  <p data-bbox="518 1003 646 1032">Image 48</p> |  <p data-bbox="1045 1003 1173 1032">Image 48</p> |
|  <p data-bbox="518 1323 646 1352">Image 49</p> |  <p data-bbox="1045 1323 1173 1352">Image 49</p> |
|  <p data-bbox="518 1621 646 1650">Image 61</p> |  <p data-bbox="1045 1621 1173 1650">Image 61</p> |

| Before Heuristic Search | After Heuristic Search |
|---|--|
|  <p data-bbox="518 728 646 757">Image 65</p> |  <p data-bbox="1045 728 1173 757">Image 65</p> |
|  <p data-bbox="518 1019 646 1048">Image 69</p> |  <p data-bbox="1045 1041 1173 1070">Image 69</p> |
|  <p data-bbox="518 1332 646 1361">Image 70</p> |  <p data-bbox="1045 1332 1173 1361">Image 70</p> |
|  <p data-bbox="518 1646 646 1675">Image 77</p> |  <p data-bbox="1045 1646 1173 1675">Image 77</p> |

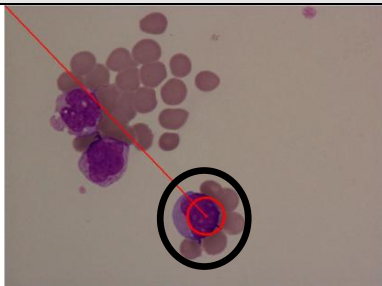
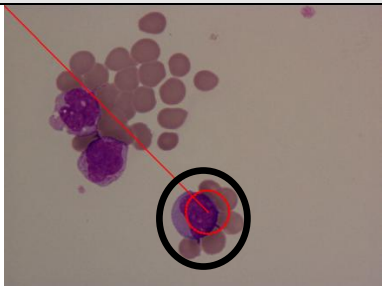
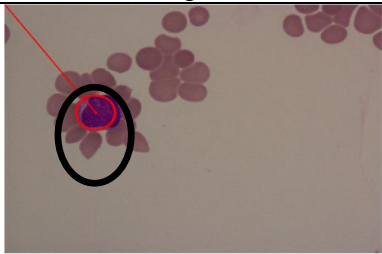

| Before Heuristic Search | After Heuristic Search |
|--|---|
|  <p data-bbox="507 703 655 741">Image 103</p> |  <p data-bbox="1034 703 1182 741">Image 103</p> |
|  <p data-bbox="507 1010 655 1048">Image 105</p> |  <p data-bbox="1034 1010 1182 1048">Image 105</p> |
|  <p data-bbox="507 1323 655 1361">Image 107</p> |  <p data-bbox="1034 1323 1182 1361">Image 107</p> |
|  <p data-bbox="507 1644 655 1682">Image 117</p> |  <p data-bbox="1034 1644 1182 1682">Image 117</p> |

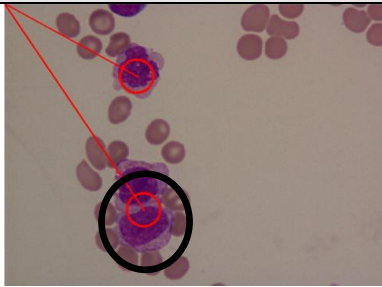
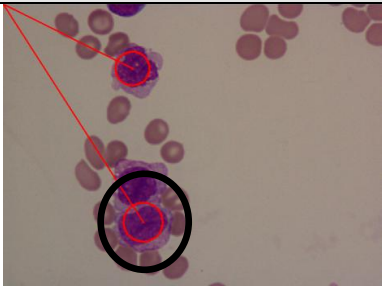
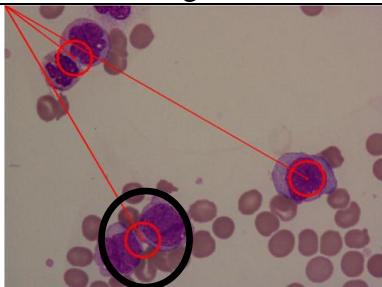
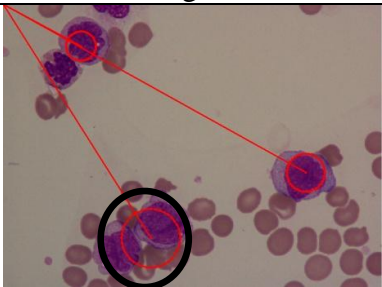


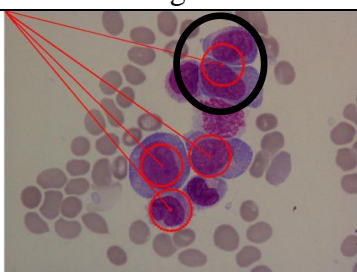
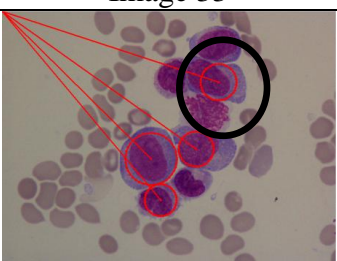
| Before Heuristic Search | After Heuristic Search |
|--|---|
|  <p data-bbox="502 723 644 757">Image 118</p> |  <p data-bbox="1029 723 1171 757">Image 118</p> |
|  <p data-bbox="502 999 644 1032">Image 119</p> |  <p data-bbox="1029 1021 1171 1055">Image 119</p> |
|  <p data-bbox="502 1344 644 1377">Image 127</p> |  <p data-bbox="1029 1344 1171 1377">Image 127</p> |
|  <p data-bbox="502 1664 644 1697">Image 128</p> |  <p data-bbox="1029 1664 1171 1697">Image 128</p> |

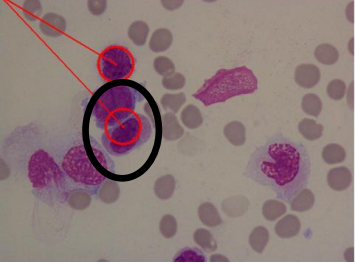
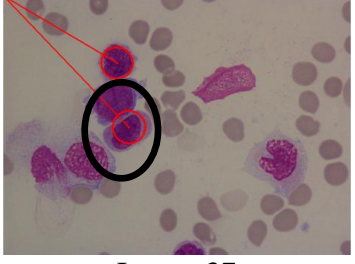
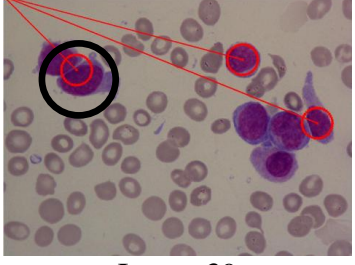
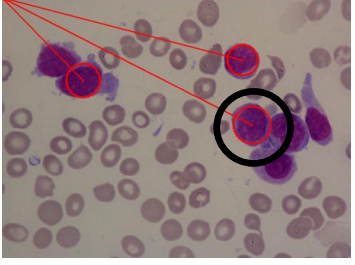
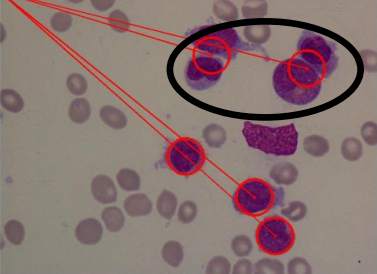
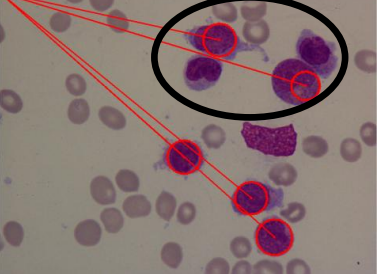
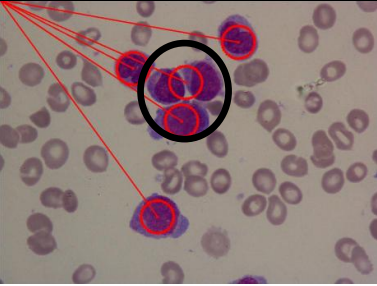
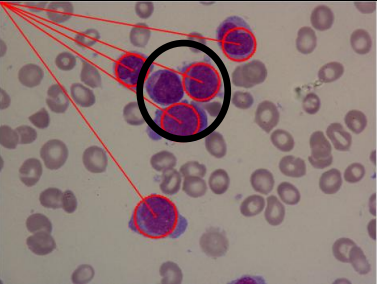
M3 Subtype

| Before Heuristics Search | After Heuristic Search |
|--|---|
|  <p data-bbox="512 734 635 770">Image 49</p> |  <p data-bbox="1038 734 1161 770">Image 49</p> |
|  <p data-bbox="512 1055 635 1093">Image 88</p> |  <p data-bbox="1038 1055 1161 1093">Image 88</p> |

M5 Subtypes

| Before Heuristics Search | After Heuristics Search |
|---|--|
|  <p data-bbox="507 1592 635 1630">Image 10</p> |  <p data-bbox="1038 1592 1166 1630">Image 10</p> |
|  <p data-bbox="507 1883 635 1921">Image 12</p> |  <p data-bbox="1038 1883 1166 1921">Image 12</p> |

| Before Heuristics Search | After Heuristics Search |
|---|--|
|  <p data-bbox="507 719 635 752">Image 20</p> |  <p data-bbox="1038 719 1166 752">Image 20</p> |
|  <p data-bbox="507 1037 635 1070">Image 22</p> |  <p data-bbox="1038 1037 1166 1070">Image 22</p> |
|  <p data-bbox="507 1355 635 1388">Image 33</p> |  <p data-bbox="1038 1355 1166 1388">Image 33</p> |
|  <p data-bbox="507 1659 635 1697">Image 36</p> |  <p data-bbox="1038 1648 1166 1697">Image 36</p> |

| Before Heuristics Search | After Heuristics Search |
|---|--|
|  <p data-bbox="507 705 635 739">Image 37</p> |  <p data-bbox="1038 705 1166 739">Image 37</p> |
|  <p data-bbox="507 1008 635 1041">Image 39</p> |  <p data-bbox="1038 1008 1166 1041">Image 39</p> |
|  <p data-bbox="507 1321 635 1355">Image 40</p> |  <p data-bbox="1038 1321 1166 1355">Image 40</p> |
|  <p data-bbox="507 1646 635 1680">Image 43</p> |  <p data-bbox="1038 1646 1166 1680">Image 43</p> |

Appendix F – Summary of the test result from WEKA

| Method | Percentage Correct | Percentage Incorrect |
|------------------------------|--------------------|----------------------|
| Classification Bayes | | |
| BayesNet | 70.59% | 29.41 |
| ComplementNaiveBayes | 60.78% | 39.22% |
| DMNBtext | 43.14% | 56.86% |
| NaiveBayes | 70.59% | 29.41% |
| NaiveBayesMultinomial | 66.67% | 33.33% |
| NaiveBayesSimple | 70.59% | 29.41% |
| Classifier Function | | |
| Logistic | 76.47% | 23.53% |
| <i>Multilayer Perceptron</i> | 92.16% | 7.84% |
| RBFNetwork | 56.86% | 43.14% |
| SimpleLogistic | 90.20% | 9.80% |
| SMO | 74.50% | 25.50% |
| Classifier Lazy | | |
| <i>IB1</i> | 94.12% | 5.88% |
| <i>IBK</i> | 94.12% | 5.88% |
| Kstar | 33.33% | 66.67% |
| LWL | 72.55% | 27.45% |

| Method | Percentage Correct | Percentage Incorrect |
|------------------------------|---------------------------|-----------------------------|
| Classifier Meta | | |
| AdaBoostM1 | 58.82% | 41.18% |
| AttributesSelectedClassifier | 78.43% | 21.57% |
| Bagging | 76.47% | 23.53% |
| ClassificationViaClustering | 54.90% | 45.10% |
| ClassificationViaRegression | 86.27% | 13.73% |
| CVParameterSelection | 43.14% | 56.86% |
| Dagging | 62.75% | 37.25% |
| Decorate | 78.43% | 21.57% |
| END | 78.43% | 21.57% |
| EnsembleSelection | 74.51% | 25.49% |
| FilteredClassifier | 66.67% | 33.33% |
| Grading | 43.14% | 56.86% |
| LogitBoost | 80.39% | 19.61% |
| MultiBoostAB | 58.82% | 41.18% |
| MultiClassClassifier | 80.39% | 19.61% |
| MultiScheme | 43.14% | 56.86% |
| ClassBalancedND | 64.71% | 35.29% |
| DataNearBalancedND | 64.71% | 35.29% |
| ND | 70.59% | 29.41% |
| OrdinalClassClassifier | 78.43% | 21.57% |
| RacedIncrementalLogitBoost | 43.14% | 56.86% |
| RandomCommittee | 88.24% | 11.76% |
| RandomSubSpace | 70.59% | 29.41% |
| RotationForest | 88.24% | 11.76% |

| Method | Percentage Correct | Percentage Incorrect |
|-------------------------|---------------------------|-----------------------------|
| Classifier Meta | | |
| Stacking | 43.14% | 56.86% |
| StackingC | 43.14% | 56.86% |
| Vote | 43.14% | 56.86% |
| ClassifierMeta | | |
| FLR | 76.47% | 23.53% |
| HyperPipes | 72.55% | 27.45% |
| VFI | 66.67% | 33.33% |
| Classifier Rules | | |
| ConjunctiveRule | 58.82% | 41.18% |
| Decision Table | 62.75% | 37.25% |
| DTNB | 64.71% | 35.29% |
| JRip | 74.51% | 25.49% |
| NNge | 80.39% | 19.61% |
| OneR | 52.94% | 47.06% |
| Part | 76.47% | 23.53% |
| Ridor | 70.59% | 29.41% |
| ZeroR | 43.14% | 56.86% |

| Method | Percentage Correct | Percentage Incorrect |
|------------------------|---------------------------|-----------------------------|
| Classifier Tree | | |
| BFTree | 76.47% | 23.53% |
| DecisionStump | 58.82% | 41.18% |
| <i>FT</i> | <i>92.16%</i> | <i>7.84%</i> |
| J48 | 76.47% | 23.53% |
| J48graft | 76.47% | 23.53% |
| LADTree | 76.47% | 23.53% |
| LMT | 90.20% | 9.80% |
| NBTree | 72.55% | 27.45% |
| RandomForest | 78.43% | 21.57% |
| Random Tree | 76.47% | 23.53% |
| REPTree | 72.55% | 27.45% |
| SimpleCart | 70.59% | 29.41% |
| UserClassifier | 43.14% | 56.86% |

Appendix G – Execution time for start and end time for twenty images

| Image | Start time | End time | Hours |
|--------------|-------------------|-----------------|--------------|
| 1 | 17:43:28 | 18:45:22 | ≈ 1 hours |
| 2 | 11:45:11 | 12:35:03 | ≈ 1 hours |
| 3 | 12:45:35 | 13:12:15 | ≈ 30 minutes |
| 4 | 13:15:27 | 15:43:54 | ≈ 2 hours |
| 5 | 15:47:26 | 17:46:57 | ≈ 2 hours |
| 6 | 18:09:22 | 19:23:40 | ≈ 1 hours |
| 7 | 19:25:57 | 19:53:28 | ≈ 30 minutes |
| 8 | 10:18:48 | 10:29:03 | ≈ 20 minutes |
| 9 | 10:30:42 | 11:44:53 | ≈ 1 hours |
| 10 | 11:50:45 | 12:20:23 | ≈ 30 minutes |
| 11 | 12:21:49 | 12:34:50 | ≈ 15 minutes |
| 12 | 12:36:21 | 12:50:44 | ≈ 20 minutes |
| 13 | 12:53:02 | 13:41:38 | ≈ 2 hours |
| 14 | 13:48:55 | 15:10:34 | ≈ 2 hours |
| 15 | 15:13:38 | 15:28:51 | ≈ 15 minutes |
| 16 | 15:31:31 | 16:01:23 | ≈ 30 minutes |
| 17 | 16:06:18 | 16:22:21 | ≈ 20 minutes |
| 18 | 16:28:31 | 16:41:54 | ≈ 20 minutes |
| 19 | 16:54:20 | 18:09:48 | ≈ 2 hours |
| 20 | 19:10:32 | 19:54:59 | ≈ 1 hours |

**Appendix H – MultiLayer Perceptron for sigmoid and nodes used
in WEKA for 51 sub-images with 10 folds validation**

| Sigmoid Node 0 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | 0.578 |
| Node 4 | -2.272 |
| Node 5 | -1.934 |
| Node 6 | 2.739 |
| Node 7 | 1.988 |
| Node 8 | -1.360 |
| Node 9 | -1.663 |
| Node 10 | -1.630 |
| Node 11 | -6.157 |
| Node 12 | -1.545 |

| Sigmoid Node 1 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | 0.477 |
| Node 4 | -1.604 |
| Node 5 | 0.575 |
| Node 6 | -7.010 |
| Node 7 | 2.694 |
| Node 8 | -4.072 |
| Node 9 | 0.874 |
| Node 10 | -3.670 |
| Node 11 | 2.016 |
| Node 12 | -3.750 |

| Sigmoid Node 2 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | -3.266 |
| Node 4 | -1.520 |
| Node 5 | 0.857 |
| Node 6 | -3.475 |
| Node 7 | 0.780 |
| Node 8 | 3.360 |
| Node 9 | 1.981 |
| Node 10 | 3.619 |
| Node 11 | -1.988 |
| Node 12 | 3.028 |

| Sigmoid Node 3 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | -2.543 |
| Node 4 | 2.574 |
| Node 5 | -1.302 |
| Node 6 | 2.121 |
| Node 7 | -5.310 |
| Node 8 | -1.592 |
| Node 9 | -3.403 |
| Node 10 | -2.158 |
| Node 11 | 5.704 |
| Node 12 | -1.079 |

| Sigmoid Node 4 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | 0.578 |
| Attrib avred | -0.933 |
| Attrib avgreen | 2.065 |
| Attrib avblue | 0.069 |
| Attrib varred | -0.483 |
| Attrib vargreen | -0.246 |
| Attrib varblue | 0.838 |
| Attrib hred | -0.180 |
| Attrib lred | 0.215 |
| Attrib hgreen | 0.041 |
| Attrib lgreen | 0.015 |
| Attrib hblue | -0.768 |
| Attrib lblue | 0.748 |
| Attrib mred | -1.382 |
| Attrib mgreen | 2.460 |
| Attrib mblue | -0.312 |

| Sigmoid Node 5 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | 0.196 |
| Attrib avred | -0.868 |
| Attrib avgreen | -0.254 |
| Attrib avblue | 1.308 |
| Attrib varred | 0.530 |
| Attrib vargreen | 0.233 |
| Attrib varblue | 0.493 |
| Attrib hred | -0.349 |
| Attrib lred | 0.380 |
| Attrib hgreen | 0.033 |
| Attrib lgreen | 0.017 |
| Attrib hblue | -3.310 |
| Attrib lblue | -0.085 |
| Attrib mred | -0.790 |
| Attrib mgreen | -0.595 |
| Attrib mblue | 1.166 |

| Sigmoid Node 6 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | 0.114 |
| Attrib avred | 1.693 |
| Attrib avgreen | 2.615 |
| Attrib avblue | -3.389 |
| Attrib varred | -1.396 |
| Attrib vargreen | -0.177 |
| Attrib varblue | -0.563 |
| Attrib hred | 0.494 |
| Attrib lred | -0.470 |
| Attrib hgreen | -0.050 |
| Attrib lgreen | 0.024 |
| Attrib hblue | -0.152 |
| Attrib lblue | 0.192 |
| Attrib mred | 1.048 |
| Attrib mgreen | 3.218 |
| Attrib mblue | -3.690 |

| Sigmoid Node 7 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | -0.730 |
| Attrib avred | 1.629 |
| Attrib avgreen | -2.394 |
| Attrib avblue | -0.019 |
| Attrib varred | 0.365 |
| Attrib vargreen | 0.201 |
| Attrib varblue | -0.822 |
| Attrib hred | -0.096 |
| Attrib lred | 0.001 |
| Attrib hgreen | -0.024 |
| Attrib lgreen | 0.039 |
| Attrib hblue | 1.167 |
| Attrib lblue | -1.110 |
| Attrib mred | 2.440 |
| Attrib mgreen | -3.017 |
| Attrib mblue | 0.643 |

| Sigmoid Node 8 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | -0.259 |
| Attrib avred | -2.658 |
| Attrib avgreen | -0.900 |
| Attrib avblue | 2.427 |
| Attrib varred | 2.140 |
| Attrib vargreen | -0.341 |
| Attrib varblue | 0.171 |
| Attrib hred | 0.538 |
| Attrib lred | 0.001 |
| Attrib hgreen | -0.024 |
| Attrib lgreen | 0.039 |
| Attrib hblue | 1.167 |
| Attrib lblue | -1.110 |
| Attrib mred | 2.440 |
| Attrib mgreen | -3.017 |
| Attrib mblue | 0.643 |

| Sigmoid Node 9 | |
|-----------------------|---------|
| Inputs | Weights |
| Threshold | -0.008 |
| Attrib avred | -1.265 |
| Attrib avgreen | -0.931 |
| Attrib avblue | 2.177 |
| Attrib varred | 1.326 |
| Attrib vargreen | -0.092 |
| Attrib varblue | 0.642 |
| Attrib hred | -0.566 |
| Attrib lred | 0.592 |
| Attrib hgreen | -0.020 |
| Attrib lgreen | 0.001 |
| Attrib hblue | 0.216 |
| Attrib lblue | -0.189 |
| Attrib mred | -0.993 |
| Attrib mgreen | -1.428 |
| Attrib mblue | 2.249 |

| Sigmoid Node 10 | |
|------------------------|---------|
| Inputs | Weights |
| Threshold | -0.355 |
| Attrib avred | -2.636 |
| Attrib avgreen | -1.028 |
| Attrib avblue | 2.462 |
| Attrib varred | 2.121 |
| Attrib vargreen | -0.294 |
| Attrib varblue | 0.293 |
| Attrib hred | 0.394 |
| Attrib lred | -0.381 |
| Attrib hgreen | -0.007 |
| Attrib lgreen | -0.019 |
| Attrib hblue | 0.492 |
| Attrib lblue | -0.510 |
| Attrib mred | -3.093 |
| Attrib mgreen | -1.706 |
| Attrib mblue | 1.919 |

| Sigmoid Node 11 | |
|------------------------|---------|
| Inputs | Weights |
| Threshold | 2.482 |
| Attrib avred | -1.557 |
| Attrib avgreen | 1.714 |
| Attrib avblue | 2.921 |
| Attrib varred | -0.791 |
| Attrib vargreen | 1.464 |
| Attrib varblue | 3.291 |
| Attrib hred | -1.606 |
| Attrib lred | 1.582 |
| Attrib hgreen | 0.019 |
| Attrib lgreen | 0.004 |
| Attrib hblue | -1.953 |
| Attrib lblue | 1.896 |
| Attrib mred | -1.400 |
| Attrib mgreen | 2.014 |
| Attrib mblue | 2.738 |

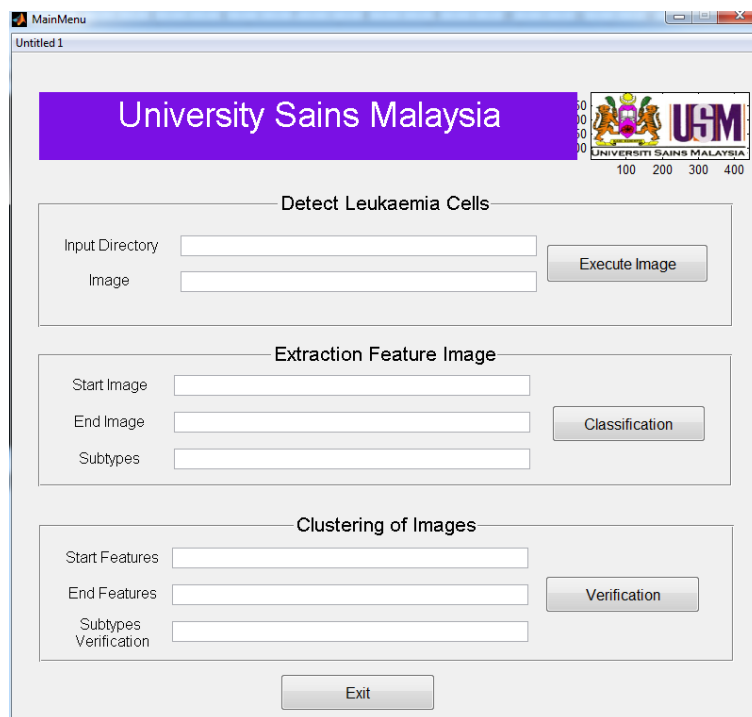
| Sigmoid Node 12 | |
|------------------------|---------|
| Inputs | Weights |
| Threshold | -0.121 |
| Attrib avred | -2.60 |
| Attrib avgreen | -0.663 |
| Attrib avblue | 2.160 |
| Attrib varred | 1.824 |
| Attrib vargreen | -0.284 |
| Attrib varblue | 0.240 |
| Attrib hred | 0.566 |
| Attrib lred | -0.590 |
| Attrib hgreen | -0.009 |
| Attrib lgreen | 0.047 |
| Attrib hblue | 0.385 |
| Attrib lblue | -0.340 |
| Attrib mred | -3.185 |
| Attrib mgreen | -1.471 |
| Attrib mblue | 1.496 |

Appendix I – User Interface

After the development of all the processes described in Chapters 4 to 7, all substeps were combined in a software application. It is a Menu Driven program with user-friendly interface.

Layout of the Menu

Below is a screenshot of the Leukaemia Detection and Classification application and a table describing individual fields that need to be filled-in by the user.



A screenshot of Leukaemia Detection and Classification program

Description of files in the figure above

| Items | Description |
|-----------------------|---|
| Input Directory | The location of the image |
| Image | Image number |
| Execute Image | Executes the detection of the leukaemia cells (from the Otsu method to the use of Heuristic Search) |
| Start Image | Choose starting image |
| End Image | Choose end image |
| Subtypes | Prediction of the subtypes of AML images |
| Classification | Extraction the features location of the images |
| Start Feature | Start number of the features. |
| End Feature | End number of the features. |
| Subtypes Verification | Identified the subtypes verification |