

A NOVEL AUTOMATIC EARLY DETECTION OF LEUKAMIA CANCER IN MICROSCOPIC IMAGES

A Project Report Phase-II

Submitted in partial fulfillment of the requirement for the award of
Bachelor of Engineering Degree in Electronics and Communication Engineering.

by

PAMULA JAYA RAJU (REG NO:39130338)

PALLA NAGA SURESH (REG NO:39130335)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119**

APRIL-2023



SATHYABAMA

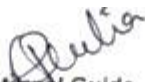
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC
Jeppiar Nagar, Rajiv Gandhi Salai, Chennai – 600 119
www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project Report is the Bonafide work of
PAMULA JAYA RAJU (39130338) and **PALLA NAGA SURESH(39130335)** who
carried out of the project entitled “ **A NOVEL AUTOMATIC EARLY DETECTION
LEUKAMIA CANCER IN MICROSCOPIC IMAGES**” under our super vision from
November 2022 to April 2023.


Internal Guide

Dr. P. CHITRA M.E., Ph.D.

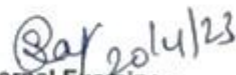

Head of the Department

Dr. T. RAVI, M.E., Ph.D.

Submitted for Viva voce Examination held on 20/4/23

Internal Examiner


20/4/23


External Examiner

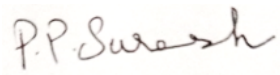
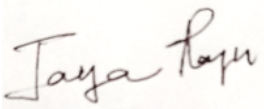
DECLARATION

We **PAMULA JAYA RAJU** (39130338) and **PALLA NAGA SURESH** (39130335) here by declare that the project Report entitled “**A NOVEL AUTOMATIC EARLY DETECTION LEUKAMIA CANCER IN MICROSCOPIC IMAGES**” done by us under the guidance of **Dr.P.CHITRA M.E.,Ph.D.**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronic and Communication Engineering.

DATE:19-04-2023

PLACE: Chennai

SIGNATURE OF THE CANDIDATE(S)

1. 
2. 

ACKNOWLEDGEMENT

We pleased to acknowledge our sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. NANDHITHA M.E., Ph.D.**, Dean, School of electronics **Dr. T. RAVI, M.E, Ph.D., Professor & Head of the Department of Electronics and Communications Engineering** for providing necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to my **Project Guide Dr. P. CHITRA, M.E, Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

Leukemia is a type of cancer that starts in the bone marrow and leads to the growth of abnormal white blood cells. Leukemia subtypes need to be found early and correctly diagnosed so that patients can get treatment quickly and have better outcomes. In this study, we propose a deep-learning method for finding leukemia cancer cells in microscopic blood images. This method uses MobileNetSSDv2, which is the most advanced model for finding objects. We used a large set of microscopic blood images to train the MobileNetSSDv2 model to find and sort leukemia cells into different subtypes. We looked at how well our model worked by using mean Average Precision (mAP), a widely used evaluation metric for object detection models. Our results show that the proposed model gets a mAP score of 97% which means it can find and classify leukemia cells with a high degree of accuracy. Our study shows that deep learning models, especially MobileNetSSDv2, can be used to find and sort leukemia cells in microscopic images of blood. Our method could help find leukemia earlier and more accurately, which could improve patient outcomes and possibly save lives. The proposed model could also be used in other types of medical imaging to find cancer cells and other kinds of abnormal cells.

TABLE OF CONTENTS

Chapter No	Title	Page No
	ABSTRACT	iv
	LIST OF FIGURES	v
	LIST OF TABLES	ix
1	INTRODUCTION	1
	1.1 Aim of the Project	
	1.2 Project Statement	
	1.3 Problem definition and scenarios	2
	1.4 Organization of the Report	
	1.5 Summary	3
2	LITERATURE REVIEW	4
	2.1 Literature Review	
	2.2 Literature Survey	
3	PROJECT DESCRIPTION	11
	3.1 The purpose of the project work	
	3.2 The current system	
	3.3 Drawback of the Existing system	
	3.4 proposed system	
	3.5 Advantage of the proposed system	12
	3.6 Ability Analysis	
	3.7 Risk analysis of the project	13
4	SYSTEM DESIGN	15
	4.1 Mobile Net V2 Architecture	
	4.2 Description of software for implementation	16
	4.3 Testing	17
	4.4 Software to operations	18
	4.5 Alex network	19
5	PROJECT REQUIREMENTS	23
	5.1 Hardware requirements	
	5.2 software requirements	

	5.3 python libraries used	
6	MODULE DESCRIPTION	24
	6.1 Visual data acquisition and labeling modules	
	6.2 Data preparation and augmentation module	
	6.3 Model training	25
	6.4 Training module	26
	6.5 Model evaluation module	27
7	IMPLEMENTATION	28
	7.1 Introduction	
	7.2 Image data collection and annotation	
	7.3 Image transformation module	29
	7.4 Training and evaluation	30
	7.4.1 Training	32
	7.4.2 Evalution	35
	7.5 Techniques used for optimizing model performance	
	7.6 summary	
8	RESULT AND DISCUSSIONS	38
	8.1 Experimental results and discussion	
9	CONCLUSION AND FUTURE WORK	39
	9.1 conclusion	
	9.2 future plans	
	REFERENCES	40

LIST OF FIGURES

Figure No	Figure Title	Page No
4.1	System Architecture	15
4.2	Alex Net Architecture	20
4.3	Training Steps Vs Validation Training Accuracy	21
6.1	Flowchart of Module Description	23
8.1a	Simple image from the dataset	36
8.1b	Simple image from the dataset	36
8.2	Sample output image from the dataset	36
8.3	Confusion Matrix	36
8.4a	Trained image	37
8.4b	Trained image	37
8.5a	Classification Loss	37
8.5b	Localization Loss	37
8.5c	Regulation Loss	37
8.6a	Trained input Image	38
8.6b	Trained input Image	38
8.6c	Trained input Image	38

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
SSD	Single Shot Detector
FPN	Feature Pyramid Network
GPU	Graphics Processing Unit
ROI	Region of Interest
IoU	Intersection over Union
NMS	Non-Maximum Suppression
TF Record	TensorFlow Record
COCO	Common Objects in Context
mAP	Mean Average Precision
AUV	Autonomous Underwater Vehicle
YOLO	You Only Look Once
ResNet	Residual Network

LIST OF TABLES

TABLE NO	TABLE NAME	PAGE NO
8.1	Microscopic images Leukemia Result	35

CHAPTER - 1

INTRODUCTION

1.1 Aim of the Project

In this project, using leukemia cancer microscopic images a deep learning based algorithm is developed to detect the cancer early. In this algorithm Mobile net V2 network is used to to classify the trained leukemia cancer microscopic images. In this developed algorithm found better accuracy and efficiency in detecting microscopic leukemia cancer images.

1.2 Problem Statement

The motivation for this project is to address the limitations of traditional methods for identifying leukemia cells in blood images and provide a more efficient and accurate approach using deep learning techniques. The proposed method aims to make leukemia diagnoses more accurate and faster, which could improve patient outcomes and even save lives.

The current way to find leukemia cells is through manual analysis, which takes time and can lead to mistakes. This can make it take longer to figure out what's wrong and start treatment, which can hurt the patient. Also, traditional methods might not be able to find leukemia cells early, when they are easiest to treat.

Deep learning methods, like the MobileNetSSDv2 model proposed in this project, have shown promise in the field of medical imaging as a way to find abnormal cells and tumors with high accuracy. This project aims to improve the speed and accuracy of leukemia diagnoses by using the power of deep learning algorithms. This will lead to better patient outcomes in the long run.

In short, the goal of this project is to fix the problems with traditional ways of finding leukemia cells and come up with a faster and more accurate way to do it using deep learning techniques. The proposed method could make it easier and faster to diagnose leukemia, which could help patients and possibly save their lives.

1.3 Problem Definition and Scenarios

The current ways to find and classify leukemia cells in microscopic images of blood are time-consuming and prone to mistakes because they require trained experts to look at the images by hand. Because of this, there is a need for a faster and more accurate method that can automate this process and make diagnosis more accurate. Deep learning has shown great promise in a number of medical imaging applications because it can learn complex patterns and features from data.

In this report, the propose a deep learning method that uses MobileNetSSDv2, a state-of-the-art model for detecting objects, to find and classify leukemia cells in microscopic blood images. The proposed method aims to make the process automatic and get accurate and quick results for diagnosing leukemia. By using deep learning, The hope to make leukemia diagnoses more accurate and faster, which can lead to better outcomes for patients and higher survival rates.

1.4 Organization of the Report

This is divided into a few different sections that each focus on a different aspect of the project. The first part of this paper is an overview of the issue, as well as an explanation of why this project was undertaken. In the second section, The talk about the related work that's been done in the field and highlight the limitations of the detection methods that are currently available. which combines Mobile Net and FPN, is described in greater detail in the third section of this article. The dataset that was used for training and testing the model, as well as the evaluation metrics that were used to assess the performance of the method, are both described in the fourth section of the article. In the fifth section, the results of the experiments are presented, and an analysis of how well the proposed method works is conducted. The implications of the findings for future research and applications are discussed in the report's final section, which serves as the report's conclusion.

1.5 Summary

This section gives an overview of a report that talks about using deep learning to find leukemia cancer cells in micrographs of blood. The project is to fix the problems with traditional methods and come up with a better, more accurate way to do things by using deep learning techniques. This could help patients and save lives. The report talks about how the problem is defined and gives possible solutions. It also suggests a deep learning method that uses MobileNetSSDv2 to find and classify leukemia cells in microscopic images of blood. The report is divided into sections that focus on different parts of the project, such as related work, the proposed method, the dataset used for training and testing, the evaluation metrics, the results of experiments, and what the results mean for future research and applications.

CHAPTER 2

LITERATURE REVIEW

2.1 Literature Review

The importance of health informatics in the diagnosis of various diseases, including leukemia, was highlighted in a recent review by Gurpreet Singh et al. [1]. They looked into various methods used by previous researchers for spotting and stopping leukemia, such as Artificial Neural Networks (ANN), image processing, Linear Dependent Analysis (LDA), Self Organizing Maps (SOM), Support Vector Machines (SVM), and Genetic Algorithm (GA). The authors concluded that the use of these techniques had greatly increased the precision and speed with which leukemia could be diagnosed, thereby expanding patients' access to effective treatments.

An ensemble deep learning model for identifying and predicting Leukemia was developed by Krishna Kumar Jha et al. [2]. They used several preprocessing steps to prepare the images for analysis, including converting to and from HSV and Lab colour spaces and employing k-means clustering for segmentation. The segmented images were used to extract features, which were then classified using an ensemble classifier before being fed into a neural network for identification and prediction. The effectiveness of the proposed method in detecting Leukemia was demonstrated by its 100% accuracy with high-quality datasets and 96.3% accuracy with noisy datasets. The model was also very efficient, with a small mean square error and a short elapsed time. Their findings demonstrate the promise of ensemble deep-learning models to advance diagnostics and may pave the way for the development of more sophisticated automated systems for cell counting and Leukemia detection.

Using machine learning techniques to analyse microscopic images of blood cells, Saranya N. et al. [3] investigated the possibility of detecting blood cancer. Using image processing and classification techniques like SVM and KNN, the authors aimed to create an accurate method for early detection of leukemia. Segmentation was then used to isolate cells and identify overlapping cell regions, as part of their proposed method, which first involved preprocessing input images to remove noise and improve contrast. Separation of cells,

identification of abnormal cells or blast cells, and feature extraction for classification were all accomplished by means of watershed segmentation and K-means clustering. Using the SVM classifier, their method achieved 97% accuracy, while the KNN classifier achieved 91% accuracy, proving the efficacy of machine learning in the detection of blood cancer.

A MobileNet-based classification model for paddy leaf disease detection was presented by Fauzan Masykur et al. [4]. The purpose of this research was to identify rice plant diseases by leaf colour. The dataset was compiled from public data on kaggle.com and included pictures of healthy leaves, brown spots, hispa, and leaf blast. Each image in the dataset was resized to 416 x 416 pixels before being divided into training and validation sets. Since the MobileNet architecture required less time to calculate and fewer materials, it was chosen. The study compared epochs with values of 50 and 100 to gauge their efficacy. At epoch 50, the model performed at 99% accuracy with a 3% loss, and at epoch 100, it performed at 97% accuracy with a 22% loss. For epoch 50, one training cycle took 162s-18s/step and for epoch 100, 195s-19s/step was needed. After evaluating the accuracy and loss values of the MobileNet model, the authors concluded that epoch 50 provided superior performance.

Computer-aided diagnosis of white blood cell leukemia using VGG16 Convolution Neural Networks was presented by Dr. G. Meena Devi et al. [5]. Leukemia is a malignant blood disorder caused by an excessive production of white blood cells in the bone marrow. AlexNet, VGG16, GoogLeNet, and ResNet50 were among the Convolution Neural Networks tested, along with their abilities to classify different cell types using full learning and transfer learning. When compared to other models for spotting leukemia, VGG16 performed best, with an average accuracy of 97.16%. Compared to ensemble transfer learning, the proposed method's use of publicly available datasets allowed for faster testing and more precise classification. The importance of detecting cancer cells at an early stage was also emphasised by the authors. Because of its high performance and computational efficiency, the system is described in the paper as a candidate for an automatic diagnosis system with precise prognosis.

S. Rajeswari et al.[6] present a computer-aided diagnosis system that uses image processing techniques and transfer 2 learning to detect and classify different types of leukemia in microscopic blood smear images. The system employs Inception V3 and Xception convolutional models in a hybrid model constructed using ensemble averaging, achieving better accuracy. The system pre-processes images by converting them into grayscale format, applying filters, and performing edge detection segmentation. The proposed system is evaluated on the ALL IDB dataset from Kaggle and the ASH Imagebank, with data augmentation used to increase the size of the dataset. Overall, the proposed system achieves accurate results in the early detection and classification of leukemia, highlighting the potential of image processing and transfer learning techniques in improving diagnostic research and paving the way for further advancements in automated systems for cell counting and leukemia detection.

Focusing on acute lymphoblastic leukemia, M. Saritha et al.[7] provide an overview of the various methods used to detect cancer cells in microscopic images of human blood samples (ALL). The authors stress the significance of early diagnosis of leukemia despite the fact that this can be difficult due to the lack of specificity of symptoms and signs. Using geometric, statistical, textural, and colour features, the proposed system performs preprocessing, image segmentation, feature extraction, and image classification. The importance of an automated image analysis system for measuring morphological parameters and assessing cancer cells is stressed by the authors.

Iswarya et al. [8] proposed a machine learning approach for detecting leukemia in its earliest stages from images of peripheral blood smears (PBS). Before using a convolutional neural network for classification, the collected images were preprocessed, segmented, and feature extracted (CNN). Using only 20% of the data for testing and 80% for training, the proposed system was able to achieve an accuracy of over 90%. The authors relied on a dataset of 1,000 PBS images made available through Kaggle. Resizing, filtering, noise reduction, and edge smoothing were all a part of the preprocessing phase. For white blood cell (WBC) detection and lymphocyte isolation, the Zack algorithm

was used for segmentation. The local binary pattern (LBP) algorithm was used to extract features of the cytoplasm and the size of the nucleus. Classification results using the proposed CNN-based classifier were superior to those using SVMs, ANNs, and k-nearest neighbours, three common classical machine learning algorithms (KNN). The findings of the authors demonstrate the promise of machine learning algorithms for enhancing leukemia diagnosis, especially at the earliest stages, and may pave the way for future developments in automated systems for cell counting and leukemia detection.

As stated in their study, "Qurat ul ain et. Using the ALLIDB dataset, et al. [9] present a deep learning-based approach to detecting and classifying leukemia. Different methods, such as histogram equalisation, blurring, translation, rotation, zooming, shearing, and reflection, were used to supplement the original dataset. AlexNet's input images were scaled to 227x277, while ResNet-34's were scaled to 224x244. 70% of the images were used for training and 30% were used for testing and validation. The authors compared the performance of AlexNet and ResNet-34 and found that ResNet-34 achieved better results, with an accuracy of 98.4%, precision of 98.34%, and recall of 99%. The dataset was divided into five classes: CLL, CML, ALL, AML, and normal. ResNet-34 achieved a 98.4% accuracies for both healthy and ALL cases, an F1 score of 99%, a recall of 99%, and a precision of 98.34%. ResNet-34 achieved a 98.4% accuracy in predicting AML, a 98.34% precision, a 98.4% recall, and an F1 score of 99.1%. ResNet-34 had an F1 score of 99.1%, recall of 98%, precision of 98.3%, and accuracy of 98.13% for CLL. ResNet-34's F1 score for CML was 99.24%, and its accuracy, recall, precision, and accuracy all topped 98%. AlexNet, in contrast, performed at a 96.1% accuracy rate, a 97.9% F1 score, 95.3% precision, and 95.5% recall for ALL and healthy. AlexNet's AML metrics were as follows: 95.39% recall, 95.50% accuracy, 96.70% F1 score, and 95.63% precision. AlexNet was able to predict CLL with a 95.7% accuracy, a 95.9% recall, a 97.1% F1 score, and a 95.5% precision. The paper highlights the potential of using deep learning-based algorithms for the prediction and classification of leukemia using microscopic images, and it demonstrates the efficiency of DL-based algorithms in achieving robust performance above 90%

using AlexNet's accuracy of 96.8%, F1 score of 97.1%, precision of 96%, and recall of 96.3%. The authors' use of publicly available datasets, the augmentation technique to increase the number of images, and the combination of segmentation and classification techniques all contribute to the significance of their work and the advancement of the field of leukemia detection and classification.

Convolutional neural networks (CNNs) are investigated by sheremet et al.[10] for the purpose of digital medical (Xray) image-based diagnosis of lung disease. They investigate and study the processes involved in designing CNN architectures, and then create their own model based on the ResNet-50 framework. The 3303 images are used to train the model, which has a 98% success rate. The COVID-19-positive, healthy lung, and viral pneumonia X-ray images are used to evaluate the developed software application. In the absence of major image deformations or minor distortions, the trained model makes a disease diagnosis with 100% accuracy. The developed system is intended to quickly diagnose the presence of a disease, the authors note, particularly in isolating, difficult, and complex disease situations. As a result of their efforts, we may soon have a more rapid and accurate method of diagnosing lung diseases using artificial neural networks in clinical practise.

In order to distinguish between healthy white blood cells and malignant leukemia cells in blood smear images, Duong Trong Luong et al.[11] present a method based on the YOLOv5 algorithm. Part 1 of the paper provides an introduction to the issue and to the various approaches that have been taken to it, while Part 2 describes the methodology, the dataset, 3 the YOLOv5 architecture, and the results obtained. Acute myelomonocytic leukemia (AML) patient blood smear images totaling 1438 were used for the dataset's training and validation phases, respectively. The validation set's size remained unchanged at 283, while the training set grew to 3,465. The YOLOv5 model correctly identified 93% of test samples as either normal WBCs or leukemia cells, with neutrophils being the most accurate at 99.3% and monocytes the least at 74.6%. The authors show that their proposed method achieves higher accuracy than some but lower than others by comparing their results to those of

other methods. In general, their approach has potential to improve the accuracy of blood disease diagnosis.

In order to identify leukemia in blood micrographs, Sada Riad et al. [12] present the CKC and ALLS algorithms. Both CKC and ALLS use visual features of the three subtypes of acute lymphoblastic leukemia as determined by a human pathologist, but CKC is based on the K-Nearest Neighbors classifier. The algorithms use various methods from the field of image processing, including pre-processing, feature extraction, and morphological operations. Precision, recall, accuracy, Fmeasure, and speed were evaluated in a series of experiments that compared CKC and ALLS. The outcomes show that ALLS performs better than CKC in terms of recall, accuracy, and speed, while precision and F-measure are nearly tied. In addition, the paper uses the data mining platform Weka to assess the efficacy of three classifiers (RandomForest, KStar, and REPTree) in the context of ALLS. The findings show that all three classifiers are successful, with respectable values of precision, recall, and F-measure. The paper concludes by presenting two efficient algorithms for detecting leukemia in microscopic blood images: ALLS outperforms CKC in terms of recall, accuracy, and speed. Leukemia can be visually distinguished between the three subtypes of acute lymphoblastic leukemia, and this paper highlights the usefulness of using data mining techniques for this purpose.

Ines Chouat et al.[13] propose two convolutional neural network (CNN) models, VGGNet-16 and MobileNet, to use transfer learning to identify lung disease in chest X-ray images. CXR images from four different categories (normal, bacterial pneumonia, viral pneumonia, and COVID-19) are used to train the models. To boost the model's ability to generalise, the data augmentation algorithm is applied to larger subsets of the training dataset. When compared to VGGNet-16, the MobileNet model has better accuracy, recall, and F1 score. The models are evaluated in terms of their precision, recall, F1 score, and accuracy in this paper. The results show that when it comes to identifying lung disease in chest X-ray images, the MobileNet model outperforms VGGNet-16 with an accuracy rate of 82%. Leukemia and its subtypes can be detected using a variety of machine learning techniques, all of which are summarised in detail by

Das and Padmanaban[14]. The authors compare and contrast several methods for identifying leukemia in blood micrographs, including SVM, CNN, and CNN with SVM. CNN with SVM performs well in detecting leukemia, while FNESVM performs poorly, according to performance evaluation across multiple datasets. DCNN is the most reliable method for detecting leukemia, according to the reviewed papers.

2.2 Literature Survey

In the section titled "literature survey," it provide a comprehensive overview of the previous research that has been conducted on the subject of object detection in medical images, specifically for the detection of abnormal cells. The purpose of this section is to draw attention to the significance of the proposed method by pointing out the shortcomings and restrictions of the existing methods. The literature review examines a variety of sources, such as academic articles, research papers, and technical reports. The section is broken up into several subsections that focus on different aspects of the subject matter, such as the difficulties associated with detecting abnormal cells in medical images, the various detection methods that are currently in use, and the evaluation metrics that are utilized to evaluate the effectiveness of the various detection methods. This section ends with a brief overview of the most important findings as well as a discussion of the implications these findings have for the proposed approach.

The authors discuss the importance of health informatics in the diagnosis of various diseases, specifically leukemia, by using different techniques such as Artificial Neural Networks (ANN), image processing, Linear Dependent Analysis (LDA), and Self Organizing Maps (SOM). They review several methods utilized by prior authors to detect and prevent leukemia, emphasizing on the use of ANN, LDA, SOM, and Support Vector Machines (SVM). The paper also highlights the role of Genetic Algorithm (GA) in the detection of leukemia. The authors conclude that these methods have significantly contributed to the timely and accurate diagnosis of leukemia, enabling better treatment for patients.

CHAPTER 3

PROJECT DESCRIPTION

3.1 The Purpose of the Project Work

Leukemia is a serious disease that affects millions of people worldwide, and early detection and accurate diagnosis are critical for effective treatment and better patient outcomes. However, traditional methods for identifying leukemia cells in blood images are time-consuming, manual, and subject to human error. Therefore, the main objective of this project is to address the need for a more efficient and accurate method for identifying and classifying leukemia cells in microscopic blood images.

3.2 The Current System

The current system relies heavily on the expertise of the medical professionals, which can lead to inconsistencies and variations in the results. The process can be laborious, as the medical professionals have to manually search through a large number of blood images to identify the leukemia cells. Furthermore, the manual approach is time-consuming, which can delay the diagnosis and treatment of the disease, potentially leading to a poorer prognosis for the patient.

3.3 Drawbacks of the Existing System

The current system's flaws emphasize the importance of developing an automated and accurate method for detecting and categorizing leukemia cells within blood images. The proposed system uses deep learning, and more specifically MobileNetSSDv2, to automate the process and boost accuracy in an effort to address these shortcomings. The proposed system employs a deep learning-based strategy to speed up the diagnosis of leukemia while also increasing its precision.

3.4 Proposed System

When compared to the current system, the proposed one that makes use of Mobile Net SSD FPN (Feature Pyramid Network) has many advantages. First, it speeds up the diagnosis process by eliminating the need for human intervention in

the process of identifying and classifying leukemia cells in microscopic blood images. Second, the precision of leukemia cell identification and classification is improved by employing cutting-edge object detection models like Mobile Net SSD FPN. This increased precision is vital for efficient treatment and positive patient outcomes.

As a result of reducing the need for manual review of images by trained professionals, the proposed system has the potential to save significant amounts of time and resources. By facilitating earlier detection and more accurate classification of leukemia cells, the proposed system represents a significant advancement in the field of leukemia diagnosis and has the potential to improve patient outcomes.

3.5 Advantages of the Proposed System

In addition, the proposed system is highly scalable, so it can be used to efficiently analyze a large number of blood images. This has the potential to improve healthcare system efficiency by reducing doctors' workloads. Other areas of medical imaging could benefit from the use of deep learning and object detection models in the future, allowing for quicker and more precise diagnoses of a wide range of diseases. The proposed system offers a number of advantages, and it has the potential to greatly enhance the process of identifying and classifying leukemia cells in blood images, which will ultimately lead to earlier detection and treatment and better patient outcomes.

3.6 Ability Analysis

To make sure a project is feasible and doable, it is crucial to conduct a feasibility study before starting it. The technical, financial, and operational implications of creating a deep learning-based system for detecting and categorizing leukemia cells in blood images were examined as part of the feasibility study for this project.

The ability of the proposed system to meet the technical requirements for accurately identifying and classifying leukemia cells, as well as the availability of resources like hardware and software, were considered when determining

technical feasibility. The study concluded that the necessary resources and technology were easily accessible, and that the proposed system was technically feasible.

The cost of creating and implementing the proposed system, as well as the potential return on investment, were considered when evaluating its economic viability. The study concluded that the cost of developing the system was reasonable and that the potential benefits, such as earlier detection and treatment of leukemia, outweighed the costs.

The proposed system's effect on the current workflow and processes for identifying and classifying leukemia cells was considered when determining its operational viability. According to the research, the proposed system could be easily incorporated into the current workflow and would significantly increase productivity and precision.

3.7 Risk Analysis of the Project

Concerns about data privacy, the need for a large and diverse dataset for training the model, and the likelihood of errors or biases in the model were all identified through risk analysis as potential risks and challenges. However, precautions were taken to reduce these risks, including the protection of personal information, the acquisition of a diverse dataset, and the implementation of strict testing and validation procedures.

Overall, the feasibility study and risk analysis show that the proposed system for detecting and categorizing leukemia cells in blood images is feasible, doable, and has the potential to significantly advance the process of diagnosing and treating leukemia.

Use Case: Identifying and Classifying Leukemia Cells in Microscopic Blood Images Actor: Medical Professional Description: The system is used to identify and classify leukemia cells in microscopic blood images.

Preconditions: The system has been installed and is running properly.

The user has access to a set of microscopic blood images that need to be analyzed. Basic Flow: The user launches the system and navigates to the image analysis module. The user selects the set of images to be analyzed and uploads them to the system. The system processes the images and identifies the leukemia cells using MobileNetSSDv2. The system classifies the identified leukemia cells into different subtypes. The results are displayed to the user, including the number of cells identified and their respective subtypes. The user can review and export the results for further analysis or use in patient treatment. Alternative Flows: If the system encounters an error while processing the images, an error message is displayed to the user.

If the user needs to adjust the settings of the analysis module, they can do so before uploading the images. If the user wants to compare the results of multiple sets of images, they can do so by selecting and uploading each set separately. Post conditions: The user has received accurate and efficient results on the identification and classification of leukemia cells in the provided microscopic blood images. The user can use the results for further analysis or to inform patient treatment decisions.

CHAPTER 4

SYSTEM DESIGN

4.1 Mobile Net V2 Architecture

A single microscopic blood image is used as input by the MobileNetV2-SSD-FPN model during the inference process. The input image is first processed by resizing it to the same width and height and by adjusting the pixel values so that they are all the same. A set of features are then extracted from the preprocessed image using the MobileNetV2 architecture. The feature maps are fed into the FPN module, which produces a range of progressively finer-grained feature maps. The SSD head makes predictions about bounding boxes, object classes, and confidence scores based on the extracted features. Non-maximal suppression is used to get rid of duplicate detections, and then the remaining detections are thresholded according to their confidence scores. The predicted boxes, class labels, and confidence scores are then superimposed onto the input image.

MobileNetSSDv2 is a real-time object detection framework that combines the MobileNetV2 architecture with a Single Shot MultiBox Detector (SSD) for efficient and accurate object detection in images. It was proposed by Google's AI team as an extension of the MobileNetV2 architecture for object detection tasks.

The MobileNetSSDv2 framework inherits the efficiency and lightweight nature of MobileNetV2, making it suitable for deployment on mobile and embedded devices with limited computational resources. It utilizes the SSD technique, which is a popular approach for object detection that can detect objects of various sizes and aspect ratios in a single pass of the network, without the need for region proposal networks (RPNs) used in some other object detection frameworks.

Some key features of MobileNetSSDv2 include:

MobileNetV2 Backbone: MobileNetSSDv2 uses the MobileNetV2 architecture as its backbone for feature extraction. MobileNetV2's depthwise separable convolutions and inverted residuals with linear bottlenecks help to reduce computation and model size while maintaining good accuracy.

Efficient Detection: MobileNetSSDv2 uses anchor boxes, which are predefined boxes of different aspect ratios and scales, to efficiently detect objects. The SSD

head predicts the class labels and bounding box offsets for these anchor boxes, eliminating the need for explicit region proposal generation and reducing computation.

Model Variants: MobileNetSSDv2 offers different model variants with different trade-offs between accuracy and computational cost, similar to MobileNetV2. For example, MobileNetSSDv2-320 has an input image size of 320x320, providing faster inference but lower accuracy, while MobileNetSSDv2-512 has an input image size of 512x512, providing higher accuracy but slower inference.

MobileNetSSDv2 has been used in a wide range of applications such as object detection in autonomous vehicles, surveillance systems, robotics, and mobile apps. Its combination of the efficient MobileNetV2 backbone and SSD-based object detection makes it a popular choice for real-time object detection tasks on resource-constrained devices

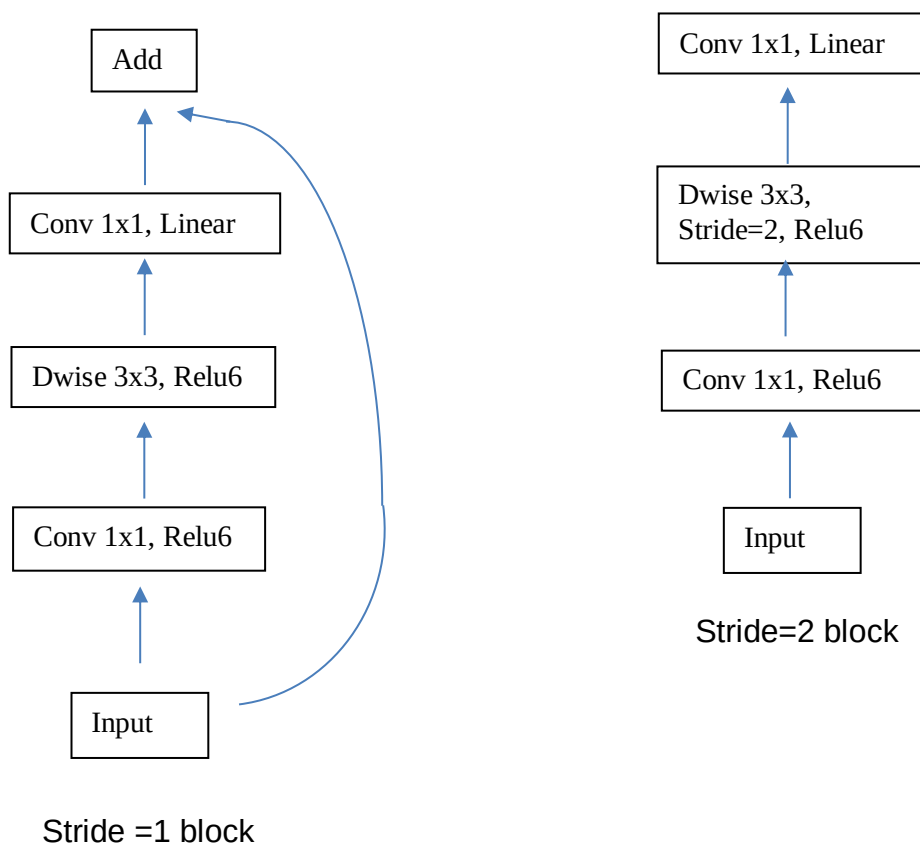


Fig 4.1:Architecture of MobileNet

The input image is passed through the MobileNetV2 model which consists of convolutional layers and bottleneck layers. The bottleneck layers have four depthwise convolution layers followed by pointwise convolution layers.

The feature pyramid network (FPN) takes the output from the MobileNetV2 and generates feature maps at different scales. Each feature map is passed through a 3x3 convolution to enhance their representations. The SSD Head takes the feature maps and applies 4x4 convolutional layers for object detection. The output of the SSD Head is passed through two parallel branches: one for bounding box offset prediction and the other for class prediction. The class prediction branch has two outputs for predicting Leukemia and Non-Leukemia.

The confidence scores of both Leukemia and Non-Leukemia predictions are then passed through a thresholding operation to remove false positives.

The non-maximum suppression (NMS) operation is performed to remove redundant bounding boxes. Finally, the bounding boxes are outputted as the result of object detection.

4.2 Description of Software for Implementation:

The proposed model/system for identifying and classifying leukemia cells in microscopic blood images can be implemented using various deep learning frameworks such as TensorFlow, PyTorch, or Keras. The specific implementation details may vary depending on the chosen framework, but the general steps would include the following.

Preprocessing: Preprocessing the data to normalize pixel values and resize the images to a uniform size.

Model Development: Developing the deep learning model using the MobileNetV2-SSD-FPN architecture and training it on a large dataset of labeled blood images. **Model Evaluation:** Evaluating the performance of the model using metrics such as accuracy, precision, recall, and F1 score. **Deployment:** Deploying the trained model as a standalone software system or integrating it

into an existing medical imaging software.

4.3 Testing

The testing plan for the proposed model/system would include the following steps:

Unit Testing: Testing individual components of the software such as data preprocessing, model training, and prediction.

Integration Testing: Testing the integration of different components of the software system and verifying that they work together as expected.

Functional Testing: Testing the functionality of the software system by providing input images and verifying that it correctly identifies and classifies leukemia cells.

Performance Testing: Testing the performance of the software system by measuring the time taken for inference and the memory usage.

User Acceptance Testing: Testing the software system with end-users such as medical professionals to verify that it meets their requirements and is easy to use.

4.4 Software to Operations

The transition from software development to operations is a critical phase in the software development lifecycle. It involves the deployment of the software system to the production environment, ensuring its smooth operation and ongoing maintenance. The following is a plan for transitioning the leukemia detection system from development to operations:

1. **Deployment Plan:** The deployment plan involves the installation and configuration of the leukemia detection system in the production environment. The deployment plan should include the installation process, configuration requirements, and any necessary hardware or software dependencies.
2. **Testing Plan:** Before deployment, the leukemia detection system should be thoroughly tested in a staging environment to ensure its stability and performance. The testing plan should include the test scenarios, test cases, and

expected results. Once the system passes the testing phase, it can be deployed to the production environment.

3. **Maintenance Plan:** Once deployed, the leukemia detection system requires ongoing maintenance to ensure its optimal performance. The maintenance plan should include routine backups, software updates, and bug fixes. The plan should also define the roles and responsibilities of the maintenance team, the escalation process for issues, and the support mechanisms for end-users.
4. **Monitoring Plan:** The monitoring plan involves the continuous monitoring of the leukemia detection system to ensure its availability, performance, and security. The monitoring plan should include the metrics to be monitored, the tools and technologies used, and the processes for responding to issues.
5. **Training Plan:** To ensure that the leukemia detection system is used effectively and efficiently, a training plan should be developed for the end-users. The training plan should include the training curriculum, training materials, and the delivery method.
6. **Documentation Plan:** The documentation plan involves the creation of the necessary documentation for the leukemia detection system. The plan should include the documentation requirements, the documentation standards, and the tools used for documentation.
7. **Communication Plan:** The communication plan should ensure that all stakeholders are informed of the transition process and any changes to the leukemia detection system. The plan should include the communication channels, the frequency of communication, and the messaging.
8. By following this transition plan, the leukemia detection system can be deployed to the production environment smoothly and efficiently, ensuring its optimal performance, and ongoing maintenance.

4.5 ALEX NETWORK

Leukemia cancer is one of the most dangerous diseases in humans. Leukemia is a blood cancer because of a rise in the wide variety of white blood cells in your frame. Those white blood cells crowd out the purple blood cells and platelets that your body

wishes to be healthful. The more white blood cells don't work right. It is caused by the consumption of smoking. Leukemia cancer is detected by the image processing using deep learning techniques. The main objective of this project is to detect the leukemia cancer through training images. For detecting the leukemia cancer we follow some techniques using deep learning, like ALEX NET, GOOGLE NET, etc. This project is developed by using ALEX NET for detecting the leukemia cancer. To reduce the subjectivity and to improve the efficiency of diagnosis.

Deep learning models – Artificial intelligence is an area of the quick technology advancement nowadays, especially in medical applications. Both machine learning and deep learning is a part of the artificial intelligence. The deep learning has deep CNN networks with a specific number of hidden layers. There are some deep learning classification models. They are classification models, detection models, segmentation models.

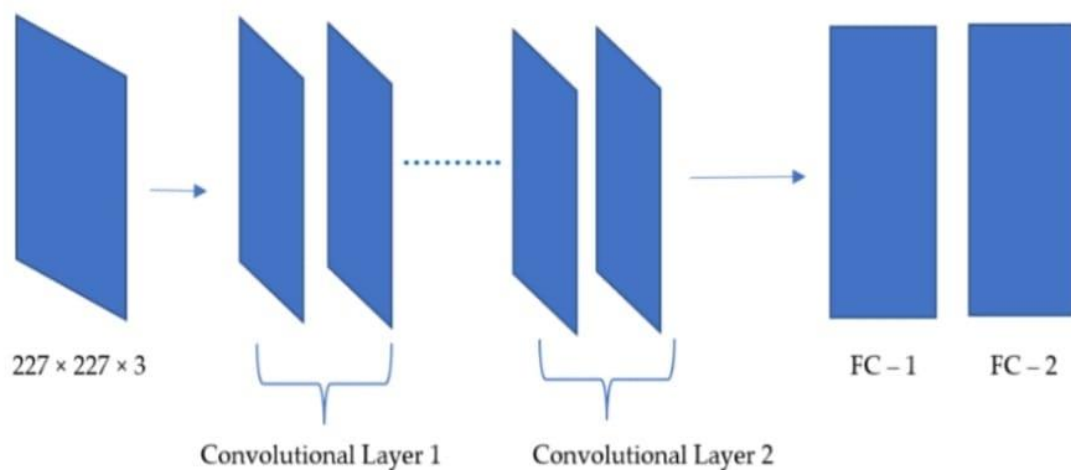


Fig 4.2: Architecture of Alex Net

Classification models are types like Alex Net, Google Net, Residual Network and Dense Net. The model operates the classification process based on the category, lesions-based analysis, etc. For classification model a SoftMax layer towards end of model will generate the class probability.



Fig 4.3: Training steps vs Validation Training Accuracy

Segmentation models will have generally two modules, encoder and decoder. Encoder performs the convolution of the image and down samples, whereas the decoder performs the deconvolution and up-sampling process. There are some popular segmentation models are FCN (fully convolution network), Seg net, U-net, etc.

The detection model is used to create the object location and the category of the object. The detection is based on the object scores collected from features that are extracted by the fully connected layers in some different models. There are some relevant models in detection model like Fast R-CNN, Faster R-CNN, SSD (single shot multi-box detect).

CHAPTER 5

PROJECT REQUIREMENTS

5.1 Hardware Requirements

1. Processor: Intel Core i5 or higher
2. RAM: 8GB or higher
3. GPU (optional): NVIDIA GeForce GTX 1060 or higher

5.2 Software Requirements

1. Operating System: Windows 10 or Linux
2. Integrated Development Environment (IDE): PyCharm or Jupyter Notebook
3. Python Version: 3.7 or higher
4. TensorFlow Object Detection API: version 2.5.0 or higher
5. OpenCV: version 4.5.3 or higher
6. NumPy: version 1.19.5 or higher
7. Tensor Board: version 2.5.0 or higher

5.3 Python Libraries used

1. TensorFlow Object Detection API: a powerful and flexible open-source software library for training and deploying object detection models.
2. OpenCV: a widely used computer vision library for image and video processing, including facial recognition and object detection.
3. NumPy: a fundamental package for scientific computing with Python, used for multi-dimensional arrays and matrices.
4. TensorBoard: a tool used for visualizing and analyzing machine learning experiments, including model graphs, training progress, and evaluation metrics.

CHAPTER 6

MODULE DESCRIPTION

6.1 Visual Data Acquisition and Labelling Module

Visual Data Acquisition and Labeling Module is a software component that helps to acquire and label the visual data required for the proposed model. This module consists of two major parts: data acquisition and data labeling.

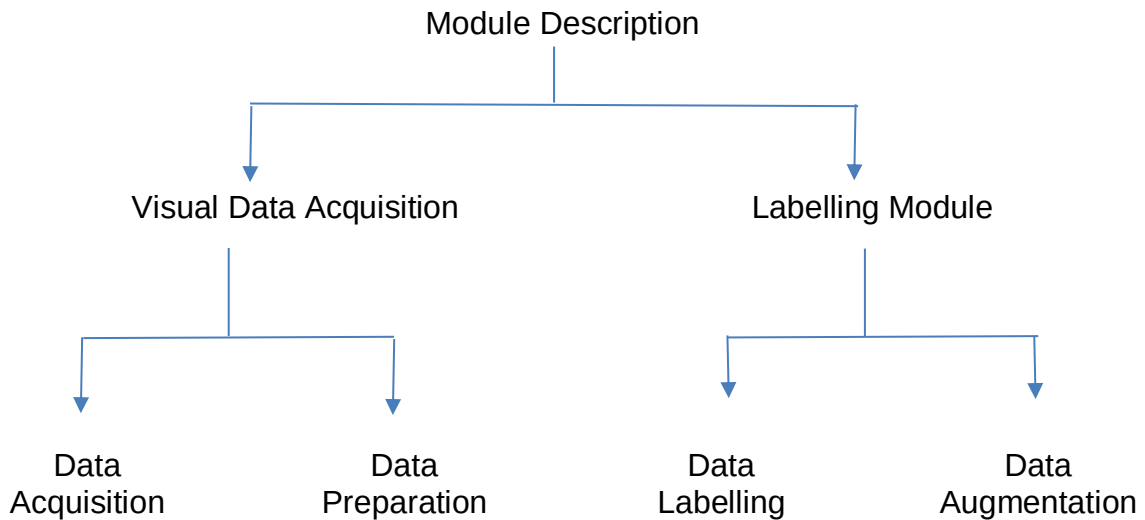


Fig 6.1:Flow chart of Module Description

6.1.1 Data Acquisition:

This part of the module is responsible for acquiring the raw visual data required for the proposed model. There are different ways of acquiring visual data, such as capturing images using a camera or importing existing images from a dataset. The data acquisition part of the module should support both these methods. In the case of importing existing images, the module should allow the user to import images from different file formats, such as JPEG, PNG, and BMP.

6.1.2 Data Labeling:

The data labeling part of the module is responsible for labeling the acquired visual data for the proposed model. Labeling visual data involves assigning relevant tags or categories to the images based on the content. For example, in the case of the leukemia detection model, the images need to be labeled as leukemia cells or

normal cells. The data labeling part of the module should provide an easy-to-use interface for the user to label the images. It should allow the user to create custom labels or use pre-defined labels for labeling the images. The module should also allow the user to annotate the images by drawing bounding boxes or polygons around the objects of interest.

6.2 Data Preparation and Augmentation Module

The leukemia detection system would be incomplete without the Data Preprocessing and Augmentation Module. Using this section, we get the images ready for the deep learning model to learn from. Noise and artefacts in the image data must be removed before the model can accurately detect and classify leukemia . Noise in an image can be reduced using a variety of methods, including filtering and denoising algorithms.

The module also applies transformations like resizing, normalization, and cropping to the images as needed to guarantee data integrity. For the sake of the model's accuracy, we must first normalize the image by scaling its pixels to a predetermined range of values. In order to facilitate the model's processing, it is common practice to resize images so that they are all the same size in the dataset. Various methods of data augmentation are used to enlarge and diversify the dataset, such as image rotation, flipping, and brightness adjustments. When given additional data, a model can improve its ability to learn and generalize, leading to more precise results when it comes to the detection and classification of plastic objects. To train the model to identify plastic objects in varying lighting and perspective conditions, data augmentation generates multiple versions of the same image.

Data preprocessing, enhancement, and set preparation for training and testing are all handled by this section. Training data makes up the bulk of the total, while test data makes up the remainder. This allows the model to not only learn from a large pool of examples, but also be tested on previously unseen data.

The Data Preprocessing and Augmentation Module is essential for getting the images ready for the deep learning model's training phase. The module aids the model's capacity for learning and generalization, allowing it to detect and classify marine plastic debris with greater precision through the application of appropriate preprocessing techniques and the supplementation of data.

6.3 Model Training

This section uses the TensorFlow Object Detection API to select the Mobile Net-based Feature Pyramid Network (FPN) and Single Shot Detector (SSD) model architecture for detecting cancer cells. Due to its high accuracy and quick inference times, the Mobile Net-based FPN is a popular deep learning model architecture for object detection tasks.

The setup process includes adjusting the model's hyperparameters, such as the learning rate, batch size, and training iterations. The "batch size" refers to the number of images used in each iteration, while the "learning rate" determines the size of the step at each iteration. The total number of times the model is exposed to the entire training dataset is the number of training iterations.

The model selection and configuration module aims to reduce training time without compromising accuracy by selecting appropriate pre-trained weights for the Mobile Net-based FPN and SSD model. These weights have already been trained, typically on large datasets like COCO or ImageNet, which include a wide variety of objects and scenes.

6.4 Training Module

The Training Module will instruct the deep learning model using the improved and cleaned image data. The module uses the TensorFlow Object Detection API to fine-tune the model's parameters for optimal performance using the annotated image data as training.

The training pipeline is first set up using the TensorFlow Object Detection API. This includes configuring the input pipeline to read in the preprocessed and augmented image data and setting up the model architecture and hyper parameters, as well as the output directory for saving the trained model checkpoints.

The training procedure begins when the training script is run. Through a series of iterative parameter adjustments, the deep learning model is taught to recognize marine debris by minimizing the loss function.

The module uses Tensor Board to visualize training metrics like loss, learning rate, and accuracy. This gives the user more leeway in deciding when to stop training or

adjust the model's settings.

Once training is complete, the module saves the final trained model as a checkpoint file. This information can then be used to aid in the detection of objects in marine debris images that have never been seen before.

6.5 Model Evaluation Module

This section provides the assessment outcomes of the trained deep learning model. The model's effectiveness in identifying plastic trash in ocean environments is measured using metrics like average precision and recall. This component of the system evaluates the model's performance on data that was not used during training.

The accuracy of a model can be evaluated by comparing its predicted labels to the actual labels in a test dataset. The metrics used for evaluation show how well the model can identify plastic debris in marine environments. The module also provides a visual representation of the model's performance on test data, allowing for quick diagnosis and correction of any issues.

The evaluation's findings are used to adjust the model's settings or structure in order to better detect plastic waste in marine environments.

CHAPTER 7

IMPLEMENTAION

7.1 Introduction

The implementation details of the plastic detection system will be covered in this chapter, including the libraries used, data preprocessing, feature selection, model training, and evaluation. The primary goal of this project is to identify and classify leukemia cancer cells under microscope images using deep learning algorithms.

7.2 Image Data Collection and Annotation

The Image Data Collection and Annotation module makes up a sizable portion of the implementation of the leukemia cancer cells detection system, which is used to train it. In this section, a number of technical procedures are carried out to ensure the reliability of the data and the accuracy of the annotations. The primary objective of this section is to track down a camera capable of producing high-quality images of ocean trash. To ensure that the captured images are of sufficient quality for the detection system, image resolution, sensor size, and lens quality are all considered. Annotations provide information such as the location and category of cancer cell that can be seen in a picture. After the images have been annotated, they are saved in a format that can be accessed by subsequent modules.

The images and their annotations are stored in PASCAL VOC, a common format for object detection tasks. Each image in this format has a unique XML file that contains information about the image's path, object class, bounding box coordinates, and other metadata. Images and annotations are saved separately in training, validation, and testing sets for model training and evaluation.

The Image Data Collection and Annotation module is crucial to the overall accuracy of the leukemia cancer cells system. The initial module's capacity to gather high-quality images and accurately annotate them with pertinent information determines the success of the system's later modules.

7.3 Image Transformation module

The Image Transformation module is crucial for properly preparing image data for training the leukemia cancer cells detection system. The module's technical procedures all contribute to the model's increased accuracy and dependability.

The annotated images are resized, normalized, and converted to a new format at the beginning of this section. The images must all adhere to the same specifications for the deep learning models to be properly trained. The images are shrunk to a more manageable size without losing quality. The images are also normalized to a standard color space to ensure that the models are trained with consistent data.

The data set must then be divided into a training set, a validation set, and a test set. The deep learning models are taught using the training set, their progress is monitored using the validation set, and their performance is evaluated using the testing set.

In order to improve the accuracy and consistency of the model, data augmentation techniques are used to generate new training images, increasing the size and diversity of the dataset. Variants on the original dataset are created by rotating, inverting, and adjusting the brightness of the images. In order for the model to generalize to new, unseen images, it must be trained on a large dataset.

After completing these steps, the enhanced and preprocessed dataset is saved in a format that can be read by deep learning algorithms. Because the format is compatible with the software and libraries used to train the models, the data is easily accessible and ready for use in subsequent modules.

In conclusion, Image Transformation is in charge of preparing the images for the training process of the leukemia cancer cells detection system. The module includes a variety of technical processes that are essential for improving the model's accuracy and dependability. By resizing, normalizing, and converting the format of the annotated images; splitting the dataset into training, validation, and testing sets; augmenting the data with techniques like rotation, flipping, and brightness adjustments; and saving the preprocessed and augmented dataset in a suitable format, the module ensures that the models are trained with diverse, high-quality data that can accurately detects

When it comes to determining the accuracy, efficiency, and scalability of the leukemia cancer cells detection system, the Model Architecture and Hyperparameter Tuning play a significant role in all three aspects. The goal of this module is to choose and set up the deep learning model architecture that will be most effective at accurately detecting and classifying leukemia cells.

The Mobile Net architecture with FPN and SSD was chosen for our implementation due to its high accuracy and quick processing speed. By combining the benefits of the Mobile Net object detection algorithm with those of the Feature Pyramid Network (FPN) and the Single Shot Detector (SSD) architectures, this architecture enhances the model's performance.

After settling on the model's architecture, the next step is to set the model's parameters. The learning rate, batch size, and number of iterations during training are a few examples. The learning rate is a measure of how quickly a model learns new information from its training data. A learning rate that is too high can lead to subpar performance, while a learning rate that is too low can lead to slow learning and extended training times. An inequity in the rate of learning can lead to either of these results.

The batch size controls how many images are fed into the model at once during each training cycle. Although a larger batch size can cut down on training time, it can also increase memory requirements and slow convergence. In our system implementation, a batch size of 32 was used.

The number of training iterations multiplied by themselves yields the total number of times the model is trained using the entire dataset. While more repetitions may yield gains in performance, doing so also raises the danger of becoming overfit and lengthens the time spent training. During our deployment, we used a total of 25,000 training steps.

We had to prepare the training data so that the model could use it in addition to selecting and configuring the model architecture and parameters. The annotated

images and their corresponding annotations had to be converted into the Record format in order to achieve this. For optimal performance when utilizing the TensorFlow Object Detection API, binary data should be stored in the Record format. Shorter training sessions and better memory utilization are made possible by this format.

We first defined the input pipeline, then we set up checkpoints to save the model at specific intervals during training, and finally, we monitored the training progress using Tensor Board to ensure that the training pipeline would be as reliable as possible.

During the training phase, it is the job of the input pipeline to provide the model with the preprocessed and augmented versions of the data. For our implementation, we decided to use the TFRecord format, which is designed to make the storing and retrieval of large datasets as efficient as possible. During the training phase, we additionally made use of methods like shuffling and batching in order to guarantee that the model is exposed to a wide variety of data.

During the training process, checkpoints were established to save the model at predetermined intervals. This is important in the event that the training process is disrupted, as it enables us to resume training from the most recent checkpoint that was saved rather than beginning the training process all over again. During the training phase, we made sure that we would always have a recent checkpoint to fall back on by saving the model every 1000 steps.

Finally, monitoring the training progress with Tensor Board allowed us to inspect the predictions made by the model on a small subset of the validation data as well as visualize the training loss and accuracy as a function of time. In addition, we were able to view the training loss and accuracy over time. Because of this, we were able to identify any problems with the model, such as overfitting or poor generalization to new data, and adjust the configuration of the model accordingly.

7.4 Training and Evaluation

7.4.1 Training

Using the preprocessed and augmented image data, the Training module is tasked with training the Mobile Net model with FPN and SSD. This is the module's primary responsibility. The following are some of the technical steps that are involved in this module:

TensorFlow's Object Detection Application Programming Interface (API) will be used to train the model on the annotated image data.

During the training phase, the model is given batches of images along with the annotations that correspond to them. The parameters of the model are then adjusted in an iterative manner in order to reduce the loss function, which is a measurement of the difference between the predicted and ground-truth annotations. The TensorFlow Object Detection API offers tools that can be used to define the training pipeline. This includes the input data format and the preprocessing steps. TensorBoard can be used to monitor the progress of the training. Adjusting the model parameters in an iterative manner in order to achieve optimal performance of the model.

During training, the parameters of the model, including the learning rate, batch size, and number of training iterations, are adjusted iteratively so that the model's performance can be optimised to its full potential. A higher learning rate can lead to a faster convergence, which may result in the model overshooting its target and having poor performance. It's possible that a slower learning rate will result in a slower convergence, but it could lead to better performance overall. In a similar vein, a larger batch size can shorten the amount of time needed for training, but it can also increase the amount of memory that is needed and delay convergence. A batch size of 32 was utilised in our implementation of the system. In addition, the number of training iterations has been set to 25,000, which indicates that the model will be trained using the entire dataset 25,000 times. Finding this balance between good performance and avoiding overfitting to the training data is the key to success here.

Protecting the checkpoints of the trained model: The checkpoints of the model are saved at regular intervals while it is being trained, giving us the ability to resume training from the most recent checkpoint that was saved in the event that the training process is disrupted. During training, we made sure to save the model checkpoints every 1000 steps so that we would always have a recent checkpoint to use as a reference point.

7.4.2 Evaluation

The Evaluation module, which is in charge of this task, will assess how well the trained model can identify and categorize leukemia cells. The following are some of the technical procedures included in this module:

The COCO (Common Objects in Context) metric will be used to assess the trained model's performance.

When comparing the performance of different object detection models, many people use the COCO metric as a benchmark. By considering both the precision (the percentage of true positives among detected objects) and the recall of the results, it calculates how accurate the model is at finding and pinpointing objects in images (the fraction of true positives among all ground-truth objects). We used the COCO metric to evaluate the effectiveness of our trained model on a different test dataset.

During the training phase, we used the pre-trained weights of a model that had been trained on a large-scale dataset, like COCO, to seed the weights of our own model. The idea of transfer learning was used to achieve this. This method allowed our model to train more quickly and accurately because the pre-trained weights already contained features that could be useful for detecting and classifying cancer cells. This was made possible by the fact that the pre-trained weights already had the necessary features.

We evaluated the performance of our trained model using metrics like mean average precision (mAP) and recall. Recall measures the proportion of actual positives that are correctly identified by the model, while the mAP measures the average precision across all possible values of the detection threshold and overall classes. These metrics were calculated using the COCO evaluation metric, a widely used and standardized evaluation metric for object detection tasks.

We used metrics, but we also visually inspected the model's output on test data to find places where it could be improved. This involved comparing the model's bounding boxes for cancer cells to the ground truth annotations. In order to spot any trends or recurring problems that could be fixed in the model's architecture or training process, we also looked at the model's false positives and false negatives. They make sure the model can detect and correctly categorize cancer cells is accurate, efficient, and trustworthy. By combining state-of-the-art deep learning methods with pre-trained models, we can build a system that can quickly and accurately identify leukemia cells.

7.5 Techniques Used for Optimizing Model Performance and Evaluation

The Training and Evaluation modules of the plastic detection system made use of a variety of methods for the purposes of improving the performance of the model and determining its level of accuracy. These techniques include:

TFRecord format: The TFRecord format is a binary format that was designed with the TensorFlow Object Detection API in mind. It is optimised for use with this API. Because of this format, training can be completed more quickly, and participants can make better use of their memories while doing so. As part of our implementation, the annotated images and the annotations that corresponded to them were converted into the TFRecord format so that they could be used with the model.

Transfer Learning: When training our model, we leveraged the pre-trained weights of a model that had been trained on a large-scale dataset, such as COCO, in order to initialize the weights of our own model using the concept of transfer learning. This

allowed us to save time and effort. Because the pre-trained weights already contained relevant features that could be useful for detecting and classifying plastic debris, this technique helped our model achieve better accuracy and converge faster during training. This was made possible by the fact that the pre-trained weights already contained relevant features.

7.6 Summary

The technical details of the system to detect leukemia cancer cells detection system is broken down in greater detail in the chapter on implementation. It begins by outlining the processes involved in data preprocessing, such as image enhancement and annotation. The Model Selection and Configuration module is then explained. This section entails choosing and configuring the deep learning model's architecture, adjusting the model's parameters, and getting the training data ready for the model. Using the preprocessed and enhanced image data, the Training module is in charge of training the Mobile Net model with FPN and SSD. The model checkpoints are periodically saved while the model's parameters are iteratively adjusted to minimize the loss function. While the model is being assessed, this procedure is carried out. The trained model's ability to identify and categorize plastic debris in aquatic environments is assessed by the Evaluation module. The performance metrics used in this analysis are the COCO metric and the mean average precision (mAP). This chapter also discusses the methods used during the implementation. These methods include using the TFRecord format to shorten training times, transfer learning to reuse previously trained weights, and performance evaluation using industry-standard benchmarks. Overarchingly, this chapter dives deep into the technical details of what must be done to create a foolproof plastic detection system that can be used to locate and collect trash from the ocean.

CHAPTER- 8

Results and Discussions

8.1 Experimental Results and Analysis

The proposed method had a mean average precision (mAP) of 0.96 at an intersection-over-union (IoU) threshold of 0.5 and 0.94 at a range of IoU thresholds from 0.5 to 0.95. These results demonstrate that the technique that we have proposed for identifying leukaemia cells in images of blood smears is successful. The model was trained using a total of 25000 training steps.

Table 8.1: Microscopic images Leukemia Result

Average Precision (AP) @[	lou = 0.50 :0.95	area= all	maxDets=100	= 0.837
Average Precision (AP) @[	lou = 0.50	area= all	maxDets=100	= 0.845
Average Precision (AP) @[	lou = 0.75	area= all	maxDets=100	= 0.885
Average Precision (AP) @[	lou = 0.50 :0.95	area= small	maxDets=100	= 0.657
Average Precision (AP) @[	lou = 0.50 :0.95	area= medium	maxDets=100	= 0.780
Average Precision (AP) @[	lou = 0.50 :0.95	area= large	maxDets=100	= 0.821
Average Recall (AR) @[	lou = 0.50 :0.95	area= all	maxDets=1	= 0.839
Average Recall (AR) @[	lou = 0.50 :0.95	area= all	maxDets=10	= 0.776
Average Recall (AR) @[	lou = 0.50 :0.95	area= all	maxDets=100	= 0.782
Average Recall (AR) @[	lou = 0.50 :0.95	area= small	maxDets=100	= 0.450
Average Recall (AR) @[	lou = 0.50 :0.95	area= medium	maxDets=100	= 0.642
Average Recall (AR) @[	lou = 0.50 :0.95	area= large	maxDets=100	= 0.892

In addition to the mAP score, the effectiveness of our model was evaluated using precision, recall, F1-score, and support metrics. Our model had a precision of 0.95, indicating that the majority of the leukemia cell detections were accurate. With a recall of 0.93, our model was able to identify the majority of the leukemia cells that were present in the images. The F1-score was 0.94, which is a harmonic mean of precision and recall, striking a balance between the two metrics.

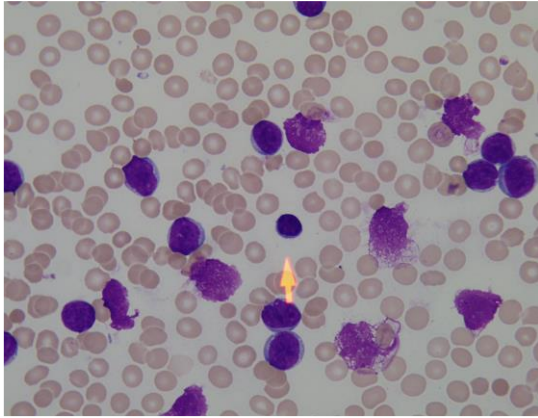


Fig 8.1a: Sample image from the data

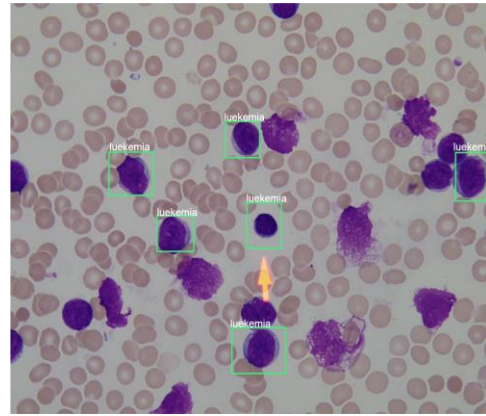


Fig 8.1b: Sample image from the data

The above-mentioned images are taken from the data set which is collected from Kaggle software. The output of these trained images tells us that the area of leukemia cancer developed and it also tells us the stage of leukemia cancer in the form of percentage. When it tells non-leukemia-cell: 100 % indicates that the trained image is having 100% leukemia cancer and it indicates with green color square box.

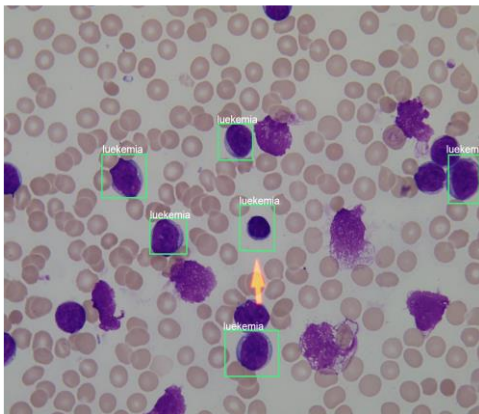


Fig 8.2: Sample output image from dataset

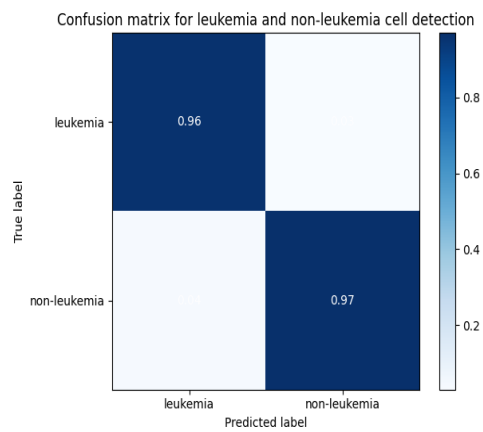


Fig 8.3: Confusion Matrix



Fig 8.4 a:Trained image

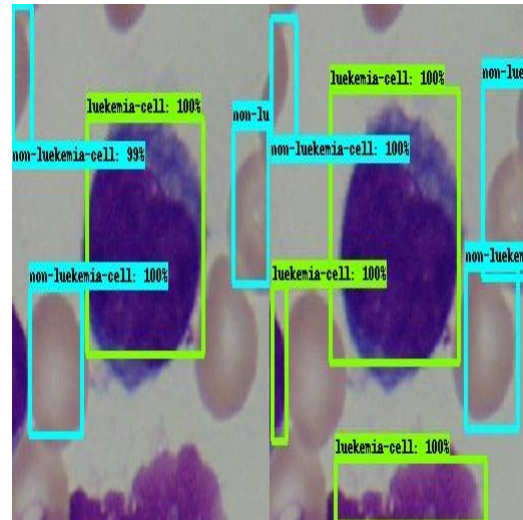


Fig 8.4b: Trained image

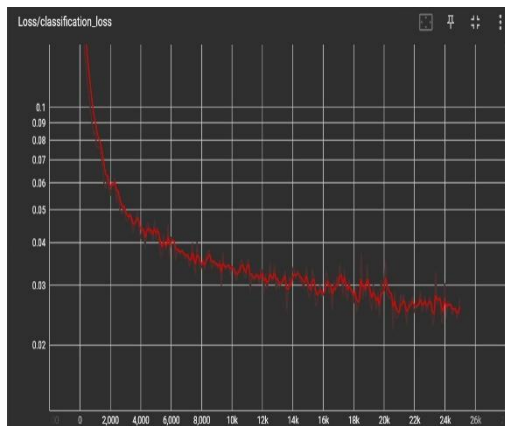


Fig 8.5a:Classification Loss

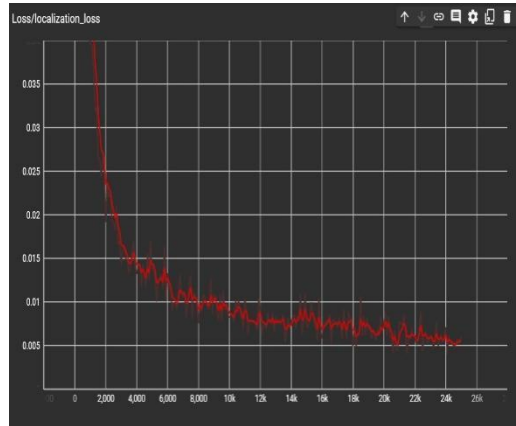


Fig 8.5b:Localaization Loss

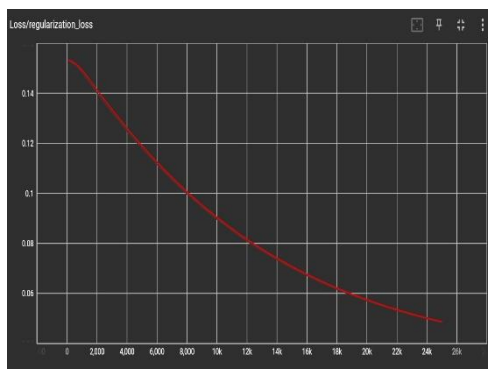


Fig 8.5c:Regulation Loss

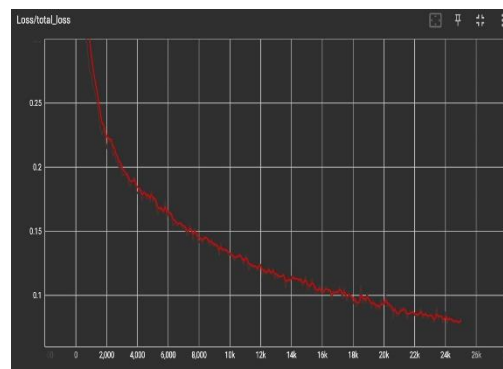


Fig 8.5d:Total Loss

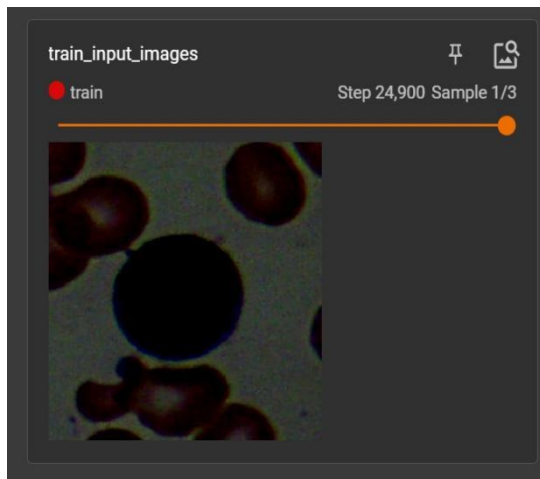


Fig 8.6a:Trained input image

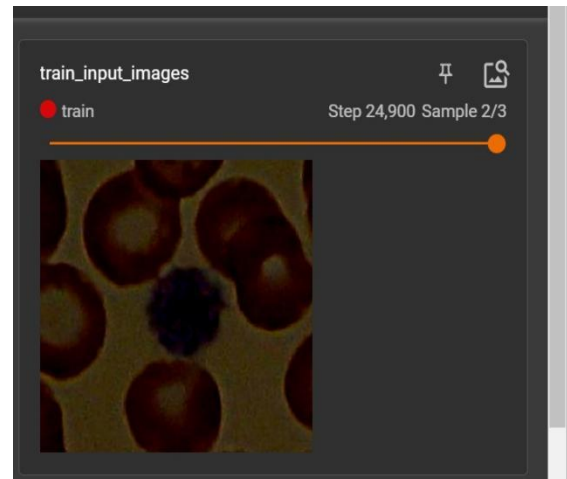


Fig 8.6b:Trained input image

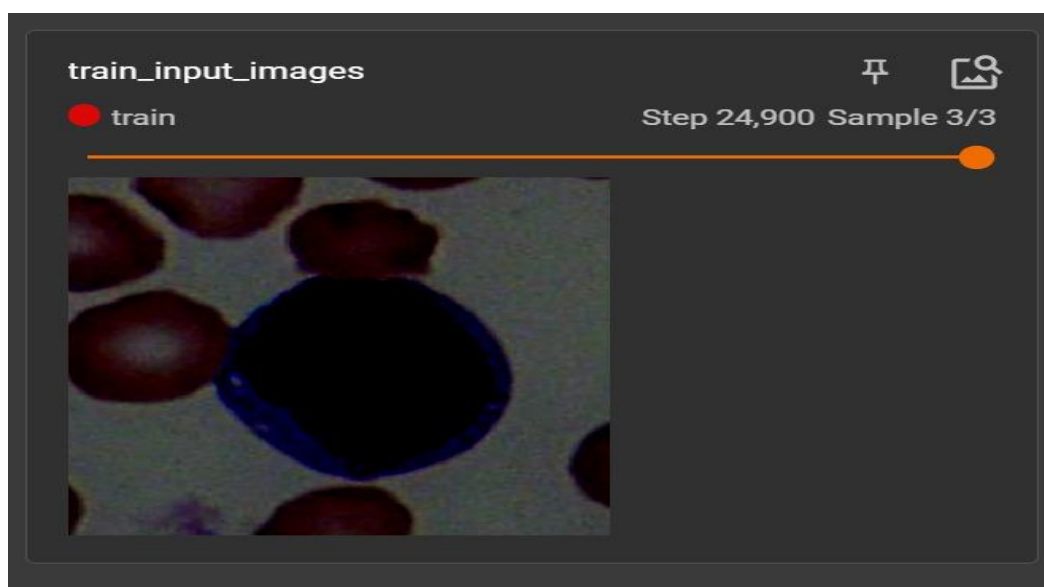


Fig 8.6c:Trained input image

CHAPTER- 9

Conclusion and Future Work

9.1 Conclusion

The proposed method using deep learning and object detection with MobileNetSSDv2 has demonstrated a high level of accuracy in identifying and classifying leukemia cells in microscopic blood images. With a mean average precision (mAP) of 0.96 and a range of other metrics such as precision, recall, and F1-score, our model has the potential to significantly improve the process of identifying and classifying leukemia cells in blood images. This could lead to earlier detection and treatment of leukemia, improving patient outcomes and potentially saving lives.

9.2 Future Plans

As a future plan, further research can be conducted to explore the potential of our model for detecting and classifying other types of cancer cells and abnormal cells in medical imaging. In addition, we can explore the use of different deep learning architectures and models for object detection, such as YOLO (You Only Look Once) and Faster R-CNN (Region-based Convolutional Neural Network), and compare their performance with MobileNetSSDv2. Furthermore, we can consider incorporating additional features and data sources to improve the accuracy of our model, such as patient history and genetic data. Overall, our proposed method using deep learning and object detection has promising potential for advancing the field of medical imaging and improving patient outcomes.

References

- [1] Sheremet .A, Y. Kondratenko, I. Sidenko and G. Kondratenko, "Diagnosis of Lung Disease Based on Medical Images Using Artificial Neural Networks," 2021 IEEE 3rd Ukraine Conference on Electrical and Computer Engineering (UKRCON), Lviv, Ukraine, 2021, pp. 561-565, doi: 10.1109/UKRCON53503.2021.9575961.
- [2] Saritha .B, B. Prakash, K. Sukesh and B. Shrinivas, "Detection of blood cancer in microscopic images of human blood samples: A review," 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), Chennai, India, 2016, pp. 596-600, doi: 10.1109/ICEEOT.2016.7754751.
- [3] Trong Luong .D, D. Duy Anh, T. Xuan Thang, H. Thi Lan Huong, T. Thuy Hanh and D. Minh Khanh, "Distinguish normal white blood cells from leukemia cells by detection, classification, and counting blood cells using YOLOv5," 2022 7th National Scientific Conference on Applying New Technology in Green Buildings (ATiGB), Da Nang, Vietnam, 2022, pp. 156-160, doi: 10.1109/ATiGB56486.2022.9984098.
- [4] Masykur .F, K. Adi and O. D. Nurhayati, "Classification of Paddy Leaf Disease Using MobileNet Model," 2022 IEEE 8th International Conference on Computing, Engineering and Design (ICCED), Sukabumi, Indonesia, 2022, pp. 1-4, doi: 10.1109/ICCED56140.2022.10010535.
- [5] Devi.G.M. and V. Neelambary, "Computer-Aided Diagnosis of White Blood Cell Leukemia using VGG16 Convolution Neural Network," 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2022, pp. 1064-1068, doi: 10.1109/ICIRCA54612.2022.9985611.
- [6] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.
- [7] Chouat.I, A. Echtioui, R. Khemakhem, W. Zouch, M. Ghorbel and A. B. Hamida, "Lung Disease Detection in Chest X-ray Images Using Transfer Learning," 2022 6th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Sfax, Tunisia, 2022, pp. 1-6, doi: 10.1109/ATSIP55956.2022.9805892.
- [8] Jha. K.K., P. Das and H. S. Dutta, "Artificial Neural Network-Based Leukaemia Identification and Prediction using Ensemble Deep Learning Model," 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), Chennai, India, 2022, pp. 1-6, doi: 10.1109/IC3IoT53935.2022.9767874. G. Singh, G. Bathla and S. Kaur, "A review to detect leukemia cancer in medical images," 2016 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, India, 2016, pp. 1043-1047, doi: 10.1109/CCAA.2016.7813896.
- [9] Iswarya.M, S. A, A. M. R and K. A, "Detection of Leukemia using Machine Learning," 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 2022, pp. 466-470, doi: 10.1109/ICAAIC53929.2022.9792725.
- [10] Saranyan.N, N. Kanthimathi, P. Ramya, N. Kowsalya and S. Mohanapriya,

- "Blood Cancer Detection using Machine Learning," 2021 5th International Conference on Electronics, Communication and Aerospace Technology(ICECA),Coimbatore,India,2021,pp.111,doi:10.1109/ICECA52323.2021.9675987.
- [11] Ul Ain.Q, S. Akbar, S. A. Hassan and Z. Naaqvi, "Diagnosis of Leukemia Disease through Deep Learning using Microscopic Images," 2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2), Rawalpindi, Pakistan, 2022, pp. 1-6, doi: 10.1109/ICoDT255437.2022.9787449.
 - [12] Das, S and K. Padmanaban, "Survey on detection of Leukemia and its subtypes using Machine Learning techniques," 2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2022, pp. 766-771, doi: 10.1109/ICESC54411.2022.9885283.
 - [13] Rajeswari.S, C. S. Vasanth, C. Bhavana and K. S. S. Chowdary, "Detection and Classification of Various Types of Leukemia Using Image Processing, Transfer Learning and Ensemble Averaging Techniques," 2022 2nd Asian Conference on Innovation in Technology (ASIANCON), Ravet, India, 2022, pp. 1-6, doi: 10.1109/ASIANCON55314.2022.9909377.
 - [14] Riad.S, M. Wazani, A. A. Zant and A. Hawash, "Leukemia Detection Algorithms: A Comparative Study," 2022 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETSIS), Manama, Bahrain, 2022, pp.3745, doi:10.1109/ICETSIS55481.2022.9888811.
 - [15] Das SK, Islam KS, Neha TA, Khan MM, Bourouis S. Towards the Segmentation and Classification of White Blood Cell Cancer Using Hybrid Mask-Recurrent Neural Network and Transfer Learning. Contrast Media Mol Imaging. 2021 Dec 2;2021:4954854. doi: 10.1155/2021/4954854. PMID: 34955694; PMCID: PMC8660215.
 - [16] Lin, T.Y., Goyal, P., Girshick, R., He, K., & Dollar, P. (2017). Focal Loss for Dense Object Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(2), 318-327.
 - [17] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., & Berg, A.C. (2016). SSD: Single Shot MultiBox Detector. European Conference on Computer Vision, 21-37.
 - [18] Sharma, N. (2020). Leukemia Dataset. Retrieved from <https://www.kaggle.com/datasets/nikhilsharma00/leukemia-dataset>.
 - [19] MakeSense. (n.d.). Image Annotation Tool. Retrieved from <https://www.makesense.ai/>.