Texas A&M University - Commerce

Department of Computer Science

Automated Blood Cell Identification and Counting

Sridevi Sowmya Grandhi

Supervisor: Derek Harter, Ph.D.

A report submitted in partial fulfilment of the requirements of
Texas A&M University - Commerce for the degree of
Master of Science in *Computer Science*

February 5, 2024

Declaration

I, Sridevi Sowmya Grandhi, of the Department of Computer Science, Texas A&M University - Commerce, confirm that this is my own work and figures, tables, equations, code snippets, artworks, and illustrations in this report are original and have not been taken from any other person's work, except where the works of others have been explicitly acknowledged, quoted, and referenced. I understand that if failing to do so will be considered a case of plagiarism. Plagiarism is a form of academic misconduct and will be penalised accordingly.

I give consent to a copy of my report being shared with future students as an exemplar.

I give consent for my work to be made available more widely to members of TAMUC and public with interest in teaching, learning and research.

Sridevi Sowmya Grandhi
February 5, 2024

Abstract

In our cutting-edge approach to automate the identification and counting of three blood cell types, we employ a sophisticated fusion of deep learning and advanced image processing techniques, particularly focusing on object detection. Traditional complete blood cell counts necessitate laborious manual counting using a haemocytometer, involving intricate laboratory equipment and chemical compounds. This antiquated method is both time-consuming and burdensome. Our innovative solution utilizes Convolutional Neural Networks (CNNs) for intricate feature extraction from microscopic blood sample images. Incorporating state-of-the-art object detection algorithms, such as YOLO (You Only Look Once) or Faster R-CNN (Region-based Convolutional Neural Network), our system precisely identifies and localizes individual blood cells, overcoming the limitations of manual counting. Image processing techniques, including contrast enhancement and morphological operations, are strategically applied to optimize image quality and facilitate accurate object segmentation. This synergistic blend of deep learning and image processing not only expedites the diagnostic process but also significantly improves the accuracy and efficiency of blood cell identification and counting. By automating this intricate task, our approach aims to revolutionize medical diagnostics, providing healthcare professionals with a rapid and reliable tool for comprehensive blood cell analysis.

Keywords: a maximum of five keywords/keyphrase separated by commas

Acknowledgements

An acknowledgements section is optional. You may like to acknowledge the support and help of your supervisor(s), friends, or any other person(s), department(s), institute(s), etc. If you have been provided specific facility from department/school acknowledged so.

Contents

List of Figures

List of Tables

List of Abbreviations

SMPCS	School of Mathematical, Physical and Computational Sciences

Chapter 1

Introduction

A complete blood cell count (CBC) is vital for assessing health, comprising red blood cells (RBCs), white blood cells (WBCs), and platelets. Manual counting methods are time-consuming and error-prone, requiring automation. Machine learning, particularly deep learning, offers robust solutions across medical applications. Applying deep learning to identify and count blood cells in smear images presents a promising avenue for accurate and efficient analysis, revolutionizing medical diagnostics. Previous models like YOLOv5 and YOLOX have pushed the boundaries of object detection with improved speed and accuracy. YOLOv5 introduced innovations like ConvBN-LeakyReLU and EfficientNet-inspired components. YOLOX further enhanced performance with methods like Cross Stage Partial Network plus CBS. While these models have advanced the field, they may face limitations in handling small objects or complex scenes

1.1 Background

The utilization of image-based methods for disease detection and diagnosis has gained significant attention in recent years. This project focuses on the development of an automated system for the identification and counting of blood cells, leveraging advanced image processing and machine learning techniques.

1.2 Problem statement

The accurate identification and characterization of blood cells, particularly red blood cells (RBCs) and white blood cells (WBCs), pose significant challenges in traditional medical diagnostics. Manual methods for blood cell analysis are often labor-intensive, time-consuming, and prone to human error, leading to variability in results.

1.3 Aims and objectives

Aims: This project aims to develop an automated system for blood cell analysis to improve accuracy, efficiency, and reliability in disease diagnosis.

Objectives: The specific objectives of this project include:

*Exploring existing methodologies for automated blood cell analysis.

*Identifying limitations in current approaches and proposing innovative solutions.

*Designing and implementing a novel solution approach combining image processing and machine learning techniques.

*Evaluating the performance of the developed system and comparing it with existing methods.

1.4 Solution approach

Briefly, the solution approach involves leveraging advanced algorithms and methodologies from image processing and machine learning domains. This includes preprocessing techniques for image enhancement, feature extraction methods, and the implementation of machine learning models for classification and counting of blood cells.

1.5 Summary of contributions and achievements

This study makes several significant contributions to the field of automated blood cell analysis. By addressing key challenges and proposing innovative solutions, we aim to: Improve the accuracy and reliability of blood cell identification and counting. Streamline the analysis process, thereby reducing time and labor requirements. Enhance the diagnostic capabilities of healthcare professionals, leading to improved patient outcomes.

1.6 Organization of the report

The report is organized into several sections for clarity and coherence. It begins with an Introduction, covering background, problem statement, aims, objectives, and solution approach. Following this, the Literature Review discusses pertinent sources and citation practices. Methodology outlines the research methodology adopted. Results present the findings obtained from the study. Discussion and Analysis critically analyze the results, highlighting their significance and limitations. Conclusions summarize the key findings and suggest avenues for future research. Finally, Appendices include supplementary materials such as data tables or additional information for interested readers. This structure aims to guide readers through the report's content efficiently and comprehensively.

Chapter 2

Literature Review

In the dynamic realm of blood cell image analysis, recent strides have transformed the landscape, automating and refining cell counting processes. Notably, one study pioneers image processing, automating red blood cell counting through a groundbreaking edge detection algorithm. Another introduces an extended Hough transform for low-parametric object detection, emphasizing ongoing modifications for precise counting. In nuclei extraction, a study clusters microscopic images, utilizing the curvelet transform and a novel color saturation gradient for discriminating atypical cases. A substantial contribution focuses on leukocyte segmentation, integrating pre-processing for robust identification. Employing circularity features, another study introduces an iterative circle detection algorithm, a breakthrough in binary image separation. The Circlet Transform emerges as a novel method, particularly for red blood cell segmentation, showcasing potential through the Circular Hough Transform. Additionally, a deep learning model proves promising in automating blood cell counting, utilizing the Blood Cell Count Dataset. Collectively, these innovations propel the field towards unparalleled efficiency and accuracy in blood cell analysis.

2.1 Example of in-text citation of references in LaTeX

Alomari et al. (2014)

2.2 Example of "risk" of unintentional plagiarism

Navigating the landscape of academic writing requires a keen awareness of the potential pitfalls, and one significant challenge is the risk of unintentional plagiarism. This occurs when writers inadvertently mismanage the incorporation of external sources, ideas, or materials into their work, leading to improper paraphrasing, summarizing, quoting, or citing. The nuances of proper attribution can be intricate, and unintentional plagiarism often stems from a lack of awareness or failure to adhere to citation rules.

Example of Unintentional Plagiarism – Citing Wrongly:

A common illustration of unintentional plagiarism is the improper citation of sources. Imagine a scenario where a writer, in the process of compiling research, encounters a compelling idea from a scholarly article. While attempting to integrate this idea into their work, they inadvertently misattribute it to another source or overlook the necessity of proper citation. This misstep results in unintentional plagiarism, as the writer fails to give due credit to the original author.

Whether due to oversight, unfamiliarity with citation guidelines, or misinterpretation of the source, such instances underscore the importance of meticulous citation practices to avoid unintentional plagiarism and uphold the principles of academic integrity.

2.3 Critique of the review

In recent advancements in blood cell image analysis, several studies have significantly contributed to automating and enhancing the accuracy of blood cell counting processes. One notable study Alomari et al. (2014), focuses on applying image processing techniques to extract blood cell images from microscopes, particularly automating the red blood cell counting process. Through the utilization of digital image processing, the study employs an edge detection algorithm to identify and count red blood cells, demonstrating a pivotal advancement in the field. Another noteworthy approach Maitra et al. (2012), involves the application of the Hough transform, a well-established feature extraction technique. Initially developed for line detection, this method has been extended for detecting low-parametric objects, such as circles. While offering a cost-effective and efficient approach, the study suggests the need for modifications to ensure accurate counting, stressing the necessity for further investigations into complete blood cell counts.

In the realm of nuclei extraction, Poomcokrak and Neatpisarnvanit (2008) employ clustering of microscopic images and the curvelet transform, proving effective in detecting detailed information and enabling discrimination between atypical and blast cases. The study introduces a novel feature, the color saturation gradient, contributing to the classification of lymphoblast cells and atypical lymphoma cells. Another significant contribution comes from Putzu et al. (2013), where it proposes a method for leukocyte segmentation and identification. This approach integrates pre-processing methods to simplify and enhance segmentation, emphasizing the multi-stage process, including shape control and nucleus-cytoplasm selection, contributing to more robust leukocyte identification.Utilizing thresholding and morphological operations, the circularity feature Sarrafzadeh et al. (2015) of blood cells is employed in an iterative structured circle detection algorithm. This introduces a new technique for binary image separation and demonstrates promising results.

Further innovations include the introduction of the Circlet Transform Soltanzadeh et al. (2012), offering a novel method for segmenting circular objects, with a specific focus on red blood cells. Utilizing the Circular Hough Transform, the method showcases potential in RBC segmentation.

2.4 Summary

In recent strides towards automating blood cell image analysis, a series of noteworthy studies have significantly advanced the field. These studies encompass diverse methodologies, such as image processing, Hough transform, clustering, and deep learning. One study pioneers the use of image processing, specifically an edge detection algorithm, to automate red blood cell counting, marking a pivotal advancement. Another approach employs the Hough transform for feature extraction, offering a cost-effective method for detecting low-parametric objects, though suggesting the need for modifications for accurate counting. Further contributions involve nuclei extraction using clustering and the curvelet transform, introducing a novel feature for discriminating atypical cases. Another study proposes a multi-stage method for leukocyte segmentation, emphasizing shape

control and nucleus-cytoplasm selection. Innovations also include the introduction of the Circlet Transform for segmenting circular objects, with a focus on red blood cells, and the application of a deep learning model trained on the Blood Cell Count Dataset, showcasing promising results in automating blood cell counting. Collectively, these studies underscore the diverse and evolving landscape of techniques contributing to the automation and accuracy of blood cell image analysis.

Chapter 3

Methodology

We mentioned in Chapter 1 that a project report's structure could follow a particular paradigm. Hence, the organization of a report (effectively the Table of Content of a report) can vary depending on the type of project you are doing. Check which of the given examples suit your project. Alternatively, follow your supervisor's advice.

3.1 Examples of the sections of a methodology chapter

A general report structure is summarised (suggested) in Table 3.1. Table 3.1 describes that, in general, a typical report structure has three main parts: (1) front matter, (2) main text, and (3) end matter. The structure of the front matter and end matter will remain the same for all the undergraduate final year project report. However, the main text varies as per the project's needs.

3.1.1 Example of a software/Web development main text structure

Notice that the "methodology" Chapter of Software/Web development in Table 3.2 takes a standard software engineering paradigm (approach). Alternatively, these suggested sections can be the chapters of their own. Also, notice that "Chapter 5" in Table 3.2 is "Testing and Validation" which is different from the general report template mentioned in Table 3.1. Check with your supervisor if in doubt.

3.1.2 Example of an algorithm analysis main text structure

Some project might involve the implementation of a state-of-the-art algorithm and its performance analysis and comparison with other algorithms. In that case, the suggestion in Table 3.3 may suit you the best.

3.1.3 Example of an application type main text structure

If you are applying some algorithms/tools/technologies on some problems/datasets/etc., you may use the methodology section prescribed in Table 3.4.

Table 3.1: Undergraduate report template structure

Frontmatter		Title Page Abstract Acknowledgements Table of Contents List of Figures List of Tables List of Abbreviations
Main text	Chapter 1	Introduction
	Chapter 2	Literature Review
	Chapter 3	Methodology
	Chapter 4	Results
	Chapter 5	Discussion and Analysis
	Chapter 6	Conclusions and Future Work
	Chapter 7	Refection
End matter		References Appendices (Optional) Index (Optional)

Table 3.2: Example of a software engineering-type report structure

Chapter 1	Introduction	
Chapter 2	Literature Review	
Chapter 3	Methodology	
		Requirements specifications
		Analysis
		Design
		Implementations
Chapter 4	Testing and Validation	
Chapter 5	Results and Discussion	
Chapter 6	Conclusions and Future Work	
Chapter 7	Reflection	

3.1.4 Example of a science lab-type main text structure

If you are doing a science lab experiment type of project, you may use the methodology section suggested in Table 3.5. In this kind of project, you may refer to the "Methodology" section as "Materials and Methods."

Table 3.3: Example of an algorithm analysis type report structure

Chapter 1	Introduction	
Chapter 2	Literature Review	
Chapter 3	Methodology	
		Algorithms descriptions
		Implementations
		Experiments design
Chapter 4	Results	
Chapter 5	Discussion and Analysis	
Chapter 6	Conclusion and Future Work	
Chapter 7	Reflection	

Table 3.4: Example of an application type report structure

Chapter 1	Introduction	
Chapter 2	Literature Review	
Chapter 3	Methodology	
		Problems (tasks) descriptions
		Algorithms/tools/technologies/etc. descriptions
		Implementations
		Experiments design and setup
Chapter 4	Results	
Chapter 5	Discussion and Analysis	
Chapter 6	Conclusion and Future Work	
Chapter 7	Reflection	

Table 3.5: Example of a science lab experiment-type report structure

Chapter 1	Introduction	
Chapter 2	Literature Review	
Chapter 3	Materials and Methods	
		Problems (tasks) description
		Materials
		Procedures
		Implementations
		Experiment set-up
Chapter 4	Results	
Chapter 5	Discussion and Analysis	
Chapter 6	Conclusion and Future Work	
Chapter 7	Reflection	

3.2 Example of an Equation in LaTeX

Eq. 3.1 [note that this is an example of an equation's in-text citation] is an example of an equation in LaTeX. In Eq. (3.1), s is the mean of elements $x_i \in \mathbf{x}$:

$$s = \frac{1}{N} \sum_{i=1}^{N} x_i. \tag{3.1}$$

Have you noticed that all the variables of the equation are defined using the **in-text** maths command $.$, and Eq. (3.1) is treated as a part of the sentence with proper punctuation? Always treat an equation or expression as a part of the sentence.

3.3 Example of a Figure in LaTeX

Figure 3.1 is an example of a figure in LaTeX. For more details, check the link:
wikibooks.org/wiki/LaTeX/Floats,_Figures_and_Captions.
Keep your artwork (graphics, figures, illustrations) clean and readable. At least 300dpi is a good resolution of a PNG format artwork. However, an SVG format artwork saved as a PDF will produce the best quality graphics. There are numerous tools out there that can produce vector graphics and let you save that as an SVG file and/or as a PDF file. One example of such a tool is the "Flow algorithm software". Here is the link for that: flowgorithm.org.

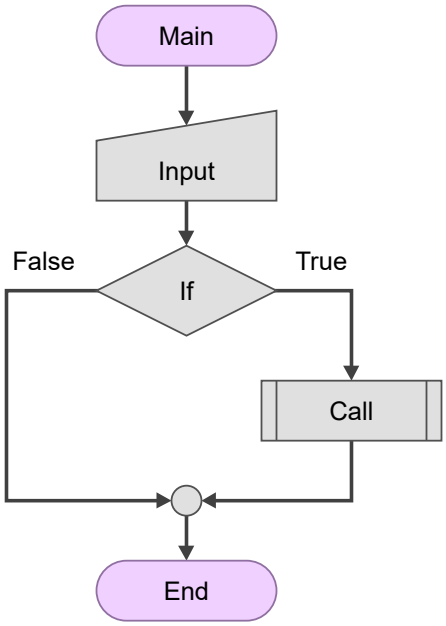

Figure 3.1: Example figure in LaTeX.

3.4 Example of an algorithm in LaTeX

Algorithm 1 is a good example of an algorithm in LaTeX.

Algorithm 1 Example caption: sum of all even numbers

Input: $\mathbf{x} = x_1, x_2, \ldots, x_N$
Output: $EvenSum$ (Sum of even numbers in $\mathbf{x}$)

1: **function** EvenSummation($\mathbf{x}$)
2: $EvenSum \leftarrow 0$
3: $N \leftarrow length(\mathbf{x})$
4: **for** $i \leftarrow 1$ to N **do**
5: **if** $x_i \mod 2 == 0$ **then** ▷ check if a number is even?
6: $EvenSum \leftarrow EvenSum + x_i$
7: **end if**
8: **end for**
9: **return** $EvenSum$
10: **end function**

3.5 Example of code snippet in LaTeX

Code Listing 3.1 is a good example of including a code snippet in a report. While using code snippets, take care of the following:

- do not paste your entire code (implementation) or everything you have coded. Add code snippets only.
- The algorithm shown in Algorithm 1 is usually preferred over code snippets in a technical/-scientific report.
- Make sure the entire code snippet or algorithm stays on a single page and does not overflow to another page(s).

Here are three examples of code snippets for three different languages (Python, Java, and CPP) illustrated in Listings 3.1, 3.2, and 3.3 respectively.

```python
import numpy as np

x  = [0, 1, 2, 3, 4, 5] # assign values to an array
evenSum = evenSummation(x) # call a function

def evenSummation(x):
    evenSum = 0
    n = len(x)
    for i in range(n):
        if np.mod(x[i],2) == 0: # check if a number is even?
            evenSum = evenSum + x[i]
    return evenSum
```

Listing 3.1: Code snippet in LaTeX and this is a Python code example

Here we used the "\clearpage" command and forced-out the second listing example onto the next page.

```java
public class EvenSum{
    public static int evenSummation(int[] x){
        int evenSum = 0;
        int n = x.length;
        for(int i = 0; i < n; i++){
            if(x[i]%2 == 0){ // check if a number is even?
                evenSum = evenSum + x[i];
            }
        }
        return evenSum;
    }
    public static void main(String[] args){
        int[] x  = {0, 1, 2, 3, 4, 5}; // assign values to an array
        int evenSum = evenSummation(x);
        System.out.println(evenSum);
    }
}
```

Listing 3.2: Code snippet in LaTeX and this is a Java code example

```cpp
int evenSummation(int x[]){
    int evenSum = 0;
    int n = sizeof(x);
    for(int i = 0; i < n; i++){
        if(x[i]%2 == 0){ // check if a number is even?
            evenSum = evenSum + x[i];
      }
    }
    return evenSum;
}

int main(){
    int x[]  = {0, 1, 2, 3, 4, 5}; // assign values to an array
    int evenSum = evenSummation(x);
    cout<<evenSum;
    return 0;
}
```

Listing 3.3: Code snippet in LaTeX and this is a C/C++ code example

3.6 Example of in-text citation style

3.6.1 Example of the equations and illustrations placement and reference in the text

Make sure whenever you refer to the equations, tables, figures, algorithms, and listings for the first time, they also appear (placed) somewhere on the same page or in the following page(s). Always make sure to refer to the equations, tables and figures used in the report. Do not leave them without an **in-text citation**. You can refer to equations, tables and figures more them once.

3.6.2 Example of the equations and illustrations style

Write **Eq.** with an uppercase "Eq" for an equation before using an equation number with (\eqref{.}). Use "Table" to refer to a table, "Figure" to refer to a figure, "Algorithm" to

refer to an algorithm and "Listing" to refer to listings (code snippets). Note that, we do not use the articles "a," "an," and "the" before the words Eq., Figure, Table, and Listing, but you may use an article for referring the words figure, table, etc. in general.

For example, the sentence "A report structure is shown in **the** Table 3.1" should be written as "A report structure is shown **in** Table 3.1."

3.7 Summary

Write a summary of this chapter.

Note: In the case of **software engineering** project a Chapter "**Testing and Validation**" should precede the "Results" chapter. See Section 3.1.1 for report organization of such project.

Chapter 4

Results

The results chapter tells a reader about your findings based on the methodology you have used to solve the investigated problem. For example:

- If your project aims to develop a software/web application, the results may be the developed software/system/performance of the system, etc., obtained using a relevant methodological approach in software engineering.
- If your project aims to implement an algorithm for its analysis, the results may be the performance of the algorithm obtained using a relevant experiment design.
- If your project aims to solve some problems/research questions over a collected dataset, the results may be the findings obtained using the applied tools/algorithms/etc.

Arrange your results and findings in a logical sequence.

4.1 A section

...

4.2 Example of a Table in LaTeX

Table 4.1 is an example of a table created using the package LaTeX"booktabs." do check the link: wikibooks.org/wiki/LaTeX/Tables for more details. A table should be clean and readable. Unnecessary horizontal lines and vertical lines in tables make them unreadable and messy. The example in Table 4.1 uses a minimum number of liens (only necessary ones). Make sure that the top rule and bottom rule (top and bottom horizontal lines) of a table are present.

Table 4.1: Example of a table in LaTeX

Bike		
Type	Color	Price (£)
Electric	black	700
Hybrid	blue	500
Road	blue	300
Mountain	red	300
Folding	black	500

4.3 Example of captions style

- The **caption of a Figure (artwork) goes below** the artwork (Figure/Graphics/illustration). See example artwork in Figure 3.1.
- The **caption of a Table goes above** the table. See the example in Table 4.1.
- The **caption of an Algorithm goes above** the algorithm. See the example in Algorithm 1.
- The **caption of a Listing goes below** the Listing (Code snippet). See example listing in Listing 3.1.

4.4 Summary

Write a summary of this chapter.

Chapter 5

Discussion and Analysis

Depending on the type of project you are doing, this chapter can be merged with "Results" Chapter as " Results and Discussion" as suggested by your supervisor.

In the case of software development and the standalone applications, describe the significance of the obtained results/performance of the system.

5.1 A section

Discussion and analysis chapter evaluates and analyses the results. It interprets the obtained results.

5.2 Significance of the findings

In this chapter, you should also try to discuss the significance of the results and key findings, in order to enhance the reader's understanding of the investigated problem

5.3 Limitations

Discuss the key limitations and potential implications or improvements of the findings.

5.4 Summary

Write a summary of this chapter.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Typically a conclusions chapter first summarizes the investigated problem and its aims and objectives. It summaries the critical/significant/major findings/results about the aims and objectives that have been obtained by applying the key methods/implementations/experiment set-ups. A conclusions chapter draws a picture/outline of your project's central and the most signification contributions and achievements.

A good conclusions summary could be approximately 300–500 words long, but this is just a recommendation.

A conclusions chapter followed by an abstract is the last things you write in your project report.

6.2 Future work

This section should refer to Chapter 4 where the author has reflected their criticality about their own solution. The future work is then sensibly proposed in this section.

Guidance on writing future work: While working on a project, you gain experience and learn the potential of your project and its future works. Discuss the future work of the project in technical terms. This has to be based on what has not been yet achieved in comparison to what you had initially planned and what you have learned from the project. Describe to a reader what future work(s) can be started from the things you have completed. This includes identifying what has not been achieved and what could be achieved.

A good future work summary could be approximately 300–500 words long, but this is just a recommendation.

Chapter 7

Reflection

Write a short paragraph on the substantial learning experience. This can include your decision-making approach in problem-solving.

Some hints: You obviously learned how to use different programming languages, write reports in LaTeXand use other technical tools. In this section, we are more interested in what you thought about the experience. Take some time to think and reflect on your individual project as an experience, rather than just a list of technical skills and knowledge. You may describe things you have learned from the research approach and strategy, the process of identifying and solving a problem, the process research inquiry, and the understanding of the impact of the project on your learning experience and future work.

Also think in terms of:

- what knowledge and skills you have developed
- what challenges you faced, but was not able to overcome
- what you could do this project differently if the same or similar problem would come
- rationalize the divisions from your initial planed aims and objectives.

A good reflective summary could be approximately 300–500 words long, but this is just a recommendation.

Note: The next chapter is "**References**," which will be automatically generated if you are using BibTeX referencing method. This template uses BibTeX referencing. Also, note that there is difference between "References" and "Bibliography." The list of "References" strictly only contain the list of articles, paper, and content you have cited (i.e., refereed) in the report. Whereas Bibliography is a list that contains the list of articles, paper, and content you have cited in the report plus the list of articles, paper, and content you have read in order to gain knowledge from. We recommend to use only the list of "References."

References

Alomari, Y. M., Sheikh Abdullah, S. N. H., Zaharatul Azma, R., Omar, K. et al. (2014), 'Automatic detection and quantification of wbcs and rbcs using iterative structured circle detection algorithm', *Computational and mathematical methods in medicine* **2014**.

Maitra, M., Gupta, R. K. and Mukherjee, M. (2012), 'Detection and counting of red blood cells in blood cell images using hough transform', *International journal of computer applications* **53**(16).

Poomcokrak, J. and Neatpisarnvanit, C. (2008), Red blood cells extraction and counting, *in* 'The 3rd international symposium on biomedical engineering', pp. 199–203.

Putzu, L., Di Ruberto, C. et al. (2013), 'White blood cells identification and counting from microscopic blood images', *World Academy of Science, Engineering and Technology* **73**, 363–370.

Sarrafzadeh, O., Dehnavi, A. M., Rabbani, H., Ghane, N. and Talebi, A. (2015), Circlet based framework for red blood cells segmentation and counting, *in* '2015 IEEE workshop on signal processing systems (SiPS)', IEEE, pp. 1–6.

Soltanzadeh, R., Rabbani, H., Talebi, A. et al. (2012), 'Extraction of nucleolus candidate zone in white blood cells of peripheral blood smear images using curvelet transform', *Computational and mathematical methods in medicine* **2012**.

Appendix A

An Appendix Chapter (Optional)

Some lengthy tables, codes, raw data, length proofs, etc. which are **very important but not essential part** of the project report goes into an Appendix. An appendix is something a reader would consult if he/she needs extra information and a more comprehensive understating of the report. Also, note that you should use one appendix for one idea.

An appendix is optional. If you feel you do not need to include an appendix in your report, avoid including it. Sometime including irrelevant and unnecessary materials in the Appendices may unreasonably increase the total number of pages in your report and distract the reader.

Appendix B

An Appendix Chapter (Optional)

...