COMPARISON OF DEEP LEARNING ALGORITHMS FOR LEUKEMIA CANCER CELL CLASSIFICATION

A THESIS SUBMITTED TO
THE FACULTY OF ARCHITECTURE AND ENGINEERING
OF
EPOKA UNIVERSITY

BY

GERTA BUSHI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JUNE, 2024

Approval sheet of the Thesis

This is to certify that we have read this thesis entitled **"Comparison of Deep Learning Algorithms for Leukemia Cancer Cell Classification"** and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

_____

Assoc. Prof. Dr. Arban Uka
Head of Department
Date: June 28, 2024

Examining Committee Members:

Prof. Dr. Betim Çiço (Computer Engineering) _____

Prof. Dr. Bekir Karlık (Computer Engineering) _____

Dr. Florenc Skuka (Computer Engineering) _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Surname: Gerta Bushi

Signature: _____

ABSTRACT

COMPARISON OF DEEP LEARNING ALGORITHMS FOR LEUKEMIA CANCER CELL CLASSIFICATION

Bushi, Gerta

M.Sc., Department of Computer Engineering

Supervisor: Prof. Dr. Bekir Karlık

Leukemia is a cancer-related disease which causes the death of individuals worldwide, regardless of age and gender. It affects the blood and bone marrow, thus leading to the abnormal production of immature white blood cells. Some of the factors that might contribute to leukemia's development might be related to genetics, radiation or chemical exposure, infections, or immune system disorders. A reliable and fast diagnosis of leukemia is crucial for a successful treatment to ensure high survival rates and low number of deaths.

Nowadays, blood tests are widely used for diagnosing leukemia. Patients undergo a complete blood count (CBC) to evaluate the count of blood cells present. In cases of leukemia, CBC reveals abnormal count of white blood cells (WBC), red blood cells (RBC) and platelets. Additionally, these blood cells are examined under a microscope. Based on the results, immature or abnormal-looking white blood cells may indicate leukemia. However, this type of diagnosis is often slow, time-consuming and less accurate, mainly because under microscopes, the shape of leukemic cells might seem similar to the shape of normal white cells, therefore making the diagnosis prone to errors.

Therefore, in this thesis, we will focus on the deep learning algorithms which have shown promising results in diagnosing leukemia cells. Some of these algorithms include Convolutional Neural Networks (CNNs), which in the context of leukemia cells diagnosis, can be trained to classify images of blood smears into normal blood

cells or leukemic blood cells. The second algorithm includes Optimized Deep Recurrent Neural Networks (ODRNNs), which can be used to analyze time-series data such as videos of cell movements or changes in cell morphology over time. The last algorithm is Transfer Learning, which is applied by fine-tuning a pre-trained neural network on a dataset of leukemia cells. This approach helps improve the performance of the model, especially when limited labelled data are available for training.

Keywords: *Leukemia, White Blood Cells, Diagnosis, Deep Learning Algorithms, Convolutional Neural Networks, Optimized Deep Recurrent Neural Networks, Transfer Learning.*

ABSTRAKT

KRAHASIMI I ALGORITMEVE TË DEEP LEARNING PËR KLASIFIKIMIN E QELIZAVE TË KANCERIT TË LEUÇEMISË

Bushi, Gerta

Master Shkencor, Departamenti i Inxhinierisë Kompjuterike

Udhëheqësi: Prof. Dr. Bekir Karlık

Leucemia është një sëmundje e lidhur me kancerin e cila shkakton vdekjen e individëve në mbarë botën, pa marrë parasysh moshën dhe gjininë. Ndikon në gjak dhe palcën e kockave, duke çuar kështu në prodhimin anormal të qelizave të bardha të papjekura të gjakut. Disa nga faktorët që mund të kontribuojnë në zhvillimin e leuçemisë mund të lidhen me gjenetikën, rrezatimin ose ekspozimin kimik, infeksionet ose çrregullimet e sistemit imunitar. Një diagnozë e besueshme dhe e shpejtë e leuçemisë është vendimtare për një trajtim të suksesshëm për të siguruar norma të larta mbijetese dhe numër të ulët vdekjesh.

Në ditët e sotme, analizat e gjakut përdoren gjerësisht për diagnostikimin e leuçemisë. Pacientët i nënshtrohen një numërimi të plotë të gjakut (CBC) për të vlerësuar numërimin e qelizave të gjakut të pranishme. Në rastet e leuçemisë, CBC zbulon numërimin anormal të qelizave të bardha të gjakut (WBC), rruazave të kuqe të gjakut (RBC) dhe pllakëzave. Përveç kësaj, këto qeliza të gjakut shqyrtohen nën mikroskop. Në bazë të rezultateve, qelizat e bardha të gjakut të papjekura ose me pamje anormale mund të tregojnë leuçemi. Megjithatë, kjo lloj diagnoze është shpesh e ngadalshme, kryesisht sepse nën mikroskop, forma e qelizave leukemike mund të duket e ngjashme me formën e qelizave normale të bardha, duke e bërë diagnozën të prirur ndaj gabimeve.

Prandaj, në këtë tezë, ne do të përqendrohemi në algoritmet të cilat kanë treguar rezultate premtuese në diagnostikimin e qelizave të leucemisë. Disa nga këto algoritme

përfshijnë Convolutional Neural Networks (CNNs), të cilat në kontekstin e diagnozës së qelizave të leucemisë, mund të trajnohen për të klasifikuar imazhet e gjakut në qelizat normale ose qelizat leukemike të gjakut. Algoritmi i dytë përfshin Optimized Deep Recurrent Neural Networks (ODRNNs), të cilat mund të përdoren për të analizuar të dhënat e serive kohore si videot e lëvizjeve të qelizave ose ndryshimet në morfologjinë qelizore me kalimin e kohës. Algoritmi i fundit është Transfer Learning, i cili aplikohet duke rregulluar një rrjet neuronal të para-trajnuar në një grup të dhënash të qelizave të leuçemisë. Kjo metodë ndihmon në përmirësimin e performancës së modelit, veçanërisht kur të dhënat e etiketuara janë në dispozicion për trajnim.

Fjalët kyçe: *Leucemia, Qelizat e Bardha të Gjakut, Diagnoza, Algoritmet Deep Learning, Rrjetet Neurale Convolutional, Optimized Deep Recurrent Neural Networks, Transfer Learning*

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to those who have supported and guided me throughout the course of my master's thesis. Their encouragement, patience, and expertise were invaluable in helping me complete this significant milestone.

First and foremost, I would like to extend my sincere thanks to my supervisor, Prof. Dr. Bekir Karlik. Your insightful guidance, constructive feedback, and unwavering support were instrumental in the successful completion of this thesis. Your expertise and dedication to my academic growth provided a solid foundation for this research. Thank you for always being available to review drafts and offer your wisdom. Your mentorship has profoundly impacted my academic journey, and I am grateful for your commitment to my success.

I am also deeply thankful to my family. To my parents, thank you for your unconditional love, encouragement, and sacrifices. Your belief in my abilities and your constant support have been my pillars of strength throughout this journey. Thank you for your understanding, patience, and emotional support during the challenging times. Your encouragement and faith in me have been a source of motivation and comfort.

Completing this thesis would not have been possible without the collective support and encouragement from my supervisor, family, and friends. Thank you all for your unwavering support and for believing in me.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Leukemia is a cancer type which is caused by the abnormal production of white blood cells. The human body consists of three types of cells: red blood cells, white blood cells and platelets, as it is shown below on Figure 1. Red blood cells, also known as erythrocytes, play a crucial role in the supply of oxygen from the heart to all tissues and make up half of the total blood volume in our body. Instead, white blood cells, also known as leukocytes, play an essential role in the immune system by defending our body from pathogens, infections, bacteria and viruses. Platelets, on the other hand, also known as thrombocytes, play a crucial role in blood clotting. When blood vessels are damaged, platelets come in handy by sealing the vessel and preventing blood loss.

Figure 1 : Types of blood cells. From left to right: red blood cell, white blood cell, platelets.

The correct categorization of white blood cells is essential in diagnosing the disease and determining the nature of it, as in our case, leukemia. In a normal human body of a healthy individual, the growth of white blood cells happens due to the body's requirement, but in a patient of leukemia, white blood cells are formed and grown abnormally and are not effective.

Due to having a dark purple color, leukemic cells are easy to get identified, but the diagnosis and further processing is quite difficult due to the variations in patterns and texture. For this reason, several deep learning algorithms have been developed to

make a correct assessment and diagnosis of leukemia and accurately differentiate between healthy cells and leukemic cells. With the evolution of computer vision and deep learning algorithms, several challenges have been solved in terms of image analysis, due to having employed automated feature engineering. Convolutional Neural Networks (CNNs) is one of the most used networks in computer vision tasks. They possess great power in self-learning and are used most heavily on image analysis tasks. Different from other methods, CNNs require only an image as input and they are able to do the classification on their own, based on their self-learning capabilities and training. However, in order for CNNs to be trained and provide accurate classifications, they need enough input sample data, which in some cases might be inadequate and insufficient. In this case, transfer learning comes into scene to exploit the potential of CNNs in classification.

For leukemia diagnosis, past studies have suggested that CNN architectures with many layers and depth levels can accurately perform leukemia detection. Moreover, it has also been stated that deep learning algorithms that have employed transfer learning are most widely used in leukemia detection and are proven to be of a high accuracy rate. Leveraging Convolutional Neural Networks (CNNs), Optimized Deep Recurrent Neural Networks (ODRNNs), and Transfer Learning, we can pave the way for a more accurate classification of blood cells and leukemia cancer diagnosis. CNNs are potent for classifying images of blood smears, thus differentiating and distinguishing between normal and leukemic blood cells. RNNs, alternatively, are potent in analyzing time-series data, deciphering behaviors and morphological changes captured in videos. As a complement to these two methods, Transfer Learning aims to exploit them by applying fine-tuning, especially in scenarios where labeled datasets are scarce.

In this thesis, we aim to make a comparison between the deep learning algorithms, CNNs, ODRNNs, and TL for an accurate identification of leukemia cells. By comparing the principles and methodologies of these algorithms, we aim to set the stage towards a more reliable and dependable framework for leukemia diagnosis.

1.1 Significance of the study

This research aims to enhance the accuracy of leukemia diagnosis by tackling deep learning algorithms, particularly Convolutional Neural Networks (CNNs), Optimized Deep Recurrent Neural Networks (ODRNNs), and Transfer Learning. By overcoming the limitations of manual microscopic examination, these algorithms have the capacity to reduce errors, and accurately predict the diagnosis. The insights gathered from this research have the potential to significantly impact healthcare, by accurately diagnosing leukemia in time and thus achieving a better quality of life for leukemia patients worldwide.

1.2 Research Objectives

To diagnose leukemia cells from blood smear images, the research objectives of this study are designed to compare and assess the effectiveness of Convolutional Neural Networks (CNNs), Optimized Deep Recurrent Neural Networks (ODRNNs), and Transfer Learning algorithms. The study tries to identify the advantages and disadvantages, strengths, and limitations of each technique through a comprehensive evaluation of algorithmic resilience across different scenarios and an examination of important performance indicators.

The research objectives of this study are as follows:

1. Evaluate the performance metrics, including accuracy and computational efficiency of the algorithms in diagnosing leukemia cells from blood smear images.
2. Compare the robustness of the algorithms across diverse datasets, including variations in sample sizes, leukemia cell morphologies and image qualities, to evaluate their applicability in real world.
3. Investigate the underlying features utilized for leukemia cell classification and the interpretability of the model predictions generated by CNNs, ODRNNs, and Transfer Learning algorithms.

1.3 Research Questions

The purpose of these research questions is to provide a better understanding of the differences between CNNs, ODRNNs and Transfer Learning algorithms. In order to examine the effectiveness and suitability of deep learning algorithms for leukemia detection, this study will embark on a comprehensive investigation.

1. In terms of efficiency, accuracy, and features utilized, how do Convolutional Neural Networks (CNNs), Recurrent Neural Networks (ODRNNs), and Transfer Learning methods compare when it comes to recognizing leukemia cells from blood smear images?
2. What factors contribute to the variability in performance and applicability of CNNs, ODRNNs, and Transfer Learning algorithms across image qualities, sample sizes, datasets, and leukemia cell morphologies, and how can these insights inform their utilization into clinical practice for leukemia diagnosis?

1.4 Theoretical Framework

This study's theoretical approach, which applies deep learning algorithms to leukemia detection, is based on several important advances in cancer, medical imaging, and artificial intelligence. Deep learning's theoretical foundations are based on concepts from neural network theory and machine learning. The main building blocks of this framework comprise Transfer Learning methods, Optimized Deep Recurrent Neural Networks (ODRNNs), and Convolutional Neural Networks (CNNs), each of which provides capabilities and methods for pattern recognition and image classification tasks. Furthermore, concepts from medical imaging are incorporated into the theoretical framework, specifically in regards to the interpretation of blood smear images for leukemia diagnosis. This includes understanding the morphological features of normal and abnormal blood cells.

Building on this theoretical foundation, computer vision research concerns help us comprehend the image processing and feature extraction methods used by deep learning algorithms. The analysis of blood smear images relies heavily on concepts

like image segmentation, edge detection, and feature representation, which enable the algorithms to detect small differences between leukemic and normal cells. To further clarify how deep learning algorithms make assessments and recognize important features in the input data, theories pertaining to model interpretation and explainability are integrated into the framework. This framework aims to clarify the details of how deep learning algorithms receive and interpret medical images, increasing trust in their diagnostic abilities.

CHAPTER 2

LITERATURE REVIEW

Leukemia is a blood and bone marrow cancer that presents a significant challenge for modern healthcare due to its variety of appearances, each with unique clinical presentations and prognoses. Its importance in healthcare is derived from its frequency as well as its complexity and severity of its effects. Leukemia is one of the most prevalent cancers in both adults and children, and it has a substantial impact on people's lives, families, and healthcare systems across the globe. In order to effectively manage and treat leukemia, early detection is essential since prompt action can greatly enhance patient outcomes and raise the chance of a successful recovery. Because leukemia is an aggressive illness with a high potential for rapid progression, the ability to detect the presence of cancer cells at an early stage is crucial.

Being identified early lowers the risk of complications and increases overall survival rates by allowing healthcare providers to rapidly undertake appropriate treatment measures. Under these circumstances, the use of cutting-edge technology like deep learning algorithms to the field of medical picture analysis has great potential to improve the effectiveness and precision of leukemia detection. Artificial intelligence's subset of deep learning has become a potent tool in the medical profession, especially for medical imaging. Deep learning algorithms can help doctors detect, diagnose, and characterize a variety of diseases, including leukemia, by using sophisticated neural networks to extract valuable information from massive amounts of medical picture data.

Deep learning algorithms can help identify abnormal cell morphology and patterns indicative of leukemia by analyzing digital images from medical examinations such as bone marrow, blood smears, and imaging scans. Furthermore, these algorithms' capacity to pick up new information and learn from large datasets could improve diagnostic speed and accuracy, which would ultimately benefit patients. Deep learning's potential in medical image analysis is still being explored by researchers,

but incorporating these cutting-edge tools into clinical practice opens up new possibilities for improving leukemia patients' prognosis and quality of care via early diagnosis and treatment.

2.1 Leukemia Cancer Cell Classification Techniques

In recent decades, leukemia cancer cell classification techniques have evolved significantly, moving from manual methods to automated approaches. One major factor contributing to this progress has been the development of deep learning algorithms. Historically, flow cytometry and manual microscopy were the traditional approaches for leukemia cell classification. Manual microscopy is the process by which skilled pathologists or laboratory staff visually inspect stained blood smears or bone marrow aspirates under a microscope. Despite being widely utilized for decades, this approach is subjective by nature and prone to variability among observers because it depends on the expertise, skills and personal judgment of the interpreter. Contrarily, flow cytometry uses fluorescently labeled antibodies to detect leukemia cells due to their surface markers.

Traditional methods for detecting leukemia cells are still in use, although they have several drawbacks. Their subjectivity is one of the main disadvantages, as it may result in inconsistent findings and possible errors in diagnosis. Moreover, accurate performance and interpretation of the tests require competent workers, as manual microscopy and flow cytometry are demanding and time-consuming procedures. Furthermore, these methods could not be sensitive or precise enough, especially when we are dealing with early stages of the disease or insufficient sample sizes.

Recently, a rise in demand has been seen in using automated techniques, such as deep learning algorithms, to detect leukemia automatically and correctly. Deep learning algorithms are a branch of artificial intelligence that draw inspiration from the architecture and operations of the human brain. These algorithms have proven to be highly effective in deciphering complex datasets. Through the training of deep neural networks on vast amounts of labeled data, scientists are able to create models that can accurately and efficiently identify leukemia cells automatically.

Deep learning's use into leukemia cell identification has enormous potential to address the drawbacks of traditional techniques. Digital images of blood and bone marrow samples can teach deep learning algorithms complex patterns and features, allowing for automated leukemia cell analysis. Furthermore, as opposed to traditional techniques, these algorithms might potentially increase the precision of results, minimize response times, and improve diagnostic accuracy.

2.2 Overview of Deep Learning Algorithms for Medical Images Classification

As deep learning algorithms can automatically extract complex patterns and characteristics from massive amounts of data, they have become highly effective tools in many fields, including medical image analysis. Optimized Deep Recurrent neural networks (ODRNNs), convolutional neural networks (CNNs), and transfer learning models are some of the most popular deep learning algorithms. Each has particular advantages for evaluating medical images along with rendering tasks including disease diagnosis and image classification simpler.

Tasks involving time-series data are ideally suited for Optimized Deep Recurrent Neural Networks (ODRNNs), a class of neural networks specifically created to handle sequential data. The capacity of recurrent connections to preserve memories of previous inputs gives ODRNNs the ability to identify temporal dependencies in the data. ODRNNs have been effectively used in a variety of healthcare applications, such as medical signal processing, where sequential data is essential for diagnosis (Lipton et al., 2015). For the analysis of dynamic medical data, involving patient health metrics, ODRNNs are especially well-suited due to their natural temporal modeling skills.

Convolutional neural networks (CNNs), on the other hand, are specialized architectures built to handle data that resemble grids, like images. CNNs can understand the spatial patterns and structures present in the data by automatically extracting hierarchical features from input photos through the use of convolutional layers. CNNs have proven to be remarkably efficient in image analysis tasks, such as segmenting and classifying medical images. CNNs have demonstrated encouraging

outcomes in a variety of medical imaging modalities, including computed tomography (CT) and magnetic resonance imaging (MRI) (Sistaninejhad et al., 2023).

Another important deep learning topic that has attracted a lot of interest in medical image classification challenges is transfer learning. Utilizing information from one task or area to enhance learning and performance on a related but distinct area is known as transfer learning. Transfer learning helps researchers make greater generalization and performance with less labeled data by optimizing pre-trained models on a target dataset. Transfer learning has been successfully used in a variety of medical imaging tasks, including as lung nodule recognition in chest radiographs and retinal lesion segmentation in fundus images, as noted by Tajbakhsh et al. (2016).

2.3 Application of Deep Learning in Leukemia Detection

The effectiveness of Optimized Deep Recurrent Neural Networks (ODRNNs) in leukemia cell detection has been examined in a number of research. For instance, Logeswari et al.'s (2024) investigation assessed how well ODRNNs performed sequential data analysis from peripheral blood smears in order to identify leukemia cells. Through the utilization of ODRNNs' temporal modeling capabilities, the researchers were able to obtain encouraging outcomes in precisely identifying abnormal cell morphology linked to leukemia.

Convolutional neural networks (CNNs) have also been extensively used in addition to ODRNNs for the identification of leukemia cells in images. In order to create an automated method for the detection and categorization of leukemia cells in microscopic pictures of blood smears, Tran et al. (2018) carried out research. The researchers were able to obtain great accuracy in distinguishing between normal and abnormal cells by training CNNs on a huge dataset of annotated photos. This would enable early diagnosis and therapy.

Additionally, by utilizing information from previously trained networks, transfer learning techniques have been used to increase the accuracy of leukemia detection models. In this regard, Loey et al. (2020) investigated the application of transfer learning to leukemia detection tasks in medical picture classification. Compared to

training from scratch, the researchers' method of fine-tuning pre-trained CNN models on leukemia-specific datasets resulted in notable gains in classification accuracy.

2.4 Performance Evaluation Metrics of Deep Learning Algorithms for Leukemia Detection

When evaluating the efficacy and dependability of deep learning models, performance evaluation measures are essential, especially when dealing with tasks like leukemia cell detection in medical image analysis. Metrics that are frequently used to assess a model's performance quantitatively include accuracy, precision, recall, and F1-score. As a general indicator of the correctness of the model, accuracy is defined as the percentage of correctly classified instances relative to the total number of instances. Contrarily, precision measures how many true positive predictions there are out of all positive predictions, demonstrating how well the model avoids false positives. Recall, also referred to as sensitivity, expresses how well the model can identify all relevant situations by calculating the percentage of genuine positive predictions among all actual positive instances. A balanced indicator of a model's performance, the F1-score, which is the mean of precision and recall, is especially helpful in situations where class distributions are unbalanced (Terven et al., 2023).

Performance metrics like sensitivity and specificity are particularly significant with regard to leukemia cell detection. The percentage of true positive predictions among all actual positive instances is known as sensitivity, or true positive rate, and it indicates how well the model can identify leukemia cells. Contrarily, specificity indicates how well the model is able to exclude out non-leukemia cells by calculating the percentage of true negative predictions among all actual instances. In order to accurately identify both positive and negative cases and reduce false positives and false negatives, leukemia detection requires striking a balance between sensitivity and specificity (Terven et al., 2023).

While performance evaluation metrics have been made accessible, there are a number of obstacles and factors to take into account when evaluating a model's performance in medical image analysis jobs. One issue that causes discrepancies in reported performance among studies is the absence of consistent benchmark datasets

and evaluation methodologies (Maier-Hein et al., 2018). Furthermore, to guarantee trustworthy performance in real-world scenarios, the robustness and generalization of deep learning models across various imaging modalities and patient populations need to be rigorously assessed (Maier-Hein et al., 2018).

2.5 Datasets and Benchmarks for Leukemia Detection

Standardized data for training, validating, and testing algorithms can be obtained by researchers through publicly available datasets. These datasets for the diagnosis of leukemia are useful tools for scientists who want to assess deep learning algorithms. The Blood Cell Image Dataset (BCID) is one example of a dataset that includes microscopic images of blood smears that were taken using several imaging modalities (Kather et al., 2019). The BCID dataset offers a comprehensive resource for leukemia detection studies, with annotated images of normal and abnormal leukocytes among other blood cell types.

An essential component of evaluating the effectiveness and performance of deep learning models for leukemia diagnosis is benchmark studies, which serve to contrast various algorithms using standardized datasets. For instance, Liu et al. (2019) carried out a benchmark study to assess how well several deep learning architectures — such as recurrent neural networks (RNNs) and convolutional neural networks (CNNs) — performed on the ALL-IDB dataset. The study assessed the algorithms according to criteria including specificity, sensitivity, and accuracy; this revealed important insights into the advantages and disadvantages of various leukemia detection techniques. Similar to this, (Kather et al., 2019) revealed how pre-trained models are useful in enhancing classification accuracy by conducting a benchmark study to examine the performance of transfer learning models on the BCID dataset.

Deep learning research in leukemia diagnosis must take dataset selection and preprocessing into account. To guarantee representative and trustworthy training data, researchers should take into consideration several aspects when choosing datasets, including data variety, sample size, and annotation quality. Deep learning models' resilience and generalization can only be improved by performing preprocessing operations such image normalization, augmentation, and noise removal. Wang et al.

(2022), for instance, stressed the value of preprocessing methods such color normalization and contrast enhancement in images in enhancing the effectiveness of deep learning models for leukemia diagnosis.

2.6 Comparative analysis of ODRNNs, CNNs, and transfer learning algorithms for leukemia detection

In order to provide insights to a reliable diagnosis of leukemia, Optimized Deep Recurrent Neural Networks (ODRNNs), Convolutional Neural Networks (CNNs), and Transfer Learning algorithms are compared. This comparison sheds light on the advantages, disadvantages, and possible uses of each technique, thus helping researchers and healthcare providers choose the best algorithms for leukemia detection tasks.

Numerous studies focused on leukemia detection have assessed and contrasted ODRNNs, CNNs, and transfer learning algorithms. Kim et al. (2022) utilized a dataset of leukemia cell images to compare how well ODRNNs, CNNs, and transfer learning models performed. The study discovered that whereas transfer learning models produced outcomes that were equivalent with less training time, CNNs surpassed ODRNNs in terms of accuracy and speed of computation.

The benefits and drawbacks of each strategy in terms of accuracy, computational efficiency, and interpretability differ based on the specifics of the assigned task and dataset. ODRNNs are ideally suited for tasks involving time-series data or sequences of different lengths since they are adept at processing sequential data and understanding temporal dependencies (Liu et al., 2019). Nevertheless, during training, ODRNNs may experience vanishing issues, which restricts their capacity to identify long-term dependencies in sequential data (Pascanu et al., 2013). Conversely, CNNs are exceptionally proficient at removing spatial features from images, thereby making it possible to analyze medical images precisely and quickly (Ronneberger et al., 2015). CNNs may, however, have difficulties capturing temporal information and may need an excessive amount of labeled data in order to be properly trained.

2.7 Identification of Challenges, Gaps and Areas for Future Research

Two major obstacles to using deep learning for leukemia detection are the lack of available data and the interpretability of the models. Ineffective model training and evaluation are caused by limited access to annotated datasets, particularly for rare leukemia subtypes, which may result in overfitting or poor performance (Liu et al., 2019). Furthermore, interpretability issues with deep learning models pose a serious challenge in healthcare situations where clear decision-making is crucial. To overcome these obstacles, cooperative efforts are needed to create interpretable deep learning models suited to leukemia detection requirements and to collect extensive annotated datasets. Improving model robustness and dependability in clinical applications requires integrating domain expertise and knowledge.

Advancement in the field of comparative analysis of deep learning techniques for leukemia diagnosis requires the identification of research gaps and areas for future research. The creation of hybrid models, which combine the advantages of ODRNNs, CNNs, and transfer learning techniques to produce better performance and generalization, is one topic of future research. Studies on the interpretability and explainability of deep learning models for leukemia diagnosis are also necessary because these models' capacity to produce results that can be understood is a prerequisite for clinical acceptability and confidence (Lundberg et al., 2018). Additionally, investigating the integration of multi-modal data sources, including clinical factors and genetic data, may improve the precision and dependability of leukemia detection algorithms.

CHAPTER 3

OVERVIEW OF CNN, TRANSFER LEARNING AND ODRNN ALGORITHMS

Leukemia diagnosis requires blood tests and bone marrow biopsies, which require manual labor and are time consuming. Therefore, the creation of an automated technology for the identification of white blood cell cancer is necessary. a classification model based on deep learning techniques and convolutional neural networks is presented in this work. The C_NMC_2019 cancer cell dataset, which consists of segments of white blood cells extracted from microscopic blood smear images, was used to train and assess this model. The model provides a satisfactory accuracy of 87% for testing and 91% for training.

According to the Mayo Foundation for Medical Education and Research, cancer is the second leading cause of mortality (MFMER). As the global population grows, so does the number of cancer-related deaths. Early cancer diagnosis is necessary to lower the death toll. The challenge is that a significant number of people fail to exhibit the early signs and symptoms of leukemia. Aspiration of the bone marrow and blood tests that reveal abnormal White Blood Cell counts are the medical procedures used to classify leukemia. The analysis of these tests is required, and it takes time considering the physicians must examine each blood sample separately. When prompt detection is crucial, this process needs to be enhanced. As a result, an automated method for leukemia detection is essential. Leukemia detection models could be constructed using image processing techniques provided by deep learning packages such as Keras. With the help of this approach, the disease should be identified promptly and without the limitations of visual assessments.

3.1 Overview of Convolutional Neural Networks

The most popular deep learning model for classifying images is the Convolutional Neural Network model. As seen in Figure 2, a CNN model typically encompasses three main layers: the Input layer, Hidden layers, and Fully Connected layers. The Convolutional Neural Network is comprised of four parts: Convolutional Layers, Rectified Linear Unit (ReLu), Pooling layers, Fully connected layers.

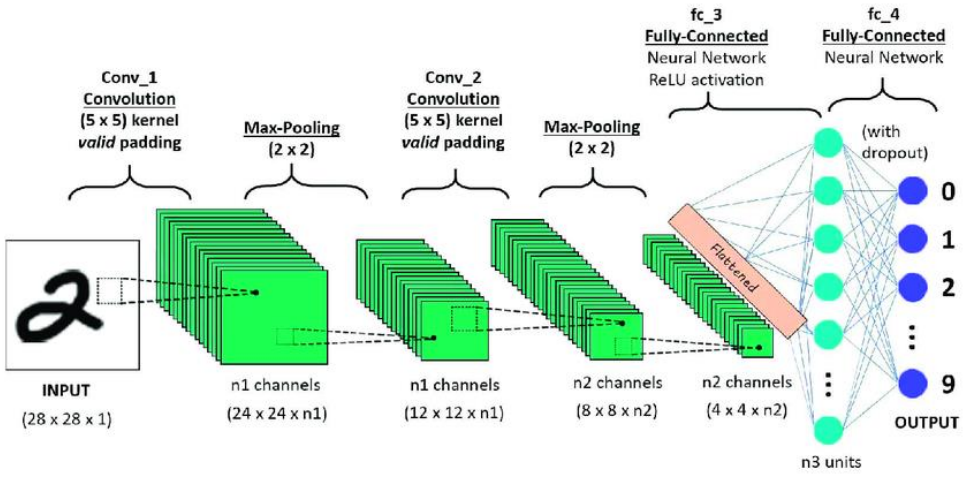

Figure 2: Convolutional Neural Networks architecture

Convolutional Layer is CNN's initial building block. Convolution, as the name implies, is the primary mathematical operation carried out; it involves applying a sliding window function to a matrix of pixels that represents a single image. Either kernel or filter refers to the sliding function that is applied to the matrix; the terms can be used interchangeably. In the convolutional layers of a CNN, the weights are represented by learnable filters or kernels. Each filter is a small matrix of weights that is convolved with the input data to extract features. During training, the network learns the values of these filter weights through backpropagation, adjusting them to minimize the difference between the predicted and actual outputs. Several equal-sized filters are applied in the convolution layer, and each filter is utilized to identify a certain pattern from the image, including the edges, curves and shapes.

Putting it simply, we employ tiny grids (also known as filters or kernels) that travel across the image in the convolution layer. Every tiny grid functions as a tiny magnifying glass, searching the image for particular patterns such as curves, shapes, or lines. It makes a new grid as it travels across the image to show where it located these patterns. For instance, different filters may perform better at identifying curves than straight lines, and vice versa. The CNN is able to obtain a clear understanding of all the various patterns that comprise the image by employing multiple filters.

For training CNN, backpropagation algorithm is employed. Convolutional neural networks (CNNs) employ backpropagation as a crucial training procedure to learn from data and gradually enhance their performance. The following is a thorough explanation of backpropagation's operation in CNN training:

1) Forward Pass: Input data, such images, are fed into the CNN during this phase.The network's input data flows through a number of layers, including pooling layers, convolutional layers, and activation functions as ReLU. For the purpose of the task at hand, like as object detection or image classification, each layer modifies the input data in a way that makes it increasingly more informative.
2) Loss Calculation: Following the forward pass, the network's output is examined against the ground truth labels. To measure the discrepancy between the actual labels and the predicted output, a loss function is computed, which in the case of classification tasks, the most commonly used is cross-entropy loss function. Backpropagation aims to reduce this loss function, which will increase the network's prediction accuracy.
3) Backpropagation: This technique entails calculating the gradients of the loss function in relation to the network's parameters. Gradients are calculated layer by layer using the calculus chain rule, starting at the output layer and working backward through the network. The gradients at each layer show the corresponding contribution of each parameter to the network's overall inaccuracy. The network's parameters are then updated using these gradients in a way that minimizes the loss function. Optimization algorithms such as stochastic gradient descent (SGD) are commonly used for this. One significant

hyperparameter that influences the convergence and stability of the training process is the learning rate, which controls the magnitude of the parameter updates.

4) Iterative Optimization: Throughout several epochs, the backpropagation procedure is done repeatedly until it progresses through the full dataset. The network takes batches of training data, computes gradients, and applies optimization methods to change parameters at the end of each epoch. By modifying its parameters to minimize the loss function, the network gains the ability to anticipate outcomes more accurately over time. Training doesn't cease until a predetermined threshold is reached, such as finishing a validation set with a suitable performance level or completing a maximum number of epochs.

Backpropagation is a crucial algorithm for CNN training since it enables the network to learn from data by repeatedly modifying its parameters in order to reduce the discrepancy between expected and actual outputs. The forward pass, loss computation, and backward pass allow CNNs to perform better on a variety of tasks, such as image classification and object detection.

Following every convolution operation, a **ReLU activation function** is performed. By teaching the network non-linear correlations between the image's features, this function strengthens the network's ability to recognize various patterns. The ReLU activation function is expressed as follows: $f(x)=\max(0,x)$. If input x is positive, it returns the input x; if not, it returns zero. When the input to a neuron is positive, ReLu keeps the output unchanged, otherwise when the input to a neuron is negative, ReLu sets the output to zero.

Pooling layer aims to extract the most important features from the complex matrix. This is accomplished by using a few aggregation processes, which decrease the feature map's (convoluted matrix) dimension and, as a result, the amount of memory needed for network training. The most common aggregation functions that can be applied are:

a) Max pooling, which is the maximum value of the feature map,
b) Sum pooling corresponds to the sum of all the values of the feature map,

c) Average pooling is the average of all the values.

In the **fully connected layers** of a CNN, each neuron is connected to every neuron in the previous layer. The strength of the connections and the way information spreads throughout the network are determined by the weights associated with these links. In order to recognize intricate relationships in the data and produce precise predictions, the network must learn the values of these weights during training.

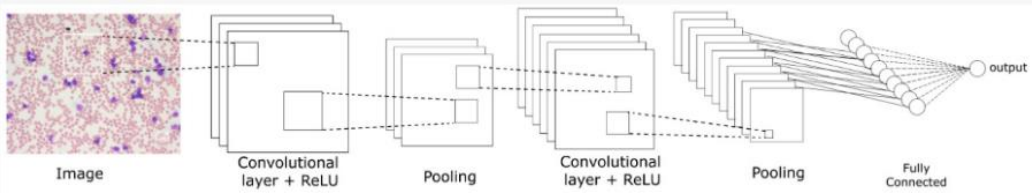

Figure 3: Architecture of CNNs

3.2 Convolutional Neural Networks for Leukemia Classification

Using Convolutional Neural Networks (CNNs) for leukemia classification is a breakthrough in medical diagnostics. CNNs are exceptionally effective at extracting characteristics from large, complicated image data, which makes them useful for tasks as identifying between several leukemia cell types. CNNs can be trained to precisely classify cells into distinct leukemia subtypes by utilizing big datasets of labeled leukemia images. This can help clinicians with diagnosis and treatment planning. CNNs can extract hierarchical representations of leukemia cells by utilizing many layers of convolutions, activation functions, pooling, and fully linked layers. This allows them to capture complex patterns and structures that may indicate distinct subtypes. CNN-based leukemia classification systems have the potential to improve patient outcomes, increase diagnostic accuracy, and deepen our understanding of the disease with additional developments in CNN structures, optimization algorithms, and data augmentation approaches. The research literature on the previous section shows that there is a scope for development of a novel classifier for Leukemia detection. In this part, we report on a novel approach to early blood leukemia diagnosis using deep

neural networks, specifically CNN, using microscopic blood smear images.

3.3 Dataset and Tools

The dataset includes 150 normal photos and 480 cancer images total in this 630-image collection. Images are retrieved online and are available in ASH Image Bank Hematology, ALL IDB, Atlas of Hematology, Atlas of Blood Smear Analysis.

3.4 Data Preprocessing

1) Remove duplicates

   The dataset was gathered from a variety of sources, and it was discovered that there were some repeats, that some images had watermarks, and that some images contained the logos of websites. In all, roughly 43 images were excluded from the dataset, making it 587 images in total.

2) Image resizing

   A scaling strategy was used to make all of the photos in the dataset 256 x 256 pixels in order to shorten the training time because the dataset has a diverse size distribution, and it was necessary to make all of the images in the dataset have the same size for training the CNN model.

3) Image filtering

   Prior to the processing phase, images must be cleaned up of noise and have their line patterns improved. This can be done by using a median filter (3 x3) and also by sharpening the image (3 x3).

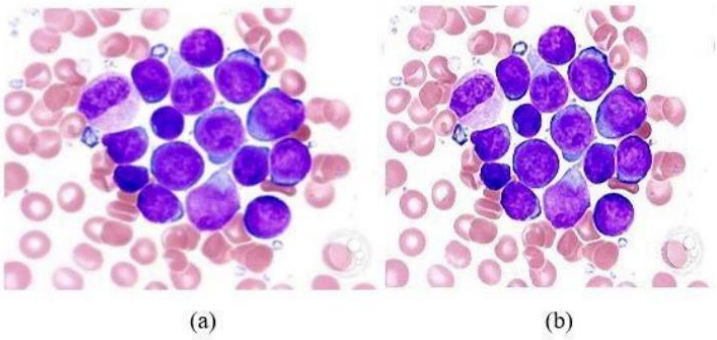

Figure 4: *(a) original image, (b) image resized by 256*256 which has been filtered by median and sharpen filters*

4) Data Augmentation

Using image data augmentation, one can create generating modified copies of the dataset's images in order to expand a training dataset. Deep-learning neural network models may be trained on more data to increase their ability to fit models and generalize what they have learned to new images. Moreover, variations of the images can be produced using augmentation techniques. The Keras deep learning neural network framework can fit models by adding image data through the ImageDataGenerator class. There are numerous varieties of augmentation methods, including the following:

a) Flipping

Reversing the rows or columns of pixels in a vertical or horizontal flip is referred to as an image flip.

b) Horizontal and Vertical Shift

When an image is shifted, all of its pixels are moved in a single direction—for example, vertically or horizontally—while maintaining the same image dimensions. This results in some pixels being clipped off the image, and there will be a section where new pixel values must be specified.

c) Random Zoom Augmentation

A zoom augmentation increases the image's size and adds new pixel values or interpolates existing ones around it.

d) Shearing

In order to create a picture without any padding or black space, shearing will automatically clip the appropriate portion from the sheared image.

e) Interpolation

Interpolation is a method for generating new data points within the range of a discrete set of existing data points. The easiest method of interpolation is nearest neighbor interpolation. This approach just finds the "nearest" neighboring pixel and assumes its intensity value, as opposed to calculating an average value using some sort of weighting criterion or producing an intermediate value based on complex procedures. Additionally, an augmented image is shown with the sample image in Figure 5.

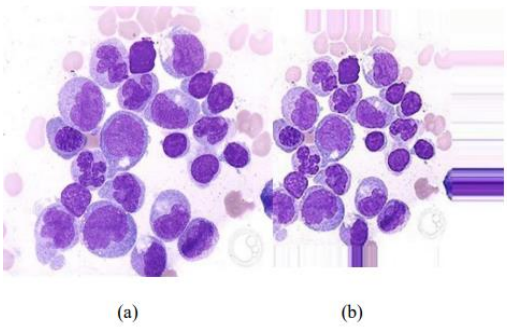

Figure 5: (a) original image and (b) augmented image

3.5 Processing Stage

Following augmentation procedures, our data are transformed into 1480 normal images and 1550 cancer images. The data is split into three data sets—a training set (60%), a validation set (20%), and a test set (20%) in order to fit the data to the models.

Training the model to be able to classify the images is the next step. Accuracy and test accuracy are our optimal parameters. The model used for classifying leukemia cells is BasicCNN Model. The input images for this model have a resolution of 128 x 128 pixels and are color (RGB) images. Three convolutional layers with max pooling layers make up this structure. Every convolutional layer is followed by a rectified linear unit (ReLu). Fully connected layers are employed trained for two categories classification using the sigmoid activation function, a constant filter size (3x3), the number of filters (128), and the stride of ones (equal 1). After 17 epochs, the obtained accuracy is 90.99% and test accuracy is 80.91%, when classifying the data set into leukemia cells or normal cells.

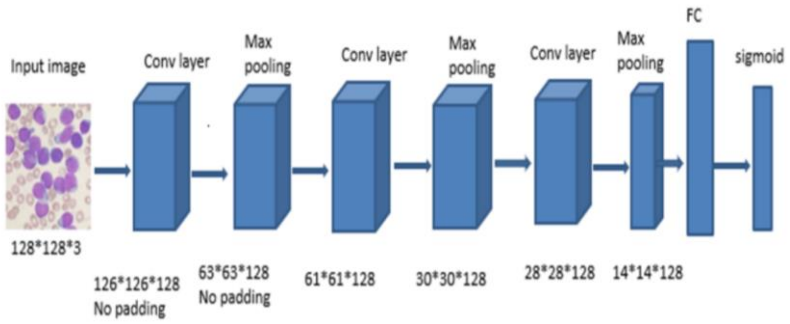

Figure 6: Basic CNN Model diagram

3.6 Results

In total for this experiment, 3030 photos were used of which 60% (1818 images) were used for training, 20% (606 images) for validation, and the remaining 20% (606 images) for model testing. In order to evaluate the performance of the model, we have a look at the statistically measured parameters.

1) Accuracy

After 17 epochs, the train accuracy for the Basic CNN model is 90.99%; as Fig. 7 illustrates, our leukemia classifier is performing exceptionally well in terms of cell classification. Figure 3.6 illustrates that the validation accuracy of the Basic CNN Model approaches 85% after 17 epochs. For new data, thus, we anticipate that our model will function with approximately 85% accuracy.

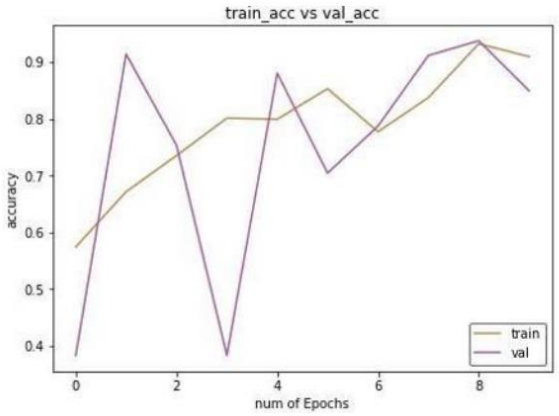

Figure 7: *Validation accuracy and train accuracy for Basic CNN Model*

2) Confusion Matrix

A confusion matrix compiles the expected results of a classification problem. For each class, the count values indicate the proportion of precise and imprecise forecasts. This is an excellent solution for presenting results in M-class classification problems because it is possible to show the relationships between the classifier outputs and the actual ones. According to Fig. 8, for the basic CNN model, there are 372 leukemia images that are predicted to be leukemia, 8 leukemia images that are predicted to be normal, 269 normal images that are predicted to be normal, and 51 normal images that are predicted to be leukemia. These accuracy levels indicate that this model performs very well when it comes to leukemia prediction, but poorly at predicting normal images.

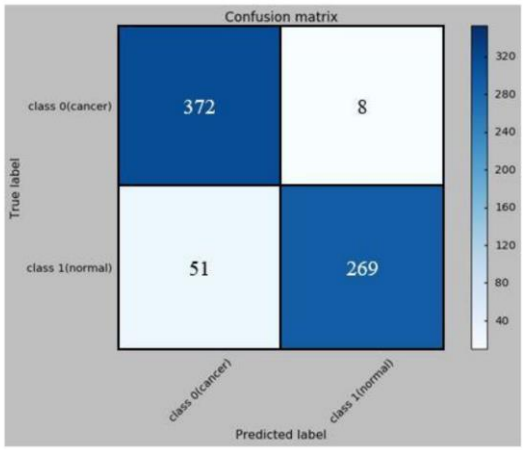

Figure 8: Confusion Matrix for Basic CNN Model

3) Precision

Precision is a performance metric that measures the percentage of positive identifications that were in fact accurate. The ratio of true positives to the total of true positives and false positives is how it is defined. As illustrated in Fig. 9, the precision of our Basic CNN model is average.

4) Recall

It is calculated by dividing the entire number of relevant samples—that is, all samples that ought to have had a positive label—by the total number of reliable positive results. As illustrated in Figure 9, in class 1, the first CNN model has a poor precision but a high recall. This indicates that there are many false positives but that the majority of positive examples are appropriately identified (low FN). However, in class 0, low recall and high precision indicate that we miss a large number of positive cases (high FN), but those we anticipate to be positive are in fact positive (low FP).

5) F1 Score

The F1 score is the harmonic mean of recall and accuracy. The range of the F1 score is [0, 1]. It indicates the classifier's accuracy (the number of instances it properly classifies) and how robust it is (a significant number of instances are recognized by it). As shown in Figure 9, the F1 Score of our model appears to be average, that means, it averagely identifies instances correctly.

6) Support

The number of samples that accurately reflect the response within that category is known as support. It offers details on the exact counts of every class in the test data.

	precision	recall	f1-score	support
class 0(Cancerous)	0.98	0.62	0.76	78
class 1(Normal)	0.84	0.99	0.91	163

Figure 9: Values of precision, recall, f1 score and support for our CNN model

3.7 Discussion

Leukemia attacks the body's blood-forming tissues, such as the bone marrow and lymphatic system. We use CNN's power to deploy the CNN Basic Model that classifies blood smears into normal and abnormal categories in order to provide the patient with the most effective treatment possible. Our dataset needed to be larger in order to be used with Deep Learning, so data augmentation is employed to solve this issue; this worked well for us because the 260 photos in our data before augmentation had been increased to 3030 images after augmentation. The model is trained using CNN, and our optimal parameters were accuracy and validation accuracy. Basic CNN Model is comprised of three convolutional layers. We have not received very good predictions from this model and the outcomes were not very accurate, due to the low number of

convolutional layers. However, the technology successfully identifies leukemia early, and processes data quickly by showing results in less than 30 seconds.

The following components comprised the detection system's design:

a) The acquisition component, which is made up of a digital camera placed above the microscope's eyepiece;
b) A pre-trained CNN model in charge of the classification system;
c) A graphical user interface that shows the image that was taken from the camera and the categorization results.

3.8 Overview of Transfer Learning

Transfer learning is a technique that involves using a model that has been trained on one task as the foundation for a model on a different task. When there is little data available for the second task or when the tasks are very similar to each other, this can be beneficial. The model can learn more quickly and efficiently on the second task by starting with the learnt features from the first task. Because the model will have already learnt general features that will probably be helpful in the second task, this can also assist prevent overfitting.

How Transfer Learning works? Here is a general overview:

1) Pre-trained Model: Begin with a model that has already undergone extensive training on an extensive dataset for a particular task. This model has been trained on large datasets on a regular basis, and it has found common features and patterns that apply to many comparable tasks.
2) Base Model: The pre-trained model is referred to as the base model. It consists of layers that have learned hierarchical feature representations by using the input data.
3) Transfer Layers: Locate a set of layers in the pre-trained model that capture general data pertinent to both the new and past tasks. These layers are usually

found near the top of the network since they are prone to learning low-level information.

4) Fine-tuning: The process of retraining the selected layers utilizing the dataset from the new challenge. The aim of fine-tuning method is to preserve the pre-training knowledge while allowing the model to adjust its parameters to better meet the requirements of the current assignment.

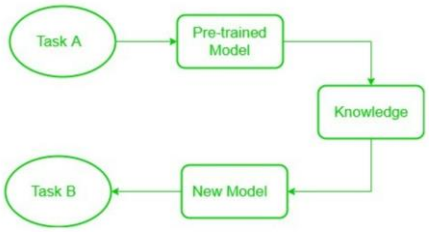

Figure 10: Transfer Learning Overview

The essence of transfer learning is that learning low-level features for task A helps with learning the model for task B. In the modern era, it is rare to find individuals training an entire convolutional neural network from scratch. Instead, it is typical to take a pre-trained model and apply its features to a new task. When working with Transfer Learning, we come across a term which is the freezing of layers. When a layer—which could be a CNN layer, hidden layer, block of layers, or any subset of all layers—cannot be trained, it is said to be fixed. Therefore, throughout training, the freeze layer weights won't be updated. Non-frozen layers adhere to standard training practices. We choose a pre-trained model as our basis model when applying transfer learning to the problem at hand. Using the pre-trained model's knowledge can now be done in two different ways. One approach is to freeze a few layers of the pre-trained model and then subsequently train the other layers on our new dataset for the new task. The second method involves creating a new model and then using some of the features extracted from the pre-trained model's layers to use them in the new model. In both scenarios, we attempt to train the remaining portion of the model after removing some of the learnt features. This ensures that the single feature that might be the same in both tasks is removed from the pre-trained model, and the remaining features are then trained to fit the new dataset.

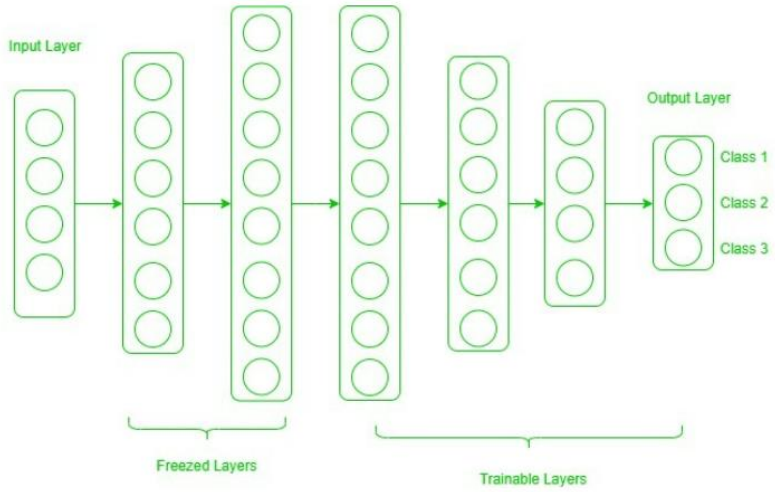

Figure 11: Freezed and Trainable Layers

In this case, a question can arise as to how we decide which layers require training and which require freezing. The simple answer is that you need to freeze layers more in order to inherit more characteristics from a pre-trained model. For example, if we have a pre-trained model that identifies some types of flowers and we need to detect some new creatures or species, in such case, we will have a new dataset with the new species that contains much features that are similar to the pre-trained model. Therefore, since the features between two models are quite similar, in order to utilize the most of its expertise in a new model, we freeze fewer layers. Now let's look at an alternative scenario: let's say we have a pre-trained model that can identify individuals in photos. If we want to use that knowledge to identify cars, however, the dataset in this scenario is completely different, so it is not a good idea to freeze a lot of layers because doing so will produce high-level features like eyes, lips, head, and noses that are useless for the new dataset of cars detection. Therefore, we train the entire model on a new dataset and just replicate low-level characteristics from the base model.

The backpropagation technique is a key mechanism in transfer learning which enables you to make use of the knowledge that pre-trained models have to offer to improve performance on an intended task. Initially, a huge dataset like ImageNet is used as the source task for a pre-trained model, which is used for image classification.

In order to reduce the error between the model's predictions and the ground truth labels, backpropagation adjusts the model's parameters during training. Backpropagation is frequently used to adjust and fine-tune the pre-trained model's parameters before deploying that knowledge to a target task. In this case, the backpropagation technique is used once more, but at a lower learning rate. This enables the model to preserve the knowledge it acquired from the source task while adjusting its parameters to the specifics of the target task. Transfer learning using backpropagation is a useful strategy when labeled data is scarce because it makes use of the features that were acquired and learned during pre-training, greatly reducing the quantity of labeled data needed for training on the target task. Furthermore, backpropagation enables the pre-trained model to be selectively frozen or fine-tuned at individual layers, providing flexibility in modifying the model architecture to fit the requirements of the target task and dataset.

3.9 Transfer Learning for Leukemia Classification

In place of more cumbersome traditional methods, this section suggests two automated classification models based on blood microscopic images that use transfer learning to identify leukemia. The first model pre-processes images of blood microscopic images and then a pre-trained deep convolutional neural network called AlexNet extracts features and classifies them using a variety of well-known classifiers. In the second model, AlexNet is optimized for both feature extraction and classification following the pre-processing of the images. Based on the tests that were run on a dataset consisting of 2,820 photos, we will notice that the second model outperforms the first due to its 100% classification accuracy. For both models, transfer learning was used, using pre-trained models. The time and effort required to create and train these networks from scratch is cut down by transfer learning. Transfer learning is applied in two ways. The first technique is using the values of the net's final fully connected layer (FC) to extract features from the input photos prior to employing another classifier. The second technique entails removing the high-level layers from the network to change its structure, which is a concept known as network fine-tuning.

The first classification model consists of three activities, which are feature extraction, classification, and image pre-processing, as illustrated in Figure 12. Pre-

processing involves a number of tasks, including scaling the images to fixed sizes, doing data augmentation to make up for the absence of huge datasets, and transforming blood images into a red-green-blue (RGB) model. Feature extraction involves taking each image and extracting a collection of features using a pre-trained AlexNet so that it may be used in classification to distinguish between photos of healthy and unhealthy cells. Many popular classifiers are used in the classification, including SVMs, K-NNs, linear discriminants and decision trees.

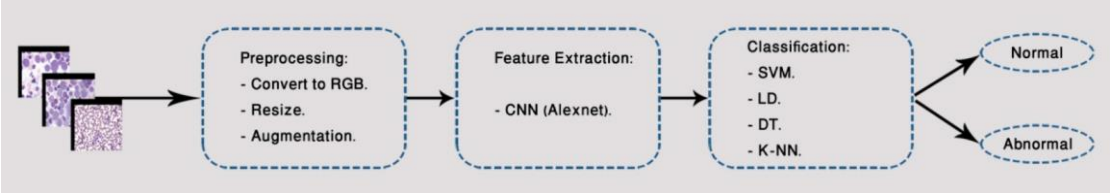

Figure 12: Diagram of the first classification model

In the image preprocessing step, the blood microscopic images are first transformed into RGB before a variety of processes are applied. After then, their dimensions are set to 227 x 227. Finally, since deep neural networks need large datasets to complete their training and testing phases, data augmentation is done to offset the lack of a large dataset. Three operations make up data augmentation: translation, reflection, and rotation. The images are translated by shifting them along the X and Y axes, with a random selection of values that are contained within the interval [15–25]. The images are mirrored along the vertical axis during the reflection process. Lastly, a random rotation angle of values bounded by the interval [25–125] is used to rotate the images right or left during the rotation procedure. Figure 13 provides examples of data augmentation on columns (a-d) which specify: original, translated, reflected, and rotated images.

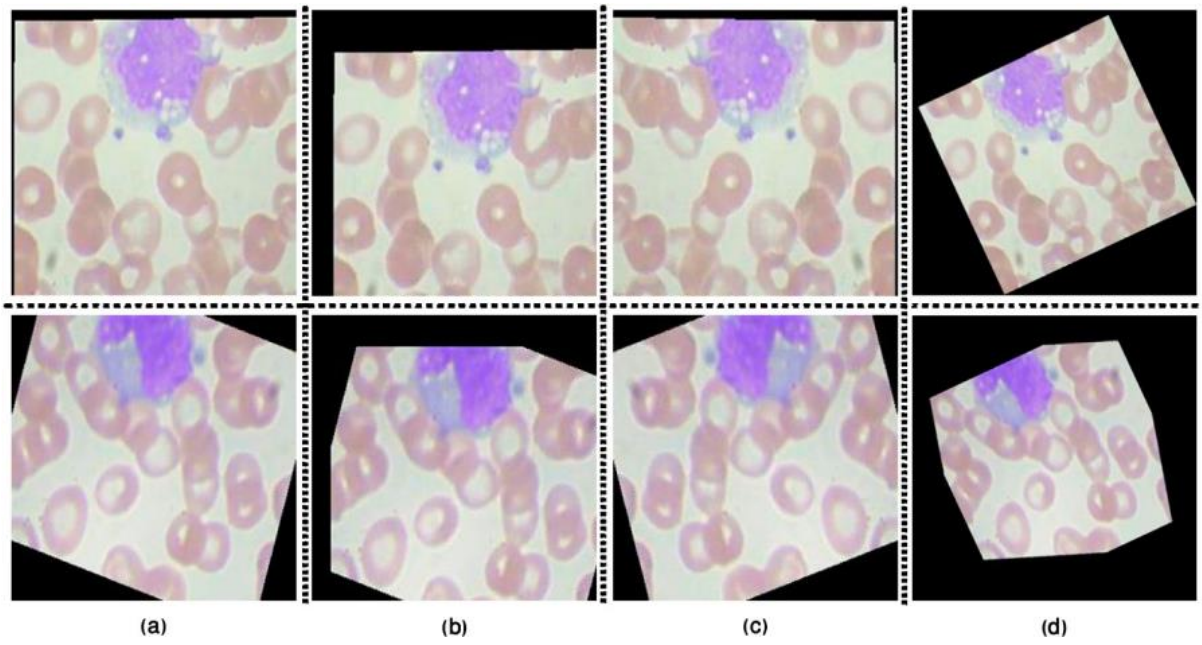

Figure 13: Data Augmentation images specifying original, translated, reflected, and rotated images

The feature extraction step is implemented with the help of AlexNet. In addition to three fully connected layers and five convolutional layers (which should be trained), AlexNet also has max-pooling layers. Several techniques are used in AlexNet to alleviate the overfitting issue, such as data augmentation and the dropout approach, which sets the output of hidden neurons to zero with a probability of 0.5. The first two fully connected layers experience dropout. Features are extracted from our initial model by calculating the values of the last fully linked layers with the feature vector length equal to 4,096.

The feature vectors from the previous stage were divided into two classes: healthy and unhealthy, using a variety of classifiers. Decision Trees with max-split of 20, Linear Discriminants, Support Vector Machines with various kernel functions, and K-NN with Euclidian distance k = 1 were among the classifiers employed.

The second classification model consists of image preprocessing step and feature extraction & classification step, as illustrated in Fig. 14 below. For this model, preprocessing step is done precisely the same as in the first model, whereas AlexNet is employed for feature extraction and classification of microscopic blood cell images.

AlexNet's architecture is adjusted to fit our task at hand. The final three layers of the original AlexNet—the final fully linked, SoftMax, and output layers—were frozen and swapped out for three more layers that were more appropriate for our classification task and then the next step was to train the network using the gathered images.

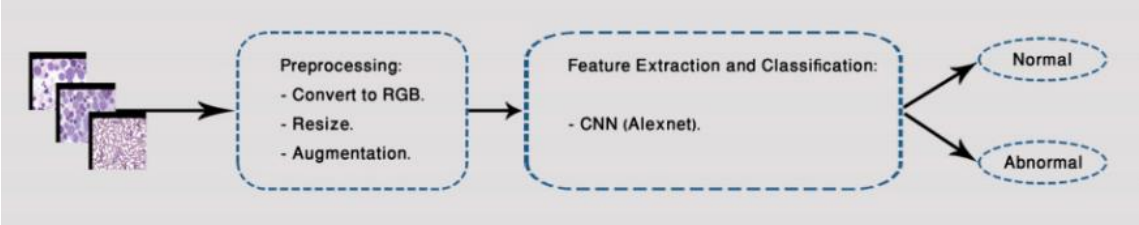

Figure 14: Diagram of the second classification model

3.10 Dataset

Our dataset is made up of 564 blood microscopic images (282 healthy and 282 leukemic). Figure 15 displays samples from the original dataset. Healthy samples are in the first row, and leukemic samples are in the second row. Using an optical laboratory microscope and a camera, samples were photographed, and the resulting photos were taken for use as a suitable dataset during the learning process. Following data augmentation, 2820 photos were obtained.

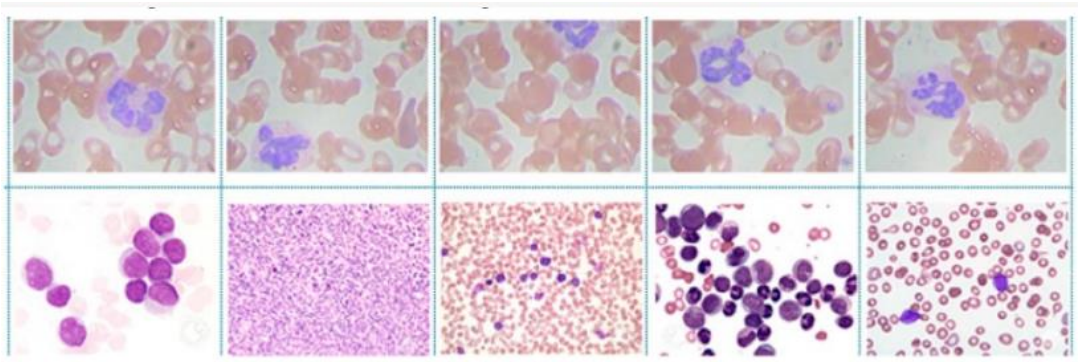

Figure 15: Samples of the dataset of healthy and unhealthy cells

3.11 Implementation and Experiments

The models were implemented using MATLAB. In order to evaluate their performance, we have listed performance metrics including Accuracy, Precision, Recall, Specificity.

$$Precision = TP/(TP + FP)$$

$$Recall = TP/(TP + FN)$$

$$Accuracy = (TP + TN)/(TP + TN + FP + FN)$$

$$Specificity = TN/(TN + FP)$$

FN stands for false negative, FP for false positive, TN for true negative, and TP for true positive. The classifiers for the first and second models were assessed using a 10-fold cross-validation method. Setting k = 1 and a maximum of 30 iterations were used to implement the K-NN classifier. The SVM classifier was employed with kernel functions that were cubic, linear, and Gaussian. The dataset was split into 80% training data and 20% test data for the second model holdout. During training, the learning rate was set to $1 \times 10{-}4$, the number of epochs to six, and the batch size to five. The NVIDIA GE FORCE 920M 4 GDDRAM graphics processing unit was used for all of the trials. The results for both models are shown in Figure 16.

Methods		Performance Metrics			
		Precision	Recall	Accuracy	Specificity
First Model	DT	95.69%	95.96%	95.82%	95.67%
	LD	99.64%	97.38%	98.51%	99.65%
	SVM-Linear	99.93%	98.72%	99.33%	99.93%
	SVM-Gaussian	99.93%	99.43%	99.68%	99.93%
	SVM-Cubic	99.93%	99.65%	99.79%	99.93%
	K-NN	99.64%	98.44%	99.04%	99.65%
Second Model	CNN (Alex Net): Cross fold	99.65%	100.00%	99.82%	99.65%
	CNN (Alex Net)	100%	100%	100%	100%

Table 1: Performance Metrics for both classification models

As we can see from the table above, the classifier which outperformed with the best results is SVM-Cubic with the highest values in all performance metrics. LD and K-NN followed with very good and similar results, and the worst performer is DT. If we look at the second model results, as illustrated in Figure 16, which employed AlexNet, outperforms the first model by scoring higher in all performance metrics. The results for both models are shown in the Figure 17 below:

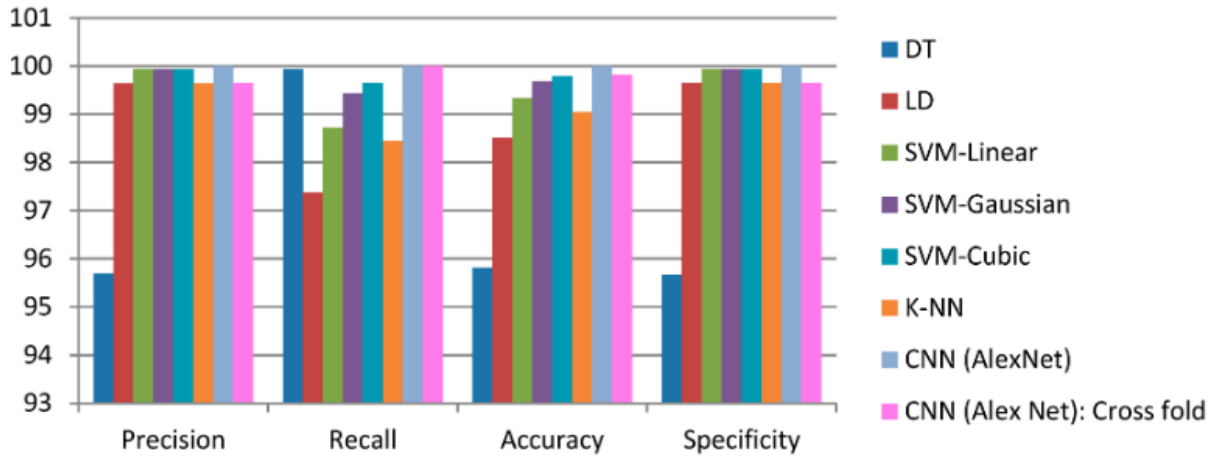

Table 2: Graphical representation of performance metrics results for both models

3.12 Discussion

Transfer learning is used in both models. In the first model, additional well-known classifiers like DT, LD, SVM, and K-NN are used for classification, while a pre-trained CNN called AlexNet is used to extract the distinguishing characteristics. Experiments showed that the SVM classifier was superior with highest results in performance metrics. AlexNet is utilized by the second model for both feature extraction and classification. This model's experiments proved that it was far superior to the first model in all the performance metrics.

3.13 Overview of Optimized Deep Recurrent Neural Networks

Optimized Deep Recurrent Neural Networks, or ODRNNs, are a major development in the field of neural network-based sequential data processing. ODRNNs are excellent for predicting patterns in historical data, natural language sequences, and other sequential data formats because they make use of deep learning concepts. ODRNNs are created with optimized architectures, which allows for more effective training and prediction than typical recurrent neural networks (RNNs). ODRNNs are highly beneficial in a variety of uses, including speech recognition, language modeling, and time-series forecasting, due to how they can efficiently capture complex temporal relationships through refined parameter optimization techniques and innovative network architectures. Moreover, ODRNNs provide improved scalability and flexibility, facilitating easy integration with a variety of data formats and network setups. Through sophisticated architectural components, ODRNNs are capable of adaptably concentrating on appropriate data while effectively allocating computational resources.

ODRNNs are known for their deep and recurrent architecture, which makes it easier to understand complex temporal dependencies in sequential data. An ODRNN is fundamentally made up of several recurrent layers, each of which has recurrent units that keep track of hidden states in order to record temporal data throughout time steps. Because of the connections between these recurrent units, information can move both inside and between layers. ODRNN architectures also frequently include optimized design choices, including long short-term memory (LSTM) cells or gated recurrent

units (GRUs), to address common problems like the vanishing gradient problem and facilitate more efficient learning of long-range dependencies. Moreover, ODRNNs have the potential to incorporate attention mechanisms, allowing input elements to be dynamically weighted according to how relevant they are to the situation at hand.

ODRNNs are trained using the backpropagation algorithm, which allows them to learn from data and gradually increase their predicting abilities. Backpropagation in ODRNNs works by iteratively modifying the network's parameters in order to reduce the difference between the desired target values/predictions and the model's predictions. The first step in this process is to send input data into the network, which then generates predictions and compares them to the ground truth. The resulting erroneous is then propagated backward across the network, layer by layer, and its gradient with respect to each parameter is calculated using the chain rule of calculus. Backpropagation through time, or BPTT, is particularly used in ODRNNs to manage the temporal dependencies present in sequential data. By gradually unrolling the network, BPTT permits errors to spread/propagated over several time steps. ODRNNs continually modify their parameters to increase their predicted accuracy through this iterative process of forward propagation and backward error propagation, eventually convergent towards the most accurate outcomes for the tasks at hand.

3.14 Optimized Deep Recurrent Neural Networks for Leukemia Classification

By analyzing microscopic images of blood samples, optimized deep recurrent neural network (ODRNN) is used to identify leukemia sickness. The recommended technique for leukemia diagnosis uses deep recurrent neural networks (DRNNs). The red deer optimization algorithm (RDOA) is then applied in order to optimize the weight collected by the DRNN. Three publicly available leukemia blood sample datasets, AML, ALL_IDB1, and ALL_IDB2, are used to assess the ODRNN model for leukemia classification. This section of the paper evaluates the efficacy of the suggested model for identification and classification of leukemia using performance metrics such as specificity, recall, accuracy, precision, and F1-score.

The first step is to obtain the free dataset from the publicly accessible datasets online. After obtaining the dataset, the next step is to handle missing values and apply

a median filter to pre-process the images of blood cells to smooth them and reduce the noise. Following the pre-processing of the images, the suggested model ODRNN for leukemia classification needs to be created. For the creation of the model, several factors need to be taken into account, such as the number of layers, units for each layer, and activation function.. To improve training efficiency, ODRNN presented an optimization technique based on RDO, which is used to optimize the above-mentioned parameters and enhance classification accuracy. Lastly, evaluation metrics have been used to evaluate these models, including precision, accuracy, specificity, recall and F1-score. A summary of the procedure is given below on Figure 18:

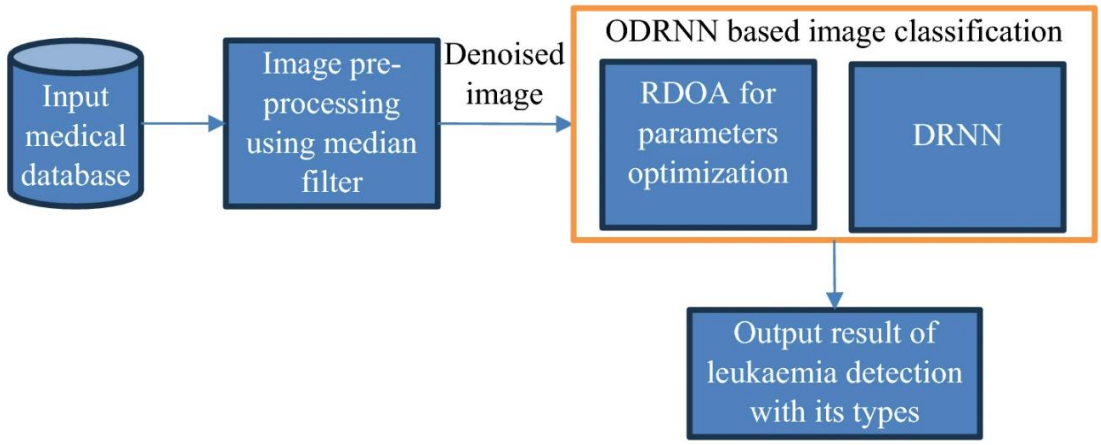

Figure 16: Overview of the methodology

3.15 Dataset

The used datasets for this model include C_NMC_2019, acute myeloid leukemia (AML), and the acute lymphoblastic leukemia—ALL database. C_NMC_2019 dataset is comprised of 15,114 ALL images of 450x450 pixels, from 118 patients. The dataset includes all leukemia's subtypes such as healthy, early, pre and pro. The images in the datasets are captured with a Zeiss camera. The AML dataset contains 10,000 images of sizes 64x64 pixels that are gathered from patient's blood smears on AML dataset.

3.16 Image Preprocessing

To improve the quality of the original image and facilitate the processing of the subsequent techniques, pre-processing step is necessary. The median filtering technique will be applied to rectify the white blood tumor cells in order to enhance the image. A distinct module in OpenCV is dedicated to image enhancement through the use of median filtering. Following the median filtering process, blood cell images have a smoother appearance, as shown in Figure 3.18. After the first processing, the results will be transformed from RGB to HSV. HSV defines the shades in terms of hue, values, color, and saturation. One benefit of HSV is that it has shades that are similar to how the human eye perceives them. Colors are created by blending primary colors from several categories, like RGB. Fig. 19 displays the color conversion results of the RGB to HSV conversion. When converting RGB to HSV color space, the blood cell image undergoes significant modifications.

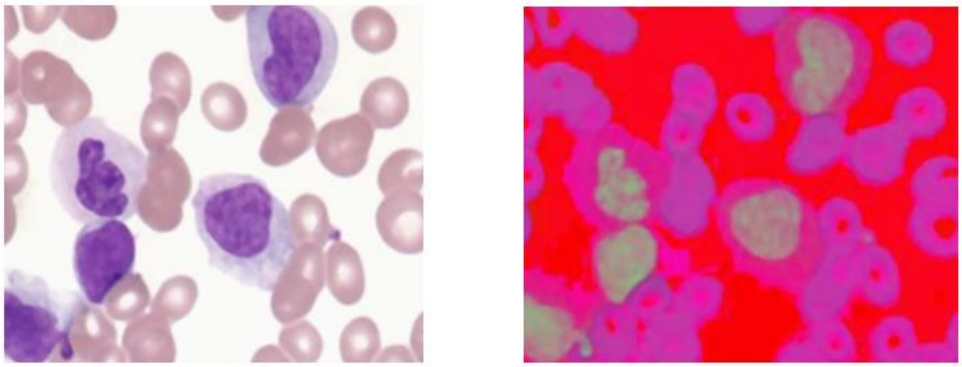

Figure 17: Blood cell images, after applying median filter (on the left), after converting from RGB to HSV (on the right)

Make use of the (Th) threshold value to eliminate things from the background. A pixel value in an image is exchanged with a MAX value of 255 (white) when it surpasses the threshold value of 0. The pixel value is replaced with 0 (black) if the shade of grey pixel count is less than the threshold. In this investigation, the threshold value was computed using a slider, and OpenCV's "in range" function was used to identify the lowest and highest values. The original data will be hidden in accordance

with the pattern after it has been obtained. The thresholding results are shown in Figure 20. After the thresholding process, white disease objects have been added to the previously red blood cell-only image, depicting illness cells.

Figure 18: Blood cell image after thresholding

3.17 DRNN Model Architecture

In DRNN the output from a previous phase is used as the input for the current phase. Neural networks simulate how neurons function when recognizing patterns. If the feed-forward network is a direct cycle, it is referred to as DRNN. Three levels are used by the DRNN: input, hidden and output, as illustrated in Figure 21. The buried layer is the main component of the DRNN that stores data about the received signal. RNN is frequently referred to as DRNN due to the large number of hidden layers it has, which vary depending on the issue specification. The problem specification determines the number of hidden layers that the DRNN integrates, adding to the network's complexity. The labelled datasets used to train the DRNN model contain data points that are assigned to distinct leukemia subtypes. The model gains the ability to link particular sequence patterns to related subtypes during training. The accuracy of the model is determined by analyzing its performance on a different test dataset after it has been successfully trained. The following are benefits of classifying leukemia using DRNNs:

A) High accuracy: can classify data with a high degree of accuracy, frequently outperforming more conventional machine learning techniques;

B) Feature extraction: By locating important features in the data which facilitate classification, DRNNs can provide important biological insights into disease processes.

C) Long-term dependencies: DRNNs may identify long-term relationships within sequences, providing researchers a more thorough comprehension of the data being studied.

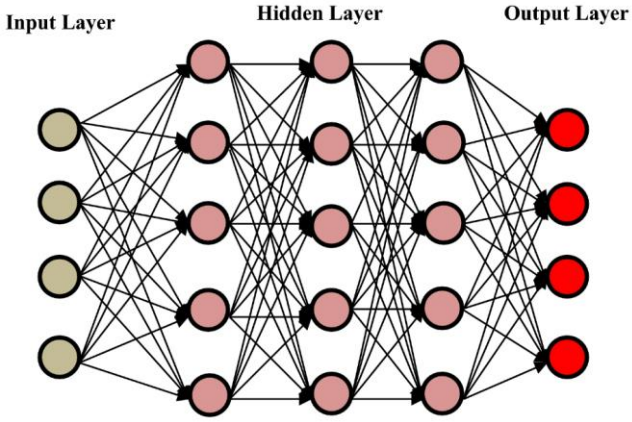

Figure 19: DRNN-based system architecture

3.18 Results and Discussion

The researchers assess the suggested identification and classification model using statistical measures linked to diseases, including specificity, recall, accuracy, precision, and F1-score. The TP (true positive), FP (false positive), TN (true negative), and FN (false negative) rates are used to represent positive classifications. The recall of a certain category of leukaemia subtype, conversely, reflects its level of predictability. There is agreement between the predicted and actual subtypes of leukaemia. While precision is the percentage of correctly classified positive leukaemia subtype predictions, the recall and accuracy measurements translate to the F1-score. Additional performance measures must be evaluated in order to put the model to the test. These measures are as following:

A) Accuracy: which specifies the total number of samples that are correctly predicted by the trained model out of all predictions made. It is calculated by dividing the sum of true positives (TP) and true negatives (TN), with the sum of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN).
B) Precision: which is the ratio of all properly predicted samples to the total number of samples classified as positive (either correctly or incorrectly). It is calculated by dividing true positives (TP) with the sum of true positives (TP) and false positives (FP).
C) Specificity: the ability of the model to predict a true negative for every available category. It is calculated by dividing true negatives (TN) by the sum of true negatives (TN) and false positives (FP).
D) Recall: the ratio of correctly categorized positive samples to the total number of positive samples. The recall assesses how well the model can identify positive samples. The more positive samples are identified, the higher the recall is. It is calculated by dividing true positives (TP) with the sum of true positives (TP) and false negatives (FN).
E) F1-Score: an alternative evaluation statistic which elaborates on a model's performance within a class, as opposed to evaluating the model's overall performance as done by accuracy. A weighted average of recall and precision can be used to interpret the F1 score, which has a maximum value of 1 and a minimum value of 0. It can be calculated as follows: 2 x Precision x Recall / Precision + Recall.

Four leukemia subtypes—ALL, CLL, AML, and CML—are classified using blood smear images by the ODRNN model. The models is evaluated using performance metrics such as accuracy, specificity, precision, recall, and F1-score. The standard dataset referenced in the dataset section was used to calculate the statistical metrics for all types of leukemia. Figure 22 demonstrates that for ALL, CLL, CML, AML, and healthy persons, the ODRNN is 100% correct. ODRNN predicts CLL with 99.91% accuracy. On the other hand, the F1-score, recall, and accuracy are all 0.99%. In contrast to ODRNN's CML accuracy of 98.96%, AML accuracy is 98.99% with a precision of 0.99%, recall of 1.0%, and F1-score of 1.0%.

ODRNN predicts the recall, F1-score, accuracy, and precision for healthy and ALL with 100%.

Leukaemia type	Precision	Recall	Specificity	F1-score	Accuracy
ALL	100	100	100	100	100
AML	98	99	98	99	98.99
CLL	99	99	99	100	99.91
CML	99	100	100	100	98.96
Healthy	100	100	100	100	100

Table 3: Statistical measurements of ODRNN model for classifying four leukemia subtypes

CHAPTER 4

METHODOLOGY

For Convolutional Neural Networks model, during the first stage of preprocessing the data, duplicate images were found and removed, which led to the removal of about forty-three images from the dataset, leaving 587 images across different sources. Next, in order to expedite training time, a scaling technique was used to ensure a uniform size of 256 x 256 pixels across all images, taking into account the dataset's heterogeneous size distribution and the requirement for uniformity when training a CNN model. In addition, images were filtered before processing in order to remove noise and improve line patterns. This was accomplished using a 3 x 3 median filter and image sharpening.

The dataset was then supplemented using data augmentation techniques, which increased the size of the training dataset and improved the generalization capacity of the CNN model. This required creating altered copies of the images using the ImageDataGenerator class in the Keras deep learning neural network framework. Various augmentation techniques were used, including flipping, shifting horizontally and vertically, random zooming, shearing, and interpolation. Moving on to the processing phase, the supplemented data was divided into training (60%), validation (20%), and test (20%) sets. It included 1550 cancer images and 1480 normal images.

The BasicCNN Model, which featured RGB color format and a resolution of 128 × 128 pixels for input images, was used to classify leukemia cells. This model architecture included fully connected layers trained for binary classification using the sigmoid activation function, a fixed filter size (3x3), 128 filters, and a stride of 1. It also included three convolutional layers with corresponding max pooling layers, each followed by a rectified linear unit (ReLu). After 17 epochs, the model was able to identify between leukemia and normal cells in the dataset with an accuracy of 90.99% and an 84.97% validation accuracy.

For Transfer Learning model, two classification algorithms are demonstrated to differentiate leukemia from healthy tissue in microscopic images. For both models, transfer learning was used, using pre-trained deep neural networks. The time and effort required to create and train these networks from scratch are eliminated by transfer learning. Transfer learning can be applied in two ways, which we have used as two approaches in our investigation. Using the values of the net's final fully connected layer (FC) to extract features from the input photos is the first approach. A different classifier thereafter is used to classify the images. The second technique entails removing the high-level layers from the network in order to alter its structure. We call this procedure "network fine-tuning".

The three primary steps of the first classification model are feature extraction, classification, and image pre-processing. Pre-processing involves a number of tasks, including scaling the images to fixed sizes, executing data augmentation to compensate for the absence of huge datasets, and converting blood images into a red-green-blue (RGB) model. Feature extraction involves taking each image and extracting a collection of features using a pre-trained AlexNet to ensure it may be used in classification to distinguish between images that are healthy and those that are impacted by leukemia. Many popular classifiers are used in the classification, including SVMs, linear discriminants (LDs), decision trees (DTs), and K-NNs. The pre-processing of the images and the subsequent feature extraction and classification are the only two processes in the second classification model. AlexNet is used in this work for both feature extraction and classification of blood microscopic images.

For Optimized Deep Recurrent Neural Networks model, the first step was to obtain the free dataset from the publicly accessible datasets available online specific to leukemia classification tasks. After acquiring the dataset, the next step would be to preprocess the data by applying a median filter to smooth the data and remove the image noise. The next step would be to define the model architecture for ODRNN. This involves specifying the number of layers in the model, the units for each layer and activation functions. The suggested model ODRNN for leukemia classification needs to be created after pre-processing. A previously trained ODRNN model was utilized by our classification model. Lastly, few evaluation metrics including

accuracy, precision, specificity, recall, F1 Score have been used to evaluate our model on a test set.

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 Results

With regards to CNN model, a total of 3030 images were employed in the CNN experiment; of those, 60% (1818 images) were used for training, 20% (606 images) for validation, and the remaining 20% (606 images) were used for testing the model. We examine the statistically measured parameters to gain insight into how successfully the model performs. The Basic CNN model's train accuracy is 90.99% after 17 epochs, and its validation accuracy is getting close to 85% after 17 epochs. Therefore, we expect our model to perform at about 85% accuracy for new data. However, after testing the model's performance on new data, we found out that the test accuracy is 80.91%. Confusion matrix is used for evaluating our model's performance, which is a list of anticipated outcomes for a classification task. The count numbers show the percentage of accurate and inaccurate projections for each class. Because it is feasible to display the relationships between the classifier outputs and the real ones, this is a great way to present findings in M-class classification issues. In our model there are 372 leukemia photos expected to be leukemia, 8 leukemia images predicted to be normal, 269 normal images predicted to be normal, and 51 normal images predicted to be leukemia for the basic CNN model. These accuracy levels show that this algorithm performs poorly in the task of predicting normal images but performs exceptionally well at predicting leukemia. A performance statistic called precision was employed as a metric, which calculates the proportion of positive identifications that were actually accurate. It is defined as the ratio of genuine positives to the sum of true positives and false positives. Our Basic CNN model has average precision. Recall is another metric used, which is computed by dividing the total number of trustworthy positive results by the total number of relevant samples, or all samples that should have had a positive label. In class 1, the first CNN model has a good recall but a poor precision. This suggests that while there are a lot of false positives, most positive

examples are correctly identified (low FN). The F1 Score and Support are the final metrics employed. The harmonic mean of recall and accuracy is known as the F1 score. The F1 score ranges from 0 to 1. It shows how robust the classifier is (a large number of cases are identified by it) and how accurate it is (the number of instances it correctly classifies). Our model's F1 Score seems to be average, indicating that it properly recognizes instances on average. Support is the quantity of samples that fairly represent the response in that category. It provides information on the precise numbers of each class in the test data.

5.2 CNN Model Image Distribution

Data Type	**Number of Images**	**Percentage**
Training	1818	60%
Validation	606	20%
Testing	606	20%
Total	**3030**	**100%**

Table 4: Image Percentage Distribution for Basic CNN Model

5.3 Basic CNN Model Performance

Metric	**Value**
Train Accuracy	90.99%
Validation Accuracy	~85%
Test Accuracy	~81%

Table 5: Model performance for Basic CNN Model

With regards to Transfer Learning models, using 10-fold cross-validation, the classifiers for the first and second models were evaluated. The K-NN classifier was implemented with a maximum of 30 iterations and a setting of k = 1. Three different kernel functions were used with the SVM classifier: cubic, linear, and Gaussian. For

the second model holdout, the dataset was divided into 80% training data and 20% test data. During the training process, the batch size was set to five, the number of epochs to six, and the learning rate to 1 × 10−4. For every trial, an NVIDIA GE FORCE 920M 4 GDDRAM graphics processor unit was utilized. SVM-Cubic had the highest values across all performance criteria, it is the classifier that performed best and produced the best results. LD and K-NN yielded excellent and comparable outcomes, but DT performed poorly. By scoring higher on all performance criteria, the second model, which used AlexNet, performs far superior to the first. In both models, transfer learning is implemented. In the first model, a pre-trained CNN known as AlexNet is utilized to extract the distinguishing features, and additional popular classifiers such as DT, LD, SVM, and K-NN are employed for classification. The SVM classifier outperformed the others in the experiments, achieving the highest performance metrics. The second model uses AlexNet for both feature extraction and classification. Experiments with this model demonstrated that it was far superior to the first model in all the performance metrics.

5.4 Transfer Learning Model Evaluation

Classifier	**Cross-Validation**	**Kernel Function (if applicable)**	**Performance Summary**
K-NN	10-fold	N/A	Excellent outcomes
SVM-Cubic	10-fold	Cubic	Best performance overall
SVM-Linear	10-fold	Linear	Good performance
SVM-Gaussian	10-fold	Gaussian	Good performance
LD	10-fold	N/A	Excellent outcomes
DT	10-fold	N/A	Poor performance

Table 6: Model Evaluation for Transfer Learning

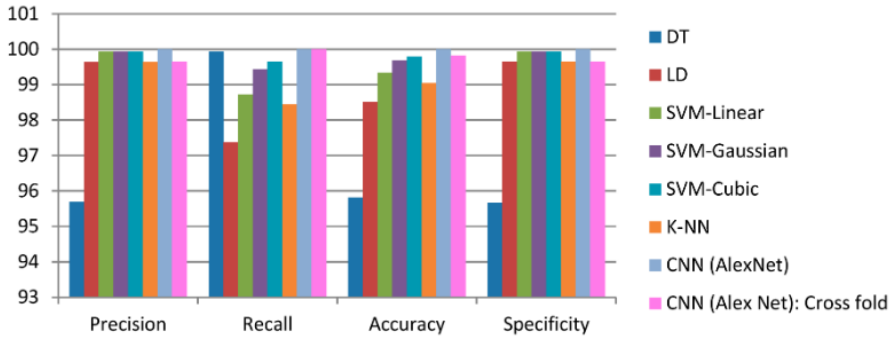

Table 7: Graphical representation of performance metrics results for both models for Transfer Learning

With regards to ODRNN model, based on images of blood smears, the proposed ODRNN model classifies four leukemia subtypes: ALL, CLL, AML, and CML. Performance metrics including as accuracy, F1-score, precision, and recall are employed to evaluate the model. Using the standard dataset described in the dataset description section, the F1-score, accuracy, precision, recall, and specificity for all types of leukemia were obtained. We can see that the model has achieved 100%% accuracy for classifying ALL leukemia cells and healthy cells. It has achieved 98.99% accuracy in classifying AML leukemia cells, 99.91% accuracy in classifying CLL leukemia cells, and 98.96% accuracy in classifying CML leukemia cells. As we can see, the model has made most correct predictions for ALL leukemia subtype and healthy cells. Regarding F1 Score metric, the model has obtained highest values of 100% for ALL, CLL, CML, and healthy cells, with AML score being 99%. Regarding specificity metric, the model has outperformed on detecting ALL, CML, healthy cells with 100% score, leaving behind AML and CLL with 98% and 99%, respectively. Regarding Recall metric, the model has achieved highest scores on ALL, CML and healthy cells with 100%, followed by AML and CLL, both 99%. Lastly for precision metric, the model has outperformed on ALL and healthy cells with 100% score, followed by CLL and CML with 99% and AML with 98%.

5.5 ODRNN Model Performance Metrics

Leukemia Subtype	Accuracy	F1-Score	Precision	Recall	Specificity
ALL	100%	100%	100%	100%	100%
AML	98.99%	99%	98%	99%	98%
CLL	99.91%	100%	99%	99%	99%
CML	98.96%	100%	99%	100%	100%
Healthy Cells	100%	100%	100%	100%	100%

Table 8: Performance metrics for ODRNN Model

Our findings are derived from an extensive review of existing literature. Additionally, we have conducted a thorough analysis using performance metrics, comparing the principles and efficacy of various algorithms in leukemia detection. Furthermore, we have compiled significant comparative data in tabular form for comprehensive evaluation.

CHAPTER 6

CONCLUSIONS

6.1 Conclusions

A cancer known as leukemia attacks the body's blood-forming tissues, such as the bone marrow and lymphatic system. Early diagnosis is crucial for the patient to receive the best treatment, so we use CNN's power to classify blood smears into normal and pathological using Basic CNN model. We looked into the use of deep CNNs in this system. By utilizing convolutional neural network classification methods and microscopic images of blood samples, a pre-trained model was deployed that could identify and categorize the samples into normal and abnormal categories. Deep learning, which looks at all of the features in tiny images rather than just focusing on changing a few key attributes as a classifier input, was used to build the system. In order to verify the precision and dependability of the system, the pretrained model is ran on a substantially expanded dataset. Since our dataset was not acquired under the same conditions and it was gathered from multiple sources, it required a larger size for usage with DL. To solve this issue, data augmentation is employed, and it performed effectively for us because the 260 images in our data before augmentation were increased to 3030 images after augmentation. Three convolutional layers with max pooling layers make up the Basic CNN model. It had an accuracy of 90% and a validation accuracy of 84.97%. Because of its limited layers, it performed poorly with our dataset. Three pieces made up the detection system's design:

1) The acquisition component, which is made up of a digital camera mounted above the microscope's eyepiece;

2) A CNN model that has been trained in advance and is in control of the classification system;

3) A graphical user interface that shows the categorization results and the image that was taken from the camera.

Leukemia treatment can be greatly aided by early detection. Two classification models that differentiate between blood microscopic images with and without leukemia were presented in the second part of our comparison section, which is about Transfer Learning. Transfer learning is used in both models. In the first model, additional well-known classifiers like DT, LD, SVM, and K-NN are used for classification, while a pre-trained CNN called AlexNet is used to extract the discriminant characteristics. Experiments showed that the SVM classifier was superior. AlexNet is utilized by the second model for both feature extraction and classification. This model's experiments demonstrated that it was more effective than the first model in a number of performance criteria.

Four leukemia types—AML, ALL, CLL, and CML—can be classified from blood smear pictures using the suggested ODRNN model in our third part of the comparison chapter. The diagnosis of leukemia is established utilizing the ODRNN model. It suggested a novel DL methodology known as ODRNN to identify leukemia through microscopic analysis of blood sample images. The suggested method for diagnosing leukemia uses ODRNN and then applies the RDOA to adjust the weights that ODRNN has learned. The roaring rates of deer serve as the basis for the RDOA optimization process. Three publicly available leukemia blood sample datasets, AML, ALL_IDB1, and ALL_IDB2, gathered from UCI Repository, are used to evaluate the suggested model. Overall, by utilizing the strength of deep learning and effective optimization to extract significant features and patterns from complicated medical data, the ODRNN presents a viable method for leukemia image detection. This could facilitate early detection, increase the accuracy of diagnoses, and eventually improve patient outcomes. Although the ODRNN offers a novel method for identifying leukemia images, it doesn't come without drawbacks. The possible complexity brought about by the hybridization of DRNN and the RDOA optimization procedure is one significant disadvantage. The complex interactions among these elements could result in higher processing requirements, which would make the model heavy on resources and difficult to use in situations requiring real-time deployment. Furthermore, the model's performance is closely linked to the optimization procedure based on deer roaring rate behavior, which adds another level of complexity to the dependency on the red deer optimization technique. Due to its complexity, the model may be more

difficult to read and comprehend the individual contributions of each component, which could restrict its applicability in clinical settings.

Model	Key Features	Pros	Cons	Train Acc.	Test Acc.
Basic CNN	Three convolutional layers with max pooling	Simple architecture, fast training	Limited layers, less effective on diverse datasets	90%	80.91%
Transfer Learning	AlexNet for feature extraction, SVM for classification (Model 1); AlexNet for both feature extraction and classification (Model 2)	Improved feature extraction, robust performance	Dependent on pre-trained models, computationally intensive	98.7% Model 1: Higher with SVM, Model 2: Superior overall	Model 1: Varies, Model 2: Superior overall
ODRNN	Deep learning with Red Deer Optimization Algorithm	High accuracy, effective optimization	High complexity, resource-intensive	High (specific value not provided)	High (specific value not provided)

Table 9: Comparison table of Basic CNN, Transfer Learning and ODRNN Models

6.2 Future Recommendations

To lessen overfitting and enhance generalization in CNNs, the amount of the dataset can be increased by means of data augmentation, synthetic data generation, or cooperative data sharing among institutions. To further improve model robustness, consider utilizing approaches like batch normalization, dropout, and ensemble procedures.

Choosing pre-trained models more closely aligned with the medical imaging domain can enhance the adaptability and usefulness of features for Transfer Learning models. Performance can be optimized while lowering the likelihood of overfitting by fine-tuning these models using a hybrid technique that blends transfer learning with a modest quantity of domain-specific data.

Complexity and training time for ODRNNs can be decreased by streamlining the model architecture and concentrating on hybrid strategies that combine CNNs for feature extraction with RNNs for sequential data. Better generalization can be ensured

by avoiding overfitting and making use of strategies like early stopping and cross-validation. Furthermore, the resource-intensive aspect of these models can be addressed by utilizing cloud-based solutions and investing in computational resources, allowing for more feasible and scalable implementations in clinical settings.

REFERENCES

[1] Lipton, Z. C., Kale, D. C., Elkan, C., & Wetzell, R. (2015). Learning to diagnose with LSTM recurrent neural networks. 18. https://doi.org/https://doi.org/10.48550/arXiv.1511.03677

[2] Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B., & Liang, J. (2016). Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Transactions on Medical Imaging, 35*(5), 1299-1312. https://doi.org/10.48550/arXiv.1706.00712

[3] Maier-Hein, L., Eisenmann, M., & Reinke, A. (2018). Why rankings of biomedical image analysis competitions should be interpreted with care. *Nature Communications*, *9*(1), 1-4. https://doi.org/10.48550/arXiv.1806.02051

[4] Liu, X., Faes, L., & Kale, A. U. (2019). A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: A systematic review and meta-analysis. *The Lancet Digital Health, 2*(6), e271-e297. http://dx.doi.org/10.1016/S2589-7500(19)30123-2

[5] Kather, J. N., Krisam, J., & Spanhol, F. A. (2019). Predicting survival from colorectal cancer histology slides using deep learning: A retrospective multicenter study. *PLoS Medicine, 16*(1),e1002730. 23. http://dx.doi.org/10.1371/journal.pmed.1002730

[6] Liu, X., Faes, L., & Kale, A. U. (2019). A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: A systematic review and meta-analysis. *The Lancet Digital Health, 3*(6), e271-e297. 27. https://doi.org/10.1016/S2589-7500(19)30123-2

[7] Bagher Sistaninejhad, Habib Rasi, Parisa Nayeri. (2023). A review paper about deep learning for medical image analysis. *Computational and Mathematical Methods in Medicine, 2023*, Article ID 7091301, 10. https://doi.org/10.1155/2023/7091301

[8] Shree, K. D., & Logeswari, S. (2024). Optimized deep recurrent neural networks for automatic detection of leukemia. *SIViP*. 17. https://doi.org/10.1007/s11760-024-03062-y

[9] Tran, T., Park, J.-H., Kwon, O.-H., Moon, K.-S., Lee, S.-H., & Kwon, K.-R. (2018). Classification of leukemia disease in peripheral blood cell images using convolutional neural network. 12. http://dx.doi.org/10.9717/kmms.2018.21.10.1150

[10] Loey, M., Naman, M., & Zayed, H. (2020). Deep transfer learning in diagnosing leukemia in blood cells. *Computers*, *9*(2), 29. https://doi.org/10.3390/computers9020029

[11] Terven, J., Cordova-Esparza, D.-M., Ramirez-Pedraza, A., & Chávez Urbiola, E. (2023). Loss functions and metrics in deep learning: A review. 53. https://doi.org/10.48550/arXiv.2307.02694

[12] Wang, R., Lei, T., Cui, R., Zhang, B., Meng, H., & Nandi, A. K. (2022). Medical image segmentation using deep learning: A survey. 25. https://doi.org/10.1049/ipr2.12419

[13] Kim, H. E., Cosa-Linan, A., Santhanam, N., et al. (2022). Transfer learning for medical image classification: A literature review. *BMC Medical Imaging, 22*, 69. https://doi.org/10.1186/s12880-022-00793-7 https://doi.org/10.48550/arXiv.1211.5063

[14] Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. *arXiv preprint arXiv:1211.5063*. 12. https://doi.org/10.48550/arXiv.1211.5063

[15] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. *arXiv preprint arXiv:1505.04597*. 8. https://doi.org/10.48550/arXiv.1505.04597

[16] Lundberg, S. M., Nair, B., Vavilala, M. S., et al. (2018). Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature Biomedical Engineering*, *2*, 749–760. https://www.nature.com/articles/s41551-018-0304-0

[17] Ahmed, A., Nagy, A., Kamal, A., & Farghl, D. (2022). Leukemia detection based on microscopic blood smear images using deep learning. *arXiv preprint arXiv:2212.09999*. 6. https://doi.org/10.48550/arXiv.2301.03367

[18] Wang, D., Khosla, A., Gargeya, R., Irshad, H., & Beck, A. H. (2016). Deep learning for identifying metastatic breast cancer. *arXiv preprint arXiv:1606.05718*. 6. https://doi.org/10.48550/arXiv.1606.05718

[19] Agaian, S., Madhukar, M., & Chronopoulos, A. T. (2014). Automated screening system for acute myelogenous leukemia detection in blood microscopic images. *IEEE Systems Journal*, *8*, 995–1004. https://doi.org/10.1109/JSYST.2014.2308452

[20] Abdeldaim, A. M., Sahlol, A. T., Elhoseny, M., & Hassanien, A. E. (2018). Computer-aided acute lymphoblastic leukemia diagnosis system based on image analysis. In *Advances in Soft Computing and Machine Learning in Image Processing* (pp. 131–147). 17. Springer. http://dx.doi.org/10.1007/978-3-319-63754-9_7