AI BLOOD SMEAR ANALYSIS

HERISH CHAUDRAY

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

AI BLOOD SMEAR ANALYSIS

HERISH CHAUDRAY

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

This report is submitted in partial fulfillment of the requirements for the Bachelor of Computer Science (Artificial Intelligence) with Honors.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA
2023/2024

DECLARATION

I hereby declare that this project report entitled

AI BLOOD SMEAR ANALYSIS

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : HERISH CHAUDRAY A/ L RAVANTHERAN Date : 19/08/2024

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

I hereby declare that I have read this project report and found this project report is sufficient in term of the scope and quality for the award of Bachelor of [Computer Science (Artificial Intelligence)] with Honours.

SUPERVISOR : TS. DR. ABDUL SYUKOR MOHAMAD JAYA Date: 5/9/2024

DEDICATION

This project is dedicated to those who have shaped my journey and provided unwavering support.

To my parents, RAVANTHERAN A/L VELLION and KALAIVENI A/P ALAGESU, whose love and encouragement have been my bedrock. Your sacrifices and belief in me have been my driving force.

To my supervisor, Ts. Dr. Abdul Syukor Mohamad Jaya who ignited my passion for research and inspired me to push boundaries. Your wisdom and guidance have been invaluable.

To my friends, whose camaraderie and understanding have made this journey enjoyable and less daunting. Thank you for your endless support and for believing in me.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

To the countless educators and role models who have inspired me along the way, thank you for imparting knowledge and fostering a spirit of inquiry and discovery.

ACKNOWLEDGEMENTS

First and foremost, I extend my heartfelt thanks to Dr. Mardziah Mohamad, Head of Hematology, Molecular and Genetic Unit at the Division of Laboratory Medicine, University Malaya Medical Centre. Her unwavering support and expert guidance have been the cornerstone of this research. Dr. Mohamad's insightful feedback has continuously propelled my work forward, and her dedication to excellence has set a standard to which I aspire. Her passion for hematology has been a source of constant inspiration, and her commitment to advancing the field has profoundly influenced my academic journey. I am deeply grateful for her mentorship and encouragement, which have been pivotal in the successful completion of this thesis. Dr. Mohamad's assistance in refining the details and ideas of my project was invaluable. Her ability to provide clarity and direction has significantly enhanced the quality of this research. I am deeply grateful for her mentorship and encouragement, which have been pivotal in the successful completion of this thesis.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRACT

White blood cells (WBCs) play a crucial role in the blood circulatory system, serving as key defenders against infections and foreign invaders. Accurate identification and classification of WBCs are essential for diagnosing various hematological conditions. This project presents HaemLyst, an innovative AI-driven web application designed to analyze blood smears and classify different types of white blood cells. HaemLyst leverages state-of-the-art machine learning algorithms and image processing techniques to achieve high accuracy in WBC classification. The application accepts user-uploaded images of blood smears, processes these images, and provides detailed predictions, including the identification of specific WBC types. Three different predictive models are employed to ensure robust and reliable results, each offering unique insights based on the same preprocessed images. In addition to predictive accuracy, HaemLyst focuses on delivering an intuitive and modern user interface. This includes clear visualization of predictions, comparison of results across models, and the display of ground truth labels for verification. The application's user-friendly design makes it accessible to both medical professionals and researchers, facilitating the rapid and accurate analysis of blood smears. Through comprehensive testing and validation, HaemLyst has demonstrated its potential to significantly enhance the efficiency and accuracy of WBC classification in clinical settings. This thesis details the development process, underlying methodologies, and performance evaluation of HaemLyst, highlighting its contributions to the field of hematology.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRAK

Sel darah putih (WBC) memainkan peranan penting dalam sistem peredaran darah, berfungsi sebagai pembela utama terhadap jangkitan dan penceroboh asing. Pengesanan dan pengelasan WBC yang tepat adalah penting untuk mendiagnosis pelbagai keadaan hematologi. Tesis ini memperkenalkan HaemLyst, aplikasi web inovatif yang didorong oleh AI untuk menganalisis smear darah dan mengelaskan pelbagai jenis sel darah putih. HaemLyst menggunakan algoritma pembelajaran mesin terkini dan teknik pemprosesan imej untuk mencapai ketepatan tinggi dalam pengelasan WBC. Aplikasi ini menerima imej smear darah yang dimuat naik oleh pengguna, memproses imej tersebut, dan memberikan ramalan terperinci, termasuk pengenalpastian jenis WBC tertentu. Tiga model ramalan yang berbeza digunakan untuk memastikan hasil yang kukuh dan boleh dipercayai, masing-masing menawarkan wawasan unik berdasarkan imej yang telah diproses. Selain ketepatan ramalan, HaemLyst memberi tumpuan kepada penyampaian antara muka pengguna yang intuitif dan moden. Ini termasuk visualisasi yang jelas bagi ramalan, perbandingan hasil antara model, dan paparan label kebenaran tanah untuk pengesahan. Reka bentuk aplikasi yang mesra pengguna menjadikannya boleh diakses oleh kedua-dua profesional perubatan dan penyelidik, memudahkan analisis smear darah yang pantas dan tepat. Melalui ujian dan pengesahan yang komprehensif, HaemLyst telah menunjukkan potensinya untuk meningkatkan kecekapan dan ketepatan pengelasan WBC secara signifikan dalam persekitaran klinikal. Tesis ini memperincikan proses pembangunan, metodologi yang mendasari, dan penilaian prestasi HaemLyst, serta menekankan sumbangannya kepada bidang hematologi.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

TABLE OF CONTENTS

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF TABLES

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF EQUATIONS

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF FIGURES

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF ABBREVIATIONS

AUC	- Area under Curve
AI	- Artificial Intelligence
API	- Application Programming Interface
CPU	- Central Processing Unit
CSV	- Comma Delimited Values
CNN	- Convolutional Neural Network
DNN	- Deep Neural Network
ERD	- Entity Relationship Diagram
FYP	- Final Year Project
Gb	- Gigabytes
GHz	- Gigahertz
GPU	- Graphics Processing Unit
HDD	- Hard Disk Drive
RAM	- Random Access Memory
ReLU	- Rectified Linear Unit
RESNET	- Residual Network
R-CNN	- Regions with Convolutional Neural Networks
ROI	- Region of Interest
[R, G, B]	- [Red, Green, Blue]
px	- Pixels
UI	- User Interface
VGG	- Visual Geometry Group
WBC	- White Blood Cells
WebApp	- Web Application
XAI	- Explainable Artificial Intelligence
XCEPTION	- Extreme Inception

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 1. INTRODUCTION

1.1 Introduction

Blood smear analysis is a cornerstone of diagnostic pathology, traditionally relying on meticulous manual visual examination of stained blood samples under a microscope by pathologists. However, the emergence of machine learning (ML) and artificial intelligence (AI) technologies offers transformative potential in enhancing diagnostic accuracy and efficiency. Pathologists cautiously scrutinize cell morphology, identifying cellular abnormalities such as variations in red blood cells (e.g., anemia, sickle cell disease), white blood cells (e.g., leukopenia, leukocytosis), and platelets (e.g., thrombocytopenia, thrombocytosis). Each abnormality provides essential diagnostic clues, aiding in the accurate identification and management of blood-related conditions. Traditional blood smear analysis, while reliable, faces challenges. Subjective visual interpretation can lead to diagnostic discrepancies, affecting patient outcomes. Additionally, manual inspection's time-intensive nature limits throughput and delays reporting (Smith, 2018). These limitations highlight the demand for more objective, standardized, and efficient diagnostic methods in clinical practice.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The integration of ML and AI technologies in blood smear analysis represents a paradigm shift towards automated, data-driven diagnostic methodologies. ML algorithms, particularly convolutional neural networks (CNNs), excel in pattern recognition tasks by analyzing large datasets of annotated blood smears (Chen et al., 2020). These algorithms can detect subtle cellular abnormalities, such as atypical lymphocytes indicative of lymphocytic leukemia, with unprecedented accuracy and speed (Wang & Zhang, 2021). AI-driven analysis offers several advantages over traditional methods. Firstly, it enhances diagnostic accuracy by minimizing the risk of human error and variability inherent in manual inspection. AI algorithms can learn from vast datasets, continuously improving their diagnostic capabilities through iterative learning processes (Lee & Kim, 2022). Secondly, AI-driven analysis accelerates diagnostic turnaround times, enabling timely interventions and personalized treatment strategies for patients (Davis, 2023). This efficiency not only reduces healthcare costs but also improves patient outcomes by facilitating early detection of diseases.

The adoption of AI technologies in healthcare necessitates careful consideration of ethical implications. Privacy concerns regarding patient data security and algorithm transparency must be addressed to maintain patient trust and compliance with regulatory standards (Davis, 2023). Moreover, the integration of AI requires substantial investments in infrastructure, training, and interdisciplinary collaboration between healthcare professionals and data scientists (Roberts, 2021). Ensuring the responsible deployment and validation of AI algorithms in clinical settings is critical to mitigate potential biases and ensure patient safety.In conclusion, the integration of machine learning and artificial intelligence technology into blood smear analysis holds immense promise for advancing diagnostic pathology. While traditional pathological methods remain foundational, AI-driven analysis offers unparalleled opportunities to enhance diagnostic precision, efficiency, and scalability in clinical practice. Future research should focus on refining AI algorithms, expanding annotated datasets, and addressing ethical considerations to maximize the potential of these technologies in transforming healthcare delivery.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Despite advancements in machine learning (ML) and artificial intelligence (AI) applications in healthcare, there exists a notable research gap in the comparative evaluation of traditional pathological methods versus AI-driven analysis in blood smear diagnostics. Current literature predominantly focuses on the technical capabilities and diagnostic accuracy of AI algorithms in detecting specific hematological abnormalities (Chen et al., 2020; Wang & Zhang, 2021). However, there is a paucity of comprehensive studies that systematically compare the diagnostic efficacy, cost-effectiveness, and practical implementation challenges of AI-driven analysis against traditional pathological methods in real-world clinical settings. Addressing this gap is crucial to substantiate the potential of AI technologies in augmenting, rather than replacing, the expertise of pathologists, thereby optimizing diagnostic workflows and improving patient care outcomes. Moreover, integration of computerized imaging methods, such as AI-powered blood smear analysis, presents a formidable challenge in disease diagnostics, despite its promise of enhanced accuracy and specificity. Though the swift uptake of these advanced technologies underscores the urgent necessity for collaboration among imaging specialists, computer scientists, and pathologists; digital pathology adheres to a structured technology adoption lifecycle, resembling that of other disruptive innovations (Figure 1), where early adopters and innovators have validated its efficacy in clinical settings. However, achieving broader adoption hinges critically on establishing compelling evidence of effectiveness to persuade the majority of stakeholders. This situation highlights the current gap in widespread acceptance, emphasizing the need for robust evidence to substantiate AI's potential to revolutionize blood smear analysis, streamline diagnostic processes, and ultimately improve patient care outcomes. Addressing these deficiencies in the existing body of research is crucial is imperative to substantiate the transformative potential of AI technologies in augmenting diagnostic precision, optimizing resource allocation, and ultimately enhancing patient outcomes in clinical practice.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

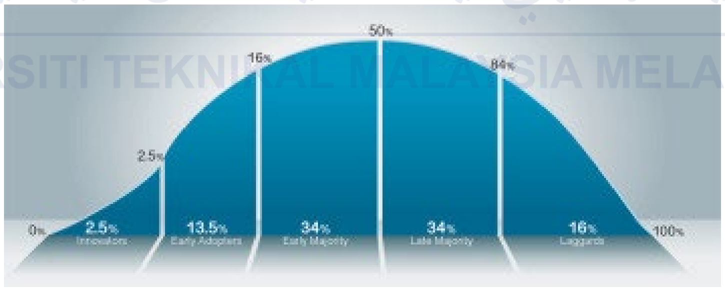


Figure 1.1: Transformation of AI Technologies

To address these critical issues, our proposed research aims to revolutionize the landscape of blood smear analysis through the integration of machine learning and artificial intelligence (AI) technology. Conventional approaches to blood smear examination are not only resource-intensive but also susceptible to errors, resulting in diagnostic delays and treatment inefficiencies. Our initiative aims to streamline this workflow by implementing sophisticated algorithms, thereby shortening the interval from blood sampling to diagnosis and markedly improving diagnostic precision. The envisioned solution, HaemLyst, harnesses cutting-edge AI methodologies to deliver meticulous and dependable outcomes. This dual-module strategy ensures a thorough and expeditious diagnostic procedure, effectively addressing the imperatives of speed and accuracy in clinical diagnostics.

1.2 Problem Statement (PS)

The process of cell detection in blood smears is highly time-consuming and demanding, often requiring the expertise of many pathologists in each department. The challenges and proposed solutions include:

i. High similarity of cells makes it difficult to differentiate each type, requiring multiple experts for cross-validation. Hence, developing an AI-powered web application, HaemLyst, that utilizes advanced image processing and machine learning techniques to automate cell detection and classification, reducing the need for expert cross-validation.

ii. The average cost of machines with centrifuges used for cell detection is RM 3.6 million, relying heavily on metrics like granularity and color. An implementation a cost-effective solution through HaemLyst, leveraging existing computer hardware and sophisticated algorithms to reduce dependency on expensive equipment, while maintaining high accuracy in cell identification. By integrating these solutions, HaemLyst aims to streamline the cell detection process, enhance diagnostic accuracy, and reduce overall costs in medical laboratories.

1.3 Objectives

i. To develop an AI Web Application for blood smear analysis.
ii. To develop a Machine Learning Model with XAI Capabilities.
iii. To evaluate performance of machine learning models.

1.4 Project Scope

1.4.1 Medical and Healthcare Sector

i. Hospitals and Clinics: For automated blood smear analysis aiding diagnosis and treatment.
ii. Medical Laboratories: Streamline blood smear examination, reducing manual workload.
iii. Research Institutions: Tool for analyzing blood smears in various studies.

1.4.2 Specific Users

i. Pathologists and Lab Technicians: Primary users need efficient and accurate analysis in a short period.
ii. Medical Students: Learning tool for studying blood cells and abnormalities.
iii. Researchers: Analyzing large datasets for hematology studies.

1.4.3 Technical Scope

i. CNN Models: Base for image classification and analysis.

1.4.4 Platforms

i. Web Application: Accessible on various OS via web browsers.
ii. Database Systems: Store user data, analysis results, and image datasets.

1.4.5 Compliance and Regulations

i. Healthcare Compliance (e.g., HIPAA): Ensure patient data privacy and security.
ii. Medical Device Regulations: Adhere to standards for clinical use.

1.5 Project Significance

Both medical professionals and patients are the main recipients of this endeavor. By utilizing sophisticated computer vision algorithms to optimize the interpretation of blood smears, numerous substantial advancements might be anticipated:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

i. Minimized Time Consumption: A primary goal of this research is to diminish the amount of time needed for cell detections. By implementing computer vision techniques, the need for manual inspection by pathologists is greatly diminished. This implies reduced waiting periods for test outcomes, enabling expedited identification and treatment determinations.

ii. Improved Accuracy and Precision: Another vital goal is to attain a high level of accuracy and exact comprehension in models for detecting blood cells. Computer vision technologies can evaluate blood smears with a level of detail and consistency that surpasses human skills. The high level of precision guarantees the identification of even minor abnormalities or uncommon cell types with certainty, resulting in more precise diagnoses and tailored treatment strategies.

iii. Cost-effectiveness: Although the upfront expenditure on advanced machinery equipped with centrifuges may appear steep, the long-term advantages surpass the expenses. Through the optimization of the analytical process, the number of resources needed for each inspection is decreased, resulting in lower operational expenses and the maximum utilization of current equipment.

iv. Accessible Expertise: By leveraging computer vision technologies, the expertise of pathologists can be extended beyond geographical restrictions. Automated systems can deliver consistent and reliable analysis to remote or underdeveloped areas with limited access to skilled medical experts.

Overall, the project's important contribution resides in its potential to change blood smear analysis, making it more efficient, accurate, and accessible. By utilizing the potential of computer

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

vision, healthcare systems may enhance patient outcomes, maximize resource use, and propel medical diagnostics into the digital age.

1.6 Expected output

The expected outcome of the research is a easy usable system that allows pathologists to identify the white blood cells present in a blood smear that is taken from an individual's blood. The system produced should be an accessible Web Application that suits a simple framework for this use. Moreover, the system shall be equipped with a well-trained model which can precisely identify and classify the cells present. This is known to be a high accuracy model that is built based on Convolutional Neural Network that is customized for this scenario.

1.7 Report Organization

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Chapter 1: Introduction

This chapter provides the overall background of the project and the advantages to pursue the project.

Chapter 2: Literature Review

This chapter discusses the various studies done in the past that are closely related to the current project. A good number of studies have been taken into account and each of them are summarized by the various methods, tools and techniques that have been used in detection of blood cells. In this chapter we have also discussed the methodology of the project that has been the flow framework of this project.

Chapter 3: Requirement analysis

In this chapter we have the analysis regarding the project. The requirement analysis chapter discusses more intensively the problem and the solution to it. The requirements such as functional, non-functional and other requirements were listed and brought up for analysis. This chapter includes the initial notation that are important for the engagement of the project

Chapter 4: Design

The design chapter covers the method that we proposed to solve the problem that we analyzed earlier in a more technical aspect. The front to backend was declared in this chapter. The entire software that was made in this project (HaemLyst) were well documented from the High Level Design, System architecture, User interface, Database configuration and the crucial AI Component that was used in the project.

Chapter 5: Results and Discussion

The results chapter presents the findings of the project, based on the methodology outlined in Chapter 3. Graphical representations of the results are provided, along with a critical analysis and discussion of the findings. The chapter relates the results to the proposed technique and provides technical justifications.

Chapter 6: Conclusion

In the conclusion chapter, the project is summarized based on its set objectives. The achievements of the project are discussed, highlighting its contributions to the university, faculty, company, or individual. The project's limitations are also stated, along with suggestions for future improvements. Finally, the chapter concludes with a reflection on whether the project meets its objectives conclusively.

1.8 Summary

This first chapter establishes the context by emphasizing the urgent and essential requirement for prompt and dependable diagnostic testing, particularly in regions with limited resources such as developing countries. This statement highlights the constraints of conventional diagnostic techniques and the difficulties encountered in identifying pathogens when they are present in small amounts. The suggested research seeks to transform blood smear analysis by incorporating machine learning and artificial intelligence (AI) technology. The research aims to speed up cell detections, enhance accuracy, and streamline diagnosis and treatment decisions by utilizing computer vision algorithms. The research aims to decrease time consumption, attain high precision in blood cell detection models, and ensure cost-effectiveness. Additionally, it seeks to enhance the accessibility of expertise by expanding the influence of pathologists through automated methods. In summary, the initiative can revolutionize blood smear analysis by improving its efficiency, accuracy, and accessibility. Through the usage of computer vision, this program shows potential for improving patient outcomes, optimizing resource allocation, and advancing medical diagnostics in the digital era.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 2. LITERATURE REVIEW

2.1. Introduction

In this chapter, a literature review for some of the existing systems related to the proposed work and the Neural Network and Machine Learning based approach classification used in the systems. Compare the existing systems with the proposed system by reviewing the existing projects architecture, techniques, modes and features.

2.2. Facts and Findings

This section will cover the project's domains, which will be helpful in comprehending the project's key goals and its conclusion.

2.2.1 Domain

In this section, we will discuss the depth of domains associated with this project, which will aid in understanding its core value and completion.

i. White blood cell

A kind of blood cell produced in the bone marrow and found in the blood and lymphatic tissues. White blood cells are components of the body's immune system. They assist the body in fighting infections and other disorders. White blood cells are classified into three types: granulocytes (neutrophils, eosinophils, and basophils), monocytes, and lymphocytes. A complete blood cell (CBC) test typically includes a measurement of the amount of white blood cells in the blood. It may be used to detect infection, inflammation, allergies, and leukemia. Also known as leukocyte and WBC.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

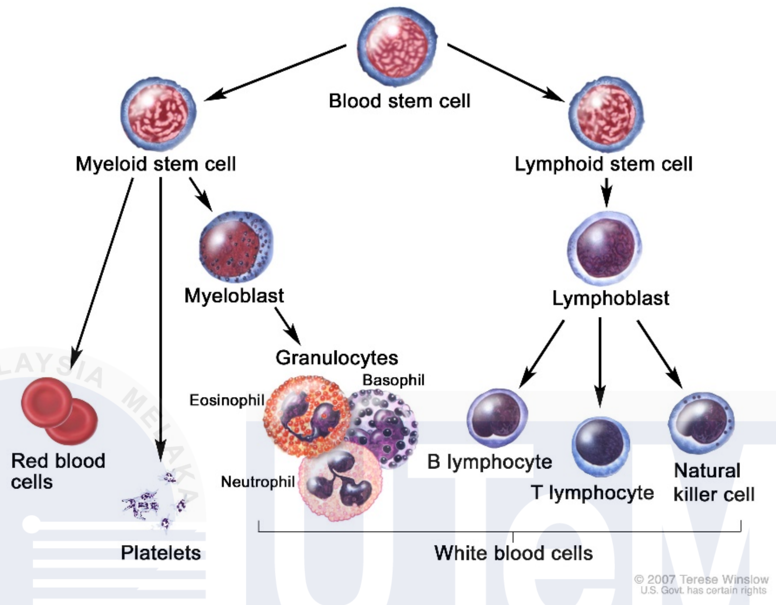


Figure 2.1 : The types of white blood cell and its hierarchy

White blood cells defend your body against infection. like your white blood cells travel through your bloodstream and tissues, they discover the source of an infection and serve like an army general, alerting other white blood cells to their location in order to protect your body from an unknown organism's onslaught. When your white blood cell army comes, it fights the invader by creating antibody proteins, which adhere to and destroy the creature.

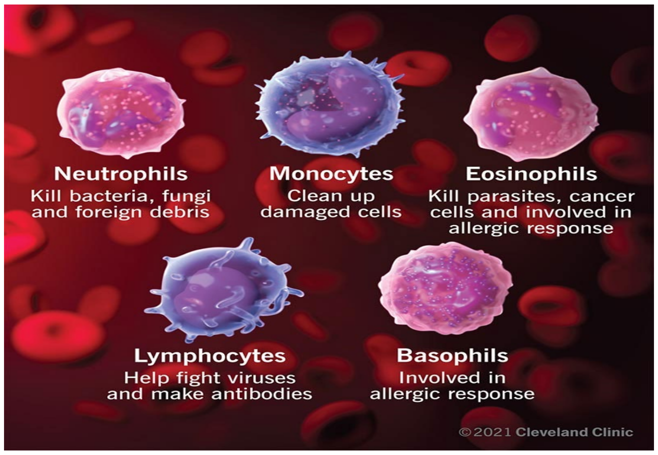

Neutrophils
Kill bacteria, fungi and foreign debris
Monocytes
Clean up damaged cells
Eosinophils
Kill parasites, cancer cells and involved in allergic response
Lymphocytes
Help fight viruses and make antibodies
Basophils
Involved in allergic response
©2021 Cleveland Clinic

Figure 2.2 : Types of white blood cell and its role

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

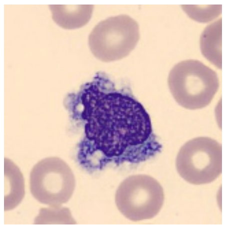
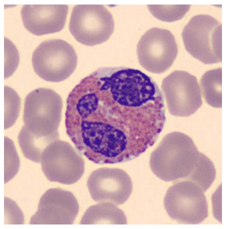
There are five different types of white blood cells. Neutrophils help your body fight infections by destroying bacteria, fungus, and foreign debris. Lymphocytes, which are made up of T cells, natural killer cells, and B cells, protect against viral infections and create proteins that aid in infection resistance. Eosinophils identify and eliminate parasites, cancer cells, and help basophils with the allergic response. Basophils cause allergic reactions such as coughing, sneezing, or a runny nose. Monocytes prevent infection by cleaning up damaged cells.

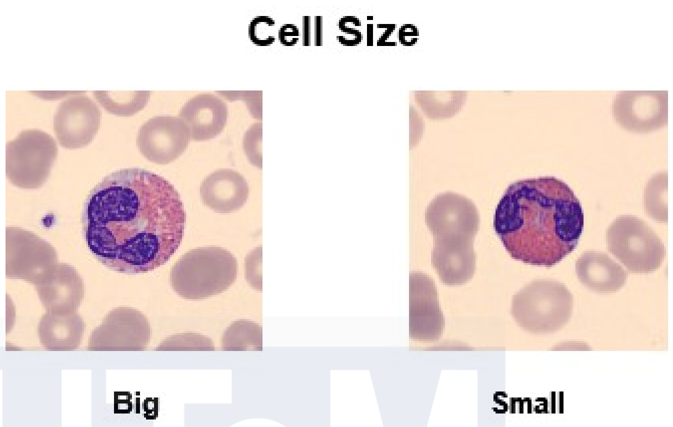
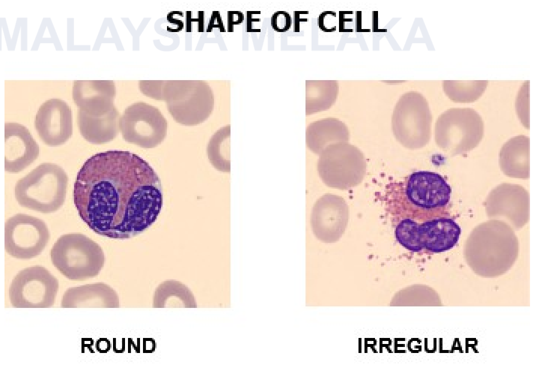
White blood cells have many different attributes that make each cell distinctive from one another. For example, cell type, cell size, cell shape, nucleus shape, nuclear-cytoplasmic ratio, chromatin density, cytoplasm vacuole, cytoplasm texture, cytoplasm colour, granule type, granule colour, granularity.

ii. Terminologies

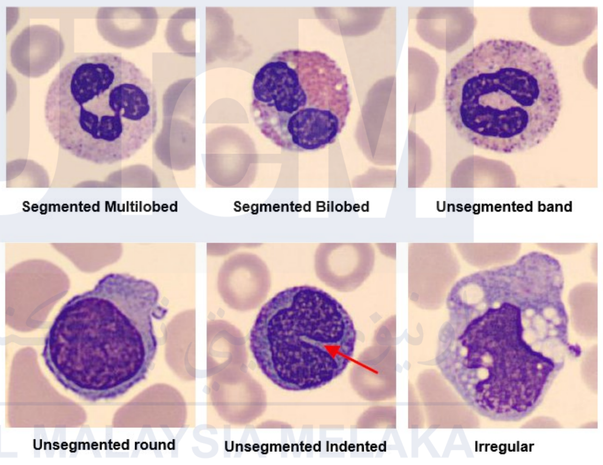
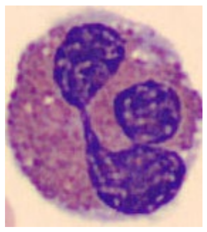
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 2.1 : Terminology used in this project

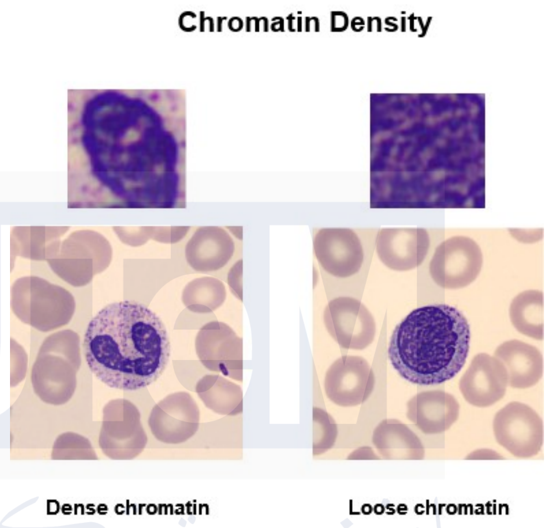
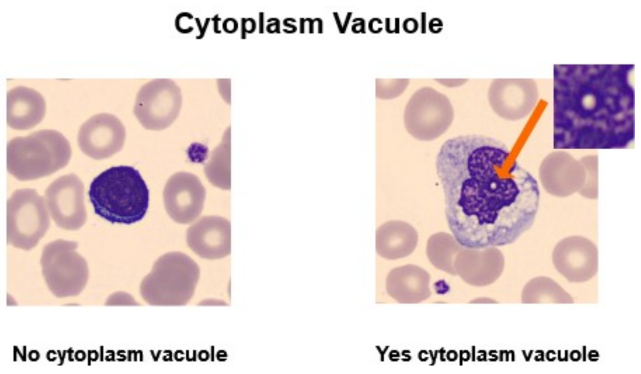
TERM	EXPLANATION	REFERENCE
Cell type	Type of WBC	 MONOCYTE  ESONOPHIL

Cell size	Size of individual cell	
Cell shape	General shape of cell	

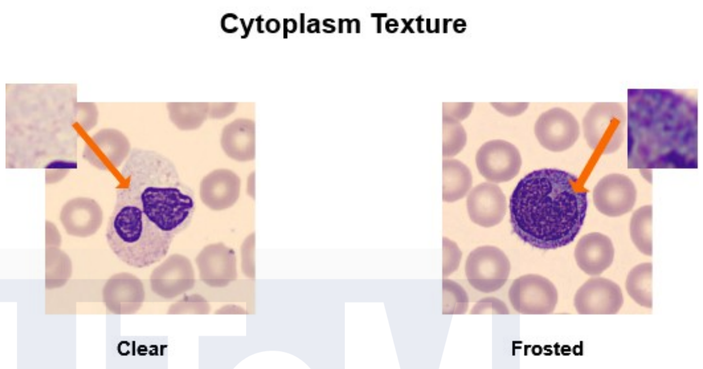
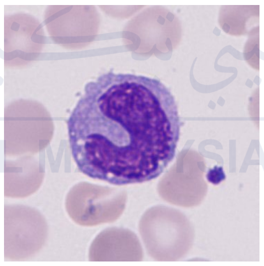
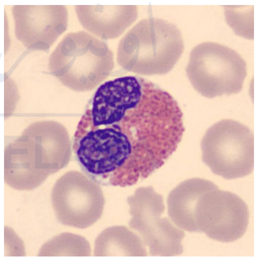
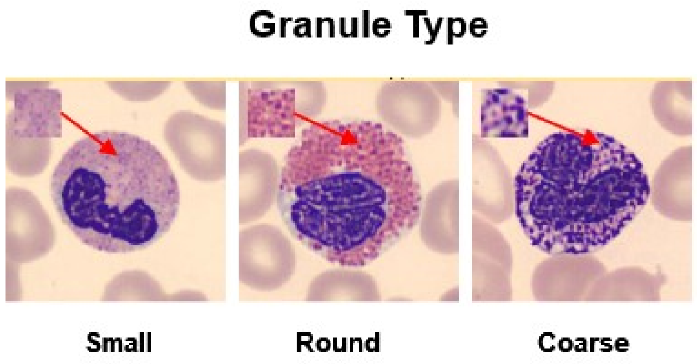
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Nucleus shape	General shape of nucleus if present	NUCLEUS SHAPE  Segmented Multilobed Segmented Bilobed Unsegmented band Unsegmented round Unsegmented Indented Irregular
Nuclear-cytoplasmic ratio	Ratio of nucleus and cytoplasm constructing the cell. *(Ratio of purple and pink area)*	

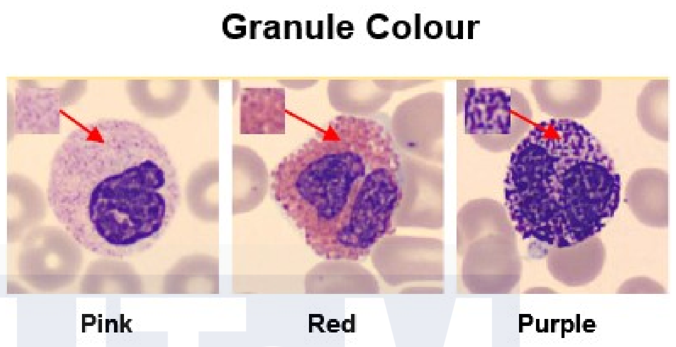
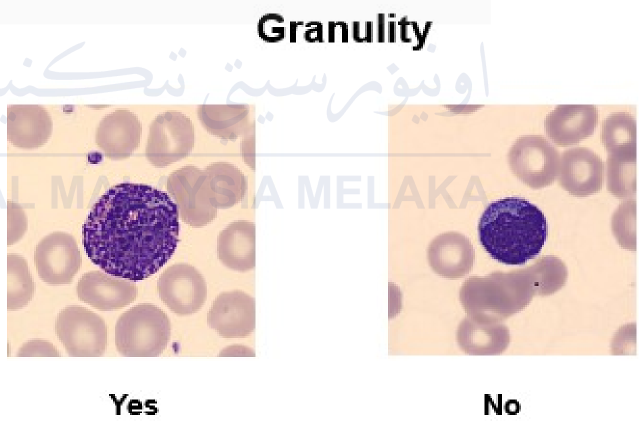
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Chromatin density	A regulatory component of gene expression in a cell population	
Vacuole presence	The presence of vacuole	 Cytoplasm Vacuole No cytoplasm vacuole Yes cytoplasm vacuole

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Cytoplasm texture	Texture of cytoplasm of the cell	
Cytoplasm colour	Colour of the cytoplasm	  Purple CYT Pink CYT
Granule type	Type of granule present	

UNIVERSITI TEKNIKAL MALAYSIA MELAKA UTeM

Granule colour	Colour of the granule	
Granularity	How fine or coarse the cell	

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.2.2 Related Work/Previous Work

Table 2.2 : Summary of Previous Work and its Review

System	Review	Hardware and software
White blood cells detection and classification based on regional convolutional neural networks. (Hüseyin Kutlu *et al.)*	White blood cells detection and classification based on regional convolutional (Hüseyin Kutlu *et al.)* neural networks utilized Regional Convolutional Neural Networks (R-CNN) to classify different cell types within the same image. The methodology involved training a Convolutional Neural Network (CNN), which is the basis of R-CNN architecture, using various architectures including AlexNet, VGG16, GoogLeNet, and ResNet50 **Overall System Performance**: The system achieved a 100% success rate in identifying White Blood Cells (WBCs). **Best Performing Architecture**: ResNet50 demonstrated the highest performance when employing transfer learning.	

Cell Type	**Accuracy Rate (%)**
Lymphocyte	99.52
Monocyte	98.40

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	Basophil: 98.48 Eosinophil: 96.16 Neutrophil: 95.04	
White blood cell subtype detection and classification based on the YOLOv3 object detection algorithm. *(Nalla Praveen et al.)*	The proposed method mainly consists of two phases. Phase 1 is WBC bounding box detection from the blood cell image, Phase 2 is classification and detection of WBC subtypes. The proposed model can classify white blood cells subtype with 90% accuracy. The experimental results of YOLOv3 model compared to faster RCNN using VGG16 for classification and detection tasks are shown in table I and table II. The YOLOv3 was able to achieve better results due to its ability to consider the contextual information about the available classes by analyzing the entire image in a single go, thereby also making it faster than any other model.	

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

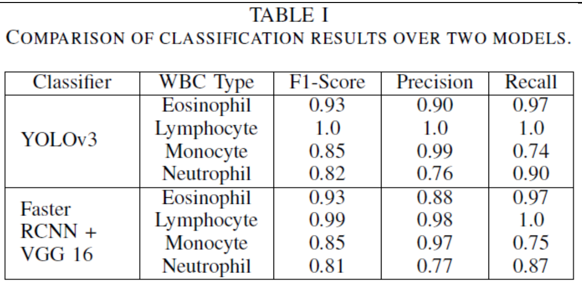


TABLE I
COMPARISON OF CLASSIFICATION RESULTS OVER TWO MODELS.

Classifier	WBC Type	F1-Score	Precision	Recall
YOLOv3	Eosinophil	0.93	0.90	0.97
	Lymphocyte	1.0	1.0	1.0
	Monocyte	0.85	0.99	0.74
	Neutrophil	0.82	0.76	0.90
Faster RCNN + VGG 16	Eosinophil	0.93	0.88	0.97
	Lymphocyte	0.99	0.98	1.0
	Monocyte	0.85	0.97	0.75
	Neutrophil	0.81	0.77	0.87

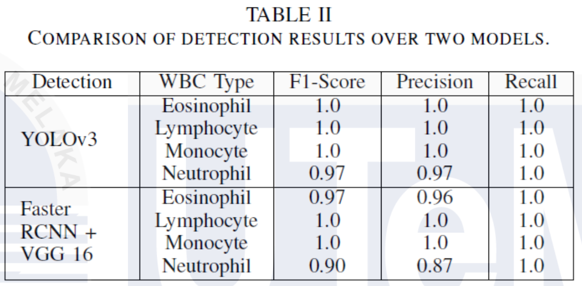
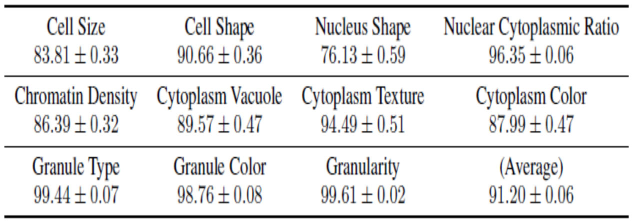


TABLE II
COMPARISON OF DETECTION RESULTS OVER TWO MODELS.

Detection	WBC Type	F1-Score	Precision	Recall
YOLOv3	Eosinophil	1.0	1.0	1.0
	Lymphocyte	1.0	1.0	1.0
	Monocyte	1.0	1.0	1.0
	Neutrophil	0.97	0.97	1.0
Faster RCNN + VGG 16	Eosinophil	0.97	0.96	1.0
	Lymphocyte	1.0	1.0	1.0
	Monocyte	1.0	1.0	1.0
	Neutrophil	0.90	0.87	1.0

WBCAtt: A White Blood Cell Dataset Annotated with Detailed Morphological Attributes (et.al Satoshi Tsutsui)	Using supervised learning with a broader dataset using various algorithms and fine-tuning processes. This expands the horizon of White Blood Cell Classsification to a more confirmatory test that pathologist can rely on. The used method provides an XAi domain for classification of White blood cell by answering why is this a Monocyte or why is this not a Monocyte. The results are presented in table below. The baseline model achieves an average macro-F-measure of 91.20 ± 0.06%. Some attributes, such as granularity, granule type, and granule color, exhibit particularly high F-measures of over 98%. On the other hand, nucleus shape is the most challenging attribute to predict, with the lowest F-measure of 76.13 ± 0.59%, which could be due to the complexity of the nucleus shape.	Label Studio Grad-CAM Python 3.8.16 Pytorch 2.0.0

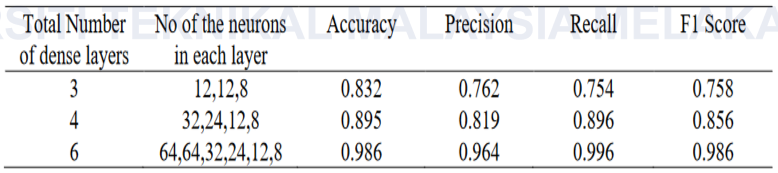
UNIVERSITI TEKNIKAL MALAYSIA MELAKA



Cell Size	Cell Shape	Nucleus Shape	Nuclear Cytoplasmic Ratio
83.81 ± 0.33	90.66 ± 0.36	76.13 ± 0.59	96.35 ± 0.06
Chromatin Density	Cytoplasm Vacuole	Cytoplasm Texture	Cytoplasm Color
86.39 ± 0.32	89.57 ± 0.47	94.49 ± 0.51	87.99 ± 0.47
Granule Type	Granule Color	Granularity	(Average)
99.44 ± 0.07	98.76 ± 0.08	99.61 ± 0.02	91.20 ± 0.06

Impact of Hidden Dense Layers in Convolutional Neural Networks to Enhance the Performance of Classification Models" by Helen Josephine et al.

The study constructed three different classification deep models to analyze and predict diabetes. The research was done based on the Deep Convolutional Neural Network and was estimated through a famous diabetes database called Pima Indian Diabetes Database (PIDD) downloaded from the UC Irvine. Machine. Learning. Repository. This dataset includes 768 instances. Dataset has 9 important attributes to identify diabetic disease. The research was done with addition of 3, 4 and 6 dense layers on the Classification Deep Model. The performance of the architecture is shown below.



Total Number of dense layers	No of the neurons in each layer	Accuracy	Precision	Recall	F1 Score
3	12,12,8	0.832	0.762	0.754	0.758
4	32,24,12,8	0.895	0.819	0.896	0.856
6	64,64,32,24,12,8	0.986	0.964	0.996	0.986

The initial deep classification model consists of 3 hidden layers, containing 12, 12, and 8 neurons, respectively. The classification performance is evaluated using four key metrics: Accuracy, Precision, Recall, and F1 Score. In the second classification model, performance improved by 7% in accuracy, **8% in precision, 15% in recall, and 10% in F1 Score.** The final test was conducted using a deep classification model with 6 hidden layers, featuring 64, 64, 32, 24, 12, and 8 neurons, respectively. The CNN demonstrated superior performance compared to other

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	classification algorithms. The final experimental results yielded 99% accuracy, 96% precision, 100% recall, and 98% F1 Score. Overall, classification accuracy improved from **83% with 3 dense layers to 99% with 6 dense layers**. The study has also concluded that more hidden layers with the deeper the network would produce better classification until the saturation point is reached.	

2.2.3 Machine Learning and Deep Learning Techniques

Table 2.3 : Technique used in the previous research

Mode	Review
Regional Convolutional Neural Networks (R-CNN)	Mask R-CNN is a state-of-the-art model for instance segmentation in images, specifically designed to identify and create masks for individual objects within an image. The architecture consists of four main components: feature extraction, region proposal, region of interest (RoI) alignment, and prediction. (Sharma, P. 2024) **Feature Extraction**: Uses ResNet as a Feature Pyramid Network (FPN) to extract features from different levels of the image. The FPN architecture allows for both high-level and low-level feature extraction, improving the accuracy of the segmentation. **Region Proposal**: Utilizes a small CNN known as the Region Proposal Network (RPN) with a sliding window approach to identify potential objects by generating bounding boxes (anchors).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	These anchors are filtered based on their Intersection-over-Union (IoU) ratio, with a threshold of 0.65 to ensure high accuracy in cell detection. **Region of Interest (RoI) Alignment**: Applies (RoIAlign) to the bounding boxes generated by the RPN to correct misalignments caused by traditional RoIPool techniques. Uses bi-linear interpolation to refine feature values, ensuring precise aggregation of the results. (Potrimba, 2024) **Prediction**: Feeds the aligned feature maps into a fully connected neural network to predict the bounding boxes and their corresponding classes. A final CNN layer is used to predict the mask for each detected object, completing the instance segmentation process. In summary, Mask R-CNN leverages ResNet and FPN for effective feature extraction, RPN for proposing regions of interest, (RoIAlign) for precise feature alignment, and a combination of fully connected and convolutional layers for accurate prediction of bounding boxes and masks. (Keylabs, 2024)
YOLOv3 Object Detection Algorithm	The proposed method for white blood cell (WBC) classification and detection is based on the YOLOv3 object detection algorithm a Convolutional Neural Network and consists of two main phases: **<u>Phase 1: WBC Bounding Box Generation</u>** **Objective**: To generate bounding boxes around WBCs in blood cell images.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	Process: Uses 364 WBC images with corresponding annotations (bounding box coordinates) for training. Images and annotations are preprocessed. YOLOv3 generates bounding boxes around WBCs with confidence scores. These bounding boxes are used as the training set for Phase 2. This phase can also be used to count red blood cells (RBCs), WBCs, and platelets in blood cell images. **Output**: Bounding boxes of WBCs detected in the images. **<u>Phase 2: Classification and Detection of WBC Subtypes</u>** **Objective**: To classify WBCs into subtypes and detect their bounding boxes. **Process**: Input consists of images and bounding boxes generated in Phase 1.The task is a multi-class classification problem with four WBC subtypes.YOLOv3 classifies each WBC and predicts its bounding box, class probabilities, and confidence score. **Output**: Predicted class, bounding box of the WBC subtype, class probabilities, and confidence score for each detected WBC. In summary, the method first detects WBCs in images (Phase 1) and then classifies these WBCs into subtypes (Phase 2), using bounding boxes and confidence scores to ensure accurate detection and classification.
Convolutional Neural Networks using ImageNet-pretrained ResNet50	The baseline model for white blood cell (WBC) classification utilizes Convolutional Neural Networks (CNNs) and employs an ImageNet-pretrained ResNet50 as the image encoder. The key aspects of this model are:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

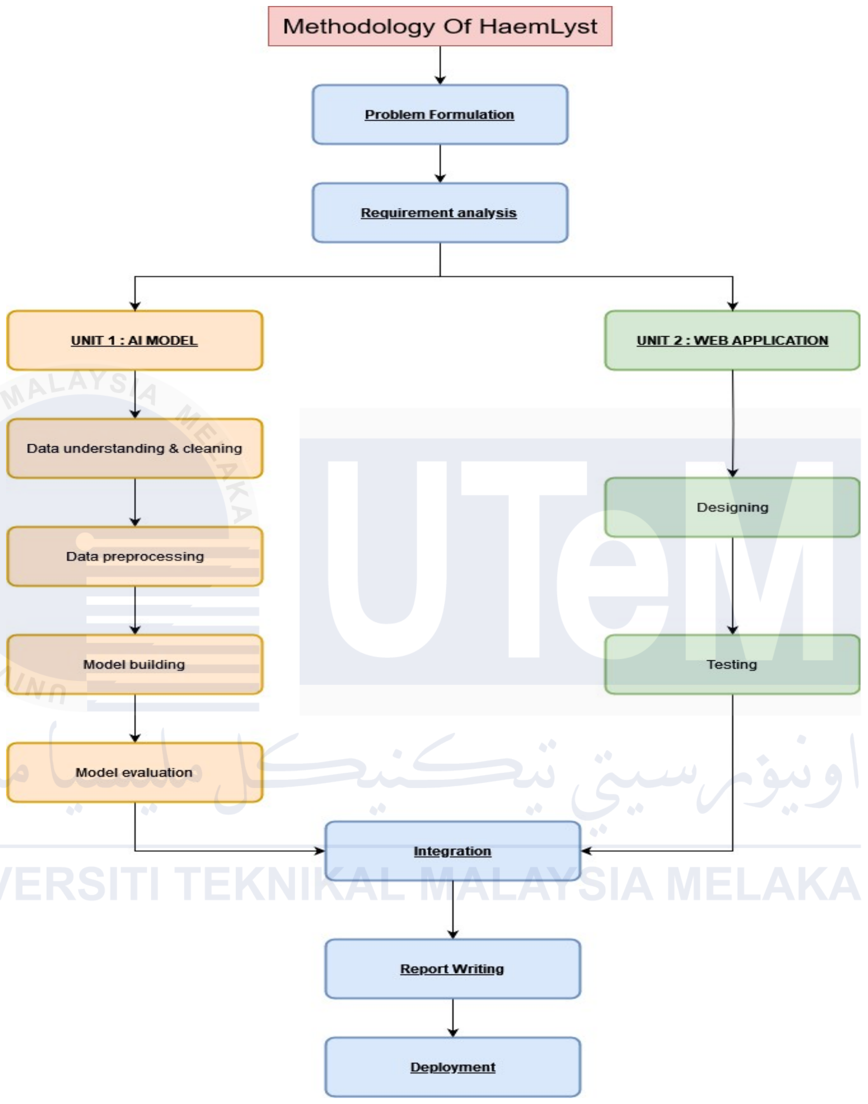
	Image Encoder: Utilizes a ResNet50 model pretrained on ImageNet to extract image representations. **Attribute Prediction**: Features multi-task heads consisting of per-attribute linear layers, with each layer dedicated to predicting a specific attribute of WBCs.

Based on the literature reviewed, we have selected Convolutional Neural Networks (CNNs) as the foundation for classifying white blood cells in microscopic images of blood smears. CNNs are ideal for this task because they use a supervised learning technique, allowing us to train the model effectively with labeled images and multiple labels, which suits our scenario well. This approach is particularly beneficial for image classification tasks as CNNs can automatically and adaptively learn spatial hierarchies of features from input images, making them highly effective for recognizing and classifying patterns within the images. Furthermore, deep learning methods like CNNs can enhance the system's accuracy and tunability, leading to more reliable results. By employing CNNs, we also pave the way for developing an Explainable AI (XAI) system, which not only aims for high accuracy but also provides insights into the model's decision-making process. Moreover, the finetuning with additional layers which were reviewed are also a part of the systems model training routine and architecture tuning to create a more precise and accurate model. This makes the system more transparent and trustworthy, essential features for medical applications where understanding the rationale behind a classification is as important as the classification itself.

2.3. Project Methodology

For this project we have adapted a modified methodology that suits the objective of the project which consists of Problem Formulation and Requirements followed by two separate units – Unit 1: Modeling and Unit 2: Web Application Development. Both these units were developed in a parallel form where both are then finalized in the integration phase.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.3 : Flow Diagram of HaemLyst Methodology

2.3.1 Problem Formulation

Problem formulation is the step in problem definition that is used to understand and decide a course of action that needs to be considered to achieve our goal (Babu, C.V.S. (2021). In this project the formulation follows as below:

i. Problem Statement

The analysis of white blood cells in a blood smear is challenging and resource-intensive, requiring significant expertise to achieve accurate results. Accurate white blood cell analysis is essential for understanding various medical conditions, necessitating precise and reliable assessments by pathologists or automated systems. This process often demands expert opinions, which may not be readily available, particularly in resource-limited settings.

ii. Problem Definition

The key challenges in white blood cell analysis include the resource intensiveness of accurate analysis, which requires advanced laboratory equipment and skilled personnel that may not be accessible in all settings. High precision and accuracy are crucial for effective diagnosis and treatment, necessitating expert involvement. Additionally, reliance on expert pathologists can lead to delays and inconsistencies, particularly in areas with a shortage of trained professionals

iii. Problem Limitations

The accuracy of AI models heavily depends on the quality and quantity of the training data. Limited access to high-quality, annotated blood smear images can impact the performance of the AI system, and variability in data sources, such as differences in staining techniques or image capture quality, may affect the consistency and reliability of the analysis. Additionally, data focused on white blood cell attributes are often limited and vague, lacking numerical values to indicate sizes, for instance. Ethical concerns related to data privacy, patient consent, and the potential for algorithmic bias must also be carefully addressed to build trust and ensure responsible use.

2.3.2 Requirements Analysis

Since we have two separate modules, we must proceed with distinct requirements analysis phases for the AI model and the web application to be developed. This separation ensures that each component is developed to meet its specific needs and functionalities efficiently.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The web application should be designed to provide a user-friendly interface for analyzing blood smears, with a particular focus on white blood cells. It must be developed to be simple and safe, ensuring that users can upload and analyze blood smear images without any risk to data security or patient confidentiality. The system should offer seamless integration with the AI model to deliver accurate and timely analysis results. Additionally, the web application should feature robust user authentication and data encryption to protect sensitive information.

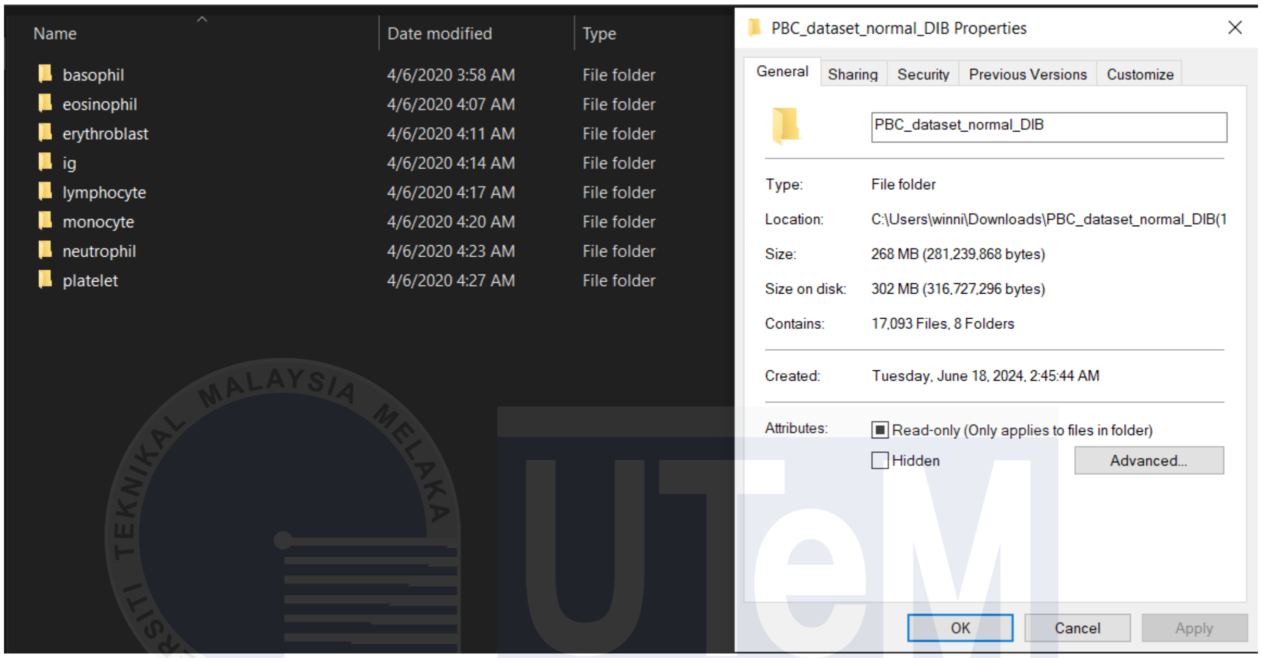
The AI model requires the development of a fine-tuned neural network capable of utilizing explainable AI (XAI) methods to ensure transparency and interpretability of its predictions. This is crucial for gaining the trust of healthcare professionals and for complying with regulatory requirements. The model should support multi-class classification to accurately distinguish between different types of white blood cells. It must be trained on a diverse and high-quality dataset to improve its robustness and reliability across various clinical scenarios.

اونيورسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.3.3 AI Model

2.3.3.1 Data Understanding and Cleaning

For this project we have two different formats of data – label in Comma Delimited Values (csv) file and images file that are in (.jpg) format. The images dataset contains a total of 17,093 images of individual normal cells, which were acquired using the analyzer CellaVision DM96 in the Core Laboratory at the Hospital Clinic of Barcelona.



Name	Date modified	Type
basophil	4/6/2020 3:58 AM	File folder
eosinophil	4/6/2020 4:07 AM	File folder
erythroblast	4/6/2020 4:11 AM	File folder
ig	4/6/2020 4:14 AM	File folder
lymphocyte	4/6/2020 4:17 AM	File folder
monocyte	4/6/2020 4:20 AM	File folder
neutrophil	4/6/2020 4:23 AM	File folder
platelet	4/6/2020 4:27 AM	File folder

PBC_dataset_normal_DIB Properties

General | Sharing | Security | Previous Versions | Customize

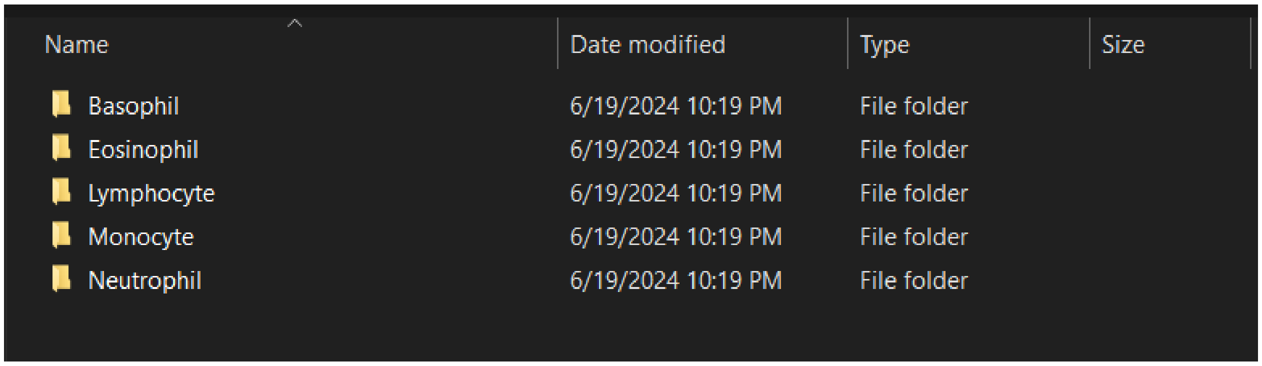
PBC_dataset_normal_DIB

Type: File folder
Location: C:\Users\winni\Downloads\PBC_dataset_normal_DIB(1
Size: 268 MB (281,239,868 bytes)
Size on disk: 302 MB (316,727,296 bytes)
Contains: 17,093 Files, 8 Folders
Created: Tuesday, June 18, 2024, 2:45:44 AM
Attributes: [■] Read-only (Only applies to files in folder)
[ ] Hidden

Figure 2.4 : Initial data present in the image's dataset

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

However, the annotations files containing 11 different attributes were provided by Satoshi Tsutsui, Winnie Pang, and Bihan Wen. WBCAtt: A White Blood Cell Dataset Annotated with Detailed Morphological Attributes only contains only 10,300 lines of data which mirrors the exact number of images which are deeply annotated. To overcome this problem, we have cleaned the image data by keeping aside the unannotated images for future uses.



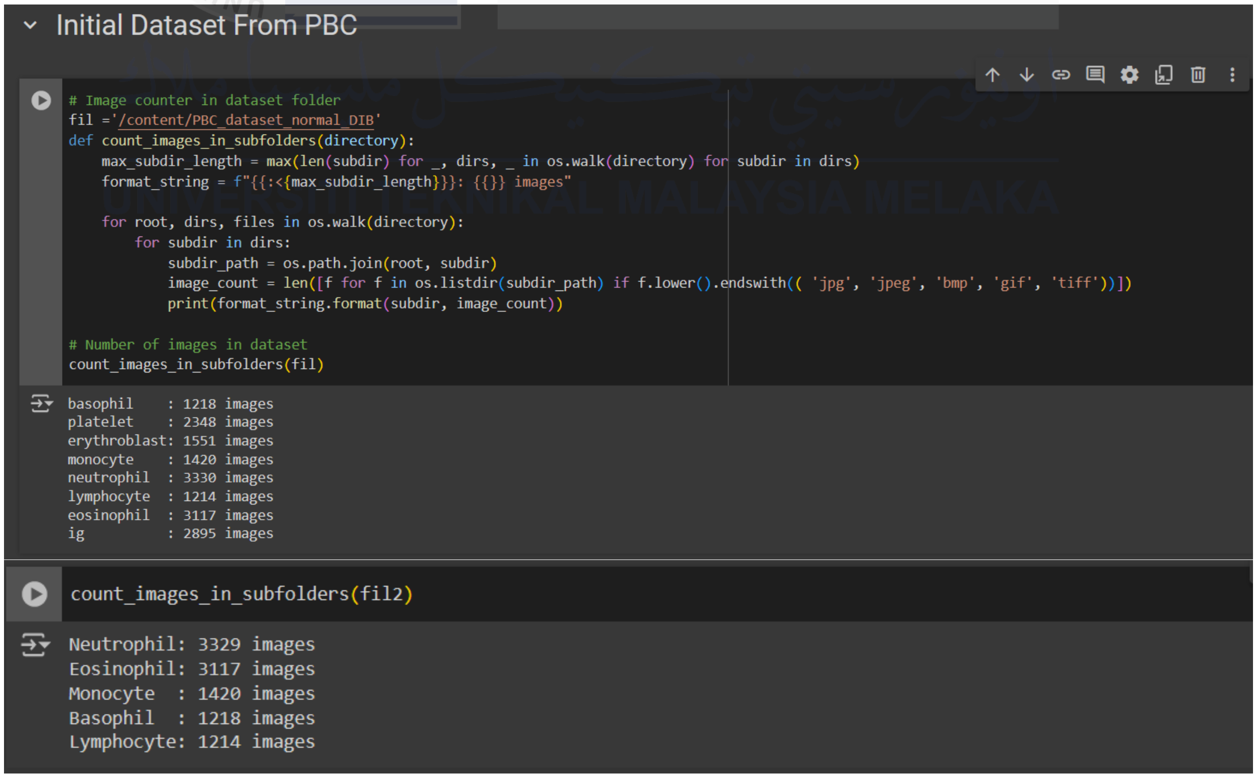
Name	Date modified	Type	Size
Basophil	6/19/2024 10:19 PM	File folder	
Eosinophil	6/19/2024 10:19 PM	File folder	
Lymphocyte	6/19/2024 10:19 PM	File folder	
Monocyte	6/19/2024 10:19 PM	File folder	
Neutrophil	6/19/2024 10:19 PM	File folder	

Figure 2.5 : Cleaned data present in the image's dataset

Table 2.4 : Comparison of before and after cleaning of dataset

INITIAL DATASET		**CLEANED DATASET**	
Basophil	: 1218 images	Basophil	: 1218 images
Eosinophil	: 3,117 images	Eosinophil	: 3,117 images
Erythroblast	: 1551 images	Lymphocyte	: 1214 images
Immature granulocyte	: 2,895 images	Monocyte	: 1420 images
Lymphocyte	: 1214 images	Neutrophil	: 3329 images
Monocyte	: 1420 images		
Neutrophil	: 3330 images		
Platelets	: 2348 images		

UNIVERSITI TEKNIKAL MALAYSIA MELAKA UTeM



```
# Image counter in dataset folder
fil ='/content/PBC_dataset_normal_DIB'
def count_images_in_subfolders(directory):
    max_subdir_length = max(len(subdir) for _, dirs, _ in os.walk(directory) for subdir in dirs)
    format_string = f"{{:<{max_subdir_length}}}: {{}} images"

    for root, dirs, files in os.walk(directory):
        for subdir in dirs:
            subdir_path = os.path.join(root, subdir)
            image_count = len([f for f in os.listdir(subdir_path) if f.lower().endswith(( 'jpg', 'jpeg', 'bmp', 'gif', 'tiff'))])
            print(format_string.format(subdir, image_count))

# Number of images in dataset
count_images_in_subfolders(fil)
```

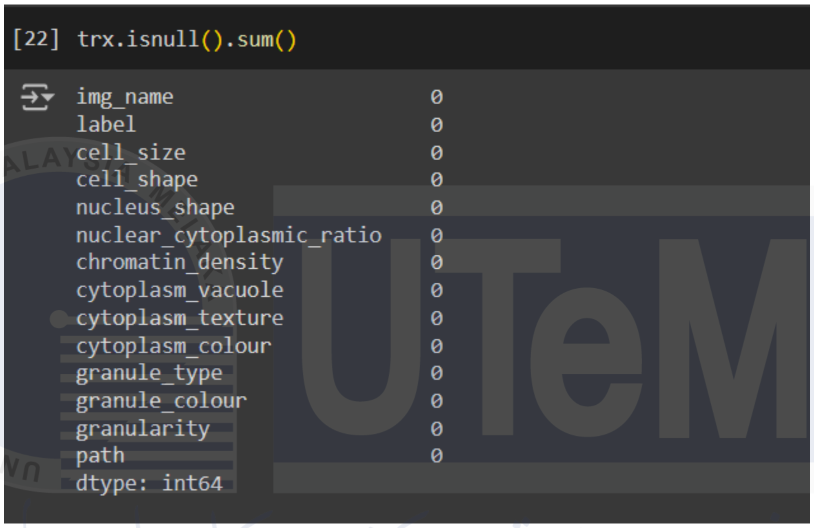
```
basophil    : 1218 images
platelet    : 2348 images
erythroblast: 1551 images
monocyte    : 1420 images
neutrophil  : 3330 images
lymphocyte  : 1214 images
eosinophil  : 3117 images
ig          : 2895 images
```

```
count_images_in_subfolders(fil2)
```

```
Neutrophil: 3329 images
Eosinophil: 3117 images
Monocyte  : 1420 images
Basophil  : 1218 images
Lymphocyte: 1214 images
```

Figure 2.6 : Code snippet of images count by cell type before and after cleaning of data

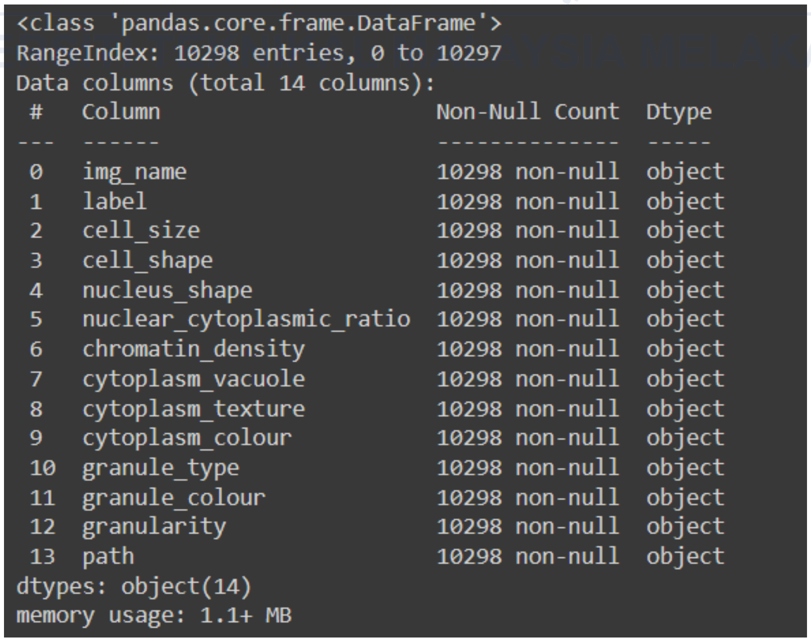
On the other hand, we attempted to clean the CSV containing the annotations, but it wasn't necessary due to the completeness and error free data. We ran a quick summary of null values and the information mining on the dataset, but it was cleaned since the initial phase. The path column was removed as it is just a specifier that is redundant to the model.



```
[22] trx.isnull().sum()

     img_name                    0
     label                       0
     cell_size                   0
     cell_shape                  0
     nucleus_shape               0
     nuclear_cytoplasmic_ratio   0
     chromatin_density           0
     cytoplasm_vacuole           0
     cytoplasm_texture           0
     cytoplasm_colour            0
     granule_type                0
     granule_colour              0
     granularity                 0
     path                        0
     dtype: int64
```

Figure 2.7 : Annotation data null count



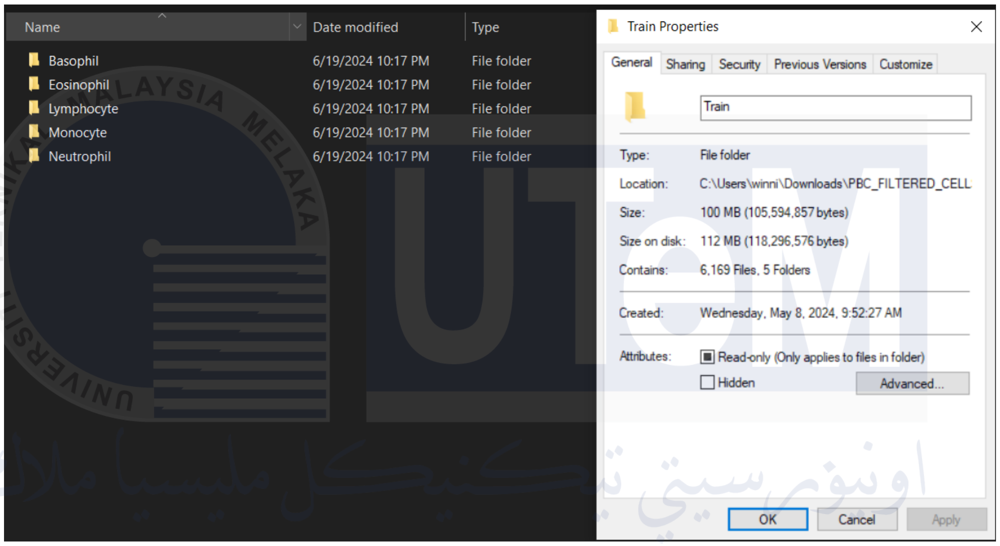
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10298 entries, 0 to 10297
Data columns (total 14 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   img_name                   10298 non-null  object
 1   label                      10298 non-null  object
 2   cell_size                  10298 non-null  object
 3   cell_shape                 10298 non-null  object
 4   nucleus_shape              10298 non-null  object
 5   nuclear_cytoplasmic_ratio  10298 non-null  object
 6   chromatin_density          10298 non-null  object
 7   cytoplasm_vacuole          10298 non-null  object
 8   cytoplasm_texture          10298 non-null  object
 9   cytoplasm_colour           10298 non-null  object
 10  granule_type               10298 non-null  object
 11  granule_colour             10298 non-null  object
 12  granularity                10298 non-null  object
 13  path                       10298 non-null  object
dtypes: object(14)
memory usage: 1.1+ MB
```

Figure 2.8 : The summary of not empty columns

2.3.3.2 Data preprocessing

i. Image dataset

The dataset of images was then further modified/ transformed into three categories – training, testing and validation. The training set contains 6169 images, training set contained 3099 images and validation set contains 1030 images, which were all randomly picked all over the label file.



Name	Date modified	Type
Basophil	6/19/2024 10:17 PM	File folder
Eosinophil	6/19/2024 10:17 PM	File folder
Lymphocyte	6/19/2024 10:17 PM	File folder
Monocyte	6/19/2024 10:17 PM	File folder
Neutrophil	6/19/2024 10:17 PM	File folder

Train Properties

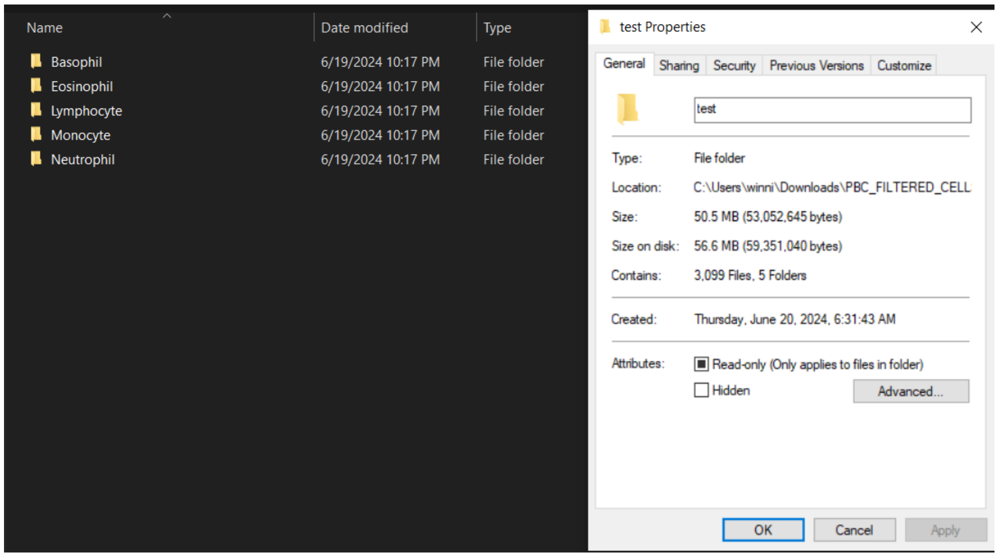
General | Sharing | Security | Previous Versions | Customize

Train

Type:	File folder
Location:	C:\Users\winni\Downloads\PBC_FILTERED_CELL
Size:	100 MB (105,594,857 bytes)
Size on disk:	112 MB (118,296,576 bytes)
Contains:	6,169 Files, 5 Folders
Created:	Wednesday, May 8, 2024, 9:52:27 AM
Attributes:	■ Read-only (Only applies to files in folder)
	☐ Hidden

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.9 :Numbers of training set and structure of folder



Name	Date modified	Type
Basophil	6/19/2024 10:17 PM	File folder
Eosinophil	6/19/2024 10:17 PM	File folder
Lymphocyte	6/19/2024 10:17 PM	File folder
Monocyte	6/19/2024 10:17 PM	File folder
Neutrophil	6/19/2024 10:17 PM	File folder

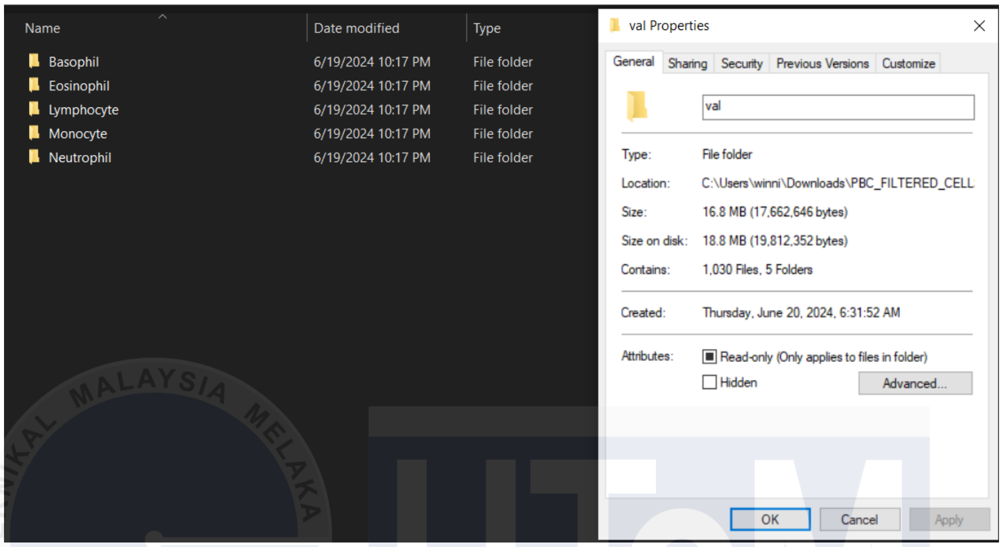
test Properties

General | Sharing | Security | Previous Versions | Customize

test

Type:	File folder
Location:	C:\Users\winni\Downloads\PBC_FILTERED_CELL
Size:	50.5 MB (53,052,645 bytes)
Size on disk:	56.6 MB (59,351,040 bytes)
Contains:	3,099 Files, 5 Folders
Created:	Thursday, June 20, 2024, 6:31:43 AM
Attributes:	■ Read-only (Only applies to files in folder)
	☐ Hidden

Figure 2.10 : Numbers of testing set and structure of folder



Name	Date modified	Type
Basophil	6/19/2024 10:17 PM	File folder
Eosinophil	6/19/2024 10:17 PM	File folder
Lymphocyte	6/19/2024 10:17 PM	File folder
Monocyte	6/19/2024 10:17 PM	File folder
Neutrophil	6/19/2024 10:17 PM	File folder

val Properties

General | Sharing | Security | Previous Versions | Customize

val

Type:	File folder
Location:	C:\Users\winni\Downloads\PBC_FILTERED_CELL
Size:	16.8 MB (17,662,646 bytes)
Size on disk:	18.8 MB (19,812,352 bytes)
Contains:	1,030 Files, 5 Folders
Created:	Thursday, June 20, 2024, 6:31:52 AM
Attributes:	■ Read-only (Only applies to files in folder) ☐ Hidden

Figure 2.11 : Numbers of validation set and structure of folder

The images dataset was split in 60:30:10 ratios as illustrated in bar chart below.

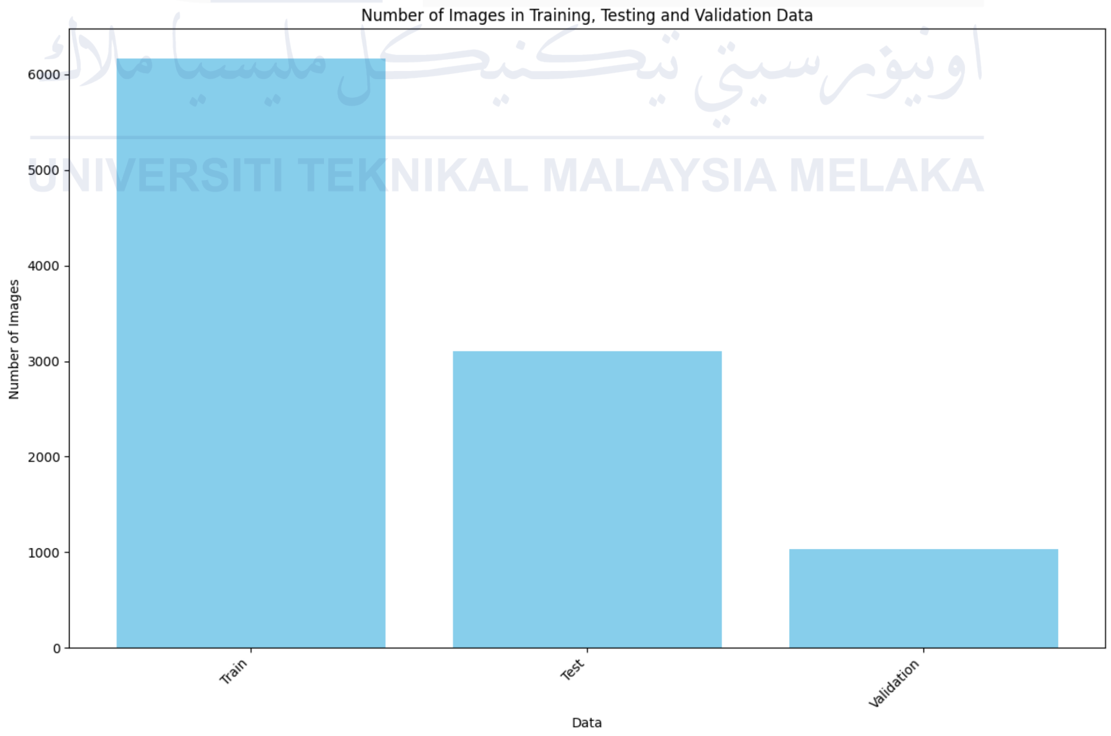


Figure 2.12 : Bar chart of distribution of images in training, testing and validation data

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Next the images were processed to array according to their pixel values. Each pixel were converted to its RGB value and converted by dividing by 255 to get values ranging from 0 to 1. Besides the pixel values were normalized by using constant mean and standard deviation to allow through training and learning process of the image.

Example of pixels chosen [R, G, B] = [128,64,32]

Augmentation / Normalization of pixels:

1. Rescale pixels to [0,1]

$$\left[\left(\frac{128}{255}\right),\left(\frac{64}{255}\right),\left(\frac{32}{255}\right)\right] = [0.502\ ,0.215\ ,0.125]$$

Equation 2.1: Pixel Rescaling

2. Normalize using the given mean and standard deviation:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

$$normallised\ value = \frac{rescaled\ value - mean}{std}$$

$$R_{norm} = \frac{0.502-0.485}{0.229} = 0.074$$

$$G_{norm} = \frac{0.251-0.456}{0.224} = -0.915$$

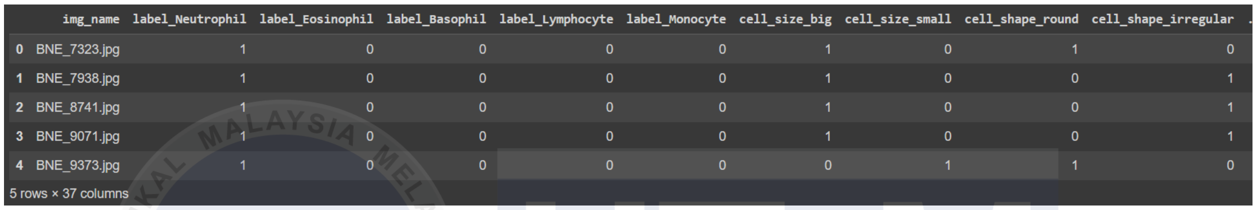
$$B_{norm} = \frac{0.125-0.406}{0.225} = -1.247$$

Equation 2.1: Normalizing Equation for all channels

The final normalized value of that pixel is [0.074, -0.915, 1.247].

ii. Annotations data(csv/label)

The attributes were encoded using the One-Hot Encoding method to either label 0 or 1 to each unique column. This way the model understands better and it's easier to interpret the result as we are working with 36 different categories for each cell.



	img_name	label_Neutrophil	label_Eosinophil	label_Basophil	label_Lymphocyte	label_Monocyte	cell_size_big	cell_size_small	cell_shape_round	cell_shape_irregular
0	BNE_7323.jpg	1	0	0	0	0	1	0	1	0
1	BNE_7938.jpg	1	0	0	0	0	1	0	0	1
2	BNE_8741.jpg	1	0	0	0	0	1	0	0	1
3	BNE_9071.jpg	1	0	0	0	0	1	0	0	1
4	BNE_9373.jpg	1	0	0	0	0	0	1	1	0

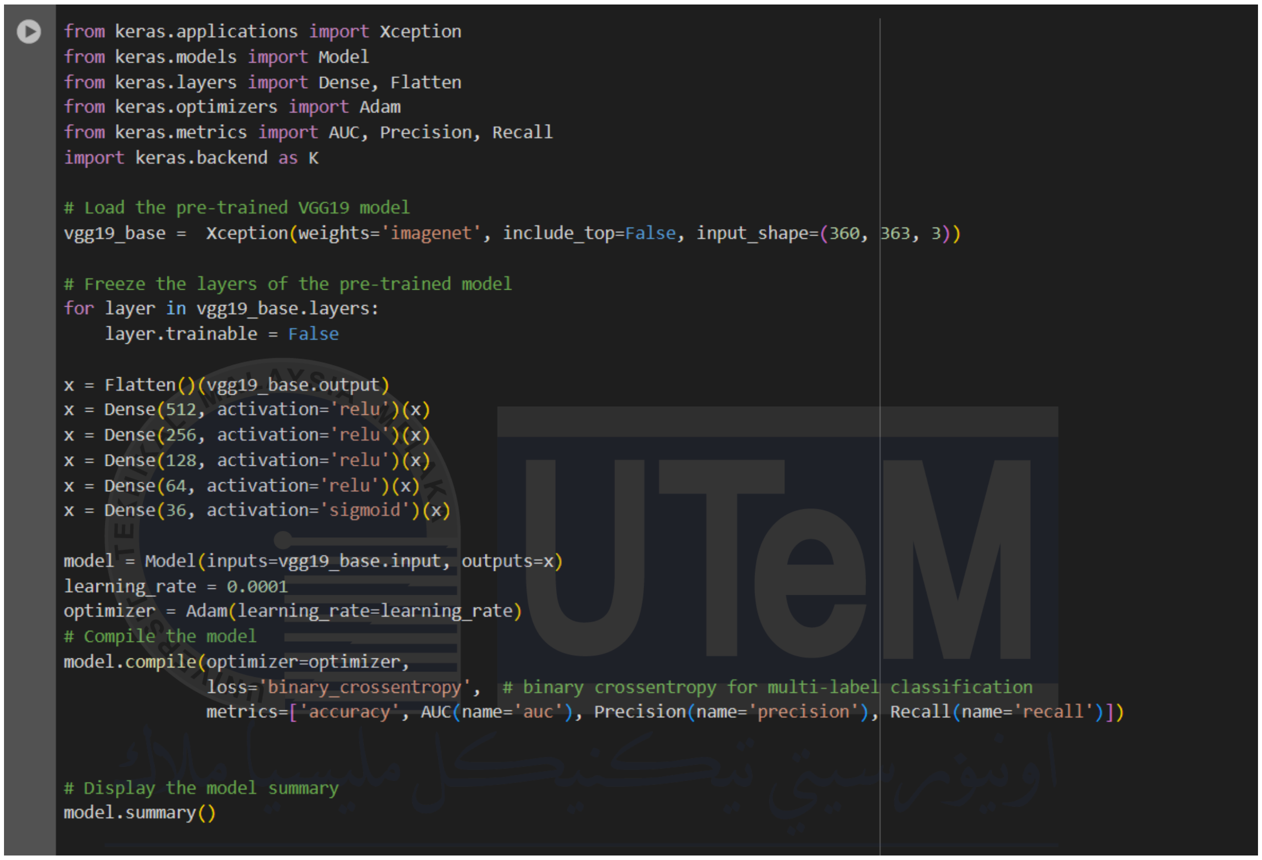
5 rows × 37 columns

Figure 2.13 : One Hot Encoded Labels

2.3.3.3 Model Building

In this phase the classification model was built using the transfer learning method through Keras and TensorFlow API. The model that was built was a headless model and based wights on the ImageNet database. Moreover, in this process we also have adapted fine tuning methodologies to ensure the model learns the dataset quickly and accurately. Numerous methods were tried including l2 regularizations to reduce the bias variance. The final tuning that we have done using the base of VGG 19 is adding 5 more dense layers alongside a final Sigmoid Activation function to produce the output. This tuning not only helped the multiclass classification problem but also increased the rate of data understanding towards the model rather than stocking onto a pattern.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



```
from keras.applications import Xception
from keras.models import Model
from keras.layers import Dense, Flatten
from keras.optimizers import Adam
from keras.metrics import AUC, Precision, Recall
import keras.backend as K

# Load the pre-trained VGG19 model
vgg19_base =  Xception(weights='imagenet', include_top=False, input_shape=(360, 363, 3))

# Freeze the layers of the pre-trained model
for layer in vgg19_base.layers:
    layer.trainable = False

x = Flatten()(vgg19_base.output)
x = Dense(512, activation='relu')(x)
x = Dense(256, activation='relu')(x)
x = Dense(128, activation='relu')(x)
x = Dense(64, activation='relu')(x)
x = Dense(36, activation='sigmoid')(x)

model = Model(inputs=vgg19_base.input, outputs=x)
learning_rate = 0.0001
optimizer = Adam(learning_rate=learning_rate)
# Compile the model
model.compile(optimizer=optimizer,
              loss='binary_crossentropy',  # binary crossentropy for multi-label classification
              metrics=['accuracy', AUC(name='auc'), Precision(name='precision'), Recall(name='recall')])


# Display the model summary
model.summary()
```

Figure 2.14 : The model building using Tensorflow and Keras API

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.3.3.4 Model evaluation

Since this is a multilabel classification problem we have engaged in evaluating the model by accuracy, recall f1-score, AUC and F1 score of every label prediction.

i. Accuracy

Accuracy measures the proportion of correct predictions (both true positives and true negatives) out of the total predictions. However, in multilabel classification, accuracy might be calculated differently as the average accuracy across all labels. Provides a general sense of how often the model is correct. However, it might not be sufficient for imbalanced datasets or when the cost of false positives and false negatives varies.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Equation 2.2 : Accuracy Equation

ii. Multilabel accuracy

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\frac{|\mathrm{y}_i \cap \hat{\mathrm{y}}_i|}{|\mathrm{y}_i \cup \hat{\mathrm{y}}_i|}$$

where N is the number of samples, y_i is the true label set for the i-th sample, and $\hat{\mathrm{y}}_i$ is the predicted label set for the i-th sample.

Equation 2.3 : Multilabel Accuracy Equation

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

iii. Recall

Recall (also known as sensitivity) measures the proportion of actual positives that are correctly identified by the model. Important when the focus is on capturing all positive instances, which is critical in scenarios where missing a positive instance is costly

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

Equation 2.4 : Recall Equation

iv. F1-Score

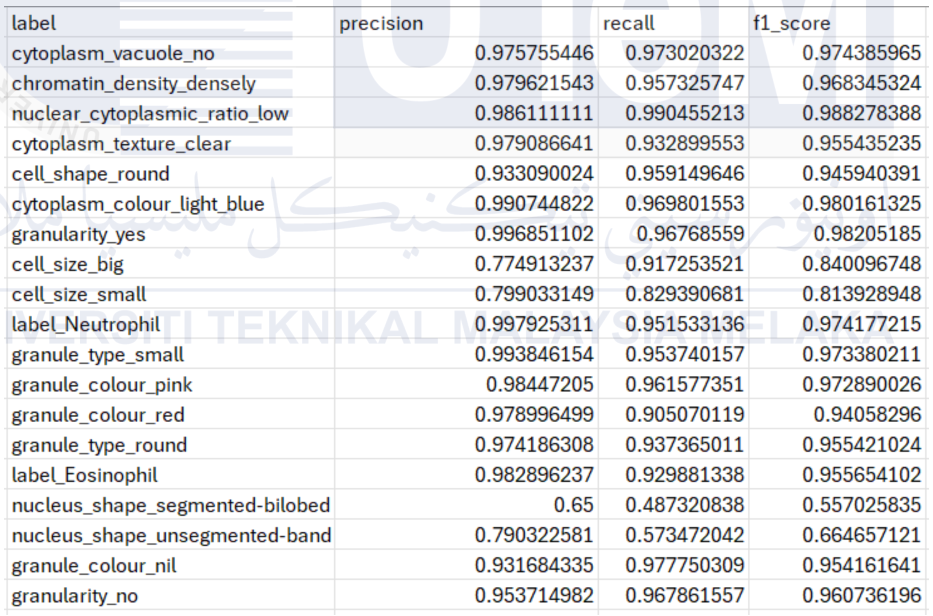
The F1-score is the harmonic mean of precision and recall, providing a balance between the two. It's particularly useful when the class distribution is imbalanced. **Multilabel F1-Score** is calculated for each label and then averaged (either macro, micro, or weighted average).

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Equation 2.5 : F1-Score Equation

v. Area Under Curve (AUC)

AUC measures the ability of the model to distinguish between classes and is typically used with ROC (Receiver Operating Characteristic) curves. Provides a measure of the model's ability to rank predictions correctly and is insensitive to class imbalance.



label	precision	recall	f1_score
cytoplasm_vacuole_no	0.975755446	0.973020322	0.974385965
chromatin_density_densely	0.979621543	0.957325747	0.968345324
nuclear_cytoplasmic_ratio_low	0.986111111	0.990455213	0.988278388
cytoplasm_texture_clear	0.979086641	0.932899553	0.955435235
cell_shape_round	0.933090024	0.959149646	0.945940391
cytoplasm_colour_light_blue	0.990744822	0.969801553	0.980161325
granularity_yes	0.996851102	0.96768559	0.98205185
cell_size_big	0.774913237	0.917253521	0.840096748
cell_size_small	0.799033149	0.829390681	0.813928948
label_Neutrophil	0.997925311	0.951533136	0.974177215
granule_type_small	0.993846154	0.953740157	0.973380211
granule_colour_pink	0.98447205	0.961577351	0.972890026
granule_colour_red	0.978996499	0.905070119	0.94058296
granule_type_round	0.974186308	0.937365011	0.955421024
label_Eosinophil	0.982896237	0.929881338	0.955654102
nucleus_shape_segmented-bilobed	0.65	0.487320838	0.557025835
nucleus_shape_unsegmented-band	0.790322581	0.573472042	0.664657121
granule_colour_nil	0.931684335	0.977750309	0.954161641
granularity_no	0.953714982	0.967861557	0.960736196

Figure 2.15 : Sample metric calculated per label from the training of model

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.3.4 Web Application

2.3.4.1 Requirement Analysis

The objective of the AI Blood Smear Analyzer is to increase the effectiveness of white blood cell identification through Artificial Intelligence and Machine Learning. The major role of this system is to aid the pathologists through a XAi that can justify its identification of a blood cell in a smear. The objectives of the system are:

i. Develop a web-based system that can analyze a blood smear mainly white blood cell in it.
ii. Develop a fine-tuned Neural Network that can lead on XAi methods.
iii. Develop an algorithm that can support multi class classification.

To achieve these objectives, the preliminary step taken is to specify the requirements of the software. The table below shows the requirements gathered from various sources and pathologists our main stakeholder for this project.

Table 2.5 : Functional and non-functional requirements of project

Requirement	Necessity
Functional requirements	• White blood cell dataset • Flask Framework • Keras and Tensorflow framework • Data about each individual cell • Image processing techniques • Image upload functionality • AI analysis and classification • Report generation and visualization • User-friendly web interface
Nonfunctional requirements	• Performance metrics • Reliability • Usability • Maintainability

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.3.4.2 System Design

In this phase, the system design is prepared which specifies hardware and system requirements, such as data layers, programming languages, network infrastructure, user interface etc. It helps define the overall system architecture.

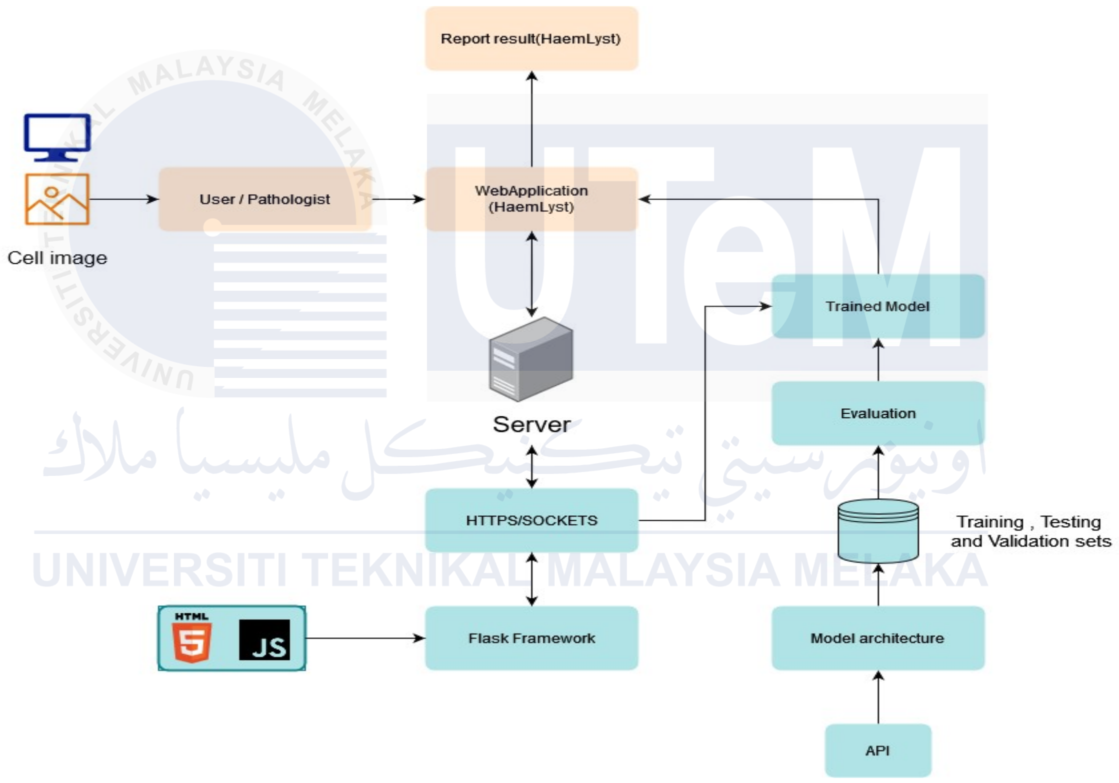


Figure 2.16 : High Level Design Diagram

The High-Level Design is illustrated in Figure 2.16. There are two ends of this system architecture, mainly the frontend (labeled as yellow in Figure 2.16) and the backend (labeled as Blue in Figure 2.16).

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

i. Frontend

The frontend primarily deals with the user experience, facilitated through the web application "HaemLyst." Both users and pathologists can upload a cell image to the application. Upon upload, the application quickly returns a prediction of the type of white blood cell present in the image along with the attributes that led to the classification.

ii. Backend

The backend comprises two parts: the web application and the model application. These components will later be integrated into a unified system during the integration phase.

iii. Web Application:

a. **Technologies Used:** HTML, CSS, JavaScript
b. **Framework:** Flask
c. **Functionality:** The web application is responsible for interpreting the frontend code and controlling the data flow through servers to display results to users.

iv. Model Application:

a. **Programming Language:** Python
b. **Libraries:** Keras, TensorFlow
c. **Functionality:** This component handles the development and fine-tuning of the CNN model. A non-relational database (Google Drive) is used to store training, testing, and validation datasets that have been cleaned for modeling purposes. The trained model is deployed as a custom blood smear analyzer, aiding in the prediction of cell types from provided images.

Summary

The system architecture ensures a seamless interaction between the user interface and the AI model. Users can upload cell images via the web application, which are then processed by the backend to provide immediate and accurate white blood cell identification and classification. The Flask framework facilitates data handling and user interactions, while the AI model, built using Keras and TensorFlow, performs the image analysis and prediction tasks. The integration of these components aims to enhance the efficiency and accuracy of white blood cell identification in blood smears.

2.3.4.3 Testing

In this phase we do Functional Testing with Black Box Testing. Functional testing is a type of software testing that verifies each function of the application works according to the requirements and specifications without focusing on the source code. It involves providing appropriate test inputs, expecting specific outputs, and comparing actual outputs with expected results. This testing checks the user interface, APIs, database, security, client/server applications, and overall functionality. Functional testing can be conducted manually or through automation and is essential for determining if the software meets its functional requirements.

Table 2.6 : Black Box testing that the web application was tested with

TEST NAME	TEST TYPE	TEST OUTCOME
Valid Image Upload	Functionality testing	The system should accept a single image in valid formats (e.g., JPEG, PNG, BMP) and classify it correctly.
Invalid Image Format Upload	Functionality testing	The system should reject images in unsupported formats (e.g., PDF, DOCX) and provide an appropriate error message.
Image Size Limit	Boundary Value Testing	System shall accept image files from [1kB – 10MB]
Image Classification Accuracy	Functionality testing	The system should correctly classify and display the outcome from the model.

2.3.5 Integration

This phase is the final phase where both units are combined into a single system. The web application that has been developed with Flask Framework is bound with the trained Convolutional Neural Network Model. The developed model will be saved in a (h5) format which contains the weights, values and the correlations. This embedded model will accept the image that is uploaded through the Web Applications image uploader module and continue to process it. The output of the model which are probabilities and classes in a machine text will be displayed to the user with visualizations with formatting to improve usability. This once again will be equipped with Black Box testing before proceeding to the deployment of the application.

2.3.6 Report Writing

The report for the HaemLyst project should provide a comprehensive overview of the entire project, covering the Introduction, Literature review, Requirement Analysis, Design, Testing and Conclusion. The aim is to document the development process, findings, and future recommendations in a structured and coherent manner.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.3.7 Deployment

The HaemLyst blood smear analyzer is deployed as a web-based application, allowing users to access it conveniently through a web browser without needing to install any software locally. This deployment method enhances accessibility, enabling users to use the application from various devices, such as computers, tablets, or smartphones, as long as they have internet connectivity. Additionally, the web application integrates with secure servers to ensure the safety and privacy of the data being analyzed, facilitating seamless and protected data transmission and storage. This setup also allows for easier updates and maintenance, ensuring the system remains up-to-date without requiring user intervention.

2.4. Requirements

All the requirements of the system to be developed in addition to deadlines and guidelines are incorporated at this part. This part will list the required software, hardware, and other requirements to deploy the system.

2.4.1 Software Requirements

i. Interpreter And Environments

   Interpreter and Environment for coding of project

   1. PyCharm
   2. Python 3.9
   3. Google Collab

ii. Package And API

   1. TensorFlow

      Build model and save model files to a smaller file
   2. Sckit-Learn

      Visualize and plot data
   3. Keras

      To build custom model and tuning
   4. Pandas

      To manipulate, transform and read data
   5. PyTorch

      To construct array and NumPy from an image
   6. Flask

      Basis to build a web app

iii. Operating System

   1. Windows 10 Home

      Operating System from Microsoft

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

iv. Document And File Editors

1. Microsoft Word

   Edit documents and make reports

2. Microsoft Excel

   Edit data files and visualize the data

v. Graphics And Diagram Making

1. Draw.io

   To make ERD, Flowcharts, process diagrams

2. Adobe Illustrator

   To make graphics images – architecture diagrams, cell images.

2.4.2 Hardware Requirements

Hardware requirement defined as the hardware will be used in this project.

i.	CPU	: Intel(R) Core (TM) i7-10870H CPU @ 2.20GHz
ii.	RAM	: 16GB
iii.	GPU	: NVIDIA GeForce RTX 3060 Laptop GPU

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.4.3 Other Requirements

No other requirements are needed during the project development.

2.5. Project Schedule and Milestones

PSM 1 MILESTONES | SEM 2 SESSION 2022/2023

TASK / WEEK	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Proposal Discussion with supervisor	■															
Proposal presentation and submission		■	■													
Progress 1 [PRJ-3] Chapter 1				■												
Progress 1 [PRJ-3] Chapter 2					■	■										
Project Progress 1 [PRJ-2]						■										
Progress 1 [PRJ-3] Chapter 3							■	■	■							
Project Progress 2 [PRJ-4]										■						
Progress 1 [PRJ-5] Chapter 4											■					
Draft Report Preparation												■	■			
Draft Report Submission														■		
Final Presentation															■	
Draft report correction																■

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.17 : Gantt chart for PSM 1

PSM 2 MILESTONES | SEM 3 SESSION 2022/2023									
TASK / WEEK	1	2	3	4	5	6	7	8	9
Discussion, correction and planning with supervisor	■								
Project Progress 1 [PRJ-3] Chapter 5		■							
Report Writing Progress [PRJ-3]			■						
Project Progress 2 [PRJ-2] Chapter 6				■					
Report Writing Progress [PRJ-3] Chapter 7					■	■			
Report Evaluation, Demonstration to Supervisor and Evaluator [PRJ-6, PRJ-10, PRJ-4, PRJ-5, PRJ -9]							■		
Correction of draft report								■	
Submission of report									■

UTeM UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 2.18 : Gantt chart for PSM 2

2.5.1 Project Scope Management

Project Management Process	
Project Title	AI BloodSmear Analysis System
Project Manager	HERISH CHAUDRAY A/L RAVANTHERAN
Project Objectives	a. To develop a Convolutional Neural Network that can correctly classify White blood Cells based on 12 different attributes that confirm a white blood cell type. b. To enhance the usage of traditional neural network via fine tuning and increase the performance of the model. c. To aid pathologists to optimize the effort and time taken to identify a white blood cell from a blood smear.
Project Deliverables	a. Web Application (HaemLyst) b. System Documentation c. Report d. d. Presentation slides
Milestones	a. **PRJ-1: PROPOSAL (WEEK1)** The initial proposal is used to define an internal or external project. b. **PRJ-2: PROJECT PROGRESS 1 (WEEK 6)** Application/System Development Progress. **c. PRJ-3: REPORT WRITING PROGRESS 1 (WEEK9)** Report writing progress for Chapter 1, Chapter 2 and Chapter 3. **d. PRJ-4: PROJECT PROGRESS 2 (WEEK 10)** Application/System Development Progress.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	e. **PRJ-5: REPORT WRITING PROGRESS 2 (WEEK11)** Report writing progress for Chapter 4. f. **PRJ-6: DEMONSTRATION (SUPERVISOR) (WEEK 14)** Demonstration of the project results. g. **PRJ-7: DEMONSTRATION (EVALUATOR)(WEEK 15)** Demonstration of the project results. h. **PRJ-8: PRESENTATION (WEEK 15)** Delivering effective and engaging presentations to the evaluator. i. **PRJ-10: REPORT EVALUATION (EVALUATOR) (WEEK 16)** PSM1 Draft report for evaluation by Evaluator j. **PRJ-9: REPORT EVALUATION (SUPERVISOR)(WEEK 16)** PSM1 Draft report for evaluation by Supervisor.
Project Constraints	b. Limited compute units on Google Collab c. Completion of project by 16 June 2024
Assumptions	a. Timely Approval of Project deliverables b. No major changes to project requirements or objectives

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2.6. Summary

This chapter reviews existing systems related to the proposed work, highlighting the neural network and machine learning approaches used in these systems, and compares them with the proposed system by examining their architectures, techniques, modes, and features. The project's domain is explored, focusing on white blood cells (WBCs), which are crucial components of the immune system produced in the bone marrow and classified into granulocytes, monocytes, and lymphocytes, each with distinct roles in fighting infections and other disorders. Understanding the attributes that differentiate these cells, such as cell size, shape, and nucleus characteristics, is essential for the project's success. The literature review reveals that previous systems, like the one by Hüseyin Kutlu, used Regional Convolutional Neural Networks (R-CNN) to classify different WBC types within the same image, achieving a 95% accuracy rate in identifying WBCs, with ResNet50 demonstrating the highest performance through transfer learning. Accuracy rates for different cell types, including lymphocytes, monocytes, basophils, eosinophils, and neutrophils, varied but were generally high, highlighting the effectiveness of CNNs in accurately classifying WBCs and informing the choice of techniques for the proposed system. The review of techniques used in previous research emphasizes the efficiency of Mask R-CNN for instance segmentation in images, which includes feature extraction, region proposal, RoI alignment, and prediction, leveraging ResNet and FPN for effective feature extraction and RPN for proposing regions of interest. For the proposed system, CNNs are chosen for their ability to learn spatial hierarchies from input images, making them ideal for image classification tasks. The project methodology involves two parallel units: Unit 1 for modeling and Unit 2 for web application development, which are integrated in the final phase, ensuring the development of a robust system for classifying WBCs in microscopic images of blood smears. The list includes hardware and software required as well as the time frame and the milestones for both PSM 1 and PSM 2

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 3. REQUIREMENT ANALYSIS

3.1. Introduction

Analysis phase is a very important part, and it can be used as requirements guide to accomplish a project. Hence, the analysis will be split into a few sections, which is problem analysis, and requirement analysis. Thus, requirement analysis will further discuss data requirement, functional requirement, non-functional requirement, and others requirement. In addition, the requirement analysis and implemented the convolutional neural network will be discussed in the proposed project.

3.2. Problem Analysis

White blood cells are a type of blood cell that is made in bone marrow and found in blood and lymph tissue. White blood cells are part of the body's immune system. They help the body fight infection and other diseases. Types of white blood cells are granulocytes (neutrophils, eosinophils, and basophils), monocytes, and lymphocytes (T cells and B cells). The vast type of white blood cells requires minute details to find the distinctive properties of a specific cell. They may cost up to 3 pathologists and averagely about 7 minutes per cell to be identified by the experts.

The diagram below illustrates the traditional methods to deeply analyze the blood cells in from am blood smear:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

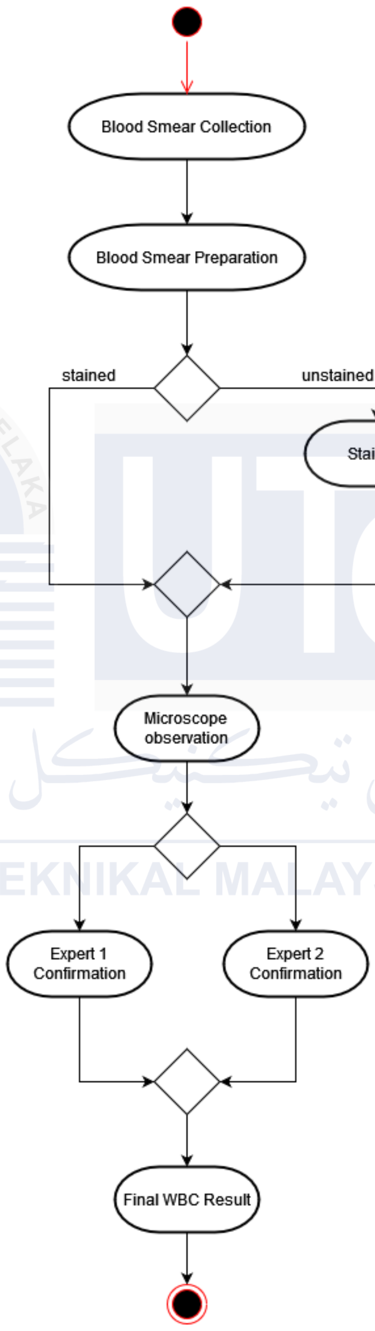


Figure 3.1 : The traditional method of depth analysis of white blood cells from a smear under the microscope

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

To enhance the traditional method of blood smear examination, which typically involves a minimum of two pathologists and a head pathologist, we propose an automated solution leveraging Machine Learning (ML) techniques, specifically through the application of computer vision and advanced image processing. This innovative approach aims to improve the efficiency, accuracy, and consistency of identifying and classifying white blood cells (WBCs) in blood smears.
Workflow of the Automated Examination System

1. **Staining and Image Acquisition**:
   i. The process begins with the preparation of the blood smear sample. This involves staining the sample to highlight different components of the blood, making it easier to identify various types of cells under a microscope.
   ii. Once the staining process is complete, high-resolution images of the blood smear are captured using a microscope equipped with a digital camera.

2. **Image Preprocessing**:
   i. The captured images are preprocessed to enhance quality and remove any noise that may interfere with accurate analysis. Techniques such as contrast adjustment, normalization, and artifact removal may be applied.
   ii. Segmentation algorithms are used to isolate the WBCs from other elements in the blood smear, such as red blood cells and platelets.

3. **Neural Network Structure for Classification**:
   i. A custom-built neural network, specifically designed for this application, is employed to analyze the preprocessed images. This neural network is trained on a large dataset of labeled blood smear images, allowing it to learn the distinguishing features of different types of WBCs.
   ii. The neural network architecture could include convolutional layers for feature extraction, followed by fully connected layers for classification.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4. **Feature Extraction and Prediction**:

As the neural network processes the image, it extracts various attributes of the identified WBCs, including:

i. **Size**: The neural network measures the dimensions of the WBCs, which can be indicative of specific cell types or abnormal conditions.

ii. **Vacuole Presence**: The detection of vacuoles (small cavities within the cells) is noted, as certain types of WBCs or pathological conditions are characterized by the presence of vacuoles.

iii. **Chromatin Density**: The density and pattern of chromatin within the cell nucleus are assessed, which helps in distinguishing between different types of WBCs and identifying abnormalities.

5. **Computation of Probabilities and Confirmatory Results**:

i. The neural network computes the probabilities for each possible classification based on the extracted features, comparing them with the known attributes of different WBC types.

ii. The system then generates a confirmatory result, indicating the most likely classification of the WBC. This prediction includes a confidence score and highlights the attributes that led to this conclusion.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.3. Requirement analysis

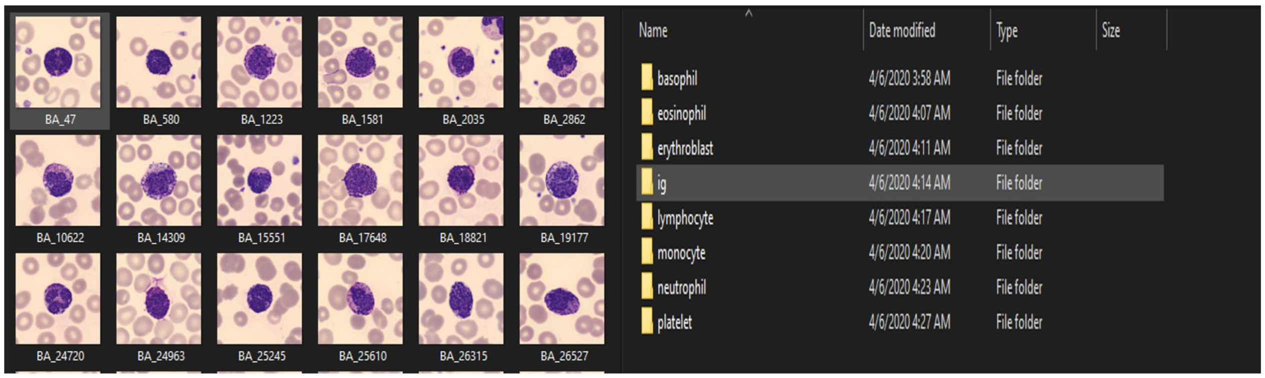
The software development life cycle (SDLC) process includes a critical phase called requirement analysis, which is essential for identifying and documenting the requirements of stakeholders and users. This phase ensures that the project aligns with the expectations and needs of all parties involved. During requirement analysis, we gather comprehensive information through various techniques such as interviews, surveys, and analysis of existing systems. This phase helps in understanding the project's scope, identifying the functionalities and features needed, and establishing clear objectives. By employing the SDLC's requirement analysis phase, we ensure that the developed software addresses the specific needs of its users, enhancing the overall effectiveness and success of the project. Additionally, this phase facilitates the identification of data requirements, functional and non-functional requirements, and other crucial elements necessary for the system's performance and usability. It also sets a solid foundation for subsequent phases in the SDLC, including design, development, testing, and deployment, ensuring a structured and efficient approach to software development.

3.3.1 Data Requirement

3.3.1.1 Image Dataset

The images dataset contains a total of 17,092 images of individual normal cells, which were acquired using the analyzer CellaVision DM96 in the Core Laboratory at the Hospital Clinic of Barcelona. The dataset is organized into the following eight groups: neutrophils, eosinophils, basophils, lymphocytes, monocytes, immature granulocytes (promyelocytes, myelocytes, and metamyelocytes), erythroblasts and platelets or thrombocytes. The size of the images is 360 x 363 pixels, in format JPG, and they were annotated by expert clinical pathologists. The images were captured from individuals without infection, hematologic or oncologic disease and free of any pharmacologic treatment at the moment of blood collection. The dataset could be downloaded from https://data.mendeley.com/datasets/snkd93bnjr/1/files/2fc38728-2ae7-4a62-a857-032af82334c3. The structure of the image's files is illustrated below:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

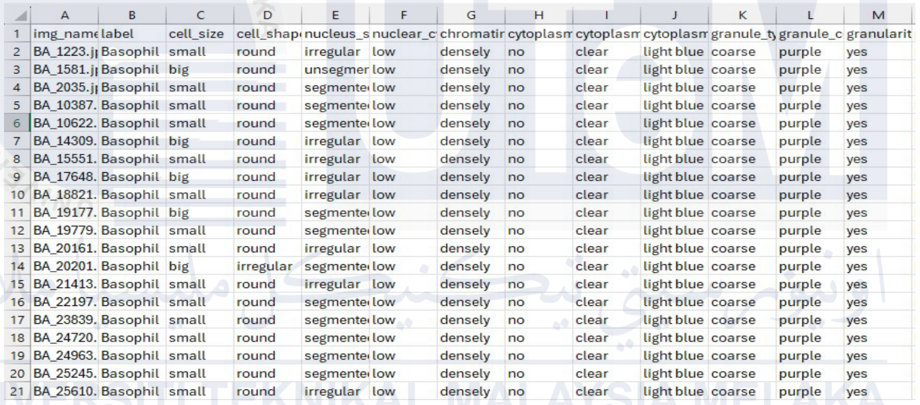


Name	Date modified	Type	Size
basophil	4/6/2020 3:58 AM	File folder	
eosinophil	4/6/2020 4:07 AM	File folder	
erythroblast	4/6/2020 4:11 AM	File folder	
ig	4/6/2020 4:14 AM	File folder	
lymphocyte	4/6/2020 4:17 AM	File folder	
monocyte	4/6/2020 4:20 AM	File folder	
neutrophil	4/6/2020 4:23 AM	File folder	
platelet	4/6/2020 4:27 AM	File folder	

Figure 3.2 :File structure of PBC_DIB white blood cell dataset

3.3.1.2 Annotations file

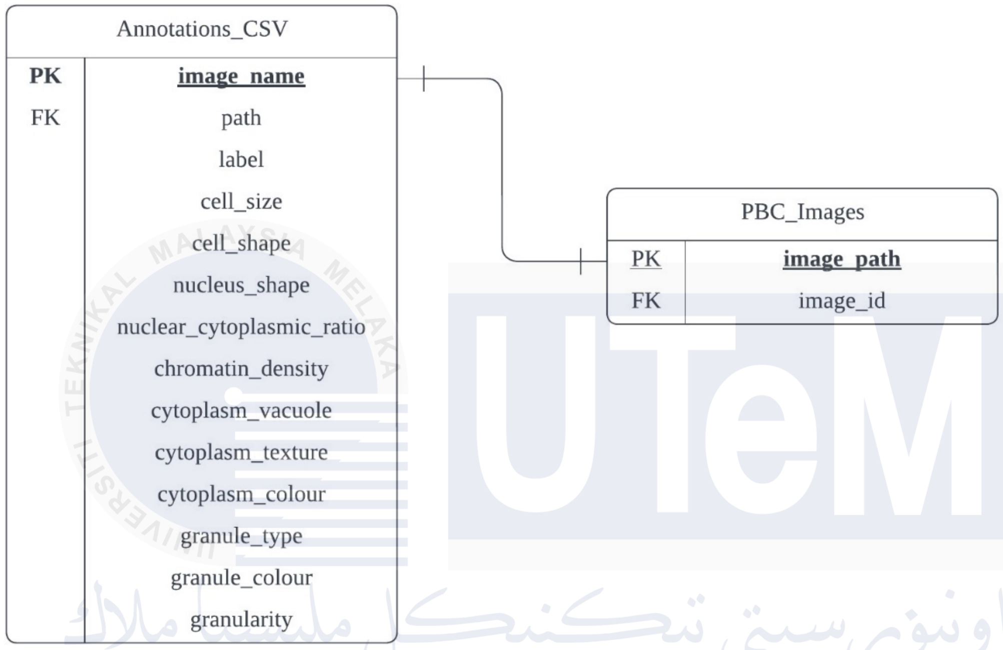
The annotations files containing 11 different attributes was provided by Satoshi Tsutsui, Winnie Pang, and Bihan Wen. WBCAtt: A White Blood Cell Dataset Annotated with Detailed Morphological Attributes. Advances in Neural Information Processing Systems (NeurIPS) 2023. 113k labels (11 attributes x 10.3k images) on white blood cell images, detailing the fine-grained concepts pathologists recognize. The csv can be sourced from https://arxiv.org/abs/2306.13531. The structure and format of the annotations.csv file is as below:



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	img_name	label	cell_size	cell_shap	nucleus_s	nuclear_c	chromatir	cytoplasm	cytoplasm	cytoplasm	granule_ty	granule_c	granularit
2	BA_1223.jp	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes
3	BA_1581.jp	Basophil	big	round	unsegmer	low	densely	no	clear	light blue	coarse	purple	yes
4	BA_2035.jp	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
5	BA_10387.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
6	BA_10622.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
7	BA_14309.	Basophil	big	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes
8	BA_15551.	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes
9	BA_17648.	Basophil	big	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes
10	BA_18821.	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes
11	BA_19177.	Basophil	big	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
12	BA_19779.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
13	BA_20161.	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes
14	BA_20201.	Basophil	big	irregular	segmente	low	densely	no	clear	light blue	coarse	purple	yes
15	BA_21413.	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes
16	BA_22197.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
17	BA_23839.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
18	BA_24720.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
19	BA_24963.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
20	BA_25245.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes
21	BA_25610.	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes

Figure 3.3 : The format of the annotations file csv

3.3.1.3 Data Model / Entity Relationship Diagram



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 3.4 : E.R.D For the Annotations file and the Images Database

3.3.2 Functional Requirements

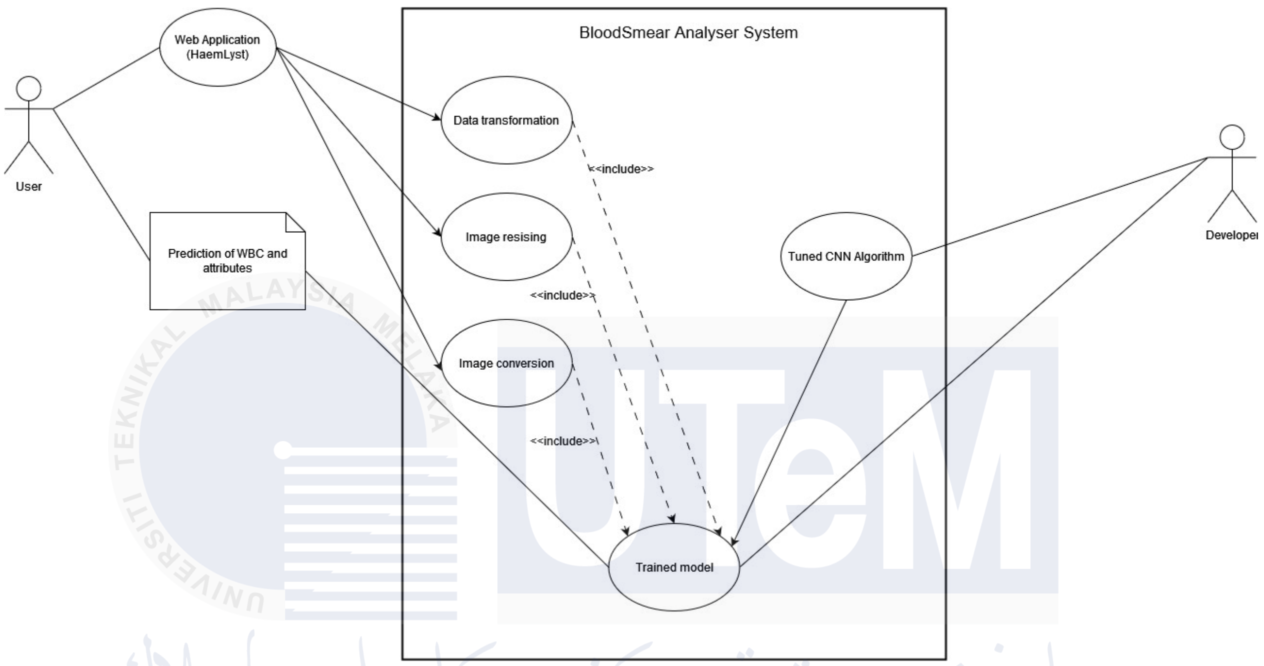


Figure 3.5 : Use case diagram of Blood Analyzer System (HaemLyst)

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The user starts off the Web Application developed in Flask using HTML, CSS and JavaScript. Once image is uploaded the BloodSmear Analyzer System (Backend) starts transforming the image into 3 channel (RGB) values of pixels between [0,255]. The image is further resized if larger than 360 pixels for model architecture. After final images is provided the system converts the images into arrays and each pixel is augmented using subtraction of mean [0.485, 0.456, 0.406] and divided by standard deviation values of [0.229, 0.224, 0.225]. The mathematical equation is as follows:

Example of pixels chosen [R, G, B] = [128,64,32]

Augmentation / Normalization of pixels:

1. Rescale pixels to [0,1]

$$\left[\left(\frac{128}{255}\right),\left(\frac{64}{255}\right),\left(\frac{32}{255}\right)\right] = [0.502\ ,0.215\ ,0.125]$$

Equation 3.1: Rescaling Formula

2. Normalize using the given mean and standard deviation:

$$normallised\ value = \frac{rescaled\ value - mean}{std}$$

$$R_{norm} = \frac{0.502-0.485}{0.229} = 0.074$$

$$G_{norm} = \frac{0.251-0.456}{0.224} = -0.915$$

$$B_{norm} = \frac{0.125-0.406}{0.225} = -1.247$$

Equation 3.2: Normalizing formula for all channels

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The final normalized value of that pixel is [0.074, -0.915, 1.247]. Some normalization of pixels yields negative values but, in this case, it is normal and emphasised as these non-absolute values help in healthy gradients evicting vanishing gradients problems. Furthermore, it also increases the models' learning process due to its large spread of values.

Finally, the processed image is then fed to the trained model that has been tuned by Developer using Convolutional Neural Network. This model now predicts the type of cell in the processed image alongside the attribute's prediction. These values with max probabilities are then returned to the User in word forms instead of numbers or encoded labels.

3.3.3 Non-functional Requirement

3.3.3.1 Requirements

a. Resource Usage: No more than 30% CPU and 79% GPU utilization during peak loads.
b. Response Time: Classification results returned within 0.8 - 1.0242 seconds.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.3.3.2 Accuracy Requirements

a. Precision: High precision in classifications, targeting an F1-macro score of 0.95.
b. Accuracy and recall: 98% accuracy in the white blood cell classification tasks.
c. Validation and Verification: Comprehensive testing with a validation dataset larger than 900 images

3.3.3.3 Data Handling Requirements

a. Data Storage Capacity: Ability to store 20 GB of image data.
b. Data Processing: Efficient handling of large image datasets, including rapid retrieval and update operations using data loaders such as Dropbox or Google Drive

3.3.4 Other requirement

Table 3.1 : Other software requirements needed for project

SOFTWARE	**USAGE**
INTERPRETER AND ENVIROMENTS	
PyCharm	Interpreter and Environment for coding of project
Python	
Google Collab	
PACKAGE AND API	
Tensorflow	Build model and save model files to a smaller file
Sckit-Learn	Visualize and plot data
Keras	To build custom model and tuning
Pandas	To manipulate, transform and read data
PyTorch	To construct array and NumPy from an image
Flask	Basis to build a web app
OPERATING SYSTEM	
Windows 10 Home	Operating System from Microsoft
DOCUMENT AND FILE EDITORS	
Microsoft Word	Edit documents and make reports
Microsoft Excel	Edit data files and visualize the data
GRAPHICS AND DIAGRAM MAKING	
Draw.io	To make ERD, Flowcharts, process diagrams
Adobe Illustrator	To make graphics images – architecture diagrams, cell images.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 3.2 : Other hardware requirements needed for project

HARDWARE	**USAGE**
SYSTEM ENVIROMENTS	
CPU: Intel(R) Core (TM) i7-10870H CPU @ 2.20GHz	Base system environments that are utilized throughout training, modeling and bug detection.
RAM: 16GB	
GPU: NVIDIA GeForce RTX 3060	
STORAGE	
BC711 NVMe SK hynix 512GB	To store data of images, edited images , annotation files , trained models, torch files and hash files.
Western Digital WD Elements 2TB HDD	

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

3.4. Summary

The analysis phase is crucial for guiding the project's requirements, encompassing problem analysis and requirement analysis, which further breaks down into data, functional, non-functional, and other requirements. The project focuses on white blood cells (WBCs), which are vital components of the immune system produced in the bone marrow. The traditional method of identifying WBCs involves multiple pathologists and is time-consuming. To improve this, an automated solution using machine learning, specifically computer vision and advanced image processing, is proposed. This involves staining and acquiring high-resolution images of blood smears, preprocessing them to enhance quality, and using a custom-built neural network for classification. The network is trained on a large dataset to learn the features of different WBCs, extracting attributes such as cell size, vacuole presence, and chromatin density to predict the cell type with a confidence score. The dataset includes 17,092 images of normal cells, annotated with

11 attributes, and sourced from the Hospital Clinic of Barcelona. The functional requirements involve preprocessing images, normalizing pixel values, and feeding them to the trained model for predictions. Non-functional requirements emphasize resource usage, response time, accuracy, data storage, and processing efficiency. Additional software requirements include PyCharm, TensorFlow, Flask, and others for development, while hardware requirements specify a powerful CPU, GPU, and adequate storage capacity for handling large datasets and training models.

UTeM
اونيۏرسيتي تيکنيکل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 4. DESIGN

4.1. Introduction

On this chapter, the system design for AI Blood Smear Analysis will be discussed in detail. The design flow of used neural network techniques, the development and deployment of the interface will be described.

4.2. High Level Design

The aim of this project is to develop a web application that can classify the white blood cells with Convolutional Neural Network and create a confirmatory and product such as XAi allowing the pathologists to get defined reasons of the classification done by the system. Multiple methods was implemented and tested using various metrics – accuracy, sensitivity , precision , F1_scores to contrast between training methods.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.2.1. System Architecture for DSS

We have implemented three different architectures to test the best performance for this multilabel classification problem for white blood cells. The architecture used in this project was VGG 19, RESNET 50 and EXCEPTION. However, to be task specific we have experimented with the PSM 1 best model VGG 19 for better accuracy and overall performance. Thus architecture used was Base VGG 19, VGG 19 added with 3 layers with Kaiming Initialization, VGG 19, RESNET 50 and XCEPTION with 5 additional layers. We have implemented the transfer learning method to train the model using ImageNet weights while doing some manual tuning and addition of layer to support the cell classification.

For each architecture we have frozen the head layers to disable the weight of ImageNet being updated during training preventing overfitting. Furthermore, tuning has been done to compensate for the froze layer by adding 5 additional layers on top of the Base of Each

Architecture. These layers were added to aid our specific task of classifying the image of cells to 12 distinctive attributes. The layers for each type of architecture are illustrated below.

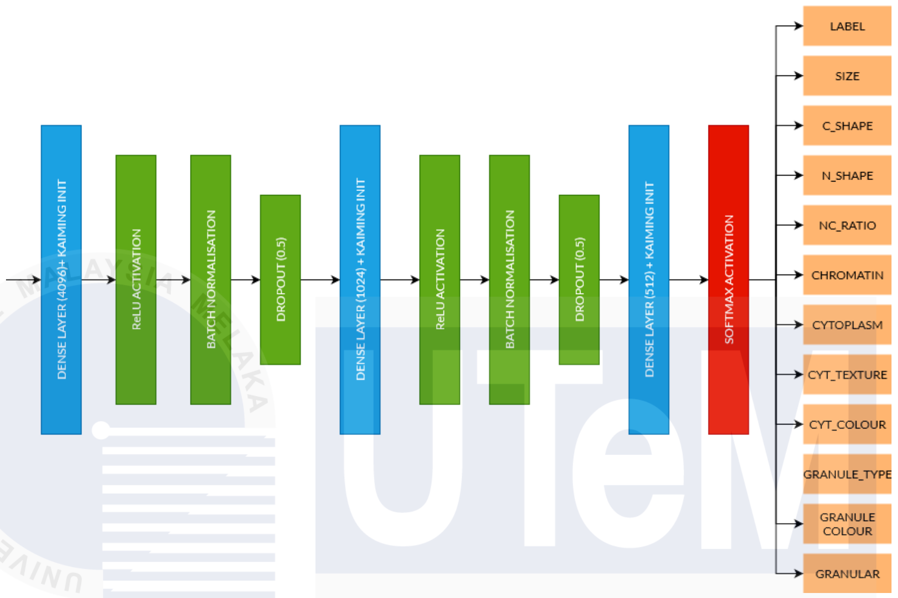


Figure 4.1: Architectural diagram for the fine tuned VGG 19 with Kaiming initializations

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 25 illustrates the extra layers that we have added to the base VGG 19 to enhance the classification problem mainly to evict low accuracies on the Nucleus Shape. In this case we have used Kaiming Initialization to set the initial weights so that the variance of the activations is consistent across layers, which helps the model learn more effectively. This helps in the weight initialization to avoid bias across dataset. For the usage of Kaiming Initialization we have also used ReLU the rectified linear unit (ReLU) or rectifier activation function which introduces the property of nonlinearity to a deep learning model and solves the vanishing gradients issue we faced beforehand.

Batch Normalization was applied after each convolutional layer to normalize the activations, which not only accelerates training by reducing the internal covariate shift but also provides a slight regularization effect. This layer helps in maintaining the stability of the network by ensuring that the input to each layer has a consistent distribution of data, which is crucial when using Kaiming Initialization. The combination of Kaiming Initialization and Batch Normalization ensures that the variance of activations remains stable, preventing the network from being too sensitive to the initialization of weights.

Dropout, on the other hand, was used as a regularization technique to prevent overfitting. By randomly dropping a fraction of the neurons during training, Dropout forces the model to generalize better, as it cannot rely on specific neurons being present. This layer is particularly effective when dealing with complex datasets where overfitting is a concern, such as in the classification of Nucleus Shape in our case.

Finally, a **Softmax** activation function was applied in the last layer to convert the model's output into probabilities for each class. This ensures that the network's predictions are interpretable as a probability distribution over the possible classes, with the class having the highest probability being chosen as the final prediction. The combination of ReLU activations in the intermediate layers and Softmax at the output provides the model with non-linearity and the ability to perform multi-class classification effectively.

Together, these enhancements—Kaiming Initialization, ReLU activation, Batch Normalization, and Dropout—create a robust architecture that improves the model's ability to learn meaningful patterns in the data while mitigating issues like vanishing gradients, overfitting, and unstable training dynamics.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Equation 4.1: SoftMax Activation Mathematical Equation

The architecture of the model can be defined as follows:

First dense layer	:	y1=ReLU(BatchNorm(Dropout(W1x+b1)))
Second dense layer	:	y2=ReLU(BatchNorm(Dropout(W2y1+b2)))
Third dense layer	:	y4=ReLU(BatchNorm(Dropout(W4y3+b4)))

Output layer : y5=Softmax(W5y4+b5)

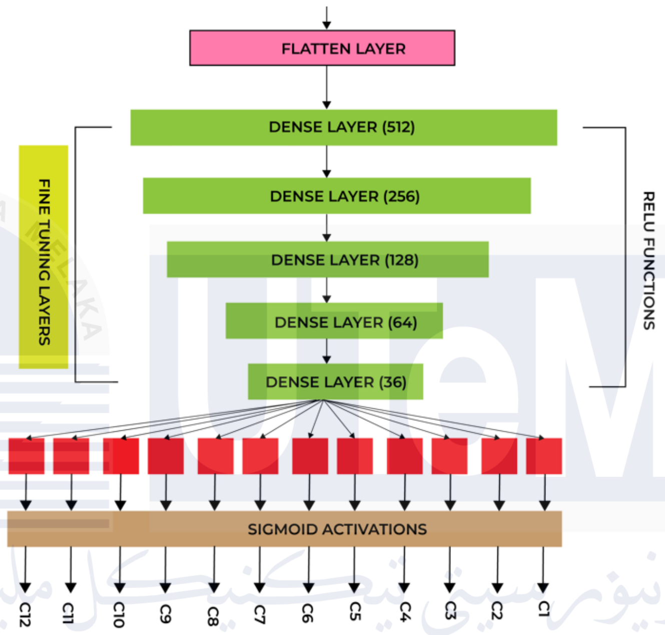


Figure 4.2 : Architectural diagram of the fine-tuned VGG 19 CNN Model

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The first 4 layers of the fine tune are equipped with ReLU activation function while the final output layer is equipped with Sigmoid Activation Function which is used for multilabel classification problems. The rectified linear unit (ReLU) or rectifier activation function introduces the property of nonlinearity to a deep learning model and solves the vanishing gradients issue. It interprets the positive part of its argument.

ReLU	$f(x) = \begin{cases} 0 \text{ for } x<0 \\ x \text{ for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 \text{ for } x<0 \\ 1 \text{ for } x \geq 0 \end{cases}$

Sigmoid	$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)\,(1 - f(x))$

Figure 4.3 : ReLU and Sigmoid function mathematical equation

The fine-tuned architectures follows:

First Dense Layer	:	$y1 = \mathrm{ReLU}(W_{1x1} + b_1)$
Second Dense Layer	:	$y2 = \mathrm{ReLU}(W_{2y1} + b_2)$
Third Dense Layer	:	$y3 = \mathrm{ReLU}(W_{3y2} + b_3)$
Fourth Dense Layer	:	$y4 = \mathrm{ReLU}(W_{4y3} + b_4)$
Fifth Dense Layer	:	$y4 = \mathrm{ReLU}(W_{5y4} + b_4)$
Output Dense Layer	:	$y5 = \sigma\,(W_{5y4} + b_5)$

The fully connected architecture for VGG19, VGG 19 added 3 layers with Kaiming Initialization, VGG 19, RESNET 50 and XCEPTION with 5 additional layers will be summarized below.

UTeM
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

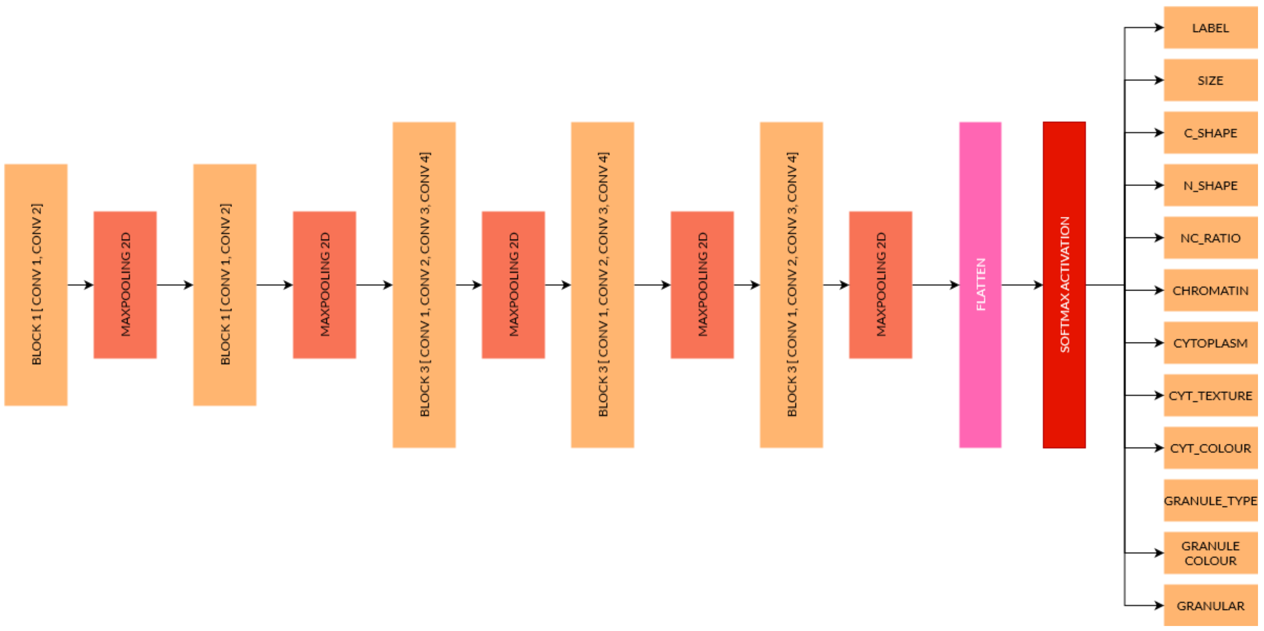


Figure 4.4 : Fully connected VGG 19 Transfer Learning Model

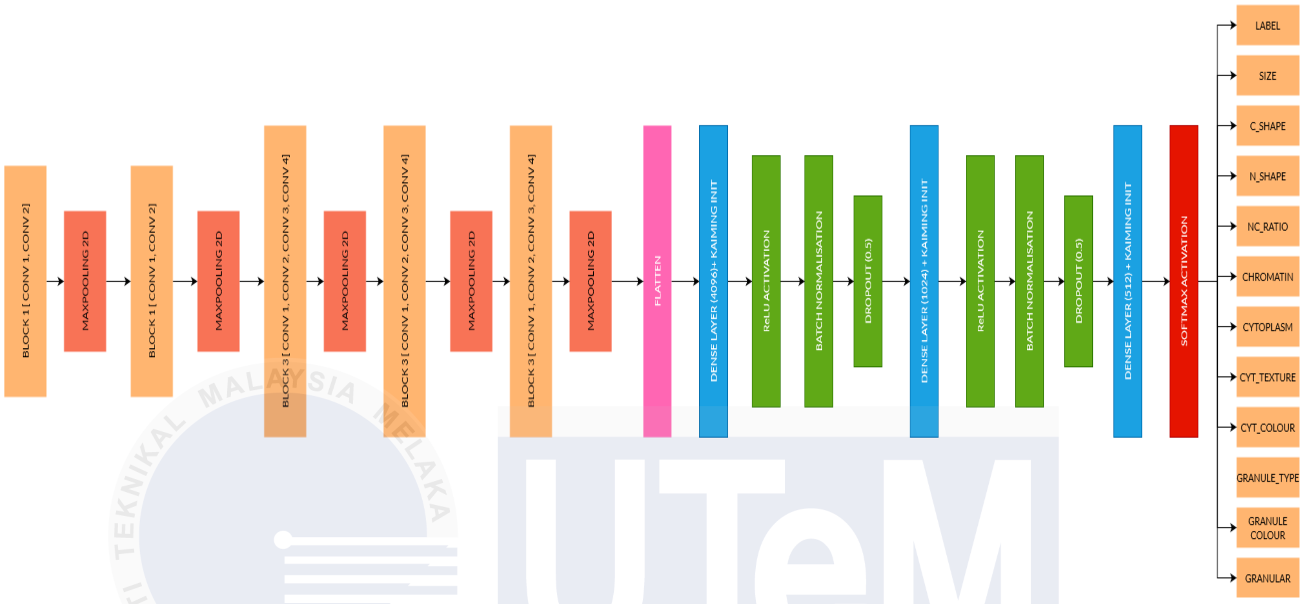


Figure 4.5 : Fully connected VGG 19 Transfer Learning Model with 3 added layers with Kaiming Initialization

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

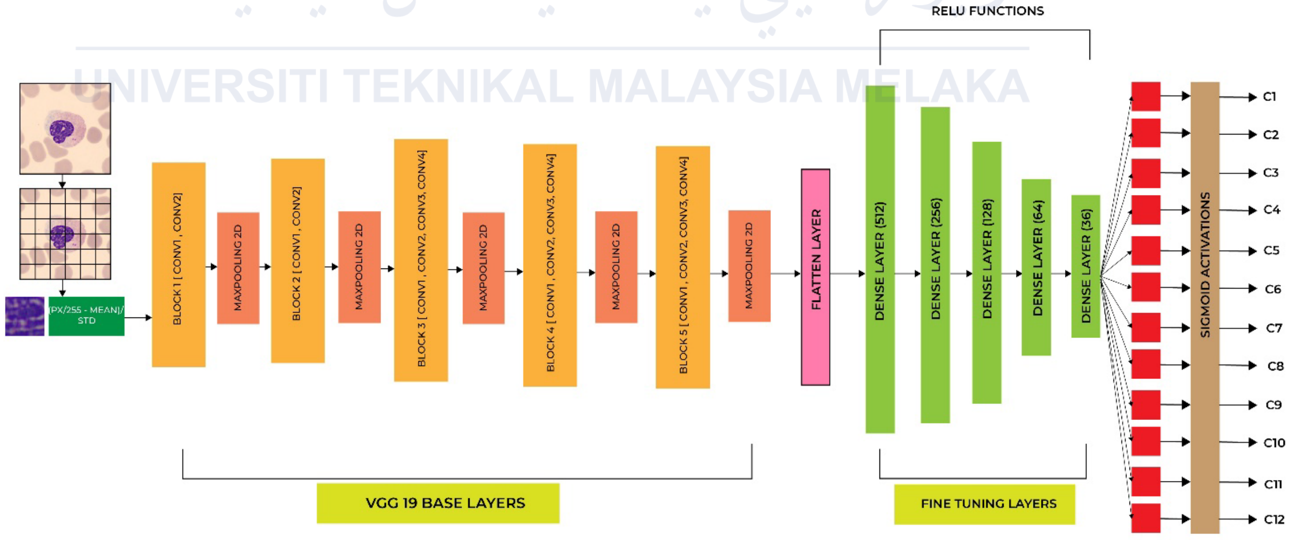


Figure 4.6 : Fully connected VGG 19 with 5 added layers Transfer Learning Model

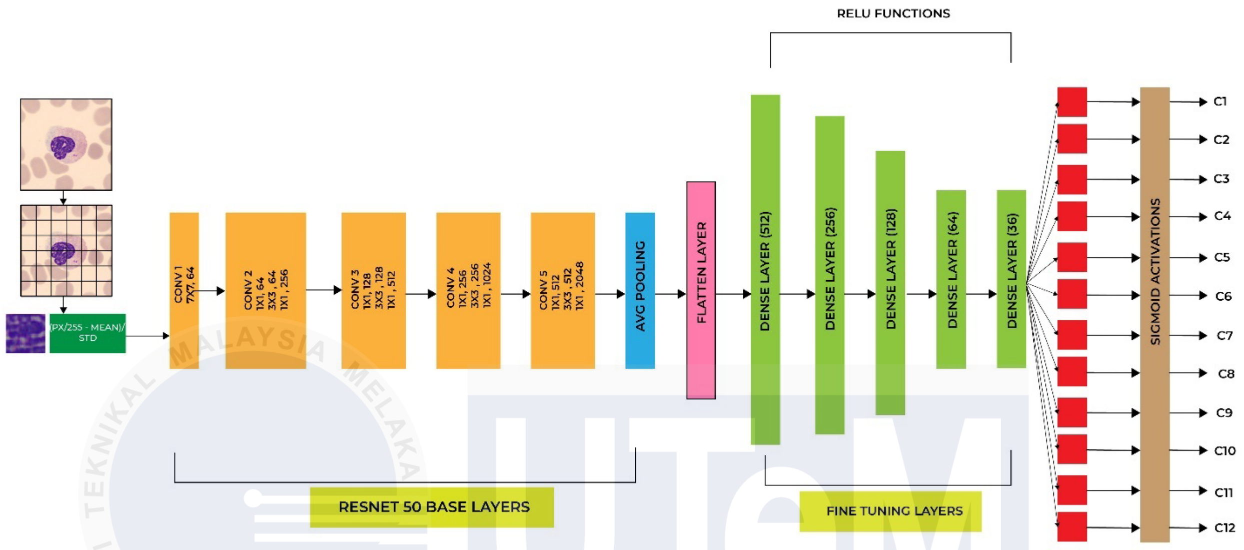


Figure 4.7 : Fully connected RESNET50 with 5 added layers Transfer Learning Model

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

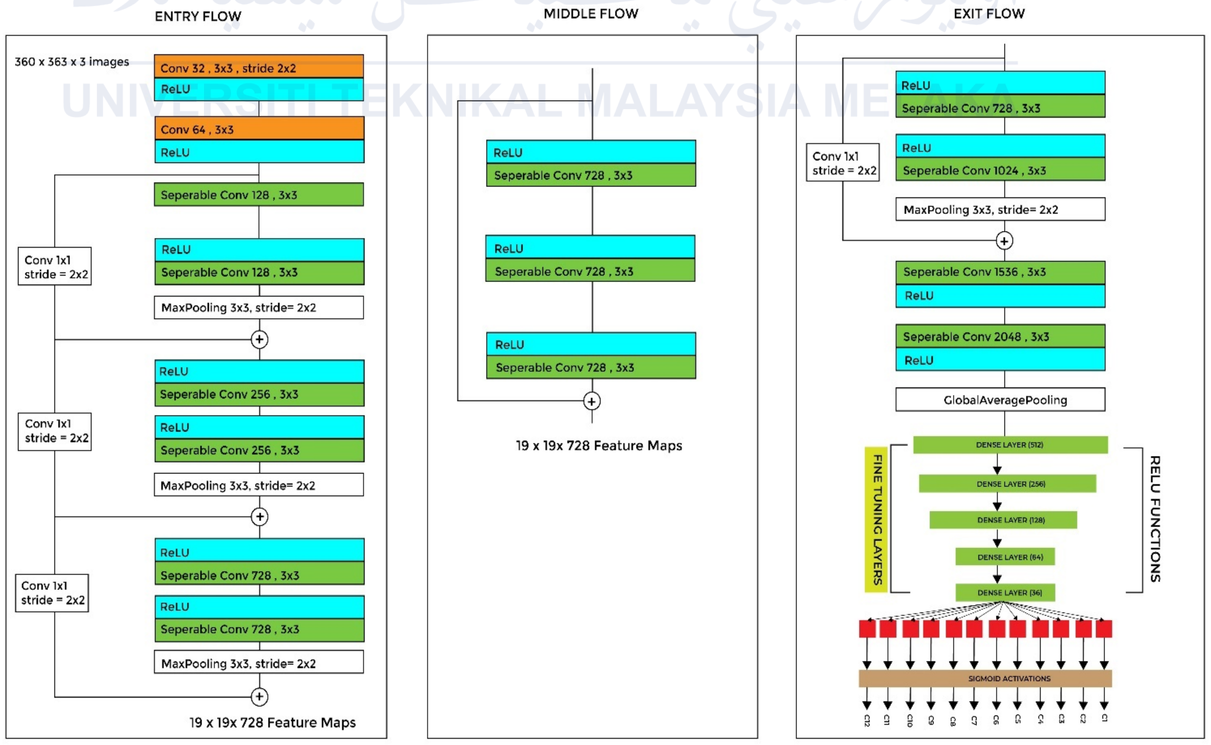


Figure 4.8 : Fully connected XCEPTION with 5 added layers Transfer Learning Model

Table 4.1 : The summary of the models that have been used

BASE VGG 19	
Number of layers	19
Architecture	Has 5 convolutional blocks followed by flattening aiding in down sampling.
Stage details	**Block 1**: 2 conv (64 filters) **Block 2**: 2 conv layers (128 filters) **Block 3**: 4 conv layers(256 filters) **Block 4**: 4 conv layers(512 filters) **Block 5**: 4 conv layers(512 filters) **Fully Connected layer**
Residual blocks	No residual blocks
Activation function	ReLU and SoftMax
Batch normalizations	None
Pooling	MaxPooling after each block and Average Pooling before Fully connected layers
Skip connections	No skip connections
Parameters	138 million
VGG 19 with 5 Added Layers	
Number of layers	19 + 5
Architecture	Has 5 convolutional blocks followed by flattening aiding in down sampling.
Stage details	**Block 1**: 2 conv (64 filters) **Block 2**: 2 conv layers (128 filters) **Block 3**: 4 conv layers (256 filters) **Block 4**: 4 conv layers (512 filters)

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

	Block 5: 4 conv layers (512 filters) **Flatten Layer followed by 5 Dense Layers**
Residual blocks	No residual blocks
Activation function	ReLU
Batch normalizations	None
Pooling	MaxPooling after each block and Average Pooling before Fully connected layers
Skip connections	No skip connections
Parameters	144 million
VGG 19 added 3 Layers with Kaiming Initialisations	
Number of layers	19 + 3
Architecture	Has 5 convolutional blocks followed by flattening and using the final layers to down sample to 2048 and 1024 before activated
Stage details	**Block 1**: 2 conv (64 filters) **Block 2**: 2 conv layers (128 filters) **Block 3**: 4 conv layers (256 filters) **Block 4**: 4 conv layers (512 filters) **Block 5**: 4 conv layers (512 filters) **Flatten Layer followed by 3 Dense Layers and Dropout, Batch Normalizing**
Residual blocks	None
Activation function	ReLU, SoftMax and Kaiming
Batch normalizations	Yes
Pooling	MaxPooling after each block and Average Pooling before Fully connected layers
Skip connections	No skip connections
Parameters	140 million

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

XCEPTION with 5 Added Layers	
Number of layers	71+5
Architecture	Has 71+5 layers using depthwise separable convolutional and residual connections
Stage details	**Entry flow** :3 convolutional blocks **Middle flow** : 8 identical blocks **Exit flow** : 3 convolutional blocks
Residual blocks	Each block has depthwise separable convulutionals followed by pointwise convulutionals
Activation function	ReLU
Batch normalizations	Applied after each layer before ReLU activations
Pooling	MaxPooling after each **Entry** and **Exit flow.** Global Average Pooling before fully connected layers.
Skip connections	Uses depthwise separable convolutional
Parameters	22.9 million
RESNET 50 with 5 Added Layers	
Number of layers	50 + 5
Architecture	Has 5 stages with 50 + 5 layers which uses residual blocks including skip connections
Stage details	Stage 1: 1 convolutional layer (7x7, 64 filters) with MaxPooling Stage2-5: Residual Blocks
Residual blocks	Each block consists of 3 layers: 1. 1x1 conv for reducing dimensions 2. 3x3 conv 3. 1x1 conv for restoring dimensions
Activation function	ReLU

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Batch normalizations	Applied after each convolution layer before ReLU activation
Pooling	Max Pooling after the initial conv layer Average Pooling before the fully connected layer
Skip connections	Identity shortcuts that add the input of a block to its output, mitigating the vanishing gradient problem
Parameters	25.6 million

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.2.2. User Interface Design for expert system/DSS/simulation

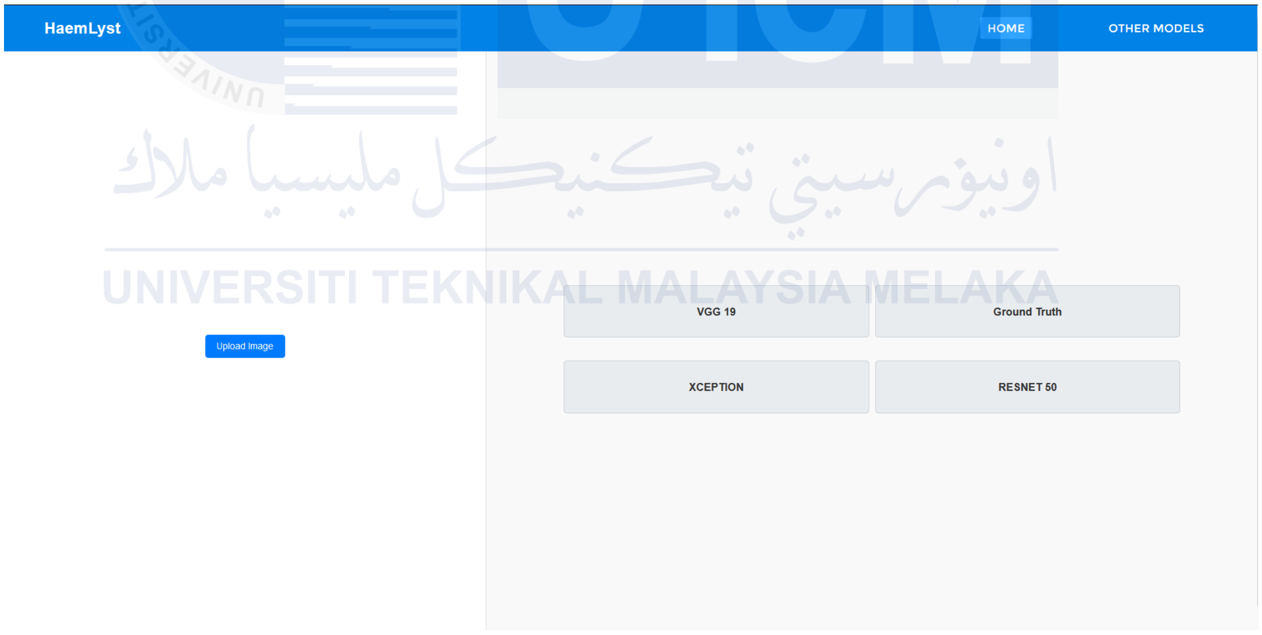
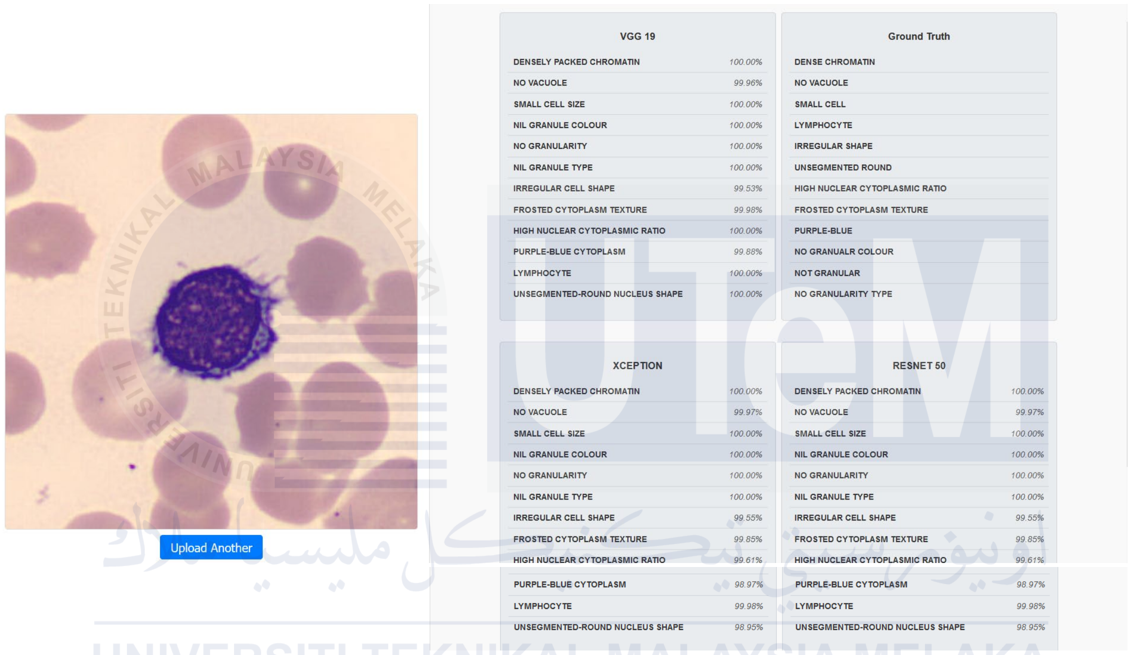


Figure 4.9 : Initial start view of HaemLyst

The HaemLyst Web Application accepts only an image as its input to be processed by the backend system. The application is built in a simple and modern look to increase the usability and the reliability of application. Only image files such as jpeg, jpg, tiff and png are allowed and read by the system. The system will prompt error message if these conditions aren't met. The image is then located by the server, and it executes the image processing on the machine to make predictions as an output.



VGG 19	
DENSELY PACKED CHROMATIN	100.00%
NO VACUOLE	99.96%
SMALL CELL SIZE	100.00%
NIL GRANULE COLOUR	100.00%
NO GRANULARITY	100.00%
NIL GRANULE TYPE	100.00%
IRREGULAR CELL SHAPE	99.53%
FROSTED CYTOPLASM TEXTURE	99.98%
HIGH NUCLEAR CYTOPLASMIC RATIO	100.00%
PURPLE-BLUE CYTOPLASM	99.88%
LYMPHOCYTE	100.00%
UNSEGMENTED-ROUND NUCLEUS SHAPE	100.00%

Ground Truth
DENSE CHROMATIN
NO VACUOLE
SMALL CELL
LYMPHOCYTE
IRREGULAR SHAPE
UNSEGMENTED ROUND
HIGH NUCLEAR CYTOPLASMIC RATIO
FROSTED CYTOPLASM TEXTURE
PURPLE-BLUE
NO GRANUALR COLOUR
NOT GRANULAR
NO GRANULARITY TYPE

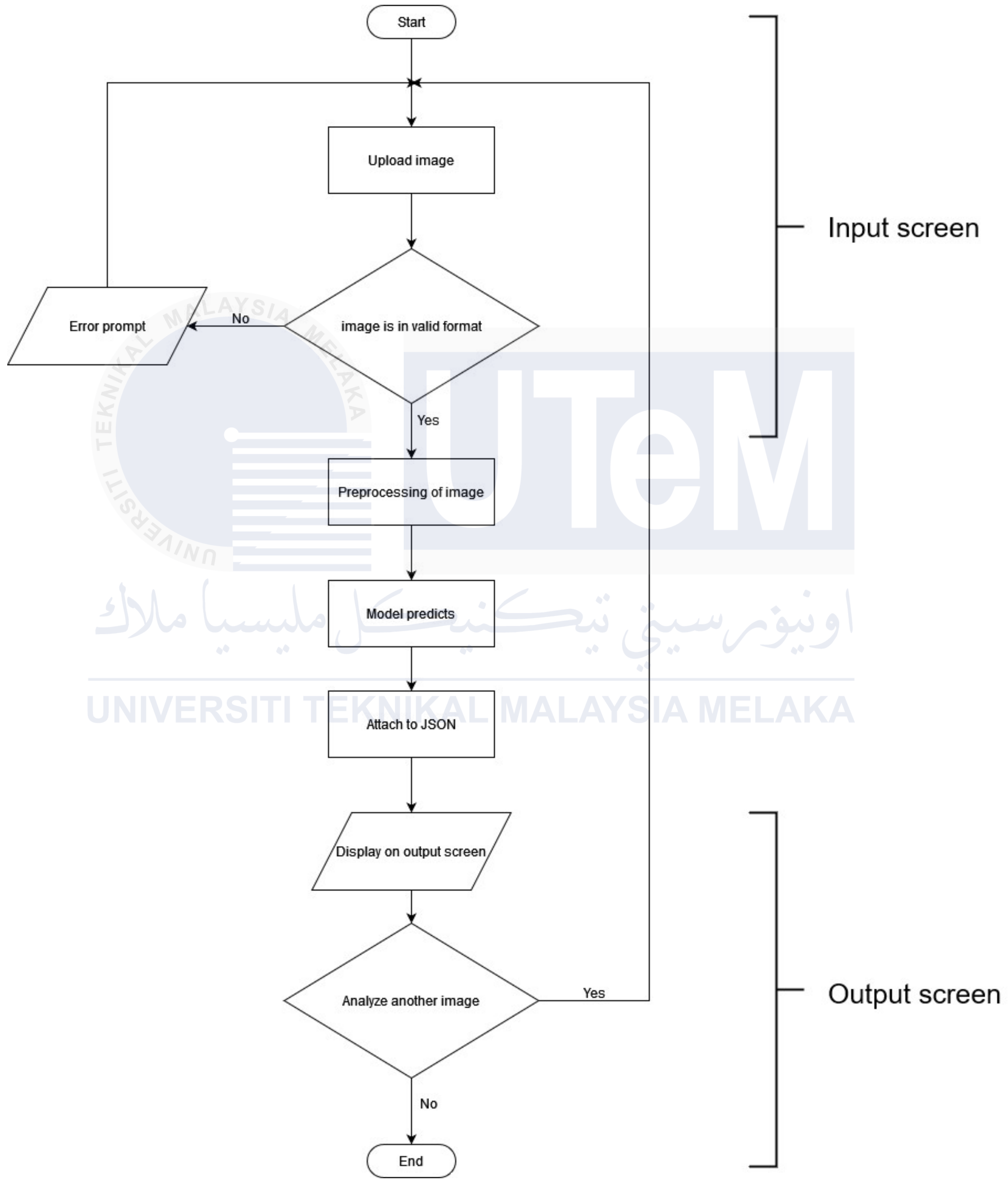
XCEPTION	
DENSELY PACKED CHROMATIN	100.00%
NO VACUOLE	99.97%
SMALL CELL SIZE	100.00%
NIL GRANULE COLOUR	100.00%
NO GRANULARITY	100.00%
NIL GRANULE TYPE	100.00%
IRREGULAR CELL SHAPE	99.55%
FROSTED CYTOPLASM TEXTURE	99.85%
HIGH NUCLEAR CYTOPLASMIC RATIO	99.61%
PURPLE-BLUE CYTOPLASM	98.97%
LYMPHOCYTE	99.98%
UNSEGMENTED-ROUND NUCLEUS SHAPE	98.95%

RESNET 50	
DENSELY PACKED CHROMATIN	100.00%
NO VACUOLE	99.97%
SMALL CELL SIZE	100.00%
NIL GRANULE COLOUR	100.00%
NO GRANULARITY	100.00%
NIL GRANULE TYPE	100.00%
IRREGULAR CELL SHAPE	99.55%
FROSTED CYTOPLASM TEXTURE	99.85%
HIGH NUCLEAR CYTOPLASMIC RATIO	99.61%
PURPLE-BLUE CYTOPLASM	98.97%
LYMPHOCYTE	99.98%
UNSEGMENTED-ROUND NUCLEUS SHAPE	98.95%

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.10 : Output screen of HaemLyst

The output of the predictions is displayed on the right half of HaemLyst where we can see the characteristics of the presented image. The retrieved image is processed and fit into the model in the machine and the predictions alongside probabilities are displayed. There are two columns and rows of output where Row 1 is the best performing Model (VGG 19) and the ground truth values of the image provided. Row 2 is the next two models (XCEPTION and RESNEST 50) that we have experimented with it which also returns prediction of the same image that has been uploaded. The ground truth is only displayed when image from testing dataset is given as it retrieves the actual values from the annotation's dataset for the corresponding images.

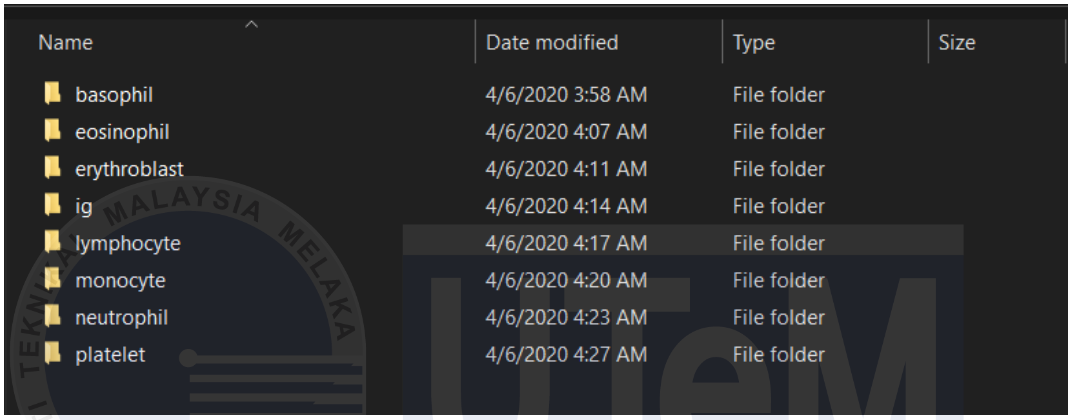


UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 4.11 : Flow chart of HaemLyst web application

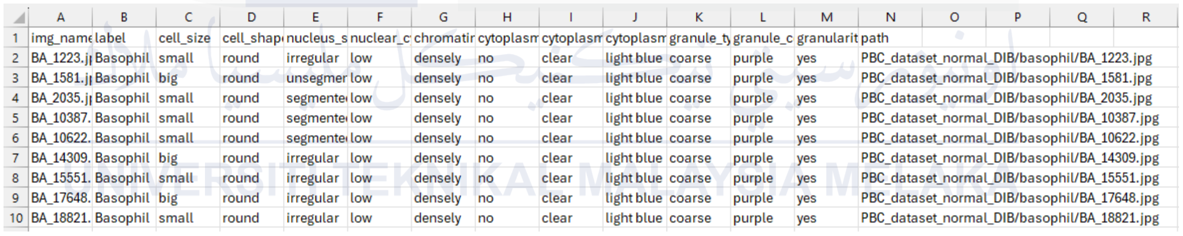
4.2.3. Database Design

A non-database design has been used for the dataset consisting of a csv file for labels and multi folders was used to store the images. The Figures below show the structure of the stored files.



Name	Date modified	Type	Size
basophil	4/6/2020 3:58 AM	File folder	
eosinophil	4/6/2020 4:07 AM	File folder	
erythroblast	4/6/2020 4:11 AM	File folder	
ig	4/6/2020 4:14 AM	File folder	
lymphocyte	4/6/2020 4:17 AM	File folder	
monocyte	4/6/2020 4:20 AM	File folder	
neutrophil	4/6/2020 4:23 AM	File folder	
platelet	4/6/2020 4:27 AM	File folder	

Figure 4.12 : Folder containing the images of cells according to names



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	img_name	label	cell_size	cell_shap	nucleus_s	nuclear_c	chromatir	cytoplasm	cytoplasm	cytoplasm	granule_ty	granule_c	granularit	path				
2	BA_1223.jp	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_1223.jpg				
3	BA_1581.jp	Basophil	big	round	unsegmer	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_1581.jpg				
4	BA_2035.jp	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_2035.jpg				
5	BA_10387.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_10387.jpg				
6	BA_10622.	Basophil	small	round	segmente	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_10622.jpg				
7	BA_14309.	Basophil	big	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_14309.jpg				
8	BA_15551.	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_15551.jpg				
9	BA_17648.	Basophil	big	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_17648.jpg				
10	BA_18821.	Basophil	small	round	irregular	low	densely	no	clear	light blue	coarse	purple	yes	PBC_dataset_normal_DIB/basophil/BA_18821.jpg				

Figure 4.13 : CSV file snippet containing the labels of each image

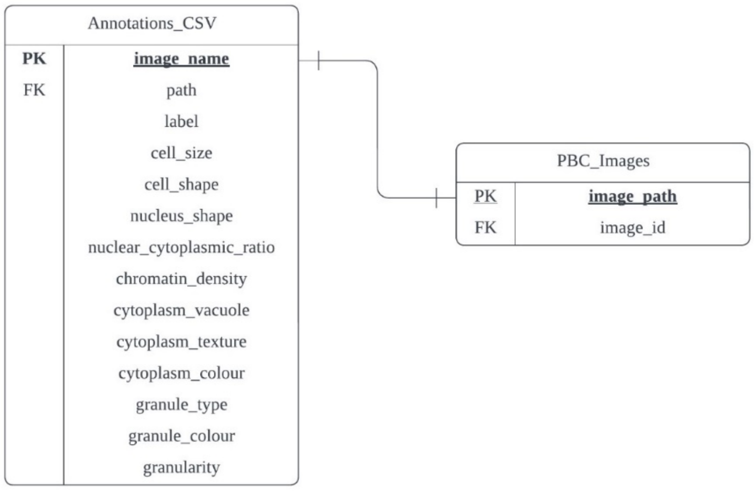


Figure 4.14 : E.R.D visualization of relation between the datasets

4.3. AI Component Design

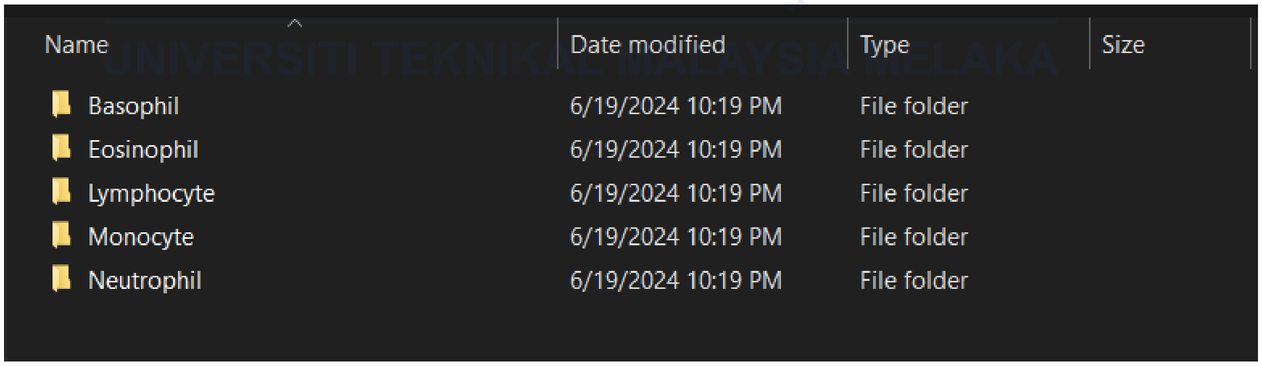
4.3.1. Dataset Description

The modified dataset has the image per category following the counts below:

Basophil	: 1218 images
Eosinophil	: 3,117 images
Lymphocyte	: 1214 images
Monocyte	: 1420 images
Neutrophil	: 3329 images

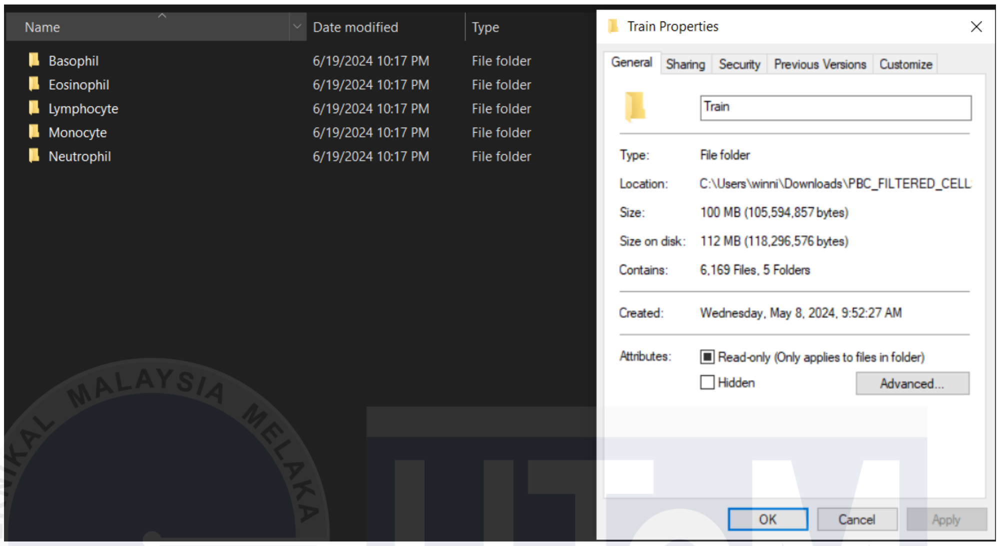
The dataset of images was then further modified/ transformed into three categories – training, testing and validation. The training set contains 6169 images, training set contained 3099 images and validation set contains 1030 images, which were all randomly picked all over the label file.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



Name	Date modified	Type	Size
Basophil	6/19/2024 10:19 PM	File folder	
Eosinophil	6/19/2024 10:19 PM	File folder	
Lymphocyte	6/19/2024 10:19 PM	File folder	
Monocyte	6/19/2024 10:19 PM	File folder	
Neutrophil	6/19/2024 10:19 PM	File folder	

Figure 4.15 : Modified folder containing the images of cells according to names

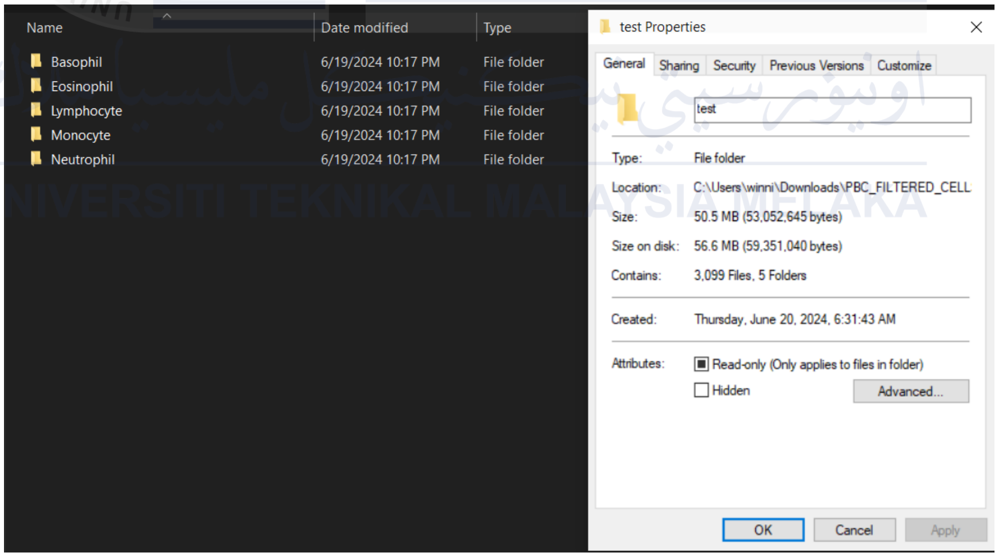


Name	Date modified	Type
Basophil	6/19/2024 10:17 PM	File folder
Eosinophil	6/19/2024 10:17 PM	File folder
Lymphocyte	6/19/2024 10:17 PM	File folder
Monocyte	6/19/2024 10:17 PM	File folder
Neutrophil	6/19/2024 10:17 PM	File folder

Train Properties

Type:	File folder
Location:	C:\Users\winni\Downloads\PBC_FILTERED_CELL
Size:	100 MB (105,594,857 bytes)
Size on disk:	112 MB (118,296,576 bytes)
Contains:	6,169 Files, 5 Folders
Created:	Wednesday, May 8, 2024, 9:52:27 AM
Attributes:	▣ Read-only (Only applies to files in folder) ☐ Hidden

Figure 4.16 : Numbers of training set and structure of folder



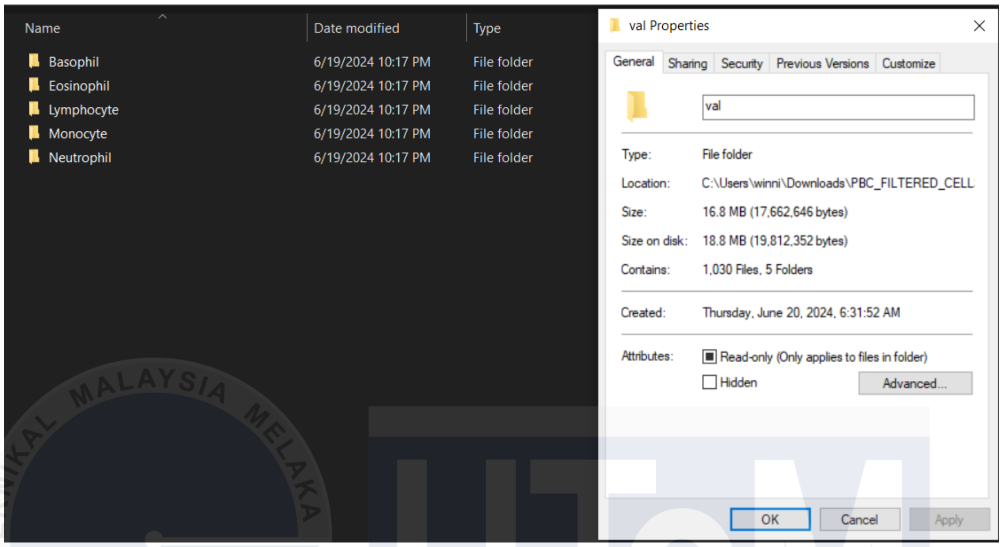
Name	Date modified	Type
Basophil	6/19/2024 10:17 PM	File folder
Eosinophil	6/19/2024 10:17 PM	File folder
Lymphocyte	6/19/2024 10:17 PM	File folder
Monocyte	6/19/2024 10:17 PM	File folder
Neutrophil	6/19/2024 10:17 PM	File folder

test Properties

Type:	File folder
Location:	C:\Users\winni\Downloads\PBC_FILTERED_CELL
Size:	50.5 MB (53,052,645 bytes)
Size on disk:	56.6 MB (59,351,040 bytes)
Contains:	3,099 Files, 5 Folders
Created:	Thursday, June 20, 2024, 6:31:43 AM
Attributes:	▣ Read-only (Only applies to files in folder) ☐ Hidden

Figure 4.17 : Numbers of testing set and structure of folder

UNIVERSITI TEKNIKAL MALAYSIA MELAKA


Name | Date modified | Type

Name	Date modified	Type
Basophil	6/19/2024 10:17 PM	File folder
Eosinophil	6/19/2024 10:17 PM	File folder
Lymphocyte	6/19/2024 10:17 PM	File folder
Monocyte	6/19/2024 10:17 PM	File folder
Neutrophil	6/19/2024 10:17 PM	File folder

val Properties

General | Sharing | Security | Previous Versions | Customize

val

Type: File folder
Location: C:\Users\winni\Downloads\PBC_FILTERED_CELL
Size: 16.8 MB (17,662,646 bytes)
Size on disk: 18.8 MB (19,812,352 bytes)
Contains: 1,030 Files, 5 Folders
Created: Thursday, June 20, 2024, 6:31:52 AM
Attributes: Read-only (Only applies to files in folder); Hidden

Figure 4.18 : Numbers of validation set and structure of folder

Figure 4.16 shows the csv file equipped with 1000 rows and 14 columns of data. All data in csv file used Natural Language – numbers and words. No usage of special characters all is encoded in UTF-8 format. Below is the table that summarizes the columns entries as a dictionary.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 4.2 : Data dictionary of the annotations file

Column name	**Representation**	**Values**
img_name	The name of the image pointed to data in row	[cell abbreviation _ index.jpg] E.g.: MO_131262.jpg
label	Type of cell in image	['Basophil', 'Neutrophil', 'Eosinophil', 'Monocyte', ‘Lymphocyte']
cell_size	Size of cell in image	[big, small]
cell_shape	Shape of cell	['round', 'irregular']
nucleus_shape	Shape of nucleus	['irregular', 'unsegmented-indented', 'segmented-bilobed', 'segmented-multilobed’,

		'unsegmented-band', 'unsegmented-round']
nuclear_cytoplasmic_ratio	Ratio of nucleus to cytoplasm	['low', 'high']
chromatin_density	Density of chromatin	['densely', 'loosely']
cytoplasm_vacuole	Presence of vacuole	['no', 'yes']
cytoplasm_texture	Texture of cytoplasm	['clear', 'frosted']
cytoplasm_colour	Colour of cytoplasm	['light blue', 'blue', 'purple blue']
granule_type	Type of granule	['coarse', 'small', 'round', 'nil']
granule_colour	Colour of granule	['purple', 'pink', 'red', 'nil']
granularity	Granularity presence	['yes', 'no']
Path	Path pointing towards the image folder location	root/category/img_name

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The CSV file used in this study has been annotated by Satoshi Tsutsui et al., as described in their paper. We filtered the images from the PBC Dataset to correspond to the labeled data, ensuring that only images with corresponding labels were used. This approach resulted in a modified dataset of images used for our analysis

4.3.2. AI Techniques

In this project we have Supervised Deep Learning using Computer Vision in the means of classifying the White Blood Cell present in the image and predicting it's the 12 attributes confirming the cell. The breakdown of the technique is Machine Learning (Supervised Learning), Deep Learning and Computer Vision.

4.3.3. Deep Learning

Deep learning is the branch of machine learning which is based on artificial neural network architecture. An artificial neural network or ANN uses layers of interconnected nodes called neurons that work together to process and learn from the input data.

In a fully connected Deep neural network, there is an input layer and one or more hidden layers connected one after the other. Each neuron receives input from the previous layer neurons or the input layer. The output of one neuron becomes the input to other neurons in the next layer of the network, and this process continues until the final layer produces the output of the network. The layers of the neural network transform the input data through a series of nonlinear transformations, allowing the network to learn complex representations of the input data.

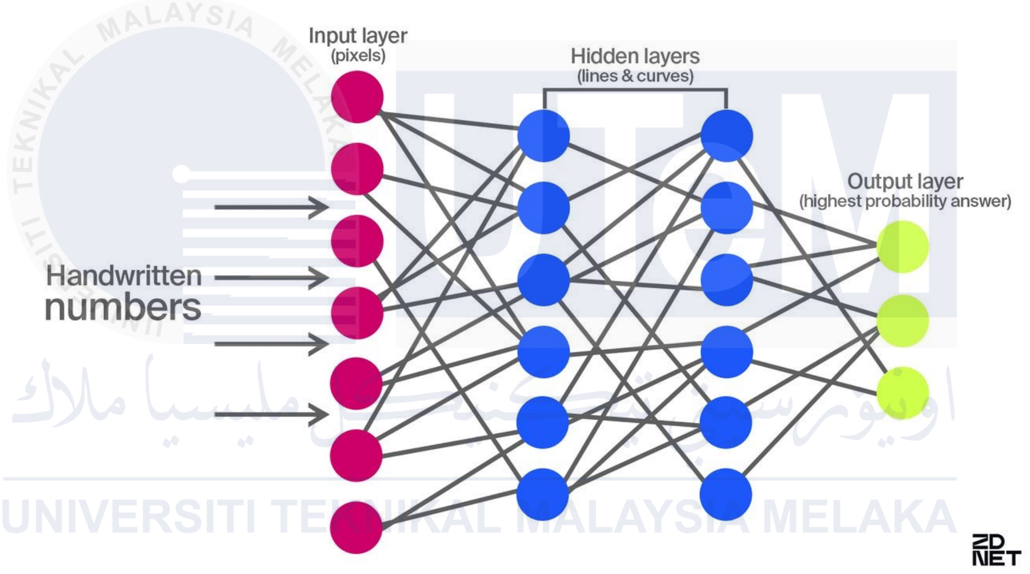


Figure 4.19 : The general structure of a neural network

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The Deep learning method we have applied in this project is Convolutional Neural Network which is tailored for image analysis suiting our primary objective of cell analysis from a microscopic image. A convolutional neural network (CNN) is a form of machine learning model that is particularly well-suited to evaluating visual data. CNNs, also known as convnets, use linear algebra principles, namely convolution operations, to extract features and find patterns in images. CNN architecture is inspired by the human brain's connectivity patterns, particularly the visual cortex, which is critical for perceiving and processing visual data. The artificial neurons in a CNN are designed to efficiently interpret visual input, allowing these models to analyze whole images.

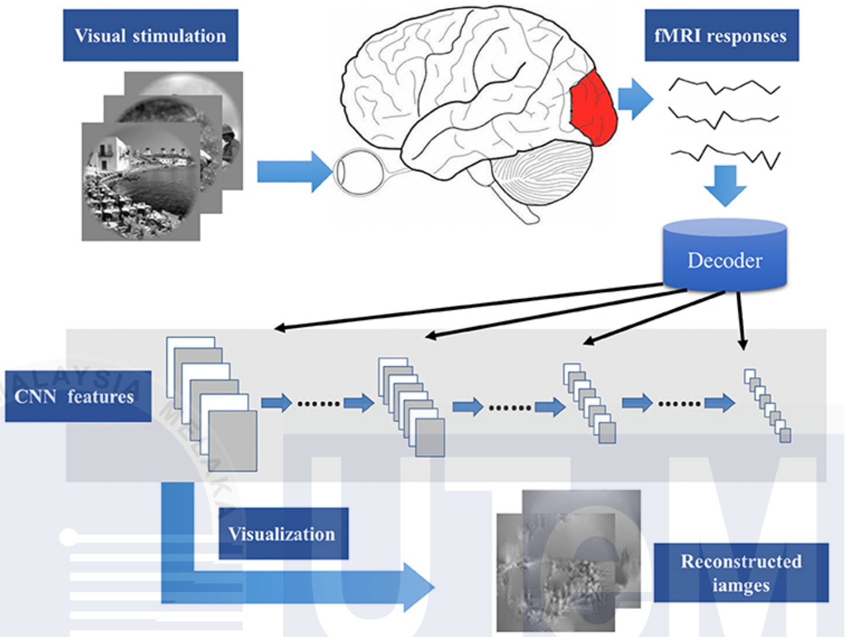


Figure 4.20: The illustration of Convolutional Neural Network

The convolutional neural network is best for our project as it is a supervised learning algorithm where we have both the image data and the sufficient annotated data for each cell.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

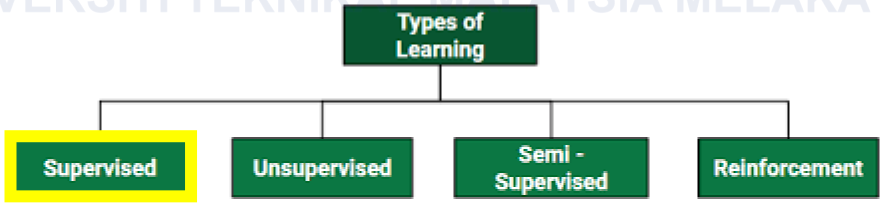


Figure 4.21 : The types of learning methods in Machine learning

In this project Supervised Learning embedded with the CNN aids to overcome this Multilabel Classification Problem. Supervised Learning combination of an input data set and the intended output is inferred from the training data. AI systems learn from a labelled dataset, where each data point is associated with a known outcome. In this project our labels are the attributes of the cell such as *Size of cell* where each image in the database is annotated with a value where the model learns the pattern for the prediction and evaluation during the implementation phase. Since our annotation data (labels) are in the form of Natural Language we must take an extra step to convert those to a machine understanding value. Thus, we have used the One-Hot encoding method to convert the labels into either 1 or 0. The One Hot encoding method allows the handling of the large categorical data present in our label data. For instance, in column [‘label’] there are 5 unique values ['Basophil', 'Neutrophil', 'Eosinophil', 'Monocyte’, ‘Lymphocyte']. Thus, encoding would allow the machine to understand the disparity between each category and its sub-categories.

4.3.4. Computer vision

Computer vision is a subfield of AI and machine learning that allows machines to see, analyze, and interpret visuals such as photographs, videos, and so on, as well as extract usable information from them that may be used to make decisions in AI applications. It can be viewed as an eye for an AI application. In this project we have used computer vision to enable the system to read the image of cell pixel by pixel. Through this we can deeply analyze the image and classify it alongside the raw data we have fed the system. The method used in this case is Image Classification and before classifying the image into different labels the images need to be preprocessed – resizing and normalization of pixels. This is vital due to the nature of a microscopic image such as cells presence in the background, color difference that is minute to human eyes and the contours of each pixel in image. To evict the errors and increase the efficiency of our model computer vision algorithm plays the most important role.
Steps of the computer vision algorithm is illustrated below:

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

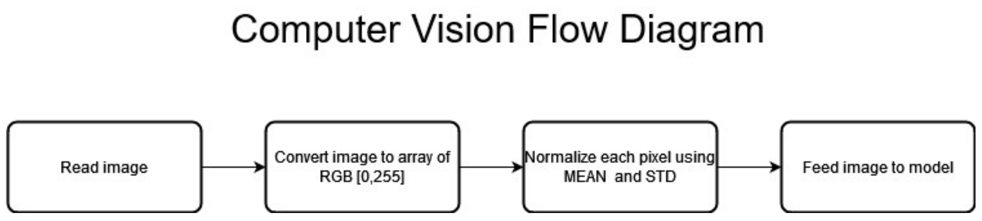

Computer Vision Flow Diagram
Read image
Convert image to array of RGB [0,255]
Normalize each pixel using MEAN and STD
Feed image to model

Figure 4.22 : Computer vision flow diagram illustrated for a cell image

4.4. Software or Hardware Design

This project does not apply the software and hardware design

4.5. Summary

In this chapter on system design for AI Blood Smear Analysis, the focus is on developing a web application utilizing Convolutional Neural Networks (CNNs) for white blood cell classification. Three prominent architectures—VGG 19, RESNET 50, and XCEPTION—are employed through transfer learning from ImageNet weights. Each model undergoes fine-tuning with additional layers optimized for the specific task of classifying 12 distinct cell attributes. The user interface (UI) of the HaemLyst web application is designed to accept image inputs, process them using the trained models, and display predictions with probabilities. The backend leverages computer vision techniques for image preprocessing, ensuring accurate analysis of microscopic cell images. The dataset structure includes folders for images and a CSV file for labels, facilitating efficient data management and annotation. Overall, the project integrates supervised deep learning techniques and computer vision to address multilabel classification challenges in cellular analysis effectively.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 5. RESULTS AND DISCUSSION

5.1. Introduction

This chapter provides an in-depth discussion of the results obtained and the technical aspects involved in the implementation of **HaemLyst**. We will explore the testing phase, including the strategies and activities employed to ensure the robustness and reliability of our system. Additionally, the chapter covers the methods used to assess the overall performance of our models in addressing the multi-label classification problem, focusing on white blood cell analysis.

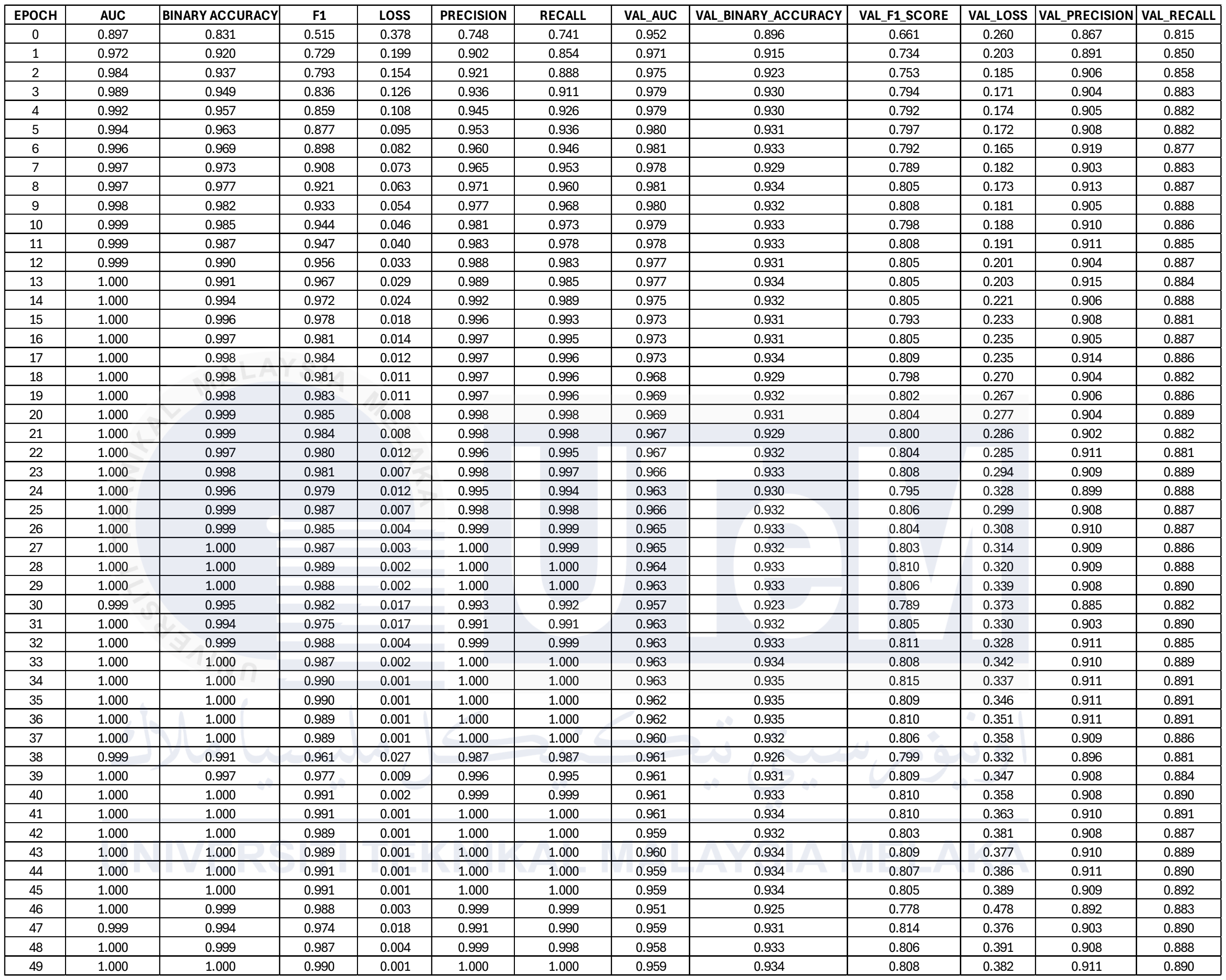
5.2. Evaluation of AI Techniques

Based on the 3 models we experimented with we have the best model which is VGG 19 with 5 added layers among the same variations of RESNET 50 and EXCEPTION. These models binary AUC, Binary Accuracy, F1-Score, Loss Precision and Recall for both validation and training we recorded and illustrated in table below.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

EPOCH	AUC	BINARY ACCURACY	F1	LOSS	PRECISION	RECALL	VAL_AUC	VAL_BINARY_ACCURACY	VAL_F1_SCORE	VAL_LOSS	VAL_PRECISION	VAL_RECALL
0	0.878	0.822	0.409	0.405	0.816	0.600	0.952	0.898	0.639	0.267	0.885	0.797
1	0.973	0.926	0.740	0.199	0.916	0.855	0.978	0.931	0.769	0.179	0.923	0.864
2	0.987	0.947	0.814	0.138	0.937	0.901	0.981	0.933	0.784	0.161	0.913	0.885
3	0.992	0.958	0.850	0.110	0.949	0.922	0.984	0.939	0.801	0.148	0.922	0.892
4	0.994	0.964	0.873	0.093	0.958	0.935	0.984	0.939	0.807	0.148	0.916	0.899
5	0.996	0.970	0.887	0.080	0.965	0.945	0.984	0.940	0.814	0.147	0.923	0.894
6	0.997	0.975	0.907	0.069	0.971	0.954	0.984	0.938	0.796	0.153	0.921	0.891
7	0.998	0.979	0.920	0.060	0.975	0.960	0.984	0.939	0.822	0.150	0.914	0.902
8	0.998	0.983	0.931	0.052	0.980	0.967	0.984	0.940	0.817	0.154	0.922	0.896
9	0.999	0.986	0.941	0.044	0.985	0.974	0.983	0.940	0.815	0.159	0.923	0.894
10	0.999	0.990	0.954	0.037	0.989	0.980	0.983	0.940	0.819	0.163	0.922	0.895
11	1.000	0.992	0.959	0.031	0.991	0.985	0.981	0.938	0.808	0.177	0.915	0.897
12	1.000	0.994	0.968	0.025	0.994	0.989	0.981	0.940	0.810	0.173	0.924	0.892
13	1.000	0.996	0.978	0.020	0.996	0.992	0.979	0.939	0.812	0.189	0.918	0.896
14	1.000	0.997	0.980	0.016	0.997	0.995	0.980	0.940	0.815	0.184	0.921	0.895
15	1.000	0.998	0.983	0.013	0.998	0.996	0.978	0.938	0.806	0.198	0.917	0.896
16	1.000	0.998	0.983	0.011	0.998	0.997	0.978	0.939	0.817	0.202	0.920	0.895
17	1.000	0.999	0.988	0.009	0.999	0.998	0.978	0.940	0.821	0.204	0.917	0.901
18	1.000	0.999	0.985	0.008	0.999	0.999	0.977	0.939	0.820	0.211	0.918	0.897
19	1.000	0.999	0.988	0.007	0.999	0.999	0.976	0.940	0.817	0.216	0.922	0.896
20	1.000	0.999	0.988	0.005	0.999	0.999	0.973	0.938	0.813	0.234	0.917	0.894
21	1.000	0.999	0.987	0.005	0.999	0.999	0.976	0.941	0.828	0.220	0.922	0.900
22	1.000	1.000	0.988	0.004	1.000	0.999	0.974	0.941	0.821	0.232	0.921	0.899
23	1.000	1.000	0.985	0.003	1.000	1.000	0.973	0.940	0.822	0.238	0.920	0.899
24	1.000	1.000	0.987	0.003	1.000	1.000	0.973	0.940	0.826	0.242	0.919	0.900
25	1.000	1.000	0.991	0.002	1.000	1.000	0.971	0.940	0.821	0.256	0.920	0.898
26	1.000	0.996	0.979	0.012	0.994	0.994	0.972	0.940	0.823	0.251	0.920	0.899
27	1.000	0.999	0.986	0.005	0.999	0.999	0.971	0.941	0.825	0.253	0.921	0.900
28	1.000	1.000	0.989	0.002	1.000	1.000	0.969	0.940	0.825	0.274	0.918	0.899
29	1.000	1.000	0.989	0.001	1.000	1.000	0.970	0.940	0.819	0.271	0.919	0.900
30	1.000	1.000	0.991	0.001	1.000	1.000	0.970	0.941	0.825	0.266	0.921	0.902
31	1.000	1.000	0.990	0.001	1.000	1.000	0.970	0.941	0.824	0.271	0.922	0.899
32	1.000	1.000	0.987	0.001	1.000	1.000	0.969	0.941	0.827	0.277	0.921	0.899
33	1.000	1.000	0.987	0.001	1.000	1.000	0.969	0.941	0.825	0.278	0.921	0.900
34	1.000	1.000	0.990	0.001	1.000	1.000	0.969	0.941	0.824	0.281	0.922	0.899
35	1.000	1.000	0.989	0.001	1.000	1.000	0.966	0.939	0.817	0.299	0.917	0.899
36	1.000	1.000	0.989	0.001	1.000	1.000	0.968	0.940	0.819	0.291	0.920	0.898
37	1.000	1.000	0.989	0.001	1.000	1.000	0.966	0.940	0.820	0.306	0.917	0.901
38	1.000	1.000	0.985	0.001	1.000	1.000	0.966	0.939	0.822	0.305	0.922	0.893
39	0.999	0.997	0.985	0.010	0.996	0.996	0.969	0.940	0.825	0.283	0.918	0.901
40	1.000	1.000	0.992	0.002	1.000	1.000	0.965	0.936	0.813	0.319	0.913	0.895
41	1.000	1.000	0.990	0.002	0.999	0.999	0.967	0.939	0.821	0.303	0.917	0.900
42	1.000	1.000	0.989	0.001	1.000	1.000	0.966	0.941	0.822	0.305	0.921	0.900
43	1.000	1.000	0.990	0.001	1.000	1.000	0.966	0.940	0.822	0.314	0.917	0.901
44	1.000	1.000	0.991	0.001	1.000	1.000	0.967	0.941	0.825	0.305	0.921	0.902
45	1.000	1.000	0.991	0.001	1.000	1.000	0.966	0.940	0.820	0.317	0.918	0.900
46	1.000	0.998	0.984	0.008	0.997	0.997	0.967	0.938	0.820	0.300	0.911	0.903
47	1.000	0.999	0.987	0.005	0.998	0.998	0.964	0.936	0.816	0.329	0.911	0.896
48	1.000	1.000	0.989	0.001	1.000	1.000	0.965	0.940	0.825	0.314	0.921	0.897
49	1.000	1.000	0.990	0.000	1.000	1.000	0.965	0.940	0.819	0.316	0.917	0.902

Figure 5.1 : Metrics of VGG 19 with 5 Added Layers for 50 epochs



EPOCH	AUC	BINARY ACCURACY	F1	LOSS	PRECISION	RECALL	VAL_AUC	VAL_BINARY_ACCURACY	VAL_F1_SCORE	VAL_LOSS	VAL_PRECISION	VAL_RECALL
0	0.897	0.831	0.515	0.378	0.748	0.741	0.952	0.896	0.661	0.260	0.867	0.815
1	0.972	0.920	0.729	0.199	0.902	0.854	0.971	0.915	0.734	0.203	0.891	0.850
2	0.984	0.937	0.793	0.154	0.921	0.888	0.975	0.923	0.753	0.185	0.906	0.858
3	0.989	0.949	0.836	0.126	0.936	0.911	0.979	0.930	0.794	0.171	0.904	0.883
4	0.992	0.957	0.859	0.108	0.945	0.926	0.979	0.930	0.792	0.174	0.905	0.882
5	0.994	0.963	0.877	0.095	0.953	0.936	0.980	0.931	0.797	0.172	0.908	0.882
6	0.996	0.969	0.898	0.082	0.960	0.946	0.981	0.933	0.792	0.165	0.919	0.877
7	0.997	0.973	0.908	0.073	0.965	0.953	0.978	0.929	0.789	0.182	0.903	0.883
8	0.997	0.977	0.921	0.063	0.971	0.960	0.981	0.934	0.805	0.173	0.913	0.887
9	0.998	0.982	0.933	0.054	0.977	0.968	0.980	0.932	0.808	0.181	0.905	0.888
10	0.999	0.985	0.944	0.046	0.981	0.973	0.979	0.933	0.798	0.188	0.910	0.886
11	0.999	0.987	0.947	0.040	0.983	0.978	0.978	0.933	0.808	0.191	0.911	0.885
12	0.999	0.990	0.956	0.033	0.988	0.983	0.977	0.931	0.805	0.201	0.904	0.887
13	1.000	0.991	0.967	0.029	0.989	0.985	0.977	0.934	0.805	0.203	0.915	0.884
14	1.000	0.994	0.972	0.024	0.992	0.989	0.975	0.932	0.805	0.221	0.906	0.888
15	1.000	0.996	0.978	0.018	0.996	0.993	0.973	0.931	0.793	0.233	0.908	0.881
16	1.000	0.997	0.981	0.014	0.997	0.995	0.973	0.931	0.805	0.235	0.905	0.887
17	1.000	0.998	0.984	0.012	0.997	0.996	0.973	0.934	0.809	0.235	0.914	0.886
18	1.000	0.998	0.981	0.011	0.997	0.996	0.968	0.929	0.798	0.270	0.904	0.882
19	1.000	0.998	0.983	0.011	0.997	0.996	0.969	0.932	0.802	0.267	0.906	0.886
20	1.000	0.999	0.985	0.008	0.998	0.998	0.969	0.931	0.804	0.277	0.904	0.889
21	1.000	0.999	0.984	0.008	0.998	0.998	0.967	0.929	0.800	0.286	0.902	0.882
22	1.000	0.997	0.980	0.012	0.996	0.995	0.967	0.932	0.804	0.285	0.911	0.881
23	1.000	0.998	0.981	0.007	0.998	0.997	0.966	0.933	0.808	0.294	0.909	0.889
24	1.000	0.996	0.979	0.012	0.995	0.994	0.963	0.930	0.795	0.328	0.899	0.888
25	1.000	0.999	0.987	0.007	0.998	0.998	0.966	0.932	0.806	0.299	0.908	0.887
26	1.000	0.999	0.985	0.004	0.999	0.999	0.965	0.933	0.804	0.308	0.910	0.887
27	1.000	1.000	0.987	0.003	1.000	0.999	0.965	0.932	0.803	0.314	0.909	0.886
28	1.000	1.000	0.989	0.002	1.000	1.000	0.964	0.933	0.810	0.320	0.909	0.888
29	1.000	1.000	0.988	0.002	1.000	1.000	0.963	0.933	0.806	0.339	0.908	0.890
30	0.999	0.995	0.982	0.017	0.993	0.992	0.957	0.923	0.789	0.373	0.885	0.882
31	1.000	0.994	0.975	0.017	0.991	0.991	0.963	0.932	0.805	0.330	0.903	0.890
32	1.000	0.999	0.988	0.004	0.999	0.999	0.963	0.933	0.811	0.328	0.911	0.885
33	1.000	1.000	0.987	0.002	1.000	1.000	0.963	0.934	0.808	0.342	0.910	0.889
34	1.000	1.000	0.990	0.001	1.000	1.000	0.963	0.935	0.815	0.337	0.911	0.891
35	1.000	1.000	0.990	0.001	1.000	1.000	0.962	0.935	0.809	0.346	0.911	0.891
36	1.000	1.000	0.989	0.001	1.000	1.000	0.962	0.935	0.810	0.351	0.911	0.891
37	1.000	1.000	0.989	0.001	1.000	1.000	0.960	0.932	0.806	0.358	0.909	0.886
38	0.999	0.991	0.961	0.027	0.987	0.987	0.961	0.926	0.799	0.332	0.896	0.881
39	1.000	0.997	0.977	0.009	0.996	0.995	0.961	0.931	0.809	0.347	0.908	0.884
40	1.000	1.000	0.991	0.002	0.999	0.999	0.961	0.933	0.810	0.358	0.908	0.890
41	1.000	1.000	0.991	0.001	1.000	1.000	0.961	0.934	0.810	0.363	0.910	0.891
42	1.000	1.000	0.989	0.001	1.000	1.000	0.959	0.932	0.803	0.381	0.908	0.887
43	1.000	1.000	0.989	0.001	1.000	1.000	0.960	0.934	0.809	0.377	0.910	0.889
44	1.000	1.000	0.991	0.001	1.000	1.000	0.959	0.934	0.807	0.386	0.911	0.890
45	1.000	1.000	0.991	0.001	1.000	1.000	0.959	0.934	0.805	0.389	0.909	0.892
46	1.000	0.999	0.988	0.003	0.999	0.999	0.951	0.925	0.778	0.478	0.892	0.883
47	0.999	0.994	0.974	0.018	0.991	0.990	0.959	0.931	0.814	0.376	0.903	0.890
48	1.000	0.999	0.987	0.004	0.999	0.998	0.958	0.933	0.806	0.391	0.908	0.888
49	1.000	1.000	0.990	0.001	1.000	1.000	0.959	0.934	0.808	0.382	0.911	0.890

Figure 5.2 : Metrics of XCEPTION with 5 Added Layers for 50 epochs

UTeM UNIVERSITI TEKNIKAL MALAYSIA MELAKA

EPOCH	AUC	BINARY ACCURACY	F1	LOSS	PRECISION	RECALL	VAL_AUC	VAL_BINARY_ACCURACY	VAL_F1_SCORE	VAL_LOSS	VAL_PRECISION	VAL_RECALL
0	0.892	0.838	0.518	0.386	0.778	0.720	0.933	0.883	0.627	0.302	0.848	0.790
1	0.952	0.898	0.678	0.258	0.868	0.819	0.954	0.896	0.690	0.253	0.862	0.820
2	0.969	0.916	0.738	0.207	0.891	0.854	0.967	0.910	0.722	0.216	0.884	0.841
3	0.978	0.928	0.775	0.177	0.906	0.874	0.970	0.914	0.719	0.207	0.901	0.833
4	0.984	0.939	0.805	0.152	0.922	0.893	0.973	0.920	0.757	0.195	0.906	0.850
5	0.986	0.942	0.815	0.143	0.925	0.899	0.972	0.919	0.744	0.204	0.901	0.849
6	0.989	0.951	0.842	0.122	0.937	0.915	0.979	0.927	0.776	0.175	0.914	0.863
7	0.991	0.956	0.859	0.111	0.944	0.924	0.975	0.922	0.774	0.193	0.894	0.871
8	0.992	0.959	0.864	0.103	0.947	0.929	0.978	0.928	0.794	0.177	0.907	0.873
9	0.994	0.962	0.876	0.096	0.950	0.935	0.976	0.925	0.777	0.192	0.899	0.874
10	0.994	0.964	0.881	0.090	0.954	0.939	0.976	0.924	0.781	0.191	0.895	0.875
11	0.995	0.969	0.895	0.081	0.960	0.947	0.978	0.929	0.797	0.182	0.900	0.885
12	0.996	0.972	0.905	0.074	0.963	0.951	0.979	0.929	0.793	0.182	0.900	0.886
13	0.996	0.972	0.904	0.072	0.964	0.953	0.978	0.930	0.788	0.186	0.905	0.882
14	0.997	0.975	0.915	0.066	0.968	0.957	0.972	0.922	0.770	0.217	0.890	0.874
15	0.997	0.977	0.915	0.061	0.971	0.961	0.978	0.931	0.796	0.191	0.910	0.880
16	0.998	0.980	0.924	0.056	0.974	0.965	0.977	0.930	0.793	0.197	0.909	0.878
17	0.998	0.980	0.931	0.054	0.974	0.965	0.975	0.926	0.801	0.207	0.895	0.880
18	0.998	0.983	0.932	0.049	0.978	0.970	0.974	0.928	0.782	0.216	0.905	0.875
19	0.999	0.983	0.937	0.047	0.978	0.971	0.974	0.927	0.797	0.223	0.899	0.881
20	0.999	0.985	0.943	0.042	0.981	0.975	0.972	0.925	0.781	0.234	0.897	0.877
21	0.999	0.986	0.948	0.040	0.983	0.976	0.975	0.931	0.802	0.217	0.905	0.886
22	0.999	0.986	0.943	0.039	0.982	0.976	0.974	0.930	0.797	0.223	0.902	0.887
23	0.999	0.989	0.956	0.034	0.986	0.981	0.975	0.930	0.807	0.216	0.900	0.889
24	0.999	0.990	0.957	0.030	0.988	0.983	0.973	0.929	0.798	0.232	0.904	0.880
25	1.000	0.992	0.964	0.025	0.990	0.987	0.972	0.928	0.800	0.240	0.904	0.878
26	1.000	0.993	0.964	0.025	0.991	0.987	0.971	0.929	0.796	0.246	0.903	0.883
27	1.000	0.992	0.963	0.025	0.990	0.986	0.972	0.927	0.804	0.243	0.895	0.884
28	1.000	0.994	0.969	0.021	0.992	0.990	0.971	0.930	0.803	0.246	0.905	0.881
29	1.000	0.994	0.972	0.019	0.993	0.990	0.967	0.925	0.785	0.282	0.892	0.880
30	1.000	0.995	0.976	0.017	0.994	0.992	0.970	0.930	0.802	0.261	0.900	0.889
31	0.998	0.987	0.950	0.041	0.981	0.979	0.969	0.926	0.790	0.263	0.899	0.877
32	1.000	0.996	0.977	0.016	0.995	0.993	0.970	0.930	0.808	0.260	0.903	0.886
33	1.000	0.998	0.980	0.011	0.997	0.996	0.970	0.931	0.805	0.265	0.905	0.885
34	1.000	0.998	0.985	0.009	0.998	0.997	0.969	0.929	0.810	0.272	0.901	0.885
35	1.000	0.998	0.983	0.010	0.998	0.996	0.968	0.930	0.807	0.283	0.900	0.888
36	1.000	0.998	0.983	0.010	0.997	0.996	0.965	0.929	0.794	0.308	0.902	0.882
37	1.000	0.998	0.985	0.008	0.998	0.997	0.966	0.930	0.801	0.296	0.903	0.886
38	1.000	0.999	0.986	0.007	0.998	0.998	0.963	0.929	0.799	0.315	0.903	0.880
39	1.000	0.998	0.983	0.007	0.998	0.997	0.964	0.929	0.796	0.309	0.903	0.882
40	1.000	0.996	0.978	0.012	0.995	0.994	0.965	0.930	0.803	0.307	0.903	0.887
41	1.000	0.998	0.986	0.008	0.997	0.997	0.962	0.929	0.804	0.330	0.901	0.885
42	0.994	0.976	0.921	0.083	0.967	0.961	0.974	0.927	0.803	0.222	0.901	0.879
43	1.000	0.995	0.976	0.019	0.995	0.992	0.971	0.929	0.802	0.248	0.903	0.882
44	1.000	0.998	0.985	0.011	0.998	0.996	0.970	0.931	0.803	0.260	0.905	0.885
45	1.000	0.999	0.985	0.009	0.999	0.998	0.969	0.930	0.806	0.269	0.902	0.887
46	1.000	0.996	0.980	0.014	0.995	0.994	0.967	0.930	0.796	0.283	0.904	0.884
47	1.000	0.999	0.988	0.006	0.999	0.999	0.966	0.930	0.801	0.294	0.904	0.883
48	1.000	1.000	0.988	0.005	0.999	0.999	0.964	0.930	0.799	0.311	0.904	0.885
49	1.000	1.000	0.990	0.004	1.000	0.999	0.964	0.930	0.806	0.306	0.901	0.884

Figure 5.3 : Metrics of RESNET 50 with 5 Added Layers for 50 epochs

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Table 5.1 : Summary of Metrics of Model Architecture With 5 Added Layers

Metrics	VGG 19 + 5 LAYERS	XCEPTION + 5 LAYERS	RESNET 50 +5 LAYERS
AUC	0.996	0.996	0.994
Binary Accuracy	0.989	0.987	0.977
F1Score	0.954	0.950	0.921
Precision	0.988	0.983	0.970
Recall	0.978	0.978	0.961
Validation AUC	0.973	0.967	0.970
Validation Binary Accuracy	0.939	0.931	0.926
Validation F1 Score	0.813	0.798	0.789
Validation Precision	0.918	0.906	0.900
Validation Recall	0.895	0.884	0.875

As table 11 illustrates, all three-model performance in training is above 97% which shows the ability of models to learn complex patterns from the dataset. But if we analyze the validation process of the models, we can infer that the models F1-Score is relatively lower when compared to those during Training. This is a major problem as the model tends to memorize patterns and mimic predictions. This phenomenon could be also called Overfitting. This is because the algorithm fits too closely with the training data that it starts making errors during testing and validation. This also means the models cannot generalize the patterns from the dataset.

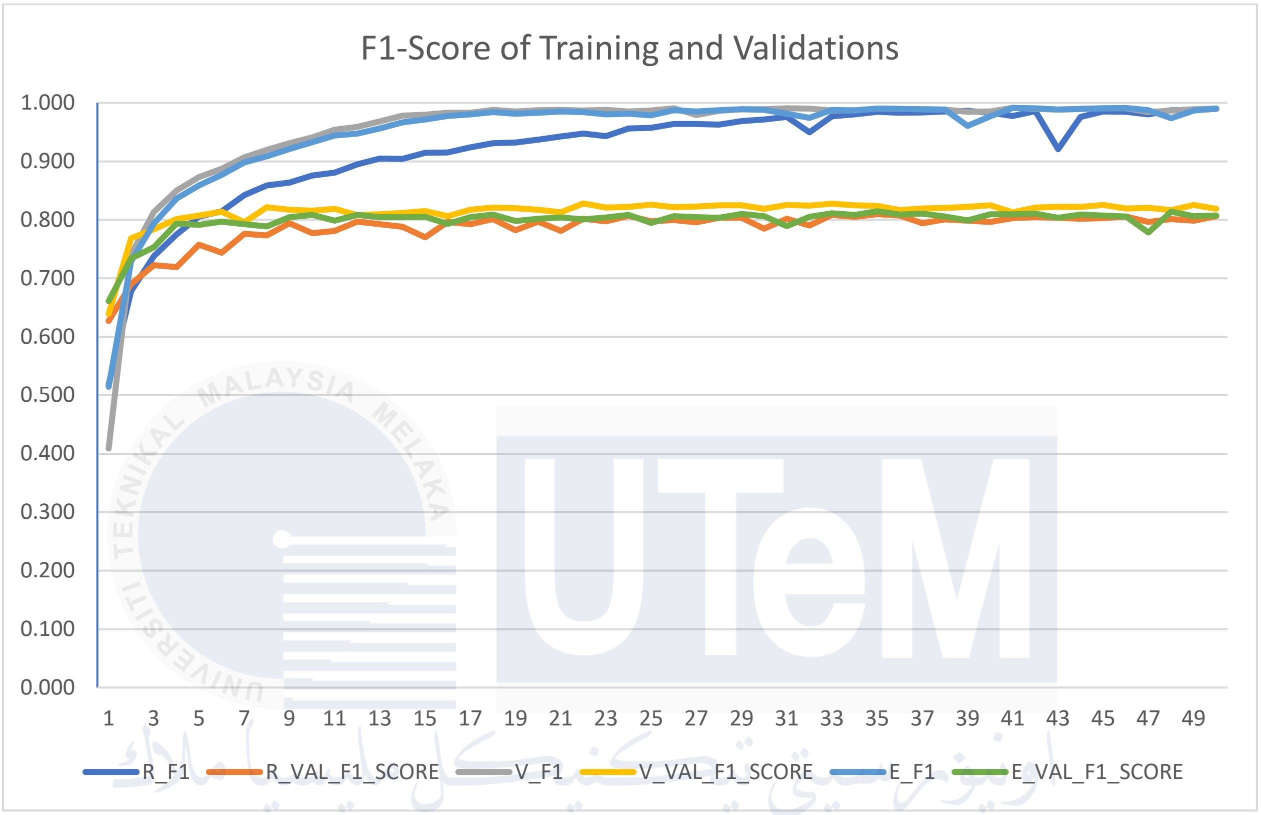
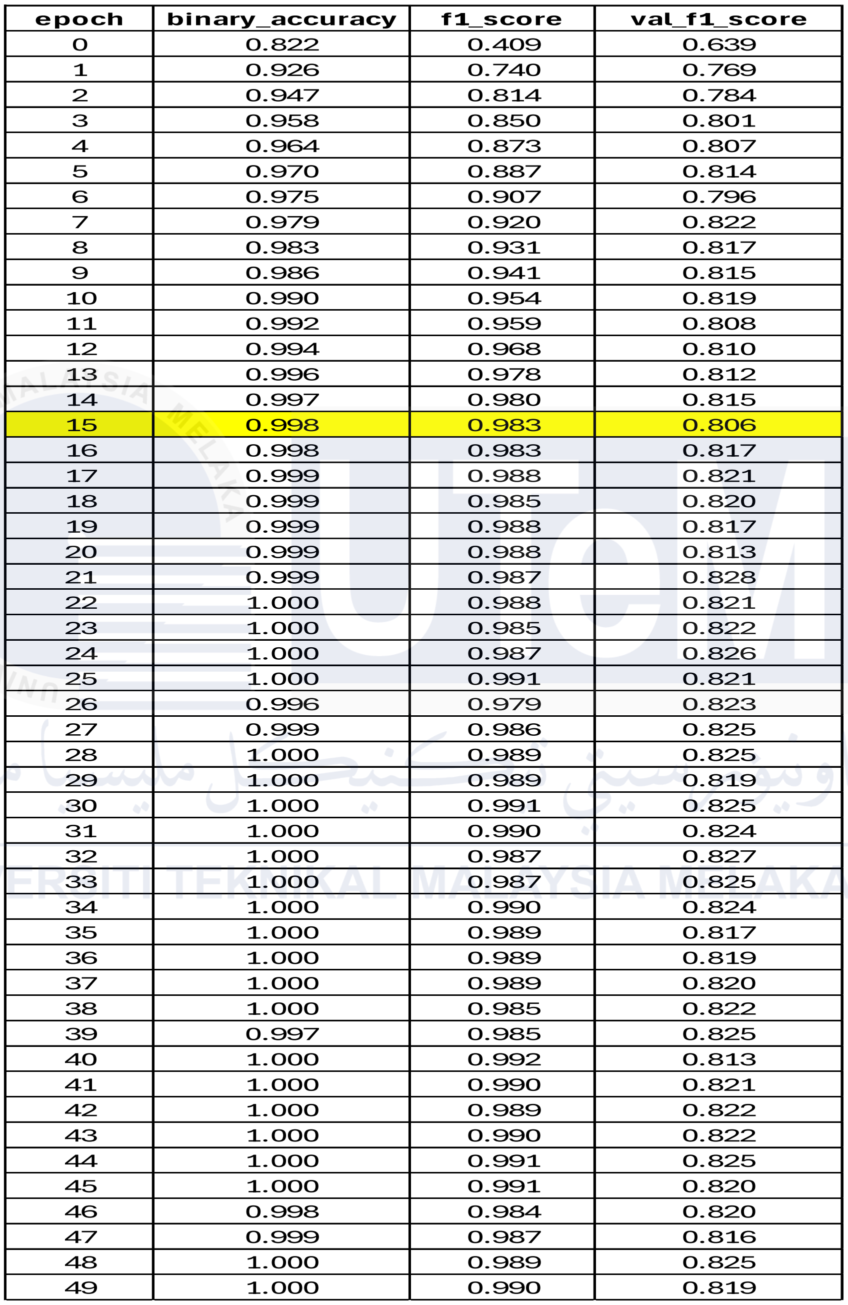


Figure 5.4 : Chart of F1-Scores between Training and Validation

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 5.4 shows the very consistent values of validation and its nature where there is no significant increment towards or closer to the Training F1-Score proves the Overfitting of the models. This is particularly due to the complex architecture of the model that directly flattens from 4096 to 512 marks on the fine-tuned layers. The overfitting started on the 15th Epoch where the model doesn't learn very vastly rather minute details and random guessing. Figure 5.4 below shows the start of the Overfitting in VGG 19 with 5 added layers.



epoch	binary_accuracy	f1_score	val_f1_score
0	0.822	0.409	0.639
1	0.926	0.740	0.769
2	0.947	0.814	0.784
3	0.958	0.850	0.801
4	0.964	0.873	0.807
5	0.970	0.887	0.814
6	0.975	0.907	0.796
7	0.979	0.920	0.822
8	0.983	0.931	0.817
9	0.986	0.941	0.815
10	0.990	0.954	0.819
11	0.992	0.959	0.808
12	0.994	0.968	0.810
13	0.996	0.978	0.812
14	0.997	0.980	0.815
15	0.998	0.983	0.806
16	0.998	0.983	0.817
17	0.999	0.988	0.821
18	0.999	0.985	0.820
19	0.999	0.988	0.817
20	0.999	0.988	0.813
21	0.999	0.987	0.828
22	1.000	0.988	0.821
23	1.000	0.985	0.822
24	1.000	0.987	0.826
25	1.000	0.991	0.821
26	0.996	0.979	0.823
27	0.999	0.986	0.825
28	1.000	0.989	0.825
29	1.000	0.989	0.819
30	1.000	0.991	0.825
31	1.000	0.990	0.824
32	1.000	0.987	0.827
33	1.000	0.987	0.825
34	1.000	0.990	0.824
35	1.000	0.989	0.817
36	1.000	0.989	0.819
37	1.000	0.989	0.820
38	1.000	0.985	0.822
39	0.997	0.985	0.825
40	1.000	0.992	0.813
41	1.000	0.990	0.821
42	1.000	0.989	0.822
43	1.000	0.990	0.822
44	1.000	0.991	0.825
45	1.000	0.991	0.820
46	0.998	0.984	0.820
47	0.999	0.987	0.816
48	1.000	0.989	0.825
49	1.000	0.990	0.819

Figure 5.5 : Overfitting phase in VGG 19 with 5 Added Layers

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

As we take a closer look at the main metrics of VGG 19 with 5 Added Layers we can see the model has excellent Binary Accuracy, AUC and F1-Score. But all those are for training and a compromise has been analyzed during the validation of white blood cell data. This is said to be as we look at the F1 Score of both training and validation there is a large difference of (~18%) which also means that the model is Overfitted. This can be further proved after Epoch 15 as the Training F1 Score continues to improve from 0.983 while validation doesn't show a consistent improvement.

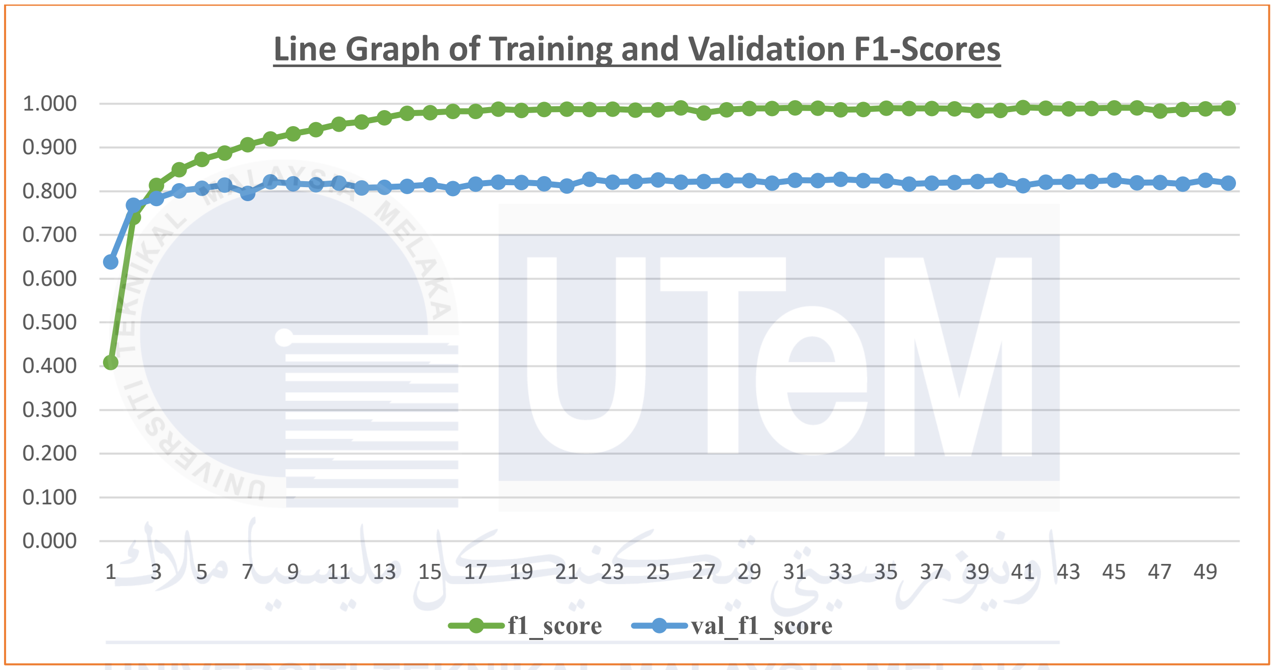
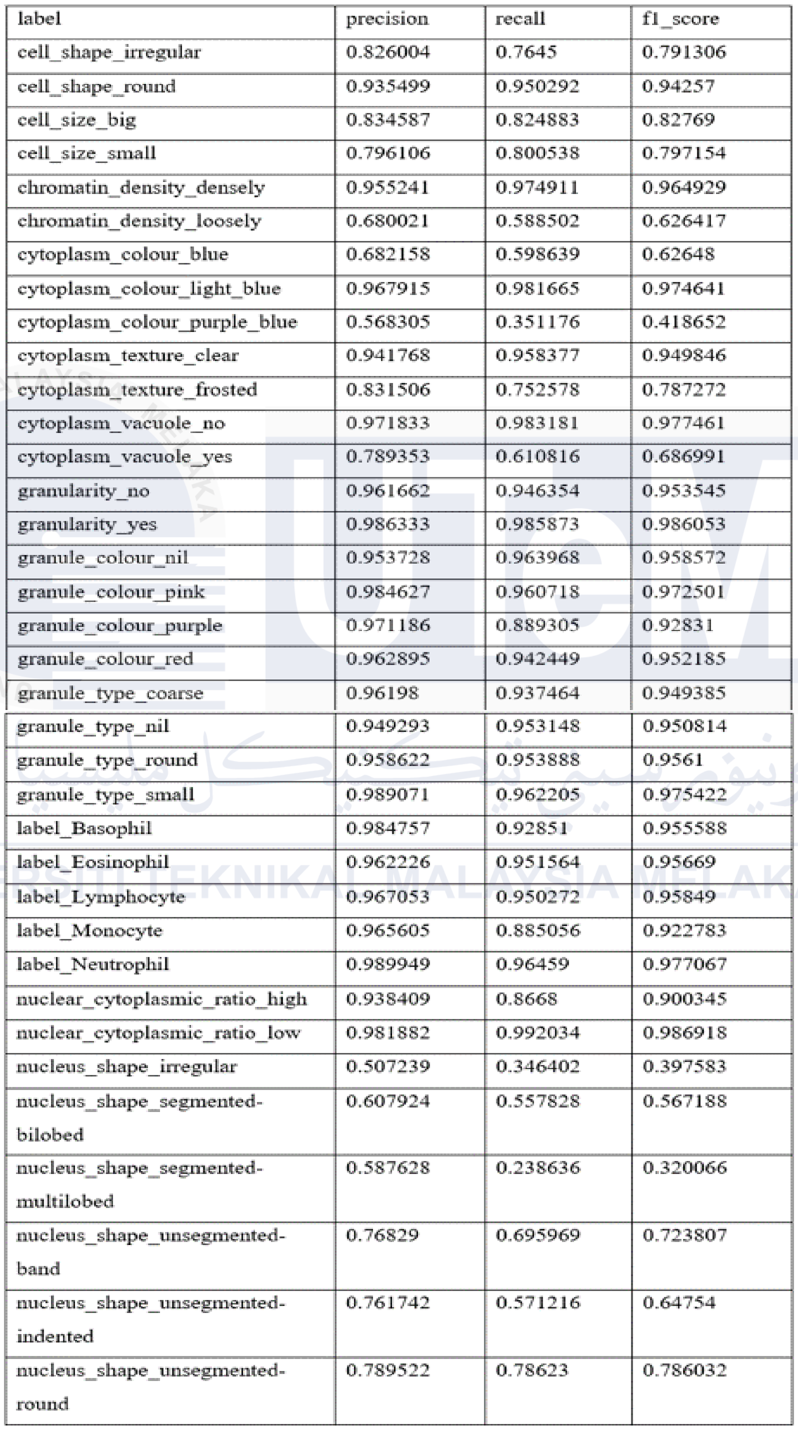


Figure 5.6 : Training and Validation F1 Scores

The chart in Figure 5.6 above illustrates the constant F1-Score of Validation where no improvements are significant through the 50 epochs while the Training F1 Score rockets up to 99% compromising on validations. This can also be a hypothesis to the lower metrics that we took account for every label using the VGG 19 with 5 Added Layers Model Architecture.



label	precision	recall	f1_score
cell_shape_irregular	0.826004	0.7645	0.791306
cell_shape_round	0.935499	0.950292	0.94257
cell_size_big	0.834587	0.824883	0.82769
cell_size_small	0.796106	0.800538	0.797154
chromatin_density_densely	0.955241	0.974911	0.964929
chromatin_density_loosely	0.680021	0.588502	0.626417
cytoplasm_colour_blue	0.682158	0.598639	0.62648
cytoplasm_colour_light_blue	0.967915	0.981665	0.974641
cytoplasm_colour_purple_blue	0.568305	0.351176	0.418652
cytoplasm_texture_clear	0.941768	0.958377	0.949846
cytoplasm_texture_frosted	0.831506	0.752578	0.787272
cytoplasm_vacuole_no	0.971833	0.983181	0.977461
cytoplasm_vacuole_yes	0.789353	0.610816	0.686991
granularity_no	0.961662	0.946354	0.953545
granularity_yes	0.986333	0.985873	0.986053
granule_colour_nil	0.953728	0.963968	0.958572
granule_colour_pink	0.984627	0.960718	0.972501
granule_colour_purple	0.971186	0.889305	0.92831
granule_colour_red	0.962895	0.942449	0.952185
granule_type_coarse	0.96198	0.937464	0.949385
granule_type_nil	0.949293	0.953148	0.950814
granule_type_round	0.958622	0.953888	0.9561
granule_type_small	0.989071	0.962205	0.975422
label_Basophil	0.984757	0.92851	0.955588
label_Eosinophil	0.962226	0.951564	0.95669
label_Lymphocyte	0.967053	0.950272	0.95849
label_Monocyte	0.965605	0.885056	0.922783
label_Neutrophil	0.989949	0.96459	0.977067
nuclear_cytoplasmic_ratio_high	0.938409	0.8668	0.900345
nuclear_cytoplasmic_ratio_low	0.981882	0.992034	0.986918
nucleus_shape_irregular	0.507239	0.346402	0.397583
nucleus_shape_segmented-bilobed	0.607924	0.557828	0.567188
nucleus_shape_segmented-multilobed	0.587628	0.238636	0.320066
nucleus_shape_unsegmented-band	0.76829	0.695969	0.723807
nucleus_shape_unsegmented-indented	0.761742	0.571216	0.64754
nucleus_shape_unsegmented-round	0.789522	0.78623	0.786032

Figure 5.7 : Individual Attribute Metrics

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

As listed on Figure 5.7 we can observe that Nucleus Shape, Cytoplasm Vacuole Presence, Cell Size and Cytoplasm Colour have a relatively lower F1-Score and Recall values. Lower recall value indicates that the model is failing to identify a significant portion of the relevant instances, which could suggest that the model is not capturing all the necessary patterns in the data. This is particularly concerning when both recall and F1 scores are low, as it may indicate that the model is overfitting and is based more on the training data and cannot generalize on the validation dataset or unseen data.

5.2.1. Overcoming the Overfitting

To accommodate the overfitting that has been on the models we have proposed a newer solution with the best architecture type that we have used previously. This solution uses a smaller architecture but with a maximized capability of parameters. The model is fine-tuned with weight decay method as an improvisation to the architecture. The weight decay (λ) was initiliased to 0.01 which adds to the loss function of the model. The hyperparameter is defined as below

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

$$\text{Loss} = \text{Original Loss} + \lambda \sum_i w_i^2$$

where:

- λ is the weight decay coefficient
- w_i are the weights of the model.

Equation 5.1 :Weight Decay Tuning Function

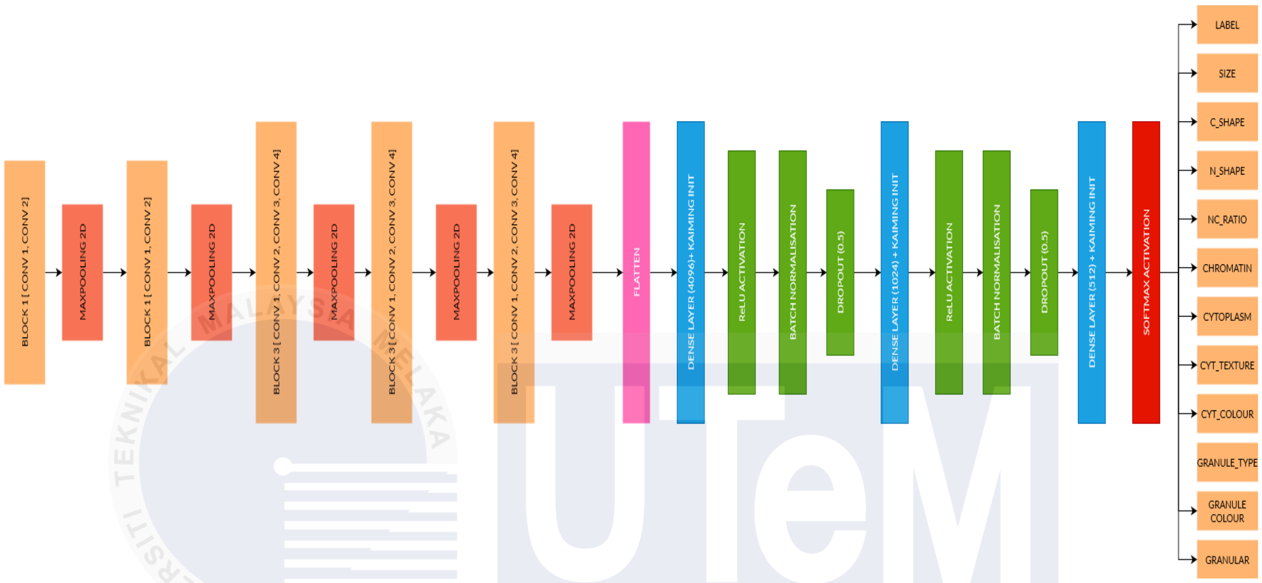
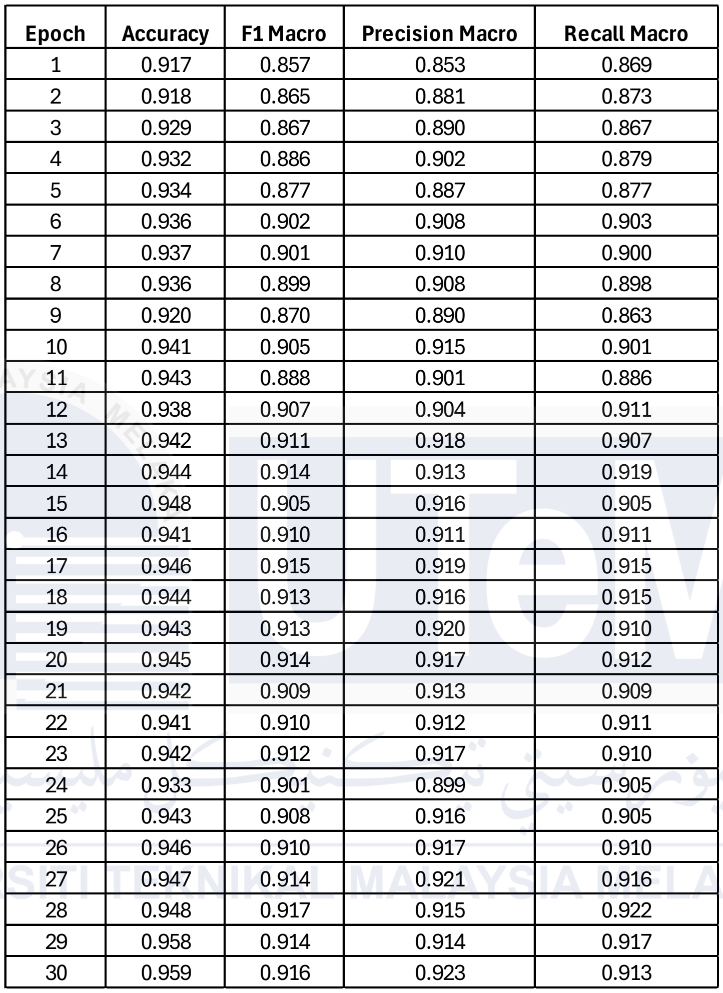


Figure 5.8 : VGG 19 added 3 layers with Kaiming Initialization.

This architecture shown in Figure 5.8 was initialized using the same idea of 5-layers but due to the lack of generalization we have downscaled the layers to 3 with weight modifications, batch normalization and dropout to make the model learn better without overfitting. The overall metrics of the model's validation on the dataset are illustrated in the figure below.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



Epoch	Accuracy	F1 Macro	Precision Macro	Recall Macro
1	0.917	0.857	0.853	0.869
2	0.918	0.865	0.881	0.873
3	0.929	0.867	0.890	0.867
4	0.932	0.886	0.902	0.879
5	0.934	0.877	0.887	0.877
6	0.936	0.902	0.908	0.903
7	0.937	0.901	0.910	0.900
8	0.936	0.899	0.908	0.898
9	0.920	0.870	0.890	0.863
10	0.941	0.905	0.915	0.901
11	0.943	0.888	0.901	0.886
12	0.938	0.907	0.904	0.911
13	0.942	0.911	0.918	0.907
14	0.944	0.914	0.913	0.919
15	0.948	0.905	0.916	0.905
16	0.941	0.910	0.911	0.911
17	0.946	0.915	0.919	0.915
18	0.944	0.913	0.916	0.915
19	0.943	0.913	0.920	0.910
20	0.945	0.914	0.917	0.912
21	0.942	0.909	0.913	0.909
22	0.941	0.910	0.912	0.911
23	0.942	0.912	0.917	0.910
24	0.933	0.901	0.899	0.905
25	0.943	0.908	0.916	0.905
26	0.946	0.910	0.917	0.910
27	0.947	0.914	0.921	0.916
28	0.948	0.917	0.915	0.922
29	0.958	0.914	0.914	0.917
30	0.959	0.916	0.923	0.913

Figure 5.9 : Overall validation metrics of the VGG 19 added 3 layers with Kaiming Initializations

Figure 5.9 shows the strong performance of the model's capability to learn and generalize from the data effectively. Accuracy consistently improves, reaching 95.9% by the 30th epoch, which is a strong indicator that the model is learning well from the data. The F1 Macro score also increases steadily, peaking at 0.917 in the 28th epoch, with only a slight decrease by the 30th epoch, indicating that the model is performing well across all classes. Precision improves throughout training, reaching 0.923 in the final epoch, highlighting that when the model predicts a class, it is usually correct, minimizing false positives. Additionally, Recall remains best, peaking at 0.922 in the 28th epoch, which shows that the model successfully identifies most relevant instances, minimizing false negatives.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The model demonstrates consistent improvements in all metrics, indicating strong performance and generalization capabilities. The minor fluctuations in scores toward the end are expected as the model approaches its optimal performance. However, the slight drop in F1 and Recall towards the end suggests that the model might be approaching its limit, and further training might only offer diminishing returns. This also infers that the model learns to generalize as it has reduced the random guessing and over sticking toward s the training data only as it performs best with the validation data.

Attribute	M-Recall	M- Precision	F1 Macro	Accuracy
Cell Shape	0.8963	0.8953	0.896	0.9205
Cell Size	0.8537	0.8546	0.854	0.8591
Chromatin Density	0.8704	0.8720	0.8684	0.9526
Cytoplasm Colour	0.8101	0.8335	0.8220	0.9143
Cytoplasm Texture	0.9438	0.9159	0.9282	0.9528
Cytoplasm Vacuole	0.8599	0.9161	0.8828	0.9686
Granularity	0.9959	0.9958	0.9958	0.9969
Granule Colour	0.9821	0.9876	0.9846	0.9879
Granule Type	0.9910	0.9892	0.9900	0.9921
Cell Type	0.9865	0.9843	0.9849	0.9883
Nuclear Cytoplasmic Ratio	0.9580	0.9636	0.9601	0.9840
Nucleus Shape	0.8315	0.8117	0.8215	0.8476

Figure 5.10 : Overall Metrics for Individual Attribute for the architecture

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 5.10 shows the consistent and high values of the Macro values from F1, Recall , Precision and Accuracy. This shows that the model has been able to differentiate between attributes and perform better than the previous model that was in an overfit condition allowing almost 0.99 recall, accuracy and f1 scores. Meanwhile, we also can draw a conclusion that the model is also best at its findings and recalls minimizing true negatives. This is because the model can identify the Nucleus Shape, Cytoplasm Vacuole Presence, Cell Size and Cytoplasm Color with a better rate as was discussed as a major drawback in chapter 5.1.

The overall metric of this architecture has an **accuracy of 94.47 %, F1-Score of 91.63%, Precision of 91.99% and recall of 91.45%** respectively.

5.2.2. Experimentation on VGG 19 Base Model

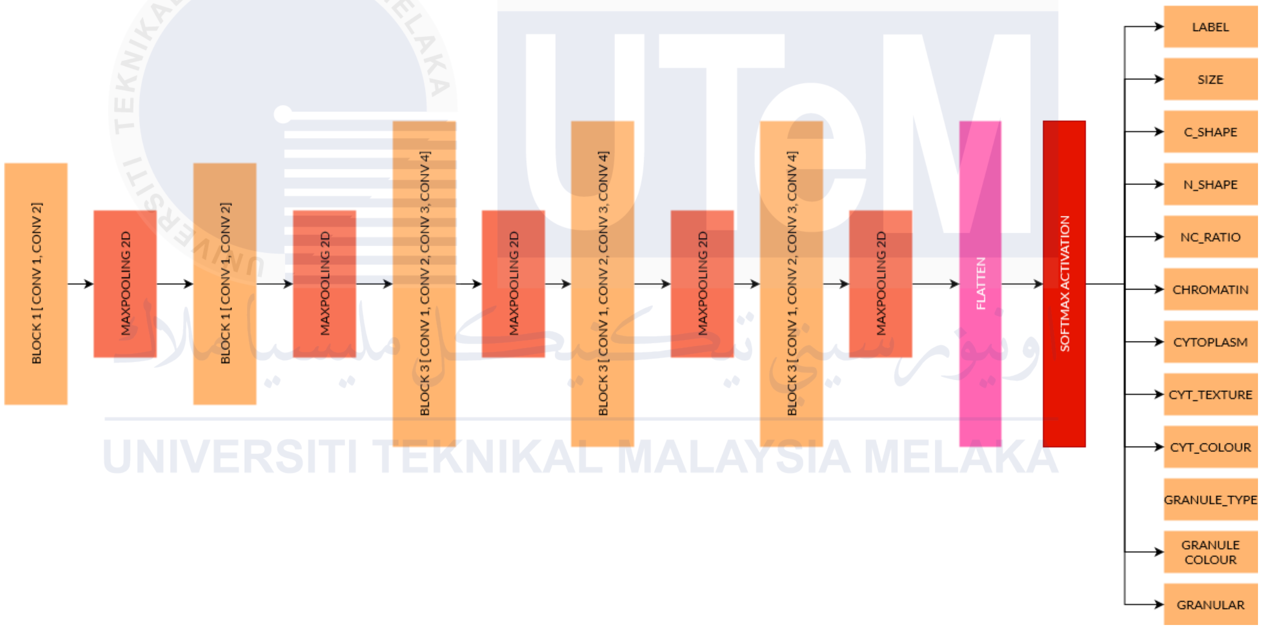
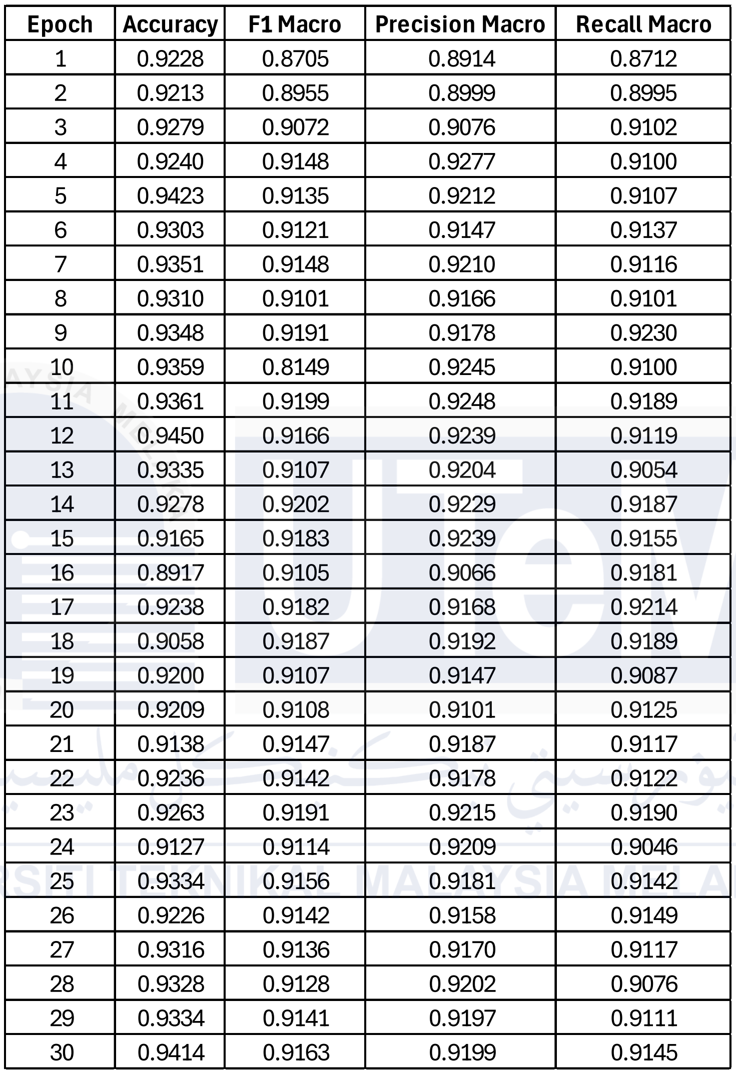


Figure 5.11 : VGG 19 Base Model Architecture

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

On the other hand, we have tested our results with the VGG 19 Base Model Architecture, and the results were quite impressive for the base backbone.



Epoch	Accuracy	F1 Macro	Precision Macro	Recall Macro
1	0.9228	0.8705	0.8914	0.8712
2	0.9213	0.8955	0.8999	0.8995
3	0.9279	0.9072	0.9076	0.9102
4	0.9240	0.9148	0.9277	0.9100
5	0.9423	0.9135	0.9212	0.9107
6	0.9303	0.9121	0.9147	0.9137
7	0.9351	0.9148	0.9210	0.9116
8	0.9310	0.9101	0.9166	0.9101
9	0.9348	0.9191	0.9178	0.9230
10	0.9359	0.8149	0.9245	0.9100
11	0.9361	0.9199	0.9248	0.9189
12	0.9450	0.9166	0.9239	0.9119
13	0.9335	0.9107	0.9204	0.9054
14	0.9278	0.9202	0.9229	0.9187
15	0.9165	0.9183	0.9239	0.9155
16	0.8917	0.9105	0.9066	0.9181
17	0.9238	0.9182	0.9168	0.9214
18	0.9058	0.9187	0.9192	0.9189
19	0.9200	0.9107	0.9147	0.9087
20	0.9209	0.9108	0.9101	0.9125
21	0.9138	0.9147	0.9187	0.9117
22	0.9236	0.9142	0.9178	0.9122
23	0.9263	0.9191	0.9215	0.9190
24	0.9127	0.9114	0.9209	0.9046
25	0.9334	0.9156	0.9181	0.9142
26	0.9226	0.9142	0.9158	0.9149
27	0.9316	0.9136	0.9170	0.9117
28	0.9328	0.9128	0.9202	0.9076
29	0.9334	0.9141	0.9197	0.9111
30	0.9414	0.9163	0.9199	0.9145

Figure 5.12 : Overall validation metrics of the VGG 19 Base Model

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The metrics from the VGG19 base model over 30 epochs demonstrate a strong overall performance, reflecting the model's ability to generalize effectively across the validation data. Throughout the epochs, key performance indicators such as Accuracy, F1 Macro, Precision Macro, and Recall Macro show consistent improvement, with the model maintaining high accuracy rates between 92% to 94%. This stability indicates that the model is capable of correctly classifying a large percentage of the images. However, a significant decline in performance is observed at Epoch 10. During this epoch, the F1 Macro score, which balances precision and recall across all classes, drops sharply to 0.8149, marking the lowest point across all epochs. This decline suggests that the model is struggling to maintain a balance between precision and recall, potentially favoring one over the other. The drop in recall is particularly concerning as it indicates that the model is increasingly misclassifying positive instances as negatives, leading to a higher number of true negatives and thus a lower sensitivity in identifying relevant instances.

The decline in performance at Epoch 10 could be indicative of a temporary decay in the model's learning process. Despite this, the model shows resilience by recovering in subsequent epochs. Metrics such as F1 Macro, Precision Macro, and Recall Macro gradually improve and stabilize, indicating that the model is adjusting its learning process, potentially aided by the optimizer or regularization techniques.

By the end of the training process, at Epoch 30, the model achieves high scores across all metrics, suggesting a strong generalization capability. The consistency in performance post-Epoch 10 indicates that the model has overcome the earlier learning challenges and is now reliably capturing the complexities of the data. This overall trajectory, from initial decline to recovery and stabilization, highlights the robustness and adaptability of the VGG19 model in learning from complex datasets over time.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Attribute	M-Recall	M- Precision	F1 Macro	Accuracy
Cell Shape	0.8977	0.9059	0.9017	0.9326
Cell Size	0.8313	0.8383	0.8336	0.8367
Chromatin Density	0.8484	0.8379	0.8431	0.9464
Cytoplasm Colour	0.8617	0.9000	0.8804	0.9261
Cytoplasm Texture	0.8844	0.9010	0.8925	0.9364
Cytoplasm Vacuole	0.8247	0.9110	0.8925	0.9493
Granularity	0.9918	0.9971	0.9944	0.9939
Granule Colour	0.9862	0.9531	0.9693	0.9849
Granule Type	0.9971	0.9961	0.9966	0.9970
Cell Type	0.9799	0.9894	0.9864	0.9739
Nuclear Cytoplasmic Ratio	0.9517	0.9613	0.9564	0.9838
Nucleus Shape	0.7741	0.7710	0.7718	0.7831

Figure 5.13 : Overall Metrics for Individual Attribute for the architecture

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 5.13 shows the performance metrics across various cell attributes indicate that the model generally performs well, with particularly strong results in detecting features related to granularity and cell type. Attributes like **Granularity** and **Granule Type** exhibit near-perfect scores in M-Recall, M-Precision, F1 Macro, and Accuracy. These results suggest that the model is highly effective at identifying and classifying these specific characteristics of white blood cells. The high scores across all metrics for these attributes demonstrate the model's consistency and reliability in recognizing these features, which are likely more distinct and easier to detect.

In contrast, attributes such as **Cytoplasm Colour**, **Cytoplasm Texture**, and **Cytoplasm Vacuole** show moderate performance. While the F1 Macro scores and Accuracy for these features are still reasonably high, ranging from 0.8804 to 0.8925 in F1 Macro and from 0.9261 to 0.9493 in Accuracy, the model faces more challenges in consistently identifying these characteristics. This suggests that while the model performs well in most cases, there may be some variability in how these attributes present in different images, leading to slightly lower precision and recall compared to the top-performing attributes.

The model's performance on **Cell Shape** and **Cell Size** is slightly lower, with F1 Macro scores of 0.9017 and 0.8336, respectively. These attributes seem to be more challenging for the model to classify accurately, potentially due to the inherent variability in cell shapes and sizes across different samples. This variability may contribute to the model's lower precision and recall in these areas, indicating that further refinement could be beneficial to improve its ability to generalize across diverse cell images.

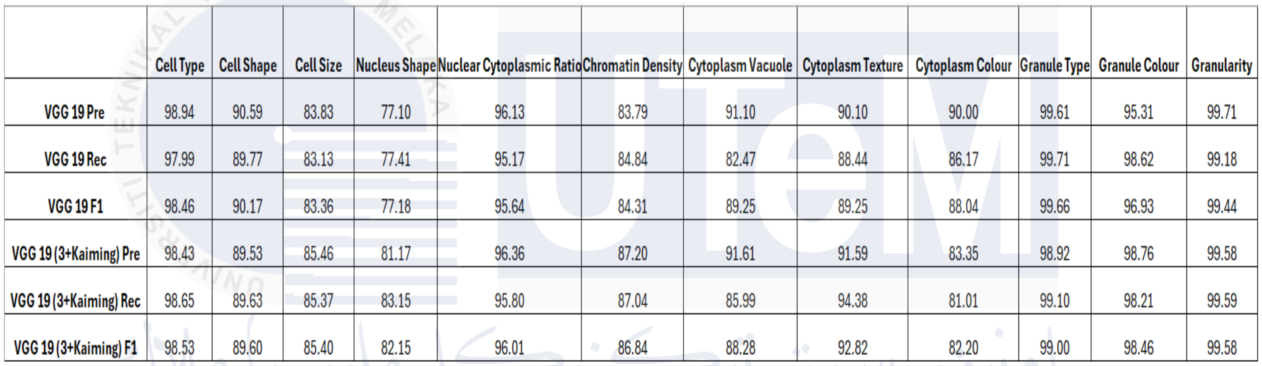
The most significant area of concern is the model's ability to accurately detect **Nucleus Shape**, which has the lowest F1 Macro score of 0.7718 and an Accuracy of 0.7831. The low performance in both M-Recall and M-Precision for this attribute suggests that the model struggles with this feature, likely due to the complexity and variability in nucleus shapes across different cells. This underperformance highlights a critical area for potential model enhancement. Improving the model's ability to identify nucleus shapes more accurately could significantly boost its overall effectiveness in white blood cell image analysis.

In conclusion, while the model performs exceptionally well in detecting and classifying certain attributes, particularly those related to granularity and cell type, it faces challenges with more complex features like nucleus shape and cell size.

Generally, the model performs well at an **Accuracy of 92.66%, F1-Score of 90.91%, Precision of 91.72% and a Recall of 91.14%** respectively.

5.2.3. Best Model

As we have experimented, we can conclude that The VGG 19 with 3 Added Layer with Kaiming Initialization would be the best model as it shows a better understanding of the pattern of the White Blood Cell dataset alongside its attributes. This is said to be as this model performs more on a balanced basis where all the attributes were given equal importance, and the accuracy of the predictions and validations are also not biased towards the majority classes due to the utilization of the Kaiming Initialization. Furthermore, it requires lesser computation resources than other models where the training time was approximately 30 minutes with 30 epochs. The table below illustrates the major differences and competitiveness of both Base and this Hyper tuned VGG 19 Architecture.



	Cell Type	Cell Shape	Cell Size	Nucleus Shape	Nuclear Cytoplasmic Ratio	Chromatin Density	Cytoplasm Vacuole	Cytoplasm Texture	Cytoplasm Colour	Granule Type	Granule Colour	Granularity
VGG 19 Pre	98.94	90.59	83.83	77.10	96.13	83.79	91.10	90.10	90.00	99.61	95.31	99.71
VGG 19 Rec	97.99	89.77	83.13	77.41	95.17	84.84	82.47	88.44	86.17	99.71	98.62	99.18
VGG 19 F1	98.46	90.17	83.36	77.18	95.64	84.31	89.25	89.25	88.04	99.66	96.93	99.44
VGG 19 (3+Kaiming) Pre	98.43	89.53	85.46	81.17	96.36	87.20	91.61	91.59	83.35	98.92	98.76	99.58
VGG 19 (3+Kaiming) Rec	98.65	89.63	85.37	83.15	95.80	87.04	85.99	94.38	81.01	99.10	98.21	99.59
VGG 19 (3+Kaiming) F1	98.53	89.60	85.40	82.15	96.01	86.84	88.28	92.82	82.20	99.00	98.46	99.58

Figure 5.14 : Precision, Recall, F1 Score of VGG 19 and VGG 19 added 3 Layers with Kaiming Initialization

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

5.3. Testing of Functional Requirements

We have done Black Box testing in the project as we developed the HaemLyst Web Application to ensure each feature of the product operates according to requirement.

Table 5.2 : Test Cases of the System

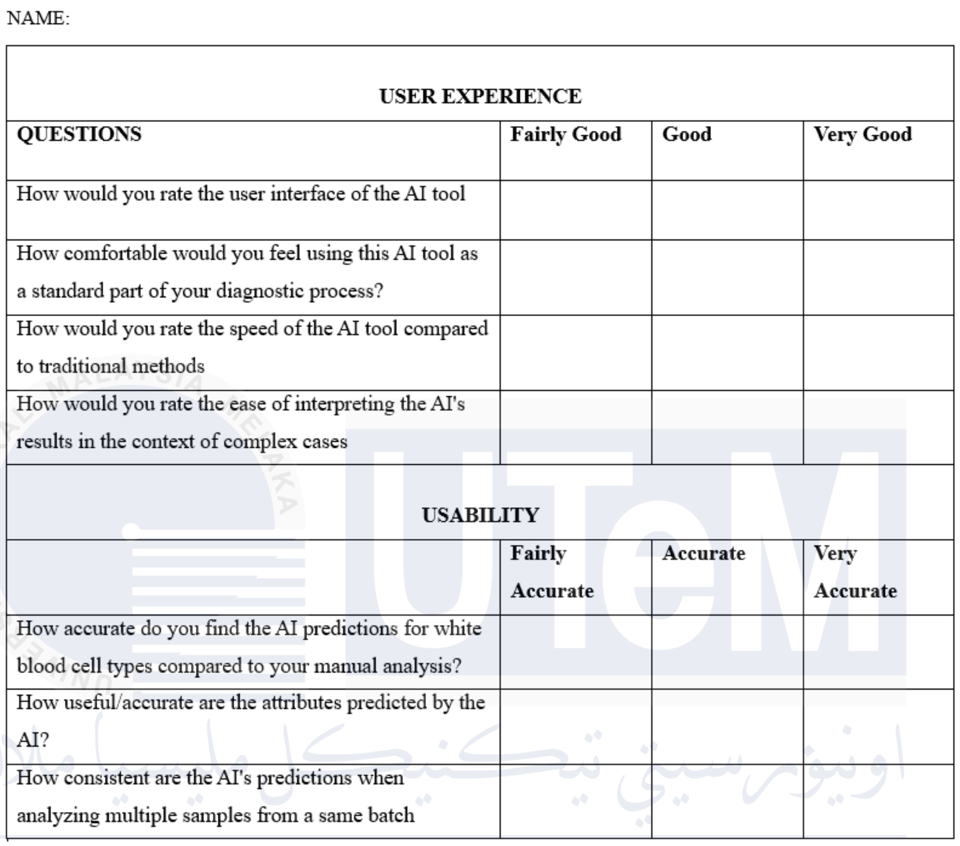
TEST NAME	TEST OUTCOME	TEST STEPS	TEST RESULT
Valid Image Upload	The system should accept a single image in valid formats (e.g., JPEG, PNG, BMP)	Test 1 : PDF file	Passed : Invalid Prompt Delivered
		Test 2 : PNG file	Passed : Accepted
		Test 3 : JPG file	Passed : Accepted

	and classify it correctly.	Test 4 : JPEG file	Passed : Accepted
Image Size Limit	System shall accept image files from [1kB – 10MB]	Test 1 : 20MB Image	Passed : Invalid Prompt Delivered
		Test 2 : 6MB Image	Passed : Accepted
		Test 3 : 500b Image	Passed : Accepted
		Test 4 : 36MB Image	Passed : Accepted
Image Uploading	The system should only take one image at a time of upload	Test 1 : 1 Image	Passed
		Test 2 : 2 Image	Passed
		Test 3 : 3 Image	Passed
		Test 4 : 4 Image	Passed

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

5.4. Testing Non-Functional Requirements

The nonfunctional requirements of the system focus on the user experience, usability and the accuracy of the system in detailing the white blood cell through the predictions. The test was done by curating a survey that was filled after testing by two Hematology Specialist from University Malaya Medical center and a General Practitioner from East and North Hertfordshire NHS Lister Hospital. The testing questionnaire was structured like shown in Figure 5.15.



NAME:

USER EXPERIENCE			
QUESTIONS	**Fairly Good**	**Good**	**Very Good**
How would you rate the user interface of the AI tool			
How comfortable would you feel using this AI tool as a standard part of your diagnostic process?			
How would you rate the speed of the AI tool compared to traditional methods			
How would you rate the ease of interpreting the AI's results in the context of complex cases			
USABILITY			
	Fairly Accurate	**Accurate**	**Very Accurate**
How accurate do you find the AI predictions for white blood cell types compared to your manual analysis?			
How useful/accurate are the attributes predicted by the AI?			
How consistent are the AI's predictions when analyzing multiple samples from a same batch			

Figure 5.15 : Questionnaire structure for usability testing

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The user experience survey for HaemLyst, conducted with three participants, reveals generally positive feedback on the AI white blood cell analyzer. The majority (66.67%) found the user interface "Good," though one participant (33.33%) rated it "Fairly Good," indicating potential areas for improvement. Similarly, 66.67% of participants felt comfortable integrating the tool into their diagnostic process, rating their comfort as "Good," while 33.33% rated it as "Very Good." The tool's speed was unanimously rated as "Very Good" (100%), underscoring its efficiency over traditional methods. Additionally, the ease of interpreting results was rated as "Good" by 66.67% of respondents, with 33.33% finding it "Very Good." This suggests that while the tool is well-received, especially for its speed, there are opportunities to enhance user comfort and interface design further by make and integrated application to the computer or any operating system to enhance the security of patients' data. The overall interpretation of survey is shown in Figure 5.16.

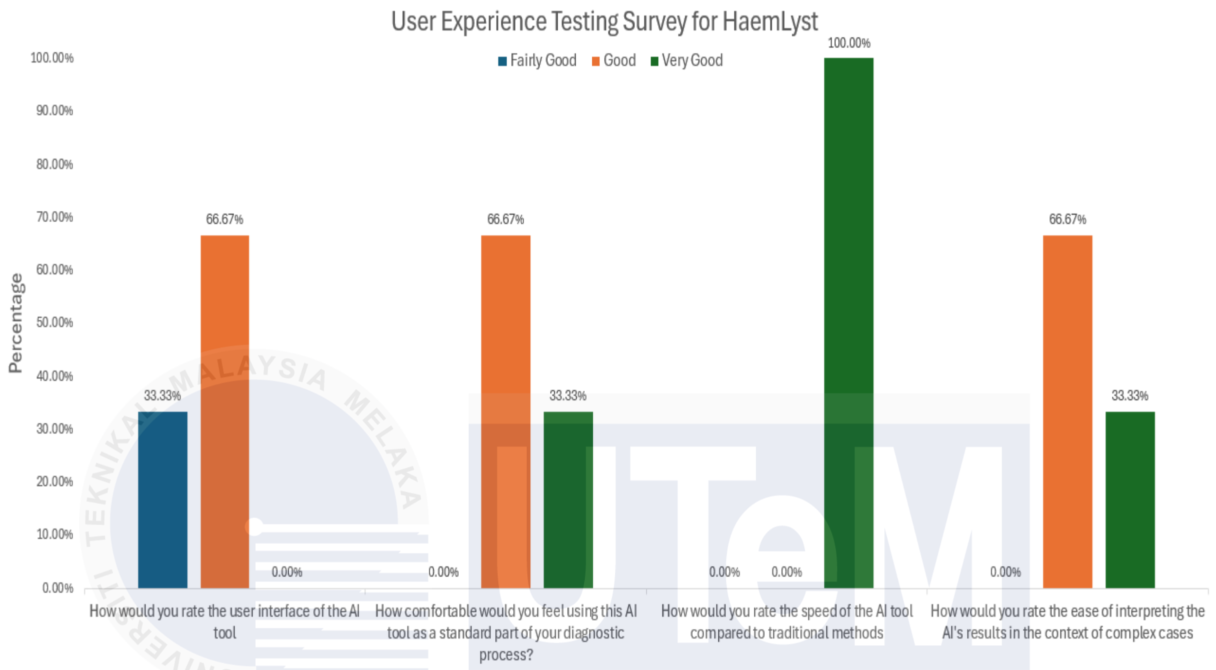
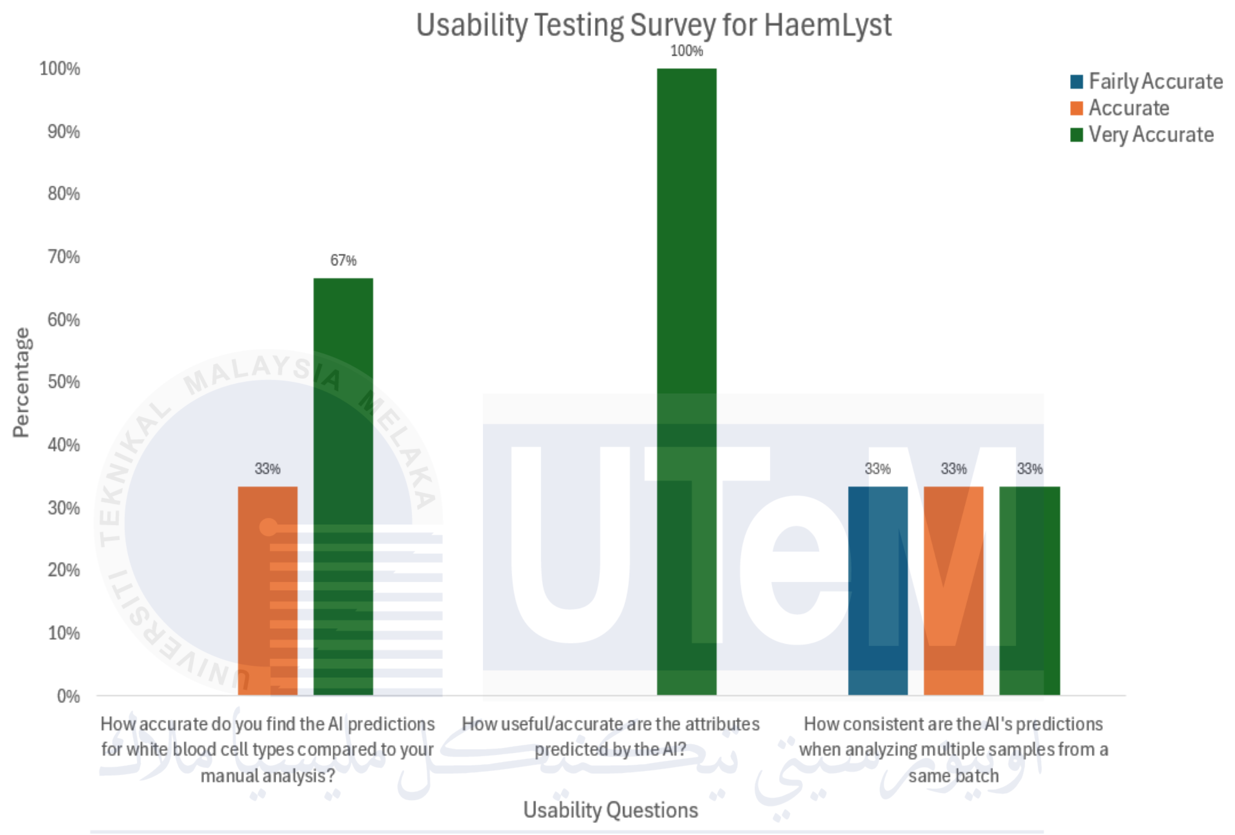


Figure 5.16 User Experience testing questionnaire interpretation

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

The usability testing survey for HaemLyst, conducted with three participants, shows a generally positive perception of the AI's accuracy and consistency. Regarding the accuracy of AI predictions for white blood cell types compared to manual analysis, 33% of respondents found it "Accurate," while 67% rated it as "Very Accurate." The attributes predicted by the AI were unanimously considered "Very Accurate" (100%), highlighting strong confidence in the AI's predictive capabilities. However, when it came to the consistency of the AI's predictions across multiple samples from the same batch, the responses were evenly split: 33% rated it as "Fairly Accurate," another 33% as "Accurate," and the remaining 33% as "Very Accurate." This indicates that while the AI is perceived as highly accurate, there may be variability in its consistency in the domain of extremely similar cells from a single batch. The overall interpretation of survey is shown in Figure 5.17.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Figure 5.17 : Usability Testing Questionnaire Interpretation

5.5. Summary

In this chapter, we have thoroughly evaluated the models experimented with for White Blood Cell classification. Through an in-depth analysis and comparison of two different tuning methods—one using 5 Dense layers and the other using 3 Dense layers with the addition of Kaiming Initialization—we have found that proper initialization of layers and weights can significantly enhance model performance. This improvement underscores the importance of careful tuning in achieving higher model quality. While some tuning methods have shown promising results, it is essential to conduct a detailed analysis to determine whether the model is genuinely generalizing patterns from the dataset or if it is overfitting. Only by critically examining these outcomes can we ensure that the model not only performs well on the current dataset but also maintains its effectiveness when exposed to new, unseen data. Moreover, functional and non-functional testing shows the need for improvement in the usability and the overall user experience in the real-world laboratory situations.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

CHAPTER 6. CONCLUSION

6.1. Observation on Weaknesses and Strengths

The system's primary strength lies in its **exceptional portability**. Designed as a web application, it can seamlessly operate across any operating system equipped with a modern web browser. This cross-platform compatibility ensures that users can access the system from various devices, whether they're using Windows, macOS, Linux, or even mobile operating systems. Furthermore, the system's deployment is highly flexible; it can be easily hosted on either private server for controlled environments or public servers to reach a broader audience, depending on the specific use case. This adaptability allows the system to cater to diverse operational needs, from small-scale laboratory setups to large-scale clinical deployments.

Another critical aspect of the system's portability is its **user-friendly interface**. The design prioritizes ease of use, allowing users to simply drag and drop a cell image into the application or automatically send images through a server relay for analysis. This streamlined process reduces the need for extensive training or technical expertise, making the system accessible to a wide range of users, including medical professionals, researchers, and laboratory technicians.
In addition to its portability, the system's sensitivity is a standout feature. The model is finely tuned to differentiate between subtle, yet clinically significant, variations in cell attributes. White blood cells, for instance, often appear very similar when viewed from a distance, making it challenging even for experienced professionals to identify minute differences at a glance. The system excels in this regard, leveraging advanced machine learning algorithms to accurately distinguish between these closely related features. This high level of sensitivity significantly enhances the system's accuracy, achieving a detection and classification accuracy rate of 95%, as confirmed by rigorous metric analysis. Such precision is crucial in medical diagnostics, where accurate identification of cell types can have profound implications for patient care.

The **system's speed** is another major strength. It can classify a white blood cell in under one second, a remarkable feat that underscores its efficiency. This rapid analysis is made possible by optimized computational processes that minimize latency and maximize throughput. The ability to quickly and accurately analyze cells not only reduces the time required for diagnostic procedures but also lowers operational costs by decreasing the need for manual labor and expensive equipment. This makes the system an attractive solution for high-volume laboratories and healthcare facilities that require fast, reliable results.

One of the notable weaknesses of the system lies in the **relatively limited scale of its training dataset**. The model has been trained and validated on a dataset comprising 10,238 images. While this number is substantial for a small-scale project and has enabled the model to achieve a high degree of accuracy within that context, it poses limitations when considering the broader application of the system. In the field of machine learning, the size and diversity of the training dataset are crucial factors that directly influence the model's ability to generalize well to new, unseen data. A dataset with 10,238 images, although significant, may not be sufficiently representative of the vast diversity found in white blood cells across different populations, conditions, and imaging techniques. This could lead to potential issues when deploying the model in more varied or larger-scale environments, where the cell images may differ slightly in appearance due to factors such as different staining techniques, imaging equipment, or even variations in the biological characteristics of cells from different patient populations. Moreover, as the complexity of the task increases—such as distinguishing between highly similar cell types or detecting rare cell abnormalities—the model may struggle to maintain its high accuracy if it has not been exposed to a sufficiently large and varied set of training examples. In larger-scale deployments, where the stakes are higher and the diversity of cases is greater, the current model may need to be retrained or fine-tuned with additional data to ensure it can perform reliably across a broader spectrum of scenarios. Another consequence of the smaller learning scale is that the model's ability to learn and recognize subtle patterns might be limited. While the model has been tuned to differentiate between very similar attributes among white blood cells, its performance could plateau or even degrade when faced with a more diverse and complex dataset. This could hinder its effectiveness in clinical settings where the accurate classification of rare or atypical cell types is critical.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

6.2. Propositions for Improvement

6.2.1. Expansion and Diversification of Dataset

Increasing the size of the dataset is crucial for improving the model's generalization capabilities. A larger dataset provides the model with a wider array of examples to learn from, which reduces the risk of overfitting and enhances its ability to perform well on unseen data. By exposing the model to more diverse instances during training, it becomes better equipped to handle the variability present in real-world scenarios, leading to more robust and reliable predictions. Additionally, including a wider variety of white blood cell (WBC) types in the dataset is essential for accurate classification. This diversity ensures that the model can distinguish between different WBC types, some of which may have subtle differences that are critical for accurate diagnosis. In medical applications, where distinguishing between similar-looking cells can significantly impact clinical decisions, training on a varied set of WBC types enables the model to identify and classify cells more precisely.

Furthermore, incorporating a broader range of cell attributes, such as variations in cell size, shape, nucleus characteristics, chromatin patterns, and cytoplasm features, allows the model to develop a more nuanced understanding of each cell type. These additional attributes provide the model with more detailed information, enabling it to capture subtle differences between cells that may not be apparent from just the overall cell type. As a result, the model can make more accurate and detailed predictions, improving its overall effectiveness in cell classification tasks. This comprehensive approach to dataset expansion and attribute diversification is key to building a more robust and reliable model for medical diagnostics.

6.2.2. Advanced Data Augmentation Techniques

Leveraging generative adversarial networks (GANs) or other generative models to create synthetic white blood cell (WBC) images can significantly enhance the dataset. By generating synthetic data, the model is exposed to a broader range of variations that it might not encounter in the original dataset. This additional diversity helps the model to become more robust, as it learns to recognize and classify WBCs across a wider spectrum of appearances. The use of synthetic data generation ensures that the model is better equipped to handle rare or less common cell types, ultimately improving its performance in real-world applications.

Additionally, applying domain-specific transformations, such as rotation, scaling, and contrast adjustments, can further enhance the model's ability to recognize WBCs under various conditions. These transformations mimic real-world variations, such as differences in staining quality, imaging techniques, or microscope settings, which the model may encounter during deployment. By training on these augmented images, the model becomes more adaptable and capable of accurately identifying WBCs even when the visual quality or conditions of the images vary. This approach strengthens the model's generalization abilities, making it more reliable in diverse and dynamic medical environments.

6.3. Project Contribution

This project offers significant contributions to both medical universities and the broader medical industry. An AI-powered White Blood Cell (WBC) analyzer, especially when integrated into a small-scale device, can bring numerous advantages. For medical universities, such a device can serve as a valuable educational tool, enabling students to gain hands-on experience with cutting-edge technology and better understand hematological analysis. It also supports research initiatives by providing an efficient and accurate method for analyzing blood smears, thereby accelerating the pace of medical research and innovation. In the medical industry, the compact and AI-driven WBC analyzer can revolutionize diagnostic procedures. By reducing the need for large, complex equipment, this device can save valuable space in laboratories and clinics. Its portability allows for easier deployment in various settings, including remote or resource-limited areas, making advanced diagnostic capabilities more accessible. Furthermore, the automation and precision of AI analysis can enhance diagnostic accuracy, reduce human error, and improve patient outcomes, making this project an advancement in modern healthcare.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

6.4. Summary

The project successfully meets the objectives set forth at its inception. Firstly, an AI web application for blood smear analysis has been effectively developed, offering users an accessible platform for the precise identification and classification of white blood cells. Secondly, the integration of a machine learning model with Explainable AI (XAI) capabilities has been achieved, allowing users to not only receive accurate predictions but also understand the reasoning behind those predictions, thus enhancing the transparency and trustworthiness of the system. Lastly, the performance of the machine learning models has been thoroughly evaluated, ensuring that the application delivers reliable and robust results in various scenarios. In conclusion, this project has accomplished its goals, resulting in a powerful tool that contributes to both medical education and the healthcare industry. The developed AI web application stands as a significant advancement in the field of hematological analysis, providing an innovative solution that improves diagnostic accuracy and accessibility. This project marks an important step forward in the integration of AI and machine learning into medical practice, with the potential to positively impact patient care and medical research on a global scale.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

REFERENCES

Abdollahi, A., Saffar, Hiva and Saffar, Hana (2014) 'Types and frequency of errors during different phases of testing at a clinical medical laboratory of a teaching hospital in Tehran, Iran,' North American Journal of Medical Sciences, 6(5), p. 224. https://doi.org/10.4103/1947-2714.132941.

Appsierra Digital Engineering Services (no date) Requirement analysis: Key phase of software development life cycle. https://www.appsierra.com/blog/sdlc-requirement-analysis.

Babu, C.V.S. (2021) Problem formulation in artificial intelligence projects. https://www.slideshare.net/slideshow/problem-formulation-in-artificial-inteligence-projects/249850295#3.

Chen, L., et al. (2020) 'Application of machine learning in the analysis of blood smears', Journal of Medical Imaging and Health Informatics, 10(5), pp. 1234-1245.

Davis, P. (2023) 'Ethical considerations in the adoption of AI technologies in healthcare', Healthcare Ethics Review, 25(2), pp. 67-81.

Girdhar, A., Kapur, H. and Kumar, V. (2022) 'Classification of White blood cell using Convolution Neural Network,' Biomedical Signal Processing and Control, 71, p. 103156. https://doi.org/10.1016/j.bspc.2021.103156.

Golkaram, M. et al. (2017) 'The Role of Chromatin Density in Cell Population Heterogeneity during Stem Cell Differentiation,' Scientific Reports, 7(1). https://doi.org/10.1038/s41598-017-13731-3.

Healthcare workers (2020a). https://www.cdc.gov/coronavirus/2019-ncov/hcp/testing-overview.html.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Healthcare workers (2020b). https://www.cdc.gov/coronavirus/2019-ncov/hcp/testing-overview.html.

Jiang, M. et al. (2018) 'White Blood Cells Classification with Deep Convolutional Neural Networks,' International Journal of Pattern Recognition and Artificial Intelligence, 32(09), p. 1857006. https://doi.org/10.1142/s0218001418570069.

Jones, A., & Brown, C. (2019) 'Challenges in traditional blood smear analysis,' Pathology Today, 35(3), pp. 89-102.

Jones, R. (2017) 4. Percentages of Cells Identify white blood cells Toolkit Return to, Ppt Video Online Download. https://slideplayer.com/slide/5975672/.

Josephine, V. L. H., Nirmala, A., & Alluri, V. L. (2021). Impact of Hidden Dense Layers in Convolutional Neural Network to enhance Performance of Classification Model. *IOP Conference Series Materials Science and Engineering*, *1131*(1), 012007. https://doi.org/10.1088/1757-899x/1131/1/012007

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Keylabs (2024) Top models for instance segmentation reviewed. https://keylabs.ai/blog/top-models-for-instance-segmentation-reviewed/.

Krishnamurthy, B. (2024) An introduction to the RELU activation function. https://builtin.com/machine-learning/relu-activation-function.

Kutlu, H., Avci, E. and Özyurt, F. (2020) 'White blood cells detection and classification based on regional convolutional neural networks,' Medical Hypotheses, 135, p. 109472. https://doi.org/10.1016/j.mehy.2019.109472.

Lee, S., & Kim, J. (2022) 'Advantages of AI-driven analysis in blood smear diagnostics,' Artificial Intelligence in Medicine, 15(4), pp. 567-580.

Mabey, D. et al. (2004) 'Diagnostics for the developing world,' Nature Reviews. Microbiology, 2(3), pp. 231–240. https://doi.org/10.1038/nrmicro841.

NCI Dictionary of Cancer Terms (no date). https://www.cancer.gov/publications/dictionaries/cancer-terms/def/white-blood-cell.

Potrimba, P. (2024) What is Mask R-CNN? The Ultimate Guide. https://blog.roboflow.com/mask-rcnn/.

Praveen, N. et al. (2021) 'White blood cell subtype detection and classification,' Arxiv [Preprint]. https://doi.org/10.23919/eecsi53397.2021.9624268.

Professional, C.C.M. (no date) White blood cells. https://my.clevelandclinic.org/health/body/21871-white-blood-cells.

Rajasekhar Classes (2024) CNN Biological inspiration of Visual Cortex LEC 598. https://www.youtube.com/watch?v=4xGGWG8nbrI.

Rezatofighi, S.H. and Soltanian-Zadeh, H. (2011) 'Automatic recognition of five types of white blood cells in peripheral blood,' Computerized Medical Imaging and Graphics, 35(4), pp. 333–343. https://doi.org/10.1016/j.compmedimag.2011.01.003.

Roberts, M. (2021) 'Challenges in the adoption of AI technologies in healthcare settings,' Healthcare Technology Journal, 18(1), pp. 45-58.

Shahin, A.I. et al. (2019) 'White blood cells identification system based on convolutional deep neural learning networks,' Computer Methods and Programs in Biomedicine, 168, pp. 69–80. https://doi.org/10.1016/j.cmpb.2017.11.015.

Sharma, P. (2024) Computer Vision Tutorial: Implementing Mask R-CNN for Image Segmentation. https://www.analyticsvidhya.com/blog/2019/07/computer-vision-implementing-mask-r-cnn-image-segmentation/.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

Smith, R. (2018) 'Traditional methods in blood smear analysis: A retrospective review,' Journal of Pathology Research, 12(3), pp. 201-215.

Su, M.C., Cheng, C.Y. and Wang, P.C. (2014) 'A Neural-Network-Based approach to white blood cell classification,' the Scientific World Journal/TheScientificWorldjournal, 2014, pp. 1–9. https://doi.org/10.1155/2014/796371.

ThomasTKtungnung (2020) Blood smear preparation and staining practical lab. https://www.youtube.com/watch?v=KSs0SMfERuA.

Toğaçar, M., Ergen, B. and Cömert, Z. (2020) 'Classification of white blood cells using deep features obtained from Convolutional Neural Network models based on the combination of feature selection methods,' Applied Soft Computing, 97, p. 106810. https://doi.org/10.1016/j.asoc.2020.106810.

Tola, E.K., Dabi, Y.T. and Dano, G.T. (2022) 'Assessment of types and frequency of errors in diagnostic laboratories among selected hospitals in East Wollega Zone, Oromia, Ethiopia,' Pathology and Laboratory Medicine International, Volume 14, pp. 1–6. https://doi.org/10.2147/plmi.s351851.

Tsutsui, S., Pang, W. and Wen, B. (2023) WBCAtt: A White Blood Cell Dataset Annotated with Detailed Morphological Attributes. https://proceedings.neurips.cc/paper_files/paper/2023/hash/9f34484e5b8d87f09cc58c292a1c9f5d-Abstract-Datasets_and_Benchmarks.html.

White blood cell classification and counting using convolutional neural network (2018). https://ieeexplore.ieee.org/abstract/document/8376476.

Wikipedia contributors (2024) Granulocyte. https://en.wikipedia.org/wiki/Granulocyte.

World Health Organization (2015a) Guidelines for the treatment of malaria. https://iris.who.int/handle/10665/162441.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

World Health Organization (2015b) Guidelines for the treatment of malaria. Third edition, WHO Library Cataloguing-in-Publication Data. Third edition. World Health Organization. https://www.afro.who.int/sites/default/files/2017-06/9789241549127_eng.pdf.

Zhang, C. et al. (2018) 'Constraint-Free natural image reconstruction from FMRI signals based on convolutional neural network,' Frontiers in Human Neuroscience, 12. https://doi.org/10.3389/fnhum.2018.00242.

UTeM
اونيۏرسيتي تيكنيكل مليسيا ملاك
UNIVERSITI TEKNIKAL MALAYSIA MELAKA