PROPOSING EFFICIENT CNN MODELS FOR THE DETECTION OF ACUTE LYMPHOBLASTIC LEUKEMIA (ALL) USING TRANSFER LEARNING

2022
MASTER THESIS
COMPUTER ENGINEERING

Hakmet Ibraheem ABED

Thesis Advisor
Assist. Prof. Dr. Adnan Saher M. AL-AJEELI

PROPOSING EFFICIENT CNN MODELS FOR THE DETECTION OF ACUTE LYMPHOBLASTIC LEUKEMIA (ALL) USING TRANSFER LEARNING

Hakmet Ibraheem ABED

Thesis Advisor
Assist. Prof. Dr. Adnan Saher M. AL-AJEELI

T.C.
Karabuk University
Institute of Graduate Programs
Department of Computer Engineering
Prepared as
Master Thesis

KARABUK
November 2022

I certify that in my opinion the thesis submitted by Hakmet Ibraheem ABED titled "PROPOSING EFFICIENT CNN MODELS FOR DETETION OF LYMPHOBLASTIC LEUMEIA (ALL) USING TRANSFER LEARNING" is fully adequate in scope and in quality as a thesis for the degree of Master of Computer Engineering.

Assist. Prof. Dr. Adnan Saher M. AL-AJEELI .....
Thesis Advisor, Department of Computer Engineering

This thesis is accepted by the examining committee with a unanimous vote in the Department of Computer Engineering as a Master of Science thesis. 25.11. 2022

Examining Committee Members (Institutions) Signature

Chairman: Assist Prof. Dr. Oğuzhan MENEMENCİOĞLU (KBU) .....

Member: Assist. Prof. Dr. Abdulkadir TAŞDELEN (AYBU) .....

Member: Assist. Prof. Dr. Adnan Saher M. AL-AJEELI (AYBU) .....

The degree of Master of Computer Engineering by the thesis submitted is approved by the Administrative Board of the Institute of Graduate Programs, Karabuk University.

Assoc. Prof. Dr. Müslüm KUZU .....
Director of the Institute of Graduate Programs

"I declare that all the information within this thesis has been gathered and presented in accordance with academic regulations and ethical principles and I have according to the requirements of these regulations and principles cited all those which do not originate in this work as well."

Hakmet Ibraheem ABED

ABSTRACT

M. Sc. Thesis

PROPOSING EFFICIENT CNN MODELS FOR THE DETECTION OF LYMPHOBLASTIC LEUKEMIA (ALL) USING TRANSFER LEARNING

Hakmet Ibraheem ABED

Karabuk University
Institute of Graduate Programs
The Department of Computer Engineering

Thesis Advisor:
Assist. Prof. Dr. Adnan Saher M. AL-AJEELI
December 2022, 66 pages

Acute lymphoblastic leukemia (ALL) is one of the main causes of mortality in the modern world. The detection of ALL has been approached in a variety of ways. The term "computer-aided diagnostics" (CAD) refers to various machine learning-based automated diagnosis systems. Early CAD systems used machine learning methods, but convolutional neural networks (CNN)-based deep learning models have gained popularity because of their autonomous feature extraction capabilities. Many different disciplines employ deep learning. One of the key industries that deep learning has revolutionized is healthcare. Different pathologists' and doctors' perspectives are other frequent problems that patients encounter. Such human mistakes often result in inaccurate or slow judgment, which may be disastrous to human life. Researchers in healthcare are using deep learning-based methodologies and have produced state-of-the-art findings to increase decision consistency, efficiency, and mistake reduction. This work suggests using deep learning (DL) and transfers learning to categorize

histological pictures for ALL diagnoses. In this work, we used transfer learning to categorize blood pathology pictures on various training photos without sacrificing performance. In order to extract features, patches are first taken from Whole Slide Images and put into the CNN. The discriminative patches are chosen based on these characteristics and supplied into the proposed architecture once pre-trained on two datasets. The suggested models perform better than the standard techniques regarding various performance metrics. Thanks to the transfer learning approach, three separate convolutional neural network models (DenseNet 169, VGG19, and a basic CNN model) were trained using two distinct data sets, ALL IDB and C-NMC ALL. The accuracy values for the three models after training on the ALL IDB data set were 99.76%, 99.49%, and 94.24%, respectively. The accuracy scores were 85.41%, 85.32%, and 80.77% when the three models were trained using the C-NMC ALL dataset.

Keywords : Acute lymphoblastic leukemia, DenseNet 169, VGG19, ALL_IDB, and C-NMC ALL dataset.

Science code : 92431

ÖZET

Yüksek Lisans Tezi

TRANSFER ÖĞRENMEYİ KULLANARAK LENFOBLASTİK LÖSEMİNİN (TÜMÜ) TESPİTİ İÇİN VERİMLİ BİR CNN MODELİ ÖNERİLMESİ

Hakmet Ibraheem ABED

Karabük Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:
Dr. Öğr. Üyesi Adnan Saher M. AL-AJEELI
Aralık 2022, 66 sayfa

Akut lenfoblastik lösemi (ALL), modern dünyada başlıca ölüm nedenlerinden biridir. ALL'nin tespitine çeşitli şekillerde yaklaşılmıştır. "Bilgisayar destekli teşhis" (CAD) terimi, çeşitli makine öğrenimi tabanlı otomatikleştirilmiş teşhis sistemlerini ifade eder. Erken CAD sistemleri, makine öğrenimi yöntemlerini kullandı, ancak evrişimli sinir ağları (CNN) tabanlı derin öğrenme modelleri, otonom özellik çıkarma yetenekleri nedeniyle popülerlik kazandı. Birçok farklı disiplin derin öğrenmeyi kullanır. Derin öğrenmenin devrim yarattığı kilit sektörlerden biri sağlık hizmetleridir. Farklı patolog ve doktorların bakış açıları da hastaların sıklıkla karşılaştığı diğer sorunlardır. Bu tür insan hataları genellikle yanlış veya yavaş karar verilmesine neden olur ve bu da insan hayatı için felaket olabilir. Sağlık alanındaki araştırmacılar, derin öğrenmeye dayalı metodolojiler kullanıyor ve karar tutarlılığını, verimliliği ve hata azaltmayı artırmak için en son teknolojiye sahip bulgular ürettiler. Bu çalışma, derin

öğrenmenin (DL) kullanılmasını önerir ve TÜM teşhisler için histolojik resimleri kategorize etmek için öğrenmeyi aktarır. Bu çalışmada, performanstan ödün vermeden çeşitli eğitim fotoğraflarındaki kan patolojisi resimlerini kategorize etmek için transfer öğrenmeyi kullandık. Özellikleri çıkarmak için önce Tam Slayt Görüntülerinden yamalar alınır ve CNN'e konur. Ayırt edici yamalar, bu özelliklere dayalı olarak seçilir ve iki veri kümesi üzerinde önceden eğitildikten sonra önerilen mimariye sağlanır. Önerilen modeller, çeşitli performans ölçütleri açısından standart tekniklerden daha iyi performans göstermektedir. Transfer öğrenme yaklaşımı sayesinde, ALL IDB ve C-NMC ALL olmak üzere iki farklı veri seti kullanılarak üç ayrı evrişimli sinir ağı modeli (DenseNet 169, VGG19 ve temel bir CNN modeli) eğitildi. ALL IDB veri setinde eğitimden sonra üç model için doğruluk değerleri sırasıyla %99,76, %99,49 ve %94,24 idi. Doğruluk puanları, üç model C-NMC ALL veri seti kullanılarak eğitildiğinde %85,41, %85,32 ve %80,77 idi.

Anahtar Kelimeler: Akut lenfoblastik lösemi, DenseNet 169, VGG19, ALL_IDB ve C-NMC ALL veri seti.

Bilim Kodu **:** 92431

ACKNOWLEDGMENT

I owe thanks and praise to God first and foremost for this success and facilitation as I bow to my beloved parents. My dear father gave me the most valuable things to make me a man of honor. My beloved mother is good at engineering my heart with her prayers. To my family that I grew up in and its extension gives me pride and honor. I owe a special thanks to my thesis supervisor, Assist. Prof. Dr. Adnan Saher M. AL-AJEELI who spared no effort in providing unlimited advice and guidance until the completion of this thesis to the fullest.

I also extend my gratitude to Karabuk University, including the wonderful professors and colleagues who accompanied us throughout our academic journey.

I dedicate this thesis to my beloved country, Iraq. And to the beautiful Turkey, which embraced this scientific experience and contributed to providing all possibilities for graduating in this distinguished way.

CONTENTS

Page

APPROVAL ii
ABSTRACT iv
ÖZET vi
ACKNOWLEDGMENT viii
CONTENTS ix
LIST OF FIGURES xii
LIST OF TABLES xiv
SYMBOLS AND ABBREVITIONS INDEX xv

PART 1 1
INTRODUCTION 1
1.1. OVERVIEW 1
1.2. MOTIVATION 3
1.3. PROBLEM STAMEMENT 4
1.4. OBJECTIVE 4
1.5. INCLUSION CRITERIA 5

PART 2 6
LITERATURE REVIEW 6

PART 3 15
DEEP LEARNING 15
3.1. CONVOLUTIONAL NEURAL NETWORK 16
3.1.1. Convolutional Layer 17
3.1.2. Max-Pooling Layer 17
3.1.3. Flatten Layer 18
3.1.4. Rectified Linear Unit Layer (ReLU) 18
3.1.5. Dropout Layer 18

Page

3.1.6. Batch Normalization Layer 19
3.1.7. Fully Connected Layer 19
3.1.8. Adam Optimization 19
3.2. CLASSIFICATION MODELS 20

PART 4 21
METHODOLOGY 21
4.1. DATASET PREPARATION 21
4.1.1. Platform Used 21
4.1.2. Initializing Libraries 21
4.2. DATASET DESCRIPTION 22
4.2.1. ALL_IDB Dataset 23
4.2.2. C-NMC ALL Dataset 24
4.3. DATA PREPROCESSING 26
4.4. DATA SPLITTING 26
4.5. DATA AUGMENTATION 27
4.6. FEATURE EXTRACTION 28
4.7. TRANSFER LEARNING 29
4.8. PROPOSED SYSTEM 30
4.8.1. DenseNet 169 Model 31
4.8.2. VGG19 Model 33
4.8.3. Simple CNN Model 35
4.9. TRAINING THE MODELS 36
4.10. MODEL EVALUATION 38

PART 5 39
RESULT AND DISCUSSION 39
5.1. ALL MODELS' ACCURACY 39
5.2. EVALUATION OF MODELS 40
5.3. CROSS-VALIDATION 41
5.4. EXPERIMENTAL RESULTS 42
5.4.1. Test Results DenseNet169 to ALL_IDB Dataset 42
5.4.2. Test Results VGG19 Model to ALL_IDB Dataset 43

Page

5.4.3. Test Results In Simple CNN Model to ALL_IDB Dataset 45
5.4.4. Test Results DenseNet169 Model to C-NMC ALL Dataset 47
5.4.5. Test Results VGG19 Model to C-NMC ALL Dataset..... 49
5.4.6. Test Results in Simple CNN Model to C-NMC ALL Dataset 50
5.5. RESULTS INTERPRETATION..... 52
5.6. COMPARISON THE RESULTS..... 53

PART 6 56
CONCLUSION..... 56
6.1. FUTURE WORK 57

REFERENCES..... 58

RESUME 66

LIST OF FIGURES

Page

Figure 1.1. Acute lymphoblastic leukemia-Image Dataset (IDB)-1 example images. (A and B) Leukemia cell, (C and D) normal cell. (source: our dataset). 2

Figure 3.1. Architecture of CNN. (source: (Hidaka, 2017). 16

Figure 4.1. Display of images of Normal and lymphoblasts. (source: our dataset). 23

Figure 4.2. Display the distribution of images in ALL_IDB dataset. 24

Figure 4.3. Display the distribution of images in ALL_IDB dataset. (Source: from our code). 25

Figure 4.4. Sample of augmented images (Source: from our code). 28

Figure 4.5. Illustration transfer learning from a previously trained model. 31

Figure 4.6. Architecture of DenseNet169 (source: (Huang, Gao, et al.,2017) 33

Figure 4.7. Illustrate of VGG19 model (source: (Jaworek-Korjakowska et al. 2019) 35

Figure 4.8. Simple CNN model schematic. 36

Figure 4.9. Principal of transfer learning. 37

Figure 5.1. A plot of model accuracy for DenseNet169 with ALL_IDB dataset. .. 42

Figure 5.2. A Plot of model loss for DenseNet169 with ALL_IDB dataset. 43

Figure 5.3. Confusion matrix for DenseNet169 with 10 epochs- 417 testing images of ALL_IDB dataset. 43

Figure 5.4. A plot of model accuracy for VGG1 with ALL_IDB dataset 44

Figure 5.5. A Plot of model loss for VGG19 with ALL_IDB dataset. 44

Figure 5.6. Confusion matrix for DenseNet169 with 10 epochs- 417 testing images of ALL_IDB dataset. 45

Figure 5.7. A plot of model accuracy for a simple CNN model with ALL_IDB. .. 46

Figure 5.8. A Plot of model loss for a simple CNN model with ALL_IDB. 46

Figure 5.9. Confusion matrix for a simple CNN model with 10 epochs- 417 testing images of ALL_IDB. 46

Figure 5.10. A Plot of model accuracy for DenseNet169 with C-NMC ALL. 47

Figure 5.11. A Plot of model loss for DenseNet169 with C-NMC ALL. 48

Figure 5.12. Confusion matrix for DenseNet169 with 10 epochs- 2133 testing images of C-NMC ALL dataset. 48

Page

Figure 5.13. A Plot of model accuracy for a VGG19 model with C-NMC ALL. 49

Figure 5.14. A Plot of model loss for VGG19 with C-NMC ALL. 50

Figure 5.15. Confusion matrix for VGG19 with 10 epochs- 2133 testing images of C-NMC ALL dataset. 50

Figure 5.16. A Plot of model accuracy for a simple CNN model with C-NMC ALL51

Figure 5.17. A Plot of model loss for a simple CNN model with C-NMC ALL..... 51

Figure 5.18. Confusion matrix for simple CNN with 10 epochs- 2133 testing images of C-NMC ALL dataset. 52

Figure 5.19. Comparing the suggested models' accuracy on the ALL_IDB dataset. 54

Figure 5.20. Comparing the suggested models' accuracy on C-NMC ALL dataset. . 54

Figure 5.21. Comparing our suggested outcomes to others. 55

LIST OF TABLES

Page

Table 2.1. Review of related literature 13
Table 4.1. Datasets characteristics 26
Table 5.1. Split of images in datasets 40
Table 5.2. Results of applying the proposed models to ALL_IDB dataset 40
Table 5.3. Results of applying the proposed models to C-NMC ALL dataset 40
Table 5.4. Performance results of DenseNet169 model to ALL_IDB. 42
Table 5.5. Performance results of VGG19 model to ALL_IDB dataset. 44
Table 5.6. Performance results of simple CNN model to ALL_IDB 45
Table 5.7. Performance results of DenseNet169 model to C-NMC ALL 47
Table 5.8. Performance results of VGG 19 model to C-NMC ALL 49
Table 5.9. Performance results of simple CNN model to C-NMC ALL 51

SYMBOLS AND ABBREVITIONS INDEX

ABBREVIATIONS

ALL : Acute lymphoblastic leukemia
DNA : Deoxyribonucleic acid
CDC : Complete blood count
WBC : White blood cell
CLL : Chronic lymphocytic leukemia
PB : Peripheral blood
AI : Artificial intelligent
CNN : Convolutional neural network
AML : Acute myeloblastic leukemia
KNN : k-nearest neighbors
SVM : Support vector machines
LDP : Local Directional Pattern
NN : Neural network
RMDL : Random Multimodal Deep Learning
CSO : Competitive swarm optimization
RGB : Red, green, blood
PCA : Principal component analysis
HLAC : Healthcare Laundry Accreditation Council
GLCM : Gray Level Cooccurrence Matrix
CAD : Computer-aided diagnostic
MLP : Multilayer perceptron
WSI : Whole-slide image
DCNN : Deep Convolutional neural network
VGG : Visual Geometry Group
GPU : Visual Geometry Group
RMS Prop : Root Mean Squared Propagation

API	: Application programming interface
TCIA	: The Cancer Imaging Archive
MSE	: Mean square error
TP	: True positive
TN	: True negative
FP	: False positive
FN	: False negative
AUC	: Area under Curve
ROC	: Receiver operating characteristic curve
CV	: Cross validation

PART 1

INTRODUCTION

1.1. OVERVIEW

Acute lymphoblastic leukaemia is the kind of leukaemia most often diagnosed in children (ALL). ALL develops when the DNA of bone marrow stem cells becomes flawed, enabling them to proliferate excessively. Lymphoblasts, which are immature stem cells, flood the body, resulting in a wide range of unusual symptoms, including bruising, bleeding from the mouth or nose, and infections [1].

Despite the fact that ALL is a treatable condition, early detection is crucial for a positive outlook. Medical professionals want a broad variety of laboratory testing. For a number of tests, lumbar puncture or bone marrow biopsy are necessary procedures. Other tests just need a single blood sample from the patient's peripheral blood and are less invasive. A full blood count is an excellent illustration of a minimally intrusive test (CBC) [2].

Everything that may be described as "lymphoblastic" can be traced back to lymphocytes, a kind of white blood cell. Leukemia cells carry out their assault on the blood at a respectably swift rate daily. In addition, they can spread to other areas of the body, such as the testicles, liver, spleen, central nervous system, and lymph nodes [1], [2].

A facility-based study by a group of doctors with various specialties found that over 5000 confirmed cases of hemorrhagic illness had been investigated between mid-January 2008 and the end of December 2012. Hematological scattering that might be

life-threatening was found in 5013 patients aged 2 to 90. There were 69.2 percent men (n = 3468) and 30.8 percent females (n = 1545), making the male to female ratio 2.2:1. At the time of the study, the average middle age was 42 years [1], [3].

ALL, a WBC malignancy often observed in children, is caused by the bone marrow's uncontrollable proliferation and overproduction of immature WBC. ALL symptoms, which include fatigue, weakness, and pain in the joints and bones, are so similar to flu and other common disorders that it is almost impossible to tell the difference between them. The three types of ALL (L1, L2, and L3) have been recognized [3] .

Chronic lymphocytic leukemia (CLL) was the most prevalent (3.7 percent; middle age 60). Patients under the age of 20 had the greatest rate of acute lymphoblastic leukemia (37.3%), whereas acute myeloid leukemia (0.2%) was the second most common kind (34 percent). At the diagnosis, patients at least 50 years old had the greatest prevalence of chronic lymphocytic leukemia and multiple myeloma [4],[5]. Since patients younger than 20 are more likely to be affected by ALL, our study primarily focuses on this age range. Figure (1.1) display examples of images which used from ALL_IDB dataset.

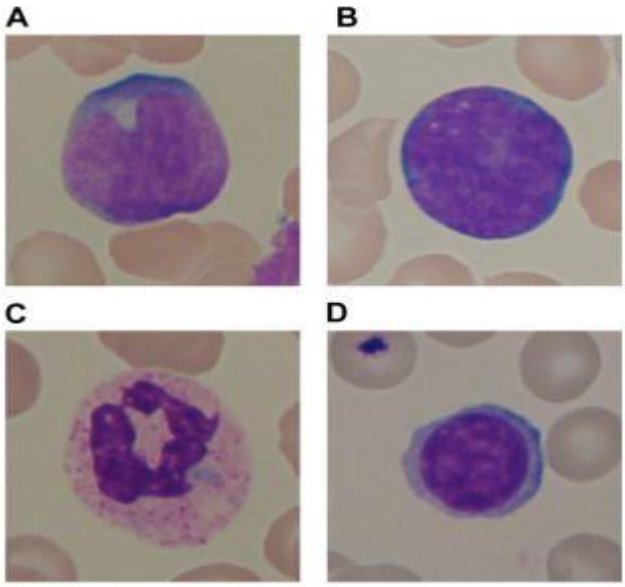

Figure 1.1. Acute lymphoblastic leukemia-Image Dataset (IDB)-1 example images. (A and B) Leukemia cell, (C and D) normal cell. (source: our dataset).

1.2. MOTIVATION

In recent years, there has been a major acceleration in the development of AI algorithms, which has resulted in the availability of computer tools that may address the issue of the objective morphological categorization of PB cells. Deep learning employs multilayered computer models to extract characteristics from incoming data to automatically discover the subjective and intuitive patterns hidden in complicated data structures, such as photos of perifral blood cells.

The fundamental objective of this thesis is to use deep convolutional neural networks to classify acute lymphoblastic leukemia automatically. Utilizing these systems, it will be possible to achieve this goal (CNN). In order to make significant progress in establishing deep learning classification systems for peripheral blood cells, this research is an important part of this purpose.

There are quite a few procedures involved in identifying ALL, and there is a significant risk of making any minor error as there is a big potential of making any slight mistake as we are all aware of the adage "To err is human." We have created a more efficient, accurate technique and less prone to error to detect the symptoms of acute lymphocytic leukemia (ALL) or the presence of immature cells in the blood. This is because most developing countries do not have digital systems. Furthermore, the approach that we are following helps us save a large amount of time, and the reason why we are focusing on time is that the sooner the ALL is identified, the sooner treatment can be begun, and as a consequence, there will be a decreased chance of mortality.

Nowadays, the role of image processing and machine learning in medical research is increasingly obvious. Many medical research and discoveries are created with the assistance of this sector of Computer Science. Many ailments, including those of the breast, the chest, and the blood, may be better understood and treated using high-quality photographs produced by medical imaging. Digital medical photographs provide the possibility for extra analysis for greater accuracy in diagnosis [6-7].

Image processing and machine learning are used in medical research as part of this thesis project. The main impetus is to find a way to determine if a patient has ALL by looking at nucleus size and other characteristics. Abnormal nuclei were identified in the blood sample using a variety of classification approaches. However, we discovered that the Convolutional Neural Network (CNN) has the highest accuracy rate for feature extraction and kernel classification (CNN). Classification is made possible with CNN.

1.3. PROBLEM STAMEMENT

We were able to narrow the issue assertions on two key reasons after investigating a broad range of historical as well as more modern studies and research:

- Pathologists often struggle to arrive at a precise diagnosis of leukemia. The present diagnostic procedures rely heavily on pathologists and their expert opinions, yet there is significant variance in the diagnosis they offer.
- It is rare to use three deep learning models or algorithms and train them on more than one acute lymphoblastic leukemia dataset. Therefore, we found shortcomings in this aspect through our review of many studies and previous research in this field.

1.4. OBJECTIVE

The first goal of our study is to use Deep Learning algorithm method to automatically diagnose and group ALL without the help of doctors or specialists. This is because we know that mistakes can happen when blood slides are used to make a diagnosis in a lab.

The second goal is to create and develop an efficient grading system utilizing three models (DenseNet 169, VGG19, and a simple CNN model). This system will use high-resolution data from blood slides to tell the different types of leukemia apart, and receiver operating feature analysis will be used to see how the network's output affects how quickly pathologists can tell the difference. It can be used as a tool to help doctors grade leukemia and as a way to help them make decisions. Its goal is to make it easier

to take the right preventative and treatment steps, with the end goal of making future diagnoses easier and more accurate.

1.5. INCLUSION CRITERIA

It was decided to look at articles that examined the use of microscopic pictures or flow cytometry in the diagnosis of leukaemia (including but not limited to AML, ALL and CLL). This analysis only included data from studies that looked at a single subject at a time. As a consequence, technical/methodological investigations and review studies were excluded.

Information on model and algorithm validation was necessary. Cross-validation, for example, may be used internally or externally to accomplish validation (for example, through the use of a new validation set or prospective validation). It was decided that only papers that provided their whole texts were eligible for inclusion; abstracts were not. According to the schedule, the publishing year will begin in January 2017 and end in January 2020. English-language content was the sole kind of content assessed for inclusion.

PART 2

LITERATURE REVIEW

We study a significant number of publications on the issue to acquaint ourselves with the methodologies employed by other academics. There has been an increase in the incidence of blood cancer, and medical procedures to detect any blast cell are too sensitive and time-demanding. Therefore, researchers had to develop a computer-aided diagnostic to identify blast cells.

Leukemic blood cells may be identified via morphological analysis of microscopic pictures, according to P. M. Gumble [6]. Because morphological analysis does not need a blood sample, it is appropriate for low-cost and remote diagnostic systems. First, leukocytes and other blood cells are found. Next, lymphocytes (those linked to acute leukaemia) are chosen, and finally, leukaemia is established. With this method, each blood cell produces two enhanced pictures, one of the cytoplasm and the other of the nucleus. One may infer from the two pictures that each kind of leukaemia has unique traits. As pointed out by P. M. Gumble, out of 66 samples that the KNN classifier successfully recognized, 72 samples were taken. The system's accuracy rate is 91.66 %.

Angelo Genovese at el. [7] proposed the first machine learning-based approach to sharpen blood sample photos adaptively. After applying DL and image processing to analyze focus quality, adaptively sharpen images, and normalize radius, classification is performed. They tested the concept using a publicly accessible database of ALL pictures and many state-of-the-art CNNs.

Kazemi et al.[8] successfully detected acute myelogenous leukaemia with a classification accuracy of 96 % by employing support vector machines. This allowed them to diagnose the disease accurately. Distinguishing characteristics including

irregularity, the nucleus-cytoplasm ratio, and the Hausdorff dimension are recovered from all the nuclei in whole pictures containing multiple cores using Image preprocessing techniques like colour segmentation. Pictures are classified as malignant or benign using an SVM classifier using a 10-fold cross-validation technique. A classifier's accuracy, sensitivity, and specificity affect how well it performs. SVM classifiers also allocate cancerous pictures to their most prevalent subgroups.

Rezato ghi et al.[9] suggested that a new approach can segment white blood cells from smear pictures. The segmentation technique offers to identify a zone of different white blood cells in HSI colour space. White blood cell nucleus and cytoplasmic granules are coloured pixels in an ellipsoidal region. White blood cells may be retrieved from a smeared picture using another morphological method. Geometrical, colour, and LDP-based texture properties are retrieved from segmented cells. Three neural networks use these properties to detect white blood cells. The categorization technique was tested on 450 white blood cell pictures. Neural networks may benefit from the use of a support vector machine. An accuracy of 96 % was achieved using the SVM., also trained NN and SVM models. The NN was most accurate 99.9 % of the time.

An aggregated deep learning model for the categorization of leukemic B-lymphoblasts was to be developed by Kasani et al. [10]. Transfer learning and data augmentation methods were used to make up for the small dataset size and improve the proposed network's performance to build a reliable and accurate deep learner. Our suggested strategy, which contained characteristics from the top deep learning models, successfully diagnosed leukemic B-lymphoblasts with a test accuracy of 96.58 %.

Rehman A et al. [11] established a robust segmentation and deep learning approach using a convolutional neural system trained on images of bone marrow. These results might be compared with other classifiers like Nave Bayesian, K-Nearest Neighbor Algorithm (KNN), and Support Vector Machine (SVM) for experimental purposes (SVM). According to the experimental results, accuracy was found to be 0.9778.

In the research done by Sivalingam et al.[12] the pictures are first preprocessed using a Type 2 fuzzy and cuckoo search-based filter. Deep fuzzy clustering is performed to

segregate cells in preprocessed blood smear images. Random Multimodal Deep Learning (RMDL) is used to detect CLL (RMDL). For RMDL training, it is recommended that you use Jaya-competitive swarm optimization (Jaya-CSO). The Jaya-CSO combines a competitive swarm optimizer with Jaya optimization (CSO) (CSO). Thus, lymphocytic leukaemia is recognized using the RMDL provided by Jaya-CSO. The recommended Jaya-CSO-based RMDL demonstrated maximum accuracy of 95.1 %, the most incredible actual negative rate of 95.4 %, and the highest valid positive rate of 95.2 %.

To help with the detection and classification of Acute Leukemia, Giao N. Pham et al.[13] used CNN to classify and extract features from raw pictures. This approach was also used in another research. This research suggested a four-layer network. Segments are acknowledged in the first three layers, and their identification is accomplished in the last Layer, which contains two layers of neural processing (Fully connected and Softmax) (Fully connected and Softmax) (Fully connected and Softmax). The experiment was carried out in Matlab, and the accuracy was 0.9643.

According to Preeti Jagadev et al. [14] study, image processing and machine learning were utilized to diagnose several types of leukemia. The smear images are first segmented using k-means clustering, marker-controlled watershed, and HSV colour-based segmentation. Heterogeneous characteristics are retrieved from segmented lymphocyte pictures because the morphological components of normal and leukemic cells vary significantly. Support Vector Machine (SVM), a machine learning classifier, predicts which of four types of leukemia a sample possesses.

Leukaemia-affected cells were investigated by Himali P. Vaghela et al.[15]. In their investigation, researchers used various techniques to detect immature cells. One involves extending a linear watershed using a clustering method and histogram equalization. Monocytes, lymphocytes, eosinophils, basophils, and neutrophils are a few examples of white blood cells that may be identified using shape-based metrics such as area, perimeter, roundness, and standard deviation. Images may be processed in a matter of minutes using image processing methods. Shape-based feature operations can more accurately identify white cells when the RGB image's contrast is

improved. To determine if a cell is CML or CLL, each is recognized and categorized (chronic Lymphocytic leukemia). This technique was used to process thirty photos. It succeeds on 28 out of 30 images, it is thus 93.33 % accurate.

Mohapatra et al. [16] could recognize lymphocytes by modifying the RGB picture color space and using a shaded C-mean clustering technique that recognizes each image pixel in the background, cytoplasm, and nucleus sections. It was determined that the cytoplasmic and nuclear features had been measured and computed. Total number Naive Bayesian, KNN, NN and SVM classifications are examined. A simple majority vote was used for the EOC classification (KNN, NN, SVM). Compared to individual ALL categories, EOC is on average 94.73% more accurate.

Deep learning networks were used by Yu et al. [17] to build a classification algorithm for WBCs. There were seven distinct WBC kinds in this sample, which evaluated this strategy against various more traditional ways. The accuracy percentage of 88.5 % of the obtained data showed that a CNN is superior to other approaches.

The Vision Transformer model was used to categorise white blood cells and was created by Cho, Priscilla et al. [18]. The Vision Transformer outperformed modern convolutional neural networks in classification performance, using less computing power during the training phase. The latent self-attention architecture further produced attention maps for a specific image. These attention maps provide cues as to which visual elements were crucial for making decisions. When they applied a convolutional neural network model and the Vision Transformer model to a dataset for classifying acute lymphoblastic leukaemia with 12,528 samples, accuracy scores of 88.4% and 86.2% were reached.

Dey, Umid Kumar, and Md Sajjatul Islam [19], researchers want to identify each person's specific leukaemia using gene expression data and three machine learning algorithms. XGBoost, Random Forest Classification, and Artificial Neural Networks are the three techniques. Principal component analysis was used on the dataset to minimize its dimensionality (PCA). The genetic expression profiles of 72 people, each with 7129 genes, were gathered via the website Kaggle. The Precision number, on the

other hand, shows how well the model predicts that a person satisfies all the conditions. For Random Forest, this percentage is 73.7 %. Both opinions are equally correct for apparent reasons.

The investigation results of Wang et al. [20] prompted the development of a technique for recognizing and categorizing WBCs that used an ensemble classifier. This classifier combines the results from many separate CNNs. In the best-case scenario, experiments employing a dataset with 3000 images for each category generated an average classification accuracy of 99.37 %.

An alternate system for classifying ALL and its subtypes were given by Pansombut et al. [21]. The researchers' CNN network, known as ConVNet, was used in this technique. The assessment was effective in reaching an accuracy of 80% using a dataset of 363 images. The results of the trials showed that this approach is much more effective than conventional methods.

Sezgin [22] employed grayscale pathology photos and established reference values for the image. By first converting to binary representation and determining the area of any remaining white patches, we may perform the diffusion process point-by-point. In the region's accounts, they had a 96.5 % success rate due to their efforts.

HLAC, wavelet transform, and Delaunay feature extraction on pathology pictures give information on cluster frequency, shape, and position for Ishikawa et al.[23]. They utilized features and the SVM technique to detect if a cell was cancerous. This research query had a 94.6 % sensitivity and an 84.9 % specificity.

Filtering data on lung nodules revealed a pattern observed by Demir et al. [24]. They employed particle swarm optimization and the Gray Level Cooccurrence Matrix (GLCM) to enhance the SVM approach to identify the outer surface texture characteristics. When performance value is examined by including external compositional aspects, they are effective with a rate of 98.3 % as a direct consequence of their efforts. However, the computer-aided diagnostic (CAD) system could not distinguish nodules next to the lung wall or beyond the grey level threshold values.

Image processing methods were used by Selcuk and Ozen [25] to differentiate between 130 healthy cells and 130 malignant cells. After the wavelet transformation, the pictures were sent through an SVM for classification. Because of this, they have a success rate that is 93.91 % of the time.

This was accomplished by Abdoulaye et al. [26] by processing the mammograms in three steps and reducing background noise. The mass's physical attributes were located, and the pattern was derived using the image's area, perimeter, and diameter. Using an AI-powered algorithm, the researchers realized that the trend they saw might be leveraged to construct a cancer detection system.

A CNN classifier was used by the researchers Pansombut and Tatdow [27] to test the feasibility of identifying lymphocytes and ALL subtypes using deep learning. The effectiveness of this strategy was assessed by contrasting it with the standard support vector machine (SVM) methodology, which required manually creating the features. Additionally, the comparison is conducted using the multilayer perceptron (MLP) and random forest classifiers, two common machine learning classifiers. The investigations show that their CNN classifier performs better when distinguishing between typical lymphocytes and pre-B cells. This illustrates that picture categorization has a big potential without requiring a sizable number of feature engineering-based preprocessing procedures.

Chong et al. [28] created a method for diagnosing malignant haematological illnesses using deep learning that just needs slide-level labelling. This approach utilised deep learning and had patchy supervision. This method increases the process efficiency by converting the whole-slide image (WSI) patches into low-dimensional feature representations. Then, each WSI's patch-level features are combined into slide-level representations using an attention-based network. These slide-level representations are the model's input, and it produces conclusive diagnostic predictions utilising them. Additionally, over two publicly accessible datasets, the performance on microscope images may achieve an average accuracy of 94.2%.

Deepika Kumar et al. [29] preprocessed the images and then extracted the most pertinent features, using cell images to train the suggested model. After the model has been trained using the ideal dense convolutional neural network (DCNN) framework, the type of cancer in the cells is predicted. The model had a sample memory of 100% and a reproducibility of measurements of 94%. Overall accuracy was measured at 97.2%. The DCNN model appears to reach performance levels comparable to those of CNN architectures created with substantially fewer parameters and processing time based on the study's findings on the returned data set.

José Elwyslan Maurcio et al. [30] proposed some fundamental alterations to the standard neural network topologies in order to achieve great performance in the malignant leukocyte categorization test. Testing involved putting the architectures VGG16, VGG19, and Xception to the test. The F1 test resulted in a score of 92.60% thanks to the suggested approach.

Because of the work of Zhao et al.[31], it is now feasible to recognize and categorize five distinct kinds of White Blood Cells automatically (WBC). Color and morphological approaches were used to detect WBCs. The SVM classification approach was used to classify the picture as eosinophil, basophil, and others, while the CNN network was used to recognize the other three species (others). Finally, they used the Random Forest (RF) approach to classify the various kinds of WBCs.

Classification of Normal vs Malignant Cells in B-ALL White Blood Cancer Microscopic Images was the ISBI issue that Christian Marzahl et al. addressed in their study [32]. They use transfer learning with varying rates of learning adaptation along with advanced augmentation techniques to address this classification challenge. This prevents the issue of overfitting the small dataset. The ResNet 18 network, after being improved with one additional regression head for the final test set, achieved a weighted F1 score of 0.8284. It received a 0.8746 score on the practise test set.

Using the ALL-IDB and Cella Vision databases, Sipes and Li [33] categorized ALL kinds of cancer. The photographs were then used in RGB color space and shrunk to 257 by 257 pixels. For training, validation, and testing, the dataset was randomly split

into 60:20:20 chunks. KNN, NN, and CNN are used for categorization. The CNN model had the highest success rate, at 92 %, while the KNN model had the lowest and most variable accuracy, at 81 %.

Table 2.1. Review of related literature.

Reference	Classifier	Description	Accuracy
P. M. Gumble [6]	KNN	Morphological analysis for 72 samples of blood.	91.66 %
Kazemi et al. [8]	SVM	Image preprocessing and colour segmentation strategy	96 %
Rezato ghi et al. [9]	SVM	Geometrical features, colour features, and LDP-based texture features	96 %
Kasani et al. [10]	VGG19	Transfer learning technique to the detection of B-lymphocytes.	96.58 %
Rehman A et al. [11]	Naive Bayesian, SVM, and KNN	Robust segmentation and deep learning approach.	97.78 %
Sivalingam et al. 12]	Jaya-CSO	Type 2 fuzzy and cuckoo search-based filter.	95.1 %
Giao N. Pham et al. [13]	CNN	Classification and extraction features from raw pictures	96.43 %
Himali P. Vaghela et al. [15]	Clustering Algorithm and Histogram Equalization	Shape-based features extraction and identification	93.33 %
Mohapatra et al. [16]	SVM, KNN	Recognize lymphocytes by altering RGB and utilizing shaded C-mean clustering.	94.73 %
Yu et al. [17]	CNN	Classification system for WBCs	88.5 %
Dey et al. [19]	XGBoost, Random Forest Classification, and ANN	Gene expression analysis	73.7 %
Wang et al. [20]	Ensemble classifier	Classification of 3000 images of blood	99.37 %
Pansombut et al. [21]	CNN	Classifying ALL and its subtypes by ConVNet technique.	80 %
Demir et al. [24]	SVM	Particle swarm optimization and the Gray Level Cooccurrence Matrix (GLCM)	98.3 %

Selcuk and Ozen [25]	SVM	Wavelet processed	93.91 %
Chong et al. [28]	ResNet50	Converting the whole-slide image (WSI) patches into low-dimensional feature representations	94.5 %
Sipes and Li [33]	KNN, CNN	Classification and feature extraction	Accuracy for KNN: 81 % Accuracy for CNN: 92

PART 3

DEEP LEARNING

Deep Learning is a subset of machine learning in which massive volumes of data are used to teach artificial neural network methods inspired by the synthesis of the human brain. Like humans acquire knowledge via experience, the deep learning algorithm will do a job often when it is adjusted significantly to improve the outcome [34],[35].

The capability of neural networks to gain information on a multilayered or "deep" foundation is what's meant when we talk about "deep learning." Deep learning can teach itself to solve any issue that requires "thinking."

Because deep learning algorithms need a significant amount of data for training purposes, recent years have seen an improvement in deep learning capabilities.

Fast learning is the goal of deep learning algorithms. For computationally demanding tasks, multi-GPU/multi-processor clusters may allow faster deep learning model training. It is possible to handle enormous volumes of data and provide more relevant outcomes by using these models [34,36].

In addition to generating more data, deep learning algorithms take advantage of the most powerful computing power currently available and the evolution of artificial intelligence. It makes AI accessible to smaller businesses as a service, enabling them to buy AI equipment and find the AI algorithms required for deep learning with little up-front investment [37,38].

3.1. CONVOLUTIONAL NEURAL NETWORK

When a network is described as "convolutional," it indicates that it employs the mathematical operation known as convolution. It is a particular instance of the linear procedure known as convolution. At least one of the layers of a convolutional network replaces matrix multiplication with convolution [38].

In addition to an input and output layer, convolutional neural networks include several layers of hidden information. To know what happens behind the scenes at CNN, you must understand how it operates. After the activation function, which is often a RELU layer, further convolutions, including pooling layers, fully connected layers, and normalization layers, are added. These extra convolutions are described as "hidden layers" because the activation function and final convolution obscure their inputs and outputs [39].

To get a more accurate final weight, backpropagation is often used during the final convolution. The layers are referred to as convolutions only as a matter of custom. In mathematics, it is a sliding dot product or cross-correlation [38],[40]. This affects the computation of values at a particular index location in the matrix. Figure (3.1) explain all layers in the CNN.

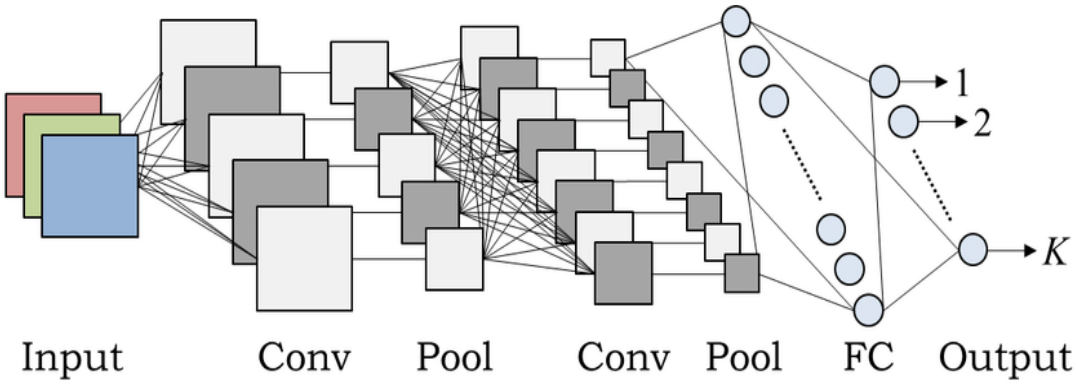

Figure 3.1. Architecture of CNN. (source: (Hidaka, 2017) [40].

3.1.1. Convolutional Layer

The fundamental components of a CNN are convolutional layers. The characteristics of the Layer include filters (or "kernels") with a restricted receptive field that extends to the depth of the input volume. During the forward pass, a 2-dimensional activation map of each filter is constructed by converging it across the width and height of the input volume. This results in a dot product between the entries of the filter and the input. Therefore, the network builds filters that are engaged at certain input points when a particular feature is identified [41].

The whole output volume of the convolution layer is created by stacking the activation maps for all filters along the depth dimension. Therefore, it is assumed that all output volume entries reflect outputs from neurons that share parameters with neurons in the same activation map [42] and neurons that examine a small amount of their input.

3.1.2. Max-Pooling Layer

The process of pooling may be thought of as "downscaling" the picture derived from the layers that came before it. One analogy would be to lower the number of pixels in a picture by decreasing its size.

Max Pooling has been the most used pooling approach to date. You may pool in a two-to-one ratio if you so want. As a consequence, the height and width of your image will be cut in half. Therefore, you must compress every four pixels (on a 2x2 grid) and remap them to a single-pixel without losing any "important" data included in the missing pixels. The largest value of these four pixels is utilized for the Max Pooling operation. Therefore, the largest value of the four pixels signifies the addition of one additional pixel. It is performed for each group of four pixels in the whole image [39],[43],[44].

3.1.3. Flatten Layer

The flattening phase is a procedure that is involved in the development of a convolutional neural network. It requires taking the pooled feature map that was produced in the stage of pooling and converting it into a one-dimensional vector. This step follows the stage of pooling. It does this by taking the output of the convolutional layers and flattening all of its structure into a single lengthy feature vector. This vector is then sent on to the dense layer, which uses it to make the final classification [39],[45].

3.1.4. Rectified Linear Unit Layer (ReLU)

A function for activation is a Rectified Linear Unit (ReLU). It is expressed mathematically as y = maximum (0, x)... ReLU is a common activation function in neural networks, especially convolutional neural networks (CNNs). ReLU is an excellent starting point if you are unsure about implementing the network activation function [46].

3.1.5. Dropout Layer

With the billions of neuron nodes communicating with each other, software-based neural networks must find ways to reduce the noise. Any messages that aren't related to the problem or training being worked on may be eliminated by a network. The elimination of a neuron node is referred to as a "dropout" [47].

Dropout Layer is often used in deep learning models to reduce the risk of overfitting. When the training data are more accurate than the test or anonymous data, a phenomenon is known as "model overfitting" occurs. This phenomenon results in the model having a lower accuracy [48].

It is feasible to eliminate some of the neurons found in both the visible and the hidden layers by using an approach known as a dropout. According to the results of the trials,

the neural network model may be made more resilient by the use of dropout, which regularises it so that it does not overfit [38],[49],[50].

3.1.6. Batch Normalization Layer

The channels of a mini-input batch are normalized using a batch normalization layer. Because of its interaction with convolutional layers and nonlinearities like ReLU layers, batch normalization layers may help lessen the impact of an unbalanced network initialization and speed up the training of convolutional neural networks.

Channel activation levels are normalized by dividing the mini-batch standard deviation by the mini-batch mean. The Layer then adds a learnable offset and scale factor, both of which are learnt [51].

3.1.7. Fully Connected Layer

Three more layers were linked to the flattened input layer; one was concealed, and the other two were output; all four were hidden from view. Dropout and Batch-Normalization are used to minimize overfitting in our model's hidden Layer of 128 nodes. A simple computation has improved the activation function of the ReLU. A basic calculation led to the discovery of the ReLU activation function, which had been determined before [40],[52].

3.1.8. Adam Optimization

Diederik P. Kingma and Jimmy Lei Ba introduced the Adam technique, also known as the Adaptive Moment Estimation Algorithm, in 2015. This approach optimizes a function by calculating moments and using them as inputs. It is a mix of the gradient descent with momentum and RMS Prop techniques, as the name indicates.

Using the Adam technique, a weighted exponential moving average of the gradient is calculated and then squared. Two decay factors govern the decay rates of these moving averages in this approach [53,54].

3.2. CLASSIFICATION MODELS

Most classifiers use a discriminative learning strategy that divides the training set into several categories. A decision boundary is a term for this separation. The classifier may utilize a linear model to establish the decision boundary for a two-dimensional feature space with classes separated by a straight line [55].

A linear model may still be useful if the feature space is three-dimensional or contains more complex two-dimensional class distributions. Curves or hyperplanes serve as the decision-making boundaries in this instance. Higher-order qualities are produced from the supplied ones to build a complicated decision boundary. Features may be squared, cubed, or multiplied in many ways to achieve this. Since the parameters are maintained constant, the model's linearity is unaffected even when nonlinear components are added [49],[56].

Despite this, linear models are not the best choice for categorizing data because their continuous output for discrete values in classification problems is uncertain. Thresholding rules remove some disparities. However, they don't work when training data is dispersed over the feature space. Each pixel may serve as an input feature to the model for classifying images [57].

PART 4

METHODOLOGY

This chapter uses convolutional neural network models to describe the image dataset and the processes used for binary classification of aberrant lymphocytes and normal WBCs. These methods were used to distinguish between abnormal lymphoblasts and WBCs typically seen in the blood.

4.1. DATASET PREPARATION

4.1.1. Platform Used

An NVIDIA GeForce GTX 1060 graphics card and Windows 10 operating system powered our Lenovo laptop for this project. The computer has an Intel(R) Core (TM) i5-7200CPU running at 2.5GHz/2.7GHz, 64-bit 6th generation, 8GB of RAM.

4.1.2. Initializing Libraries

The project was executed utilizing a variety of software packages:

- **Keras**
  It is a Google-developed, high-level API for deep learning and neural network development. Python, a programming language, is used to expedite the building neural networks. This technology may generate many neural networks in the background [49],[58].

- **Tensorflow**

  TensorFlow is an open-source artificial intelligence library. There is supplementary assistance for conventional machine learning. TensorFlow was created to do intricate mathematical computations. By importing this library into our Python environment [58],[59], it may be used for various instructional objectives.

- **Scikit learning**

  It is the best Python-based machine learning library. The Python implementation of machine learning and data mining covers a variety of methods, including classification, regression, grouping, and dimension reduction, API to ensure uniformity [60],[61].

- **Numpy**

  The library's use of array structures and linear algebra algorithms is crucial to deep learning. Scientific Python packages like SciPy and Matplotlib often use NumPy. This combination often supplants MatLab, a popular scientific computing environment [62].

- **Matplotlib**

  This Python research package contains the array structures and linear algebra needed for deep learning. NumPy is often used by scientific Python libraries like SciPy and Matplotlib. MatLab, a well-known scientific computing environment, often uses this combination[62],[63].

4.2. DATASET DESCRIPTION

ALL_IDB and C-NMC-ALL datasets were employed as the two primary data and information resources throughout the training of the models suggested by this research. Utilization of the internet as a means to get access to any of these two data collections is a possibility.

You may get each of these data sets by searching on the internet for those specific terms. These data sets include a range of blood photographs showing healthy cells and photos of acute lymphoblastic leukaemia. In addition to that, the visual representations of blood are included in these data sets. A portion of these data sets is shown in figure (4.1), where each sample is represented as either being healthy or diseased.

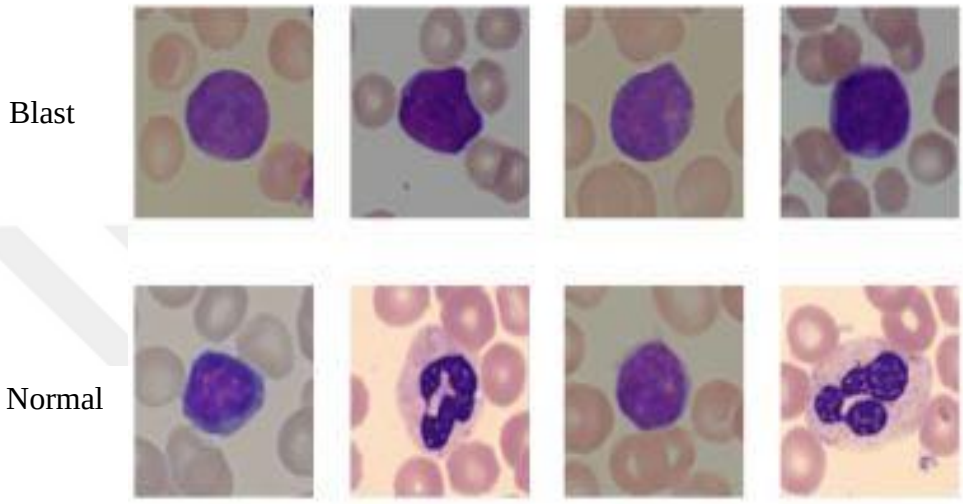

Figure 4.1. Display of images of Normal and lymphoblasts. (source: our dataset).

4.2.1. ALL_IDB Dataset

Images of lymphocyte cells, healthy people, and patients with acute lymphocytic leukaemia were used in the study. A copy of the ALL_IDB dataset had been downloaded, and the file's owner had given their consent [64]. Public access is given to the ALL_IDB dataset, which contains microscopic images of peripheral blood cells. These photos were created from scratch to aid segmentation, evaluation, and classification. Data in the database is generally reliable since oncologists have annotated it. The ALL_IDB database has two different versions (ALL_IDB1 and ALL_IDB2).

The ALL-IDB2 database was used to conduct the experiments in this paper. ALL_IDB2 is a collection of cutout regions of interest taken from blood smear photographs of healthy persons and leukaemia patients that are a part of the ALL-IDB1 dataset. These patients have been picked at random. The ALL_IDB2 dataset comprises

a subset of 260 segmented photos, with the remainder featuring cancerous cells and fifty per cent including normal leukocytes.

ALL_IDB dataset includes 2083 photographs, with 1034 showing people in good health and 1049 showing those in bad health (blast). Figure (4.2) reveals the dataset's distribution of images (blasts and normal cells).

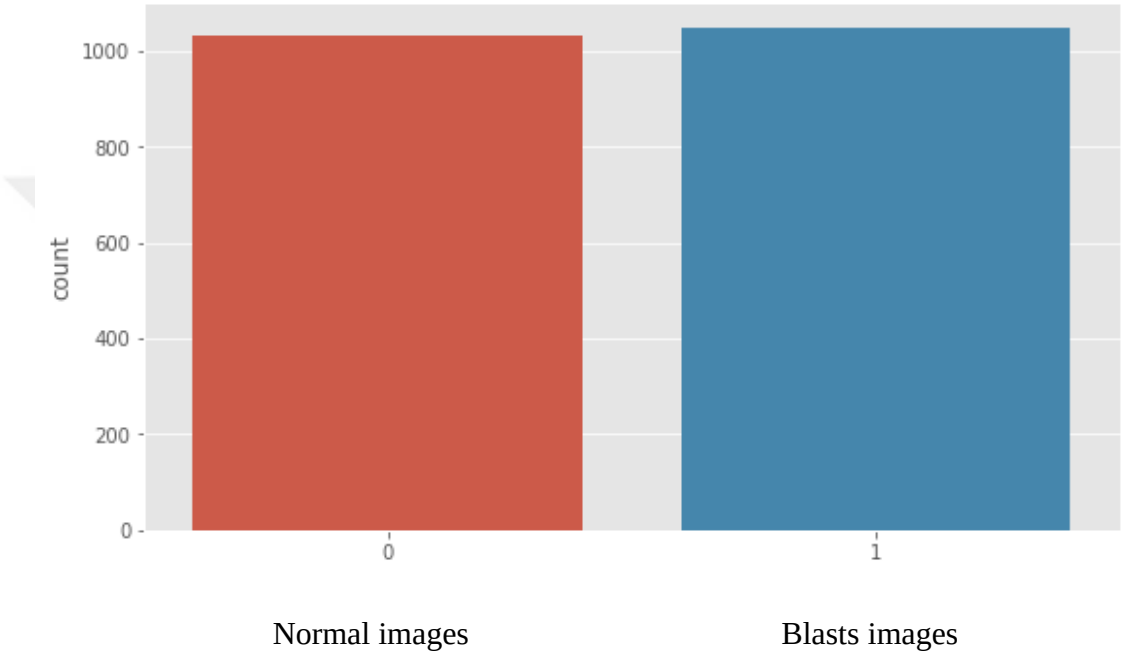

Figure 4.2. Display the distribution of images in ALL_IDB dataset.

4.2.2. C-NMC ALL Dataset

The second dataset, called C-NMC ALL, has 10.661 images, of these, 3.389 are healthy images, while 7.272 are sick images known as blasts cells.

For B-Lineage acute lymphoblastic leukemia patients[65], compiled and curated a collection of microscopic images of cancer cells and healthy cells (B-ALL).

The collection, generated at the subject level, includes images of healthy people and cancer patients. This is the most comprehensive data set ever accessible to the public

on B-ALL cancer. The Cancer Imaging Archive (TCIA) in the United States has C-NMC, which helps develop B-ALL cancer diagnostic tools.

Immediately after Wright staining, microscope slides were created. To study the falls, the researchers employed a bright-field microscope with a resolution between 300 and 500 micrometres. It captured the images at 2592 x 1944 pixels using the Canon PowerShot G5 camera attached to the microscope. subimages of individual leukocytes with a resolution of 257 x 257 were cropped and saved as TIFs from these photographs. For every photo classified as normal leukocytes (Y = 1), there was a picture classified as abnormal lymphoblast (Y = 1). PATIENTS had no views, whereas healthy persons had at least one photo. Figure (4.3) reveals the dataset's distribution of images (blasts and normal cells). All images include erythrocytes in the centre, with various backlights, and are saved as PNGs (table 4.1).

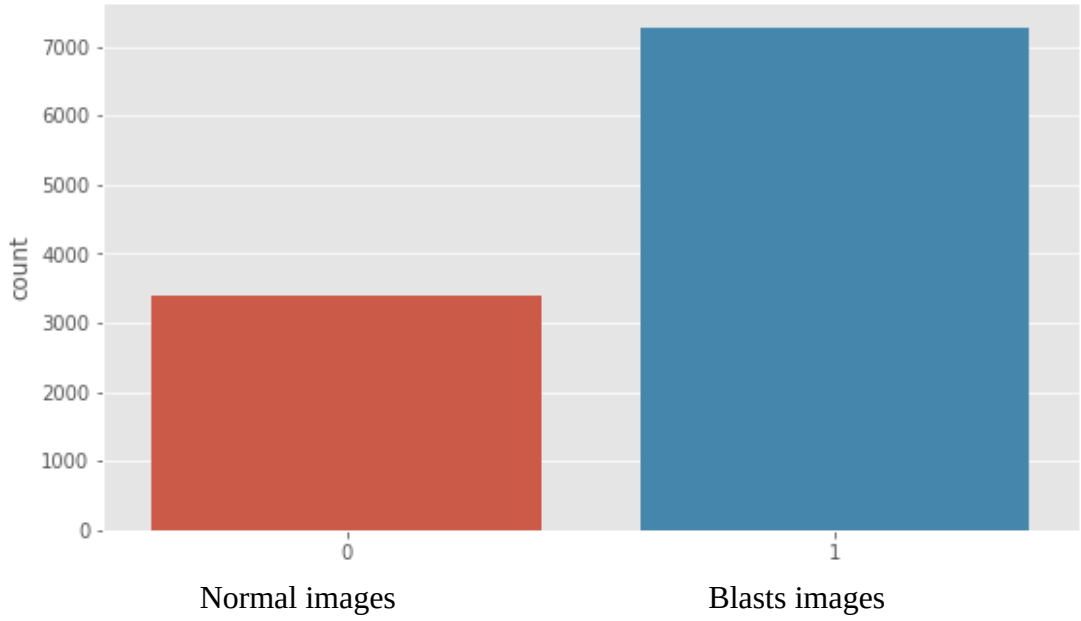

Figure 4.3. Display the distribution of images in ALL_IDB dataset. (Source: from our code).

Table 4.1. Datasets Characteristics.

	Normal images	Blasts	Total	Resolution
IDB-1	1034	1049	2083	257 x 257
C-NMC ALL	3389	7272	10661	257 x 257

4.3. DATA PREPROCESSING

To improve the efficiency of the model training process, the conventional preprocessing method of removing the mean of the cell pictures is used. The mean cell image from the training set was calculated, and then this value was subtracted from every photo in both the training set and the test set (Figure 4.4). The visuals are produced using the RGB color space, and the values of each pixel may take any value between 0 and 255 at any given time. The data did not need further standardization since they were already within a limited range.

4.4. DATA SPLITTING

A DL model's assessment may be complex. The dataset is often split into two independent sets for testing and training. Two groups are employed to train and test the model: one for the training and one for the testing. The model's performance and accuracy are both evaluated using an error measure. Contrarily, traditional approaches may provide wildly different answers for the same test set. K-fold cross-validation [66] may be used to resolve this issue. It separates the data into many folds so that each can be used as a testing set.

In this study, the pictures included within each of the two data sets were divided into three groups: a validation set (consisting of 5 %), a test set (composed of 20 %), and a training set (consisting of 75 %).

4.5. DATA AUGMENTATION

The proposed technique uses TensorFlow data augmentation functions like random flip (up-down and left-right). Using augmentation is the only way to teach an image to be accurate in real-world situations. Due to a feature that randomly flips photos up and down, our model can accurately identify leukaemia-affected cells even if the training dataset does not include any of these cells [67].

Depending on the relationship between the two images, one picture is shifted to the other. Tf. Random. Uniform () transposes float32 data if the spatial value is greater than or equal to 0.75. In addition, the pixel size affects several colors, brightness, and contrast factors. The test shots may seem brighter or more saturated than the example photographs. As a result, this data enrichment is quite beneficial [68].

Cropping capabilities enhanced the model's prominence in the cell by shrinking and reshaping all photographs to fit the model. Tensor-formatted images have to be decoded from bitmap image files (bmp). A tensor is a multidimensional array. Expressing a video or photo using an array of sensors is possible. In this way, the model can understand the idea better. Eliminate any data duplication and sorting throughout the training phase to ensure objectivity. When a model is trained with the same image repeatedly, it is more likely to make predictions about test cases based on the same idea [69].

Using NumPy, all photos and labels were initially turned into a single data structure. The training and augmentation batch sizes are the same. Matplotlib software was used to plot the images once the augmentation technique had been confirmed. Figure (4.4) shows the datasets after they've been augmented.

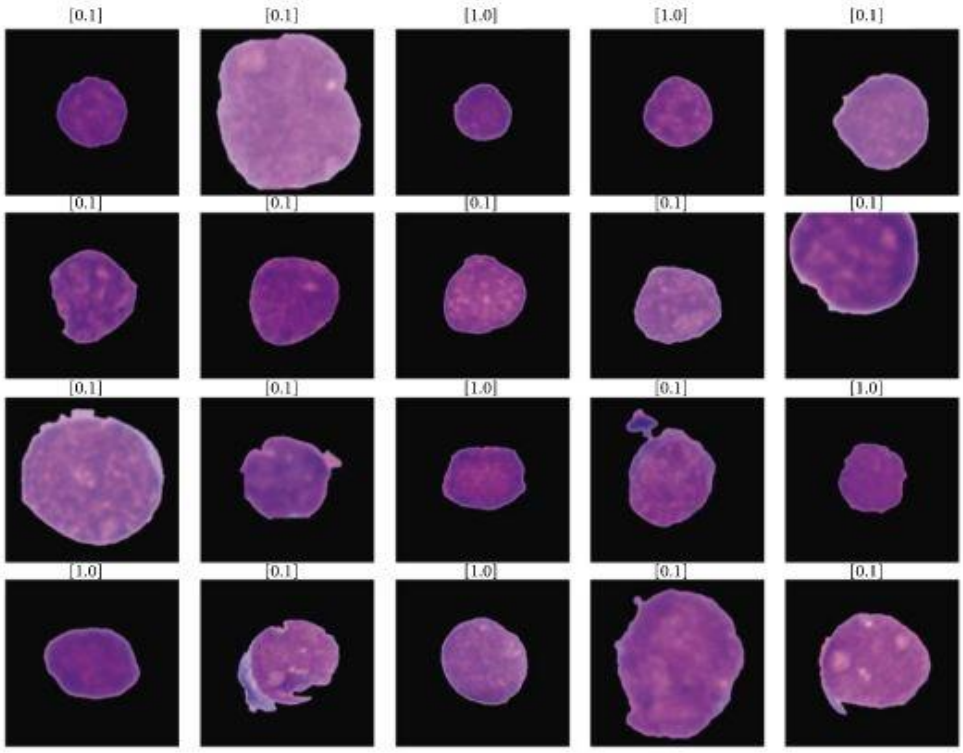

Figure 4.4. Sample of augmented images (Source: from our code).

CNN-based image segmentation investigations have employed information replication to reduce the model error rate. There is a greater variety of microscopic blood cell images in the retrieved datasets, although only a few leukaemia subtypes may be represented there.

Because of the limited sample datasets accessible to deep learning systems, overfitting may occur. As a result, data augmentation and image alteration are used to expand the dataset's size. After applying image processing techniques to both datasets, the total number of observations has increased. The number of samples for each kind of leukaemia before and during therapy significantly increased.

4.6. FEATURE EXTRACTION

The input training data is fed into convolution layers, which are utilised to extract features. The feature extraction procedure is aided by a set of filters that are applied to each convolutional layer. A CNN model's convolution layers, in general, teach more complicated features as the model's depth grows. For example, the first convolution

layer gathers the most fundamental features of the training data, whereas the last convolution layer gathers the most complex aspects [70].

To examine how well models can learn features from whole color images and the raw pixel values of those photographs, the standard feature extraction processes used in cell biology were sidestepped. Methods similar to these may be discovered in the discipline of cell biology [71] .

4.7. TRANSFER LEARNING

Deep CNNs are employed in many applications because they can learn complicated visual representations. A large amount of data is required for feature extraction in medical image-related concerns. Overfitting occurs because of a lack of leukemia categorization data. To get around this problem, the researchers in this study used a transfer learning method. Even with a short dataset, transfer learning may be able to overcome the problem of insufficient samples [72-74].

To enhance a network, you may use transfer learning to move parameters from one domain to another. Figure 5 depicts both traditional machine learning (ML) and transfer knowledge. Formal machine learning (ML) models are built from the ground up, meaning more data is needed to get an extraordinary performance score. However, because of the transfer learning approach, the model already has information from the source work, and very little data is needed for the target job to receive high scores in performance matrices [74].

Data from the source domain must be weighted differently to be useful in the target domain. Transfer learning may beat scratch learning since the pre-trained model already contains a lot of basic knowledge. Transfer learning is a strategy that utilizes the transferred environment to acquire insight into both low- and mid-level characteristics. A little data from the new domain is needed to meet this study's goals [75],[72].

This research used transformation learning instead of a massive number of data points. Throughout the training, specific well-known pre-training models were used. Machine learning models typically fail to generalize effectively when applied to untrained data. While it fails in certain circumstances, it performs somewhat better than terribly in others. Cross-validation, a resampling strategy, ensures that the model works well on unknown data. This resampling procedure divides the data into k groups of comparable size [76].

The rest of the k-1 sets were used to train the model, with the first set acting as a testing ground. Afterwards, the test error rate may be calculated. Data from the first iteration may be used to train the second iteration and identify errors. There are k groups that must be handled. In cases of class imbalance or problems with classification, the k-fold class CV might be helpful.

The relative category ratio is preserved as close as possible with each training and validation. As the stratification increases, so does the issue. Compared to K-fold cross-validation, conventional K-fold cross-validation has a more significant variance and takes longer to compute because the MSE (mean square error) will include fewer variables when additional data points are utilized for nullification (variance) [76] [77].

4.8. PROPOSED SYSTEM

After collecting attributes from the photos, we classify raw blood cell images using CNN. The bulk of CNN layers are convolutional, pooling, and completely connected. To get an output value from neurons using convolution layers, which first compute a weighted sum of inputs and then apply bias to that weighted sum, the rectifier linear unit (ReLU) is required. The value of ReLu, which determines whether a neuron is active or inactive, may be determined by solving the equation ReLu (z) = maximum (0, z).

Using pooling layers to limit the number of calculations and parameters reduces the spatial size of the representation, which ultimately results in control overfitting. Each

neuron is connected to all activations in the Layer below in layers with total connectivity.

In this study, the transfer learning method was used by training the models DenseNet 169 and VGG 19 on the PCam dataset, as mentioned in (Vulli, Adarsh, et al. 2022).) we used three separate CNN models and trained each using the transfer learning approach. Figure (4.6) show the method of training the model using the learning transfer mechanism from training a previously trained model.

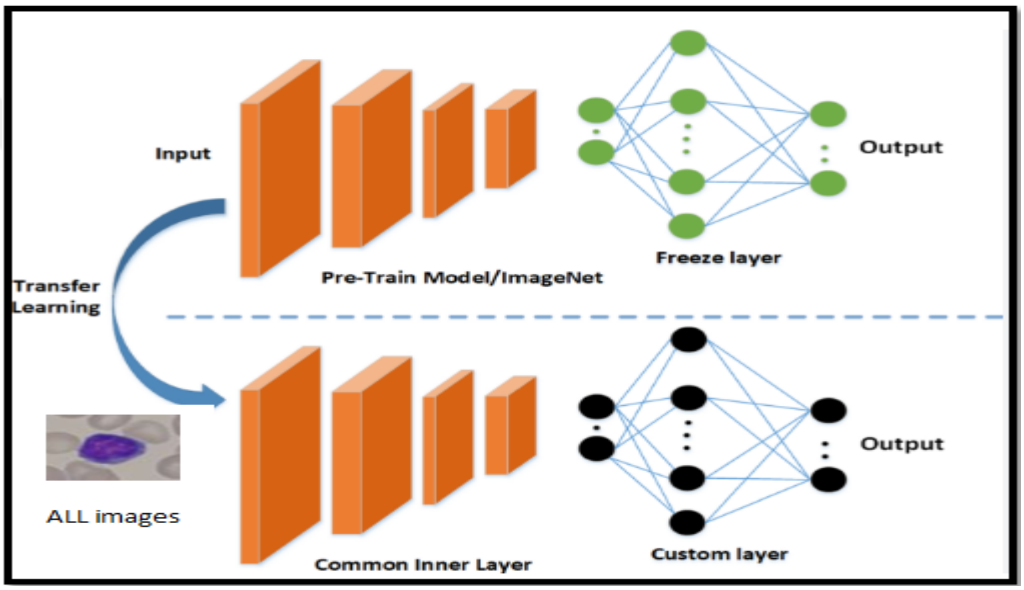

Figure 4.5. Illustration transfer learning from a previously trained model.

4.8.1. DenseNet 169 Model

The common issue of vanishing gradients in CNN-based architectures has been solved using DenseNet. The previous Layer's output and the next Layer's output are concatenated. A picture is sent into a neural network, which then uses the image to anticipate what will happen next. When feature maps are being created, the output of the previous convolutional Layer is passed on to the next convolutional Layer. Sending this feature map to the next convolutional Layer. As a result, each Layer is directly related to the others [78].

To fit the DenseNet architecture, the typical CNN design is modified by adding connections between each Layer. DenseNet's structure is composed of dense chunks. Consequently, each successive Layer receives the output feature maps of all layers before it as input [79].

A feature map in DenseNet 169 is transmitted from one Layer to another, with each Layer gaining extra input from all previous layers. The aggregate data from the layers above is sent to the Layer below. Because each Layer accumulates feature maps from all previous layers, the network may be compact regarding the number of channels [80].

DenseNet, with 169 layers, is recommended as a solution. First, the pre-trained DenseNet model is loaded for feature extraction. The transfer learning approach is used to load a pre-trained model.

In this study, the previously trained model in [81] was used on the Pcam dataset A more accurate description is that transfer learning applies previously learned patterns to new ones by leveraging their information (features, weights) [82].

Using this technique displays its value when the quantity of data is exceedingly unequal, and the information acquired by another task (acute lymphoblastic leukaemia classification) may lead to better utilization. Second, the final Layer of the pre-trained DenseNet model is turned into a fully connected layer with a two neurons classifier for binary classification. Figure (4.6) appears the illustrated layers of DenseNet169.

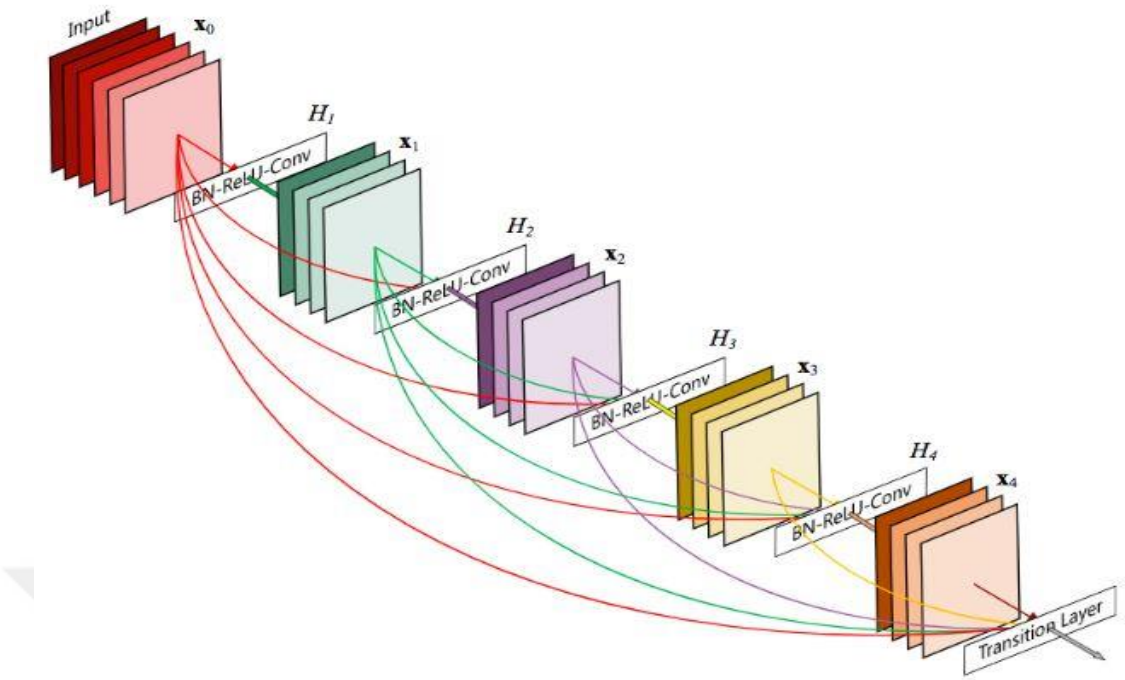

Figure 4.6. Architecture of DenseNet169 (source: (Huang, Gao, et al.,2017)[82]).

4.8.2. VGG19 Model

VGG19 model needs a large amount of training data before being trained from scratch. As a concept, transfer learning helps us employ current workplace models. You may leverage an existing model to transfer learning to fresh pictures in several ways. Only the fully connected classifier will be trained since we utilized pre-trained models to extract features from our limited dataset.

The pre-trained version of the VGG-19 model was supplied by K. Simonyan and A. Zisserman of the University of Oxford in their 2014 publication [83]. It has been discovered that the VGG-19 CNN architecture may reach very high accuracy levels while processing big picture datasets. Google stumbled onto this. The VGG-19 model is made up of around 143 million distinct parameters. The VGG-19 network has 19 trainable layers, some fully linked and convolutional layers. Max pooling and dropout are two further layers [84], [85].

A convolutional layer operates and then passes the result at each image position (feature map) [83]. Filters using convolutional layers often use 3 x 3 feature extractors that may be trained to extract specific features.

ReLU activation function and the maximum-pooling method follow each convolutional layer stack in the neural network's architecture. The ReLU is presently the most often utilized nonlinear activation function since its parameter, x, reflects the input to a neuron.

$$f(x) = \max(0, x) \tag{4.1}$$

But the issue of vanishing gradients may be overcome by utilizing the ReLU function instead of the faster sigmoid function. Afterwards, max-pooling is used to reduce the number of data points. An equal-length stride and a 2-by-2 filter set the stage.

For each convolution around a certain subregion, the filter produces the greatest number in that region [86]. Afterwards, the densely-connected classifier and dropout layer were utilized in the classification procedure, which used convolutional layers ranging from the layer of conv1 to the layer of conv5. Each neuron in a thick layer receives information from all the neurons in the Layer above it, forming a seamless network. While a convolutional layer depends on consistent features with a limited number of repetitions, a densely connected layer provides learning features based on the previous Layer's properties. The densely coupled Layer must have an activation function [86],[87].

An efficient network may be achieved by setting activations to zero in this Layer. P - dropout rate reduces the risk of overfitting when training. The use of this Layer is not included in the testing [87].Figure (4.7) represents for VGG19 model.

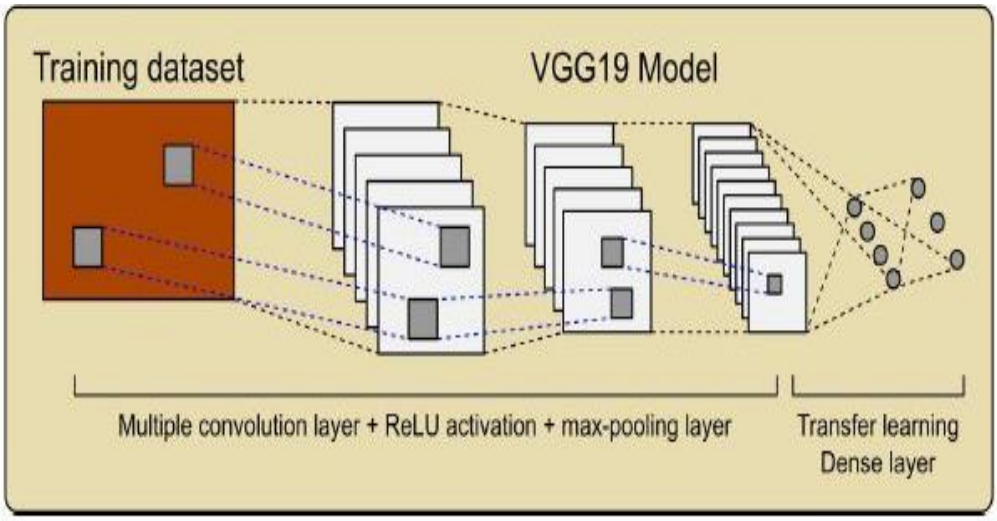

Figure 4.7. Illustrate of VGG19 model (source: (Jaworek-Korjakowska et al. , 2019) [83].

4.8.3. Simple CNN Model

This model can diagnose and categorize photos of acute lymphoblastic leukaemia and delivers a greater prediction accuracy with the same precision we achieved when training the other two models. Moreover, this model gives us the same accuracy that we obtained when training the other two models (DenseNet 169, VGG 19). This gave us an exciting task to create a simple CNN model consisting of a few layers. This model is able to detect acute lymphoblastic leukaemia and categorize photos of the disease.

The proposed model includes the following layers: a Conv2D +Relu layer with an image size of 225 x 225 pixels; a max-pooling layer with an image size of 127 x 127 pixels; a Conv2D +Relu layer with an image size of 125 x 125 pixels; a max-pooling layer with an image size of 62 x 62 pixels; and finally, a Conv2D +Relu layer with an image size of After that, the image, which has a pixel resolution of 60 by 60, is imported into the Conv2D +Relu layer. The flatten Layer is the last stage that must be completed before receiving the output. This Layer reduces the data to a one-dimensional array and puts it into two dense layers. Figure (4.8) displays a graphical depiction of the proposed CNN model architecture in the form of a diagram.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 255, 255, 32)      896
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 127, 127, 32)      0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 125, 125, 64)      18496
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 62, 62, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 60, 60, 128)       73856
_________________________________________________________________
flatten (Flatten)            (None, 460800)            0
_________________________________________________________________
dense (Dense)                (None, 128)               58982528
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 129
=================================================================
Total params: 59,075,905
Trainable params: 59,075,905
Non-trainable params: 0
```

Figure 4.8. Simple CNN model schematic.

4.9. TRAINING THE MODELS

A persistent shortage of data hinders research in the medical field. Transfer learning may be the answer to this problem. It's possible to reduce the amount of data needed by using transfer learning algorithms [88], which use a previously trained model to teach a new one. Both trainable and non-trainable layers are included in the pre-training model. When using a fresh dataset to retrain a pre-trained model, the original layers are replaced with the new ones.

A model's training history must be examined while it is being tested. This example of TensorFlow uses a history function named plot metrics (history). History. Keys may be used to get information on training loss, accuracy, training F1, and training F1 score values ().

An accuracy measure is used to assess the classifier's performance. The answer is 1 if the total quantity of accurate data is divided by the total amount of properly categorized data. One of the most used measures for measuring research accuracy is True Positive (TP), False Positive (FP), and True Negative (TN) (FN).

$$Accurac = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.2}$$

In a word, the model's accuracy in testing and validation represents its performance throughout training and validation. The loss function relies on the formula employed in the training or validation phase to calculate the loss. Calculating loss is often done using binary cross-entropy, squared error loss, or absolute error loss. Cross-entropy was used in the proposed approach. Binary cross-entropy is the inverse of the logarithm of the chance that a forecast is accurate. (Figur 4.9) represents the diagram for pretianed of the proposed models.

$$\text{logloss} = -\frac{1}{N}\sum_{i}^{N}\sum_{j}^{M} yij \log(pij) . \tag{4.3}$$

The number of rows is N, and the number of classes is M in this case.

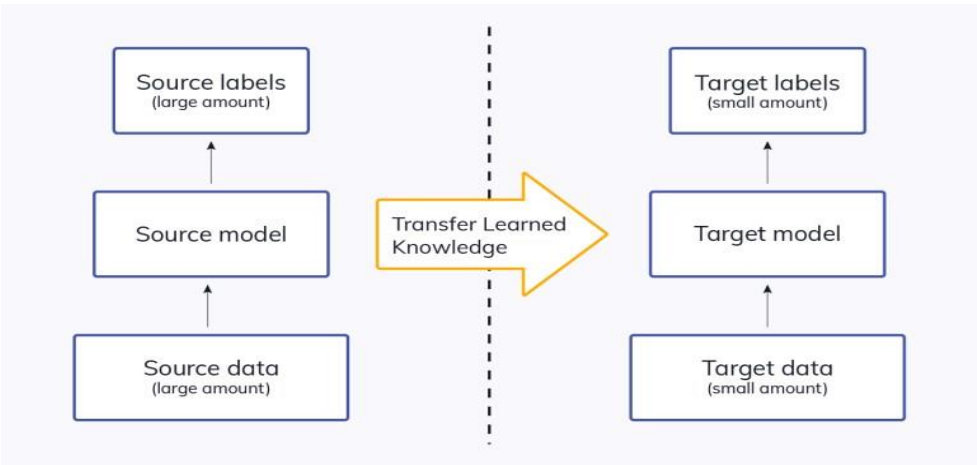

Figure 4.9. Principal of transfer learning.

.

4.10. MODEL EVALUATION

The models themselves need to be assessed for the process of creating models to be effective. It helps identify which model most accurately represents our data and how well the model that is finally chosen will operate in the years to come.

In data science, evaluating a model's performance by utilizing the same data used to train it is discouraged since it may lead to too optimistic and overfit models. Evaluation of models in data science may be carried out using either the hold-out or the cross-validation method. Both of these methods are described here. Both methods employ a test set concealed from the model during the effectiveness analysis of a model to avoid overfitting the data to the model's parameters[89].

PART 5

RESULT AND DISCUSSION

In this chapter, we will compare and evaluate the accuracy rates of the three algorithms we applied to the two datasets (ALL_IDB and C-NMC-ALL) for acute lymphocytic leukaemia to discover the factors that contributed to the accuracy rates. This will be presented through plots, photos and graphs highlighting the differences in the findings between all those models.

5.1. ALL MODELS' ACCURACY

These findings were accomplished using three different models: the DenseNet 169 model, the VGG 19 model, and a fundamental CNN model. Putting these models into practice allowed us to attain the desired outcomes. Classification models are trained with the use of cancer cells (blast) and health data samples(normal), with cancer cells accounting for 75% of the training data, health data samples accounting for 20% of the test data, and health data samples accounting for 5% of the validation test data, respectively. The image data for the two distinct data sets have been separated into the categories presented in table (5.1) in line with the percentages discussed earlier in this section. Tables (5.2 and 5.3) show the differences in results between the models used. Including the effects of accuracy, sensitivity, specificity and AUC rations after the experimental performance on the two datasets.

Table 5.1. Split of images in datasets.

DATASET	NO. OF IMAGES	TRAINING	TESTING	VALIDATION
ALL_IDB	2083	1562	417	104
C-NMC ALL	10661	7996	2133	532

Table 5.2. Results of applying the proposed models to ALL_IDB dataset.

MODEL	ACCURACY	SENSITIVITY	SPECIFICITY	AUC
DenseNet 169	99.76 %	99.52 %	100 %	99 %
VGG 19	99.49 %	100 %	99.51 %	99 %
Simple CNN	94.24 %	97.93 %	91.03 %	93 %

Table 5.3. Results of applying the proposed models to C-NMC ALL dataset.

MODEL	ACCURACY	SENSITIVITY	SPECIFICITY	AUC
DenseNet 169	85.41 %	83.70 %	91.57 %	81 %
VGG 19	85.32 %	85.18 %	80.98.%	79 %
Simple CNN	80.77 %	84.08 %	72.33 %	80 %

5.2. EVALUATION OF MODELS

Model evaluation is an essential part of any project involving any learning, whether it is machine learning or deep learning. A model can produce excellent results while it's being tested. Still, it might produce poor results when measured against any assessment technique, such as an accuracy score, a Receiver Operating Characteristic Graph (ROC), or a Confusion Matrix. This is because accuracy scores, ROC graphs, and Confusion matrices are all examples of assessment techniques. This is because there are a wide variety of elements, each of which has the potential to impact a different aspect of the quality of the outcomes.

We assessed the effectiveness of our models as a part of this more comprehensive body of work by measuring the accuracy of their cross-validation, creating receiver operating characteristic graphs (ROC), and generating classification reports.

5.3. CROSS-VALIDATION

When a prediction function's parameters are known, adjustments are made to the function. Then the process is evaluated using the same data samples [27], there is a possibility of a methodological error occurring. It is more likely for a model to achieve a training accuracy of 1 if the model would always assign the same labels to samples that it has previously analyzed. On the other hand, it can fail if it comes across any test data unfamiliar.

The term "overfitting" describes situations like the one described above. To solve this issue, we can implement the process of cross-validation as a solution. Although it is not always possible, the best practice is to use various approaches to arrive at a cross-validation score. This is the case the majority of the time. The most basic technique is the K-fold CV method, which divides the information into a total of K groups. The procedure that is described further down is used for every K fold that is created.

- The K-10 folds of the training data are used throughout the model's training process.
- The results of the simulation are then interpreted using the model.
- The evaluation is performed using the data sets that are still accessible. Because we had fewer data samples than usual, we used a CV of 10 folds, which again took significant time to compute the cross-validation score. In most cases, a CV of 5 folds is adequate to determine whether or not a hypothesis is valid.

5.4. EXPERIMENTAL RESULTS

5.4.1. Test Results DenseNet169 to ALL_IDB Dataset

The results of applying the DenseNet169 model to the ALL_IDB dataset revealed that it achieved an accuracy of 99.76 %, a sensitivity of 99.52 %, a specificity of 100 %, and an area under the curve (AUC) of 99.75 %, respectively, as shown in table (5.4), and the results backed up this argument. Figure (5.1) contains a graphic representing the model's accuracy, while Figure (5.2) includes a picture representing the loss of the model DenseNet169. The confusion matrix generated by our model is shown in Figure (5.3), which contains two classes, either blast or normal, which places it within the context of the ALL IDB dataset; the data distribution is revealed when we applied the test set with 20% of the total data to it, and through that, we got all the results.

Table 5.4. Performance results of DenseNet169 model to ALL_IDB.

MODEL	ACCURACY	SENSITIVITY	SPECIFICITY	AUC
DenseNet 169	99.76 %	99.52 %	100 %	99.75 %

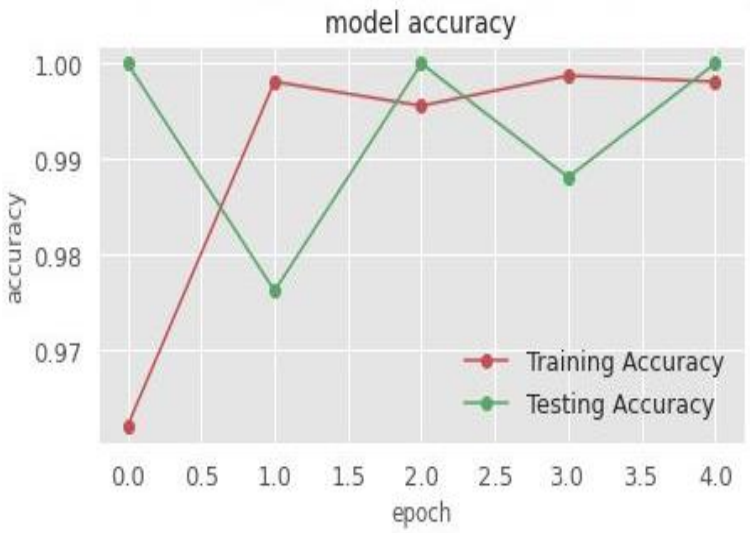

Figure 5.1. A plot of model accuracy for DenseNet169 with ALL_IDB dataset.

Figure 5.2. A Plot of model loss for DenseNet169 with ALL_IDB dataset.

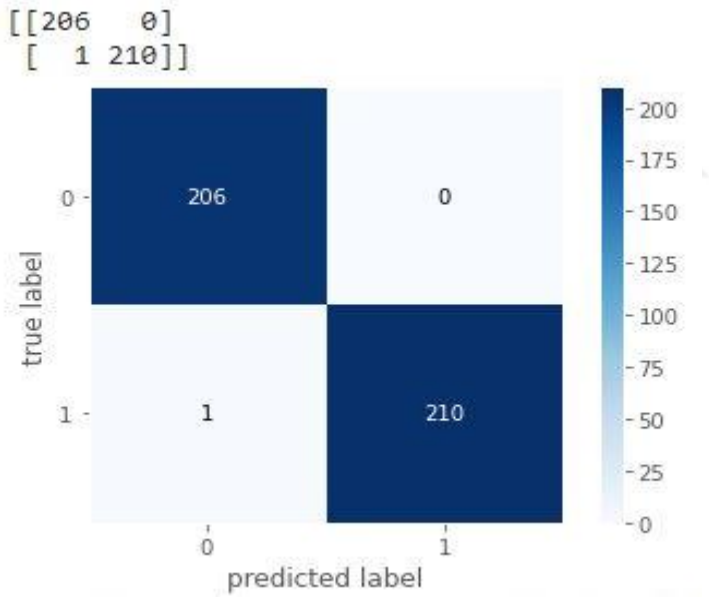

Figure 5.3. Confusion matrix for DenseNet169 with 10 epochs- 417 testing images of ALL_IDB dataset.

5.4.2. Test Results VGG19 Model to ALL_IDB Dataset

The results of applying the VGG19 model to the ALL_IDB dataset revealed that it achieved an accuracy of 99.49 %, a sensitivity of 100 %, a specificity of 99.51 %, and an area under the curve (AUC) of 99.76 %, respectively, as shown in table (5.5), and

the results backed up this argument. Figure (5.4) contains a graphic representing the model's accuracy, while Figure (5.5) includes a picture representing the loss of the model DenseNet169. The confusion matrix generated by our model is shown in Figure (5.6), which contains two classes, either blast or norma, which places it within the context of the ALL IDB dataset, the data distribution is revealed when we applied the test set with 20% of the total data to it, and through that, we got all the results.

Table 5.5. Performance results of VGG19 model to ALL_IDB dataset.

MODEL	ACCURACY	SENSITIVITY	SPECIFICITY	AUC
VGG 19	99.49 %	100 %	99.51 %	99.76 %

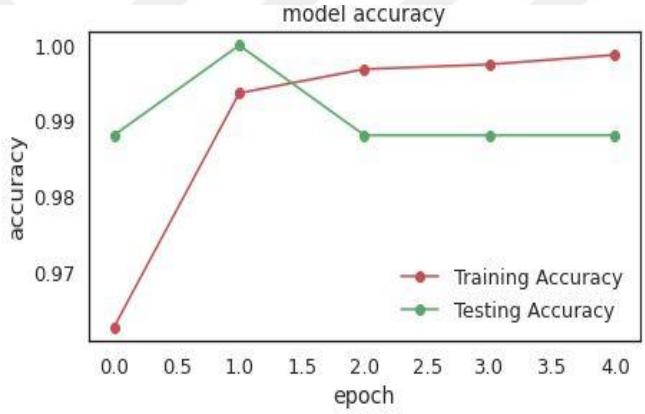

Figure 5.4. A plot of model accuracy for VGG1 with ALL_IDB dataset

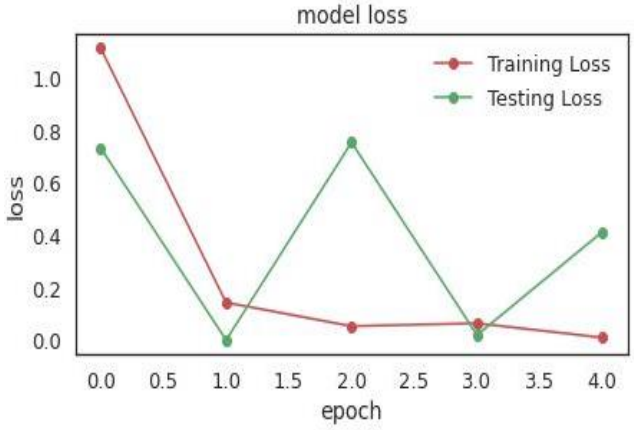

Figure 5.5. A Plot of model loss for VGG19 with ALL_IDB dataset.

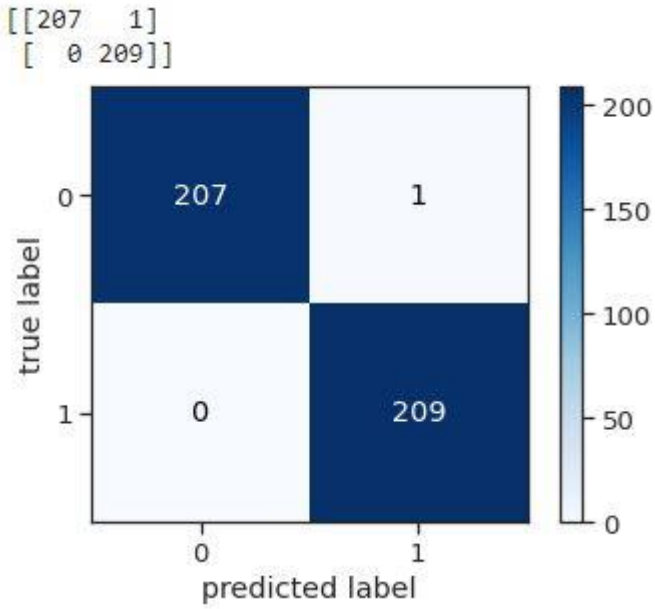

Figure 5.6. Confusion matrix for DenseNet169 with 10 epochs- 417 testing images of ALL_IDB dataset.

5.4.3. Test Results In Simple CNN Model to ALL_IDB Dataset

The results of applying a simple CNN model to the ALL_IDB dataset revealed that it achieved an accuracy of 94.24 %, a sensitivity of 97.93 %, a specificity of 91.03 %, and an area under the curve (AUC) of 98 %, respectively, as shown in table (5.6), and the results backed up this argument. Figure (5.7) contains a graphic representing the model's accuracy, while Figure (5.8) includes a picture representing the loss of the model DenseNet169. The confusion matrix generated by our model is shown in Figure (5.9), which contains two classes, either blast or norma, which place it within the context of the ALL IDB dataset. The data distribution is revealed when we applied the test set with 20% of the total data to it, and through that, we got all the results.

Table 5.6. Performance results of simple CNN model to ALL_IDB.

MODEL	ACCURACY	SENSITIVITY	SPECIFICITY	AUC
Simple CNN	94.24 %	97.93 %	91.03 %	98 %

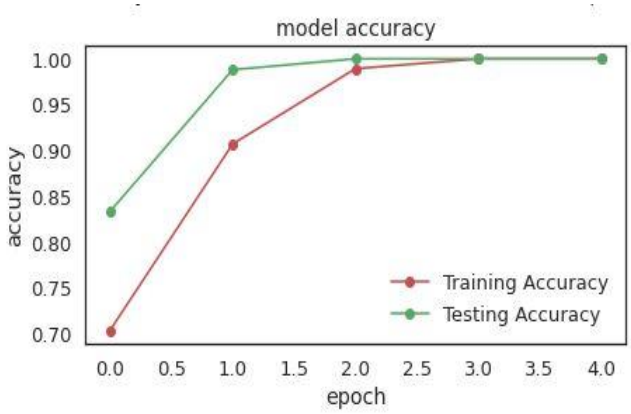

Figure 5.7. A plot of model accuracy for a simple CNN model with ALL_IDB.

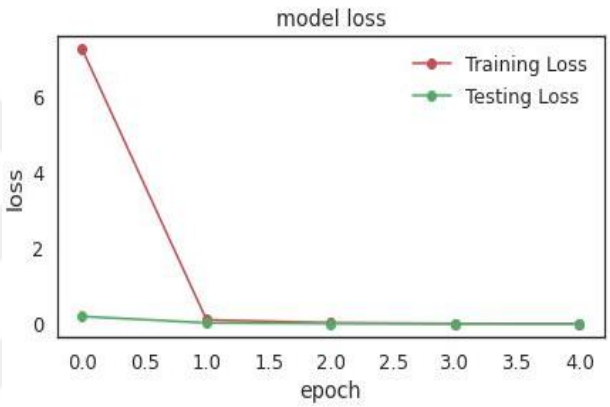

Figure 5.8. A Plot of model loss for a simple CNN model with ALL_IDB.

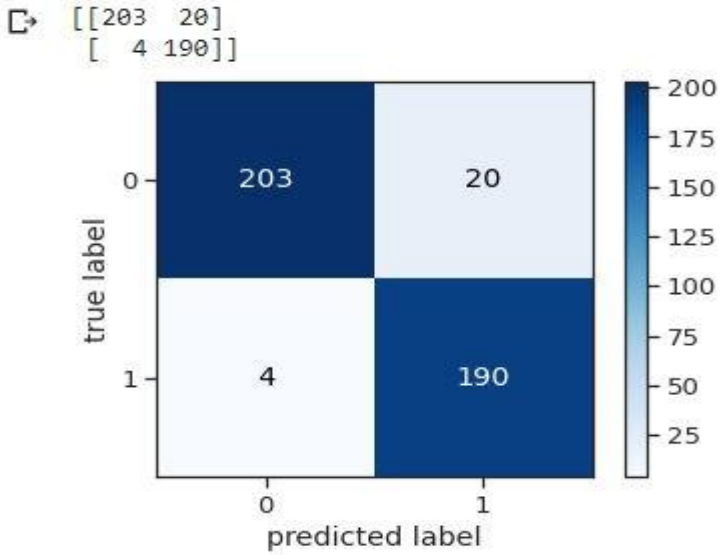

Figure 5.9. Confusion matrix for a simple CNN model with 10 epochs- 417 testing images of ALL_IDB.

5.4.4. Test Results DenseNet169 Model to C-NMC ALL Dataset

The results of applying the DenseNet169 model to the C-NMC ALL dataset revealed that it achieved an accuracy of 85.41 %, a sensitivity of 83.70 %, a specificity of 91.57 %, and an area under the curve (AUC) of 61 %, respectively, as shown in table (5.7), and the results backed up this argument. Figure (5.10) contains a graphic representing the model's accuracy, while Figure (5.11) includes a picture representing the loss of the model DenseNet169. The confusion matrix generated by our model is shown in Figure (5.12), which contains two classes, either blast or norma, which places it within the context of the C-NMC ALL dataset. The data distribution is revealed when we applied the test set with 20% of the total data to it, and through that, we got all the results.

Table 5.7. Performance results of DenseNet169 model to C-NMC ALL.

MODEL	ACCURACY	SENSITIVITY	SPECIFICITY	AUC
DenseNet 169	85.41 %	83.70 %	91.57 %	81 %

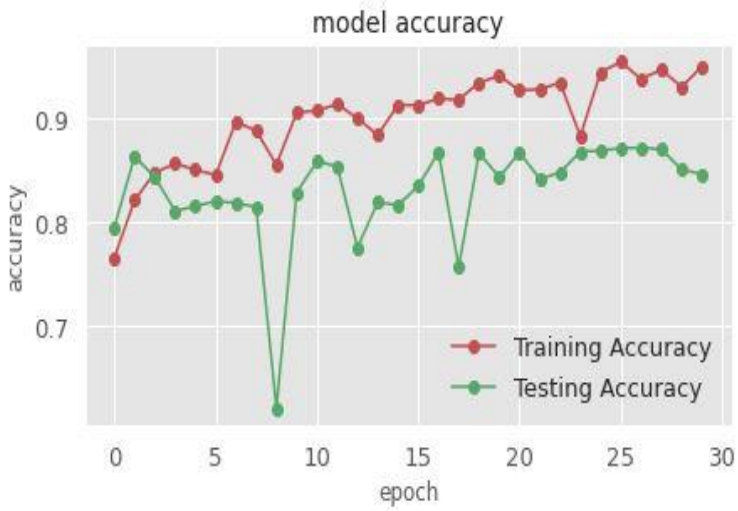

Figure 5.10. A Plot of model accuracy for DenseNet169 with C-NMC ALL.

Figure 5.11. A Plot of model loss for DenseNet169 with C-NMC ALL.

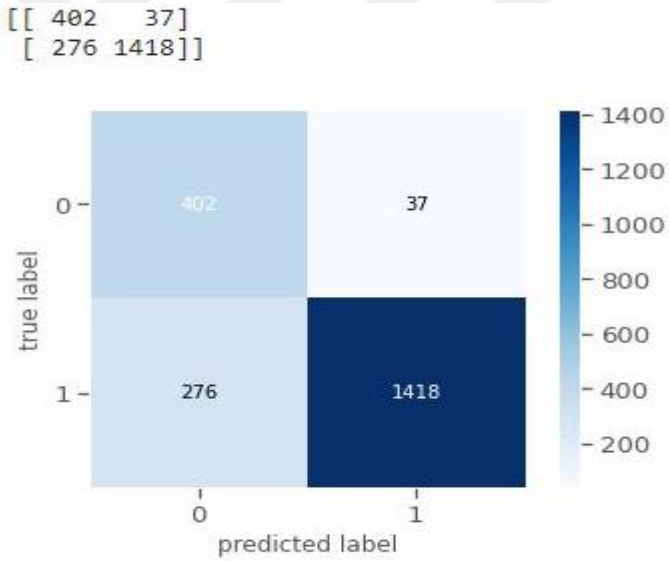

Figure 5.12. Confusion matrix for DenseNet169 with 10 epochs- 2133 testing images of C-NMC ALL dataset.

We tried to train the model by increasing the number of epoch times by 20, 40 and 50 in order to get a higher accuracy rate, but after testing, we found out that the best percentage that appeared to us was with 10 epochs, which achieved a rate of 86.87%.

5.4.5. Test Results VGG19 Model to C-NMC ALL Dataset

The results of applying the VGG19 model to the C-NMC ALL dataset revealed that it achieved an accuracy of 85.32 % a sensitivity of 83.84 %, a specificity of 80.98.% and an area under the curve (AUC) of 84 %, respectively, as shown in table (5.8), and the results backed up this argument. Figure (5.13) contains a graphic representing the model's accuracy, while figure (5.14) includes a picture representing the loss of the model DenseNet169. The confusion matrix generated by our model is shown in figure (5.15), which contains two classes, either blast or norma, which place it within the context of the C-NMC ALL dataset. The data distribution is revealed when we applied the test set with 20% of the total data to it, and through that, we got all the results. Figure (5.14) represents the ROC curve for the VGG19 model.

Table 5.8. Performance results of VGG 19 model to C-NMC ALL.

MODEL	ACCURACY	SENSITIVITY	SPECIFICITY	AUC
VGG 19	85.32 %	85.18 %	80.98.%	79 %

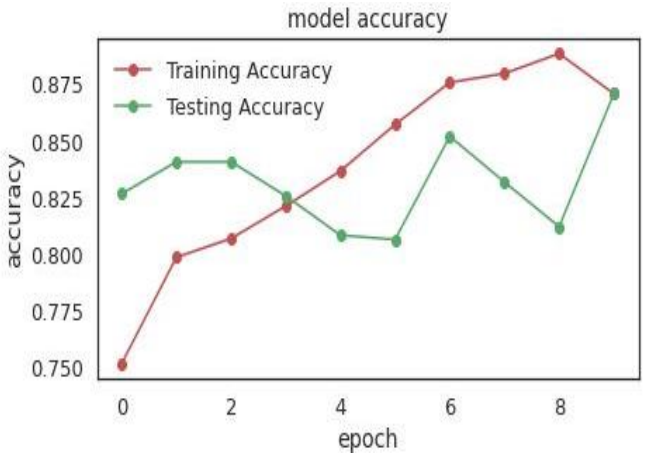

Figure 5.13. A Plot of model accuracy for a VGG19 model with C-NMC ALL.

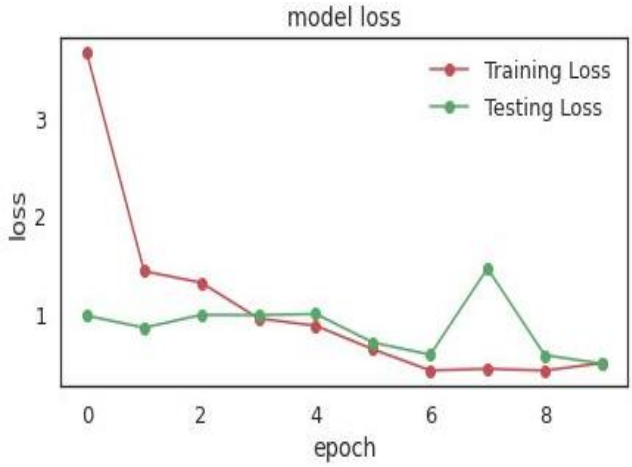

Figure 5.14. A Plot of model loss for VGG19 with C-NMC ALL.

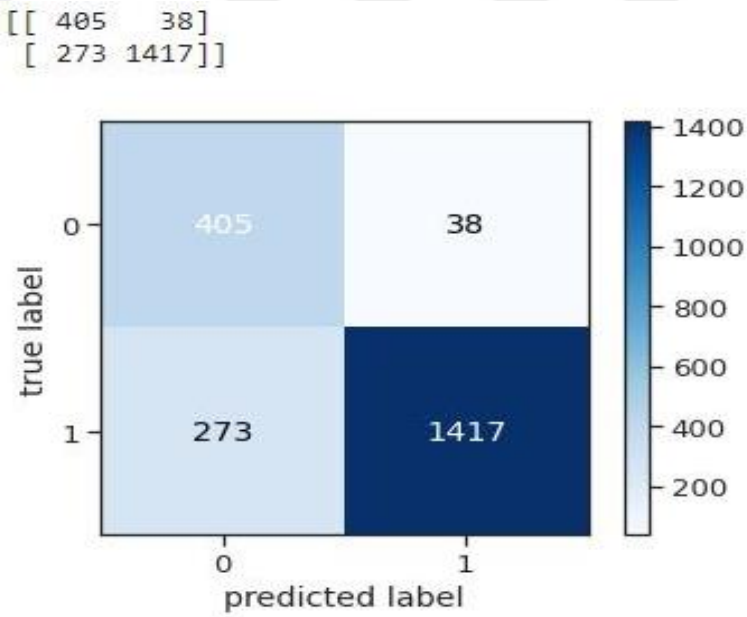

Figure 5.15. Confusion matrix for VGG19 with 10 epochs- 2133 testing images of C-NMC ALL dataset.

5.4.6. Test Results in Simple CNN Model to C-NMC ALL Dataset

The results of applying a simple CNN model to the C-NMC ALL dataset revealed that it achieved an accuracy of 80.77 %, a sensitivity of 84.53 %, a specificity of 69.34 %, and an area under the curve (AUC) of 96 %, respectively, as shown in table (5.9), and the results backed up this argument. Figure (5.15) contains a graphic representing the model's accuracy, while Figure (5.16) includes a picture representing the loss of the model DenseNet169. The confusion matrix generated by our model is shown in Figure

(5.15), which contains two classes, either blast or norma, which place it within the context of the C-NMC ALL dataset. The data distribution is revealed when we applied the test set with 20% of the total data to it, and through that, we got all the results. Figure (5.17) represents the ROC curve for the simple CNN model.

Table 5.9. Performance results of Simple CNN model to C-NMC ALL.

MODEL	ACCURACY	SENSITIVITY	SPECIFICITY	AUC
Simple CNN	80.77 %	84.53 %	69.34 %	94 %

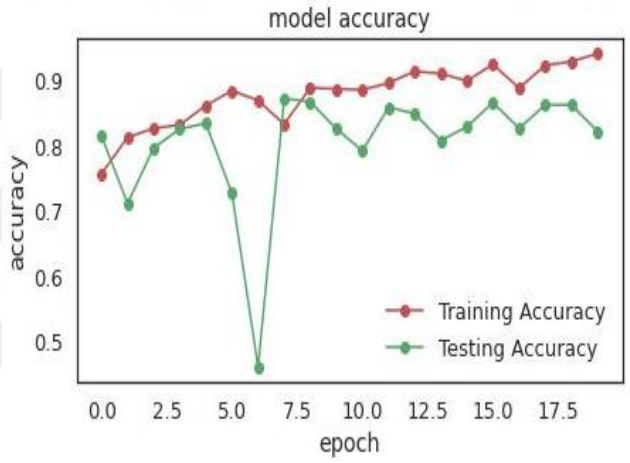

Figure 5.16. A Plot of model accuracy for a simple CNN model with C-NMC ALL

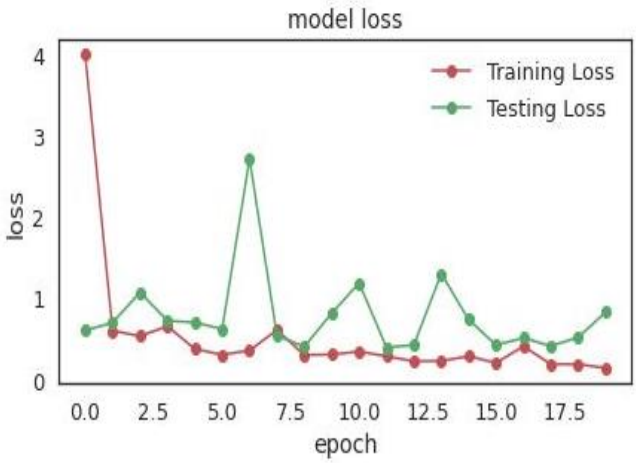

Figure 5.17. A Plot of model loss for a simple CNN model with C-NMC ALL.

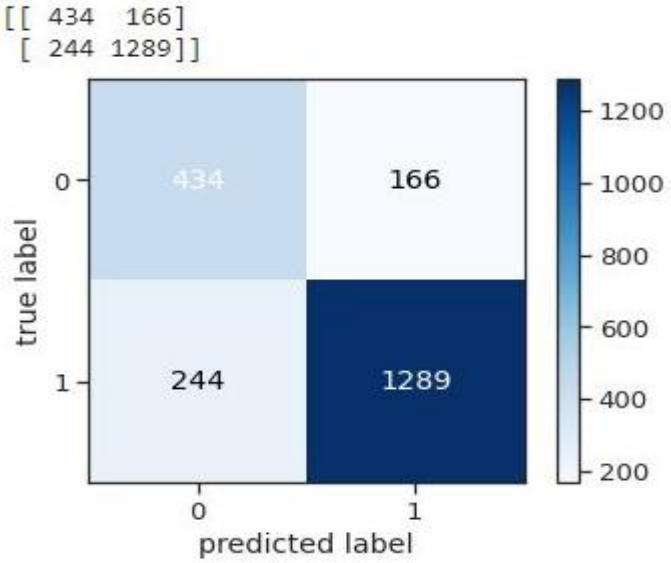

Figure 5.18. Confusion matrix for simple CNN with 10 epochs- 2133 testing images of C-NMC ALL dataset.

5.5. RESULTS INTERPRETATION

After collecting the results from training the three proposed models on the two data sets, it became clear that the ALL IDB data set has superior accuracy, sensitivity, specificity, and AUC ratios than the C-NMC ALL data set. This became immediately apparent when comparing the results from the three models after training on the two data sets. The DenseNet169 model achieved an accuracy rate of 99.76 %, which is higher than what was achieved by the VGG19 and Simple CNN models, which achieved an accuracy rate of 99.49 % and 94.24 %, respectively, and the reason for this is the efficiency of the layers of the model DenseNet 169 and the method of transfer learning to him. When training on the C-NMC ALL data set, the DenseNet169 model achieved an accuracy rate of 85.41 %, which is higher than that of the VGG19 and simple CNN models, which achieved an accuracy of 85.32% and 80.77%, respectively. According to what we have been able to learn, this may be caused by several different causes, including the following:

- The results of training the proposed models on the first data set appeared high, and the reason for this, we believe, is because the first data (ALL IDB) was well trained, and this was reflected in the performance of the three models through the transfer of learning. Even though the number of images in the first

dataset ALL IDB is much less than the number of images in the second data set C-NMC ALL, training the proposed models on the first data set appeared high.

- Even though the data set (ALL IDB) only has a small number of images, it is considered somewhat balanced. This is shown by how the images are spread out in figure (4.2), which is different from how the images are spread out in the C-NMC ALL dataset, shown in figure (4.3).
- The model (DenseNet 169) achieved high accuracy rates when trained on the two data sets compared to the other two models when trained on both data sets, and the reason, in our opinion, is that we achieved a good fit and avoided as much as possible over-allocation, as well as because of the strength and efficiency of the model and the number of its layers, as it proved It is possible to achieve high prediction accuracy rates when trained with any data set. This was evident when we reduced errors in the training and test sets while obtaining high accuracy in the training and test sets by controlling and controlling with variance and bias.

5.6. COMPARISON THE RESULTS

The varying degrees of precision that we discovered via the graphs are discussed in this section. The accuracy results we acquired may be shown in figure (5.19). These results were achieved by training the three suggested DenseNet 169 VGG19 CNN models on the ALL IDB dataset. The accuracy results we achieved may be shown in figure (5.20). These results were acquired by training the three suggested models on the C-NMC ALL data set.

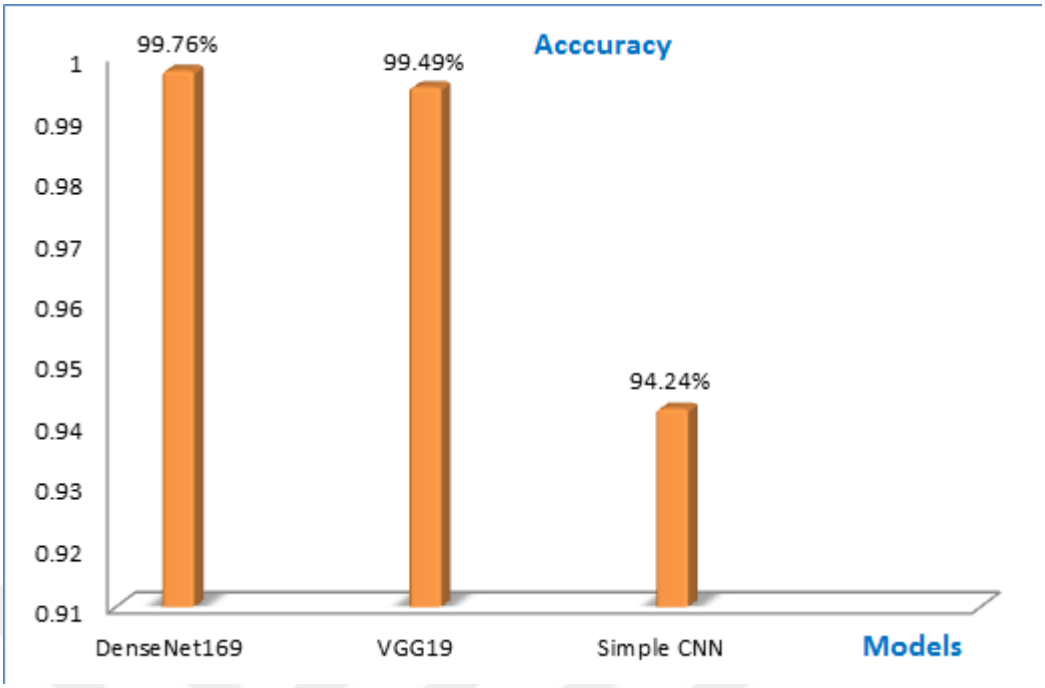

Figure 5.19. Comparing the suggested models' accuracy on the ALL_IDB dataset.

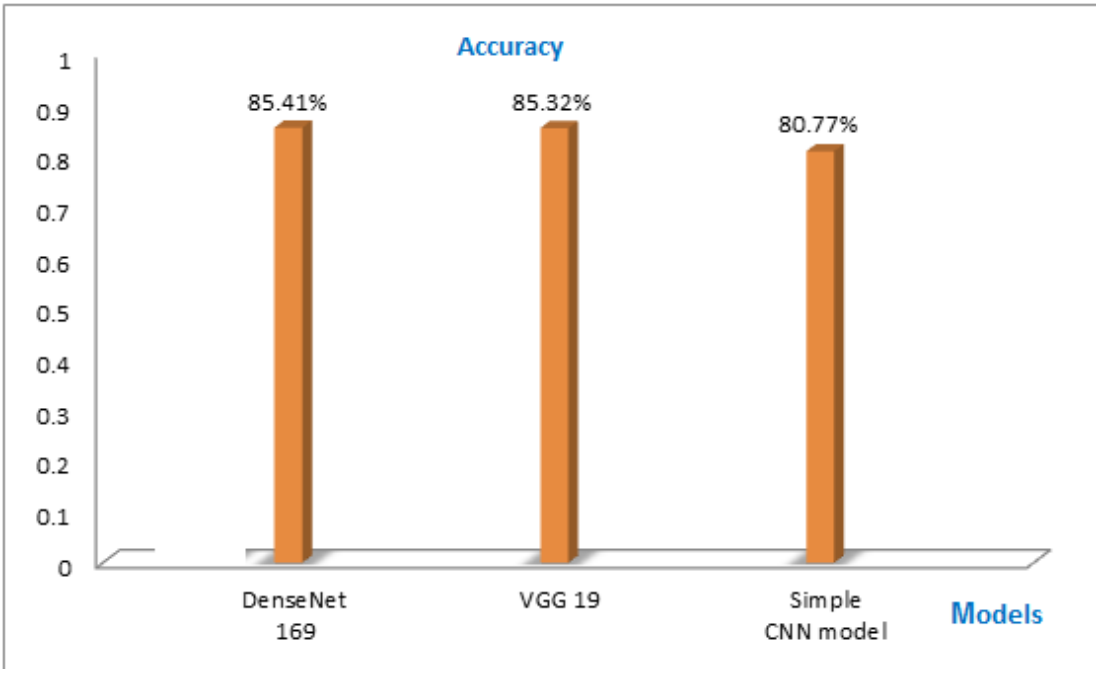

Figure 5.20. Comparing the suggested models' accuracy on C-NMC ALL dataset.

In figure(5.21), we examine the accuracy results from 11 research publications chosen randomly and compare them with one of our suggested models (DenseNet 169), which attained the greatest accuracy rates. These results were compared to one another to determine which model achieved the highest accuracy rates. Note that when selecting

those earlier studies, we took special care to ensure that the fields in which they worked were comparable to the one in which we are doing our study, which is the identification of lymphatic leukaemia using deep learning models. This precaution allowed us to select the most relevant studies for our investigation. It has been shown that the accuracy rate of the system that we have provided is higher than that of the study that was done in the past.

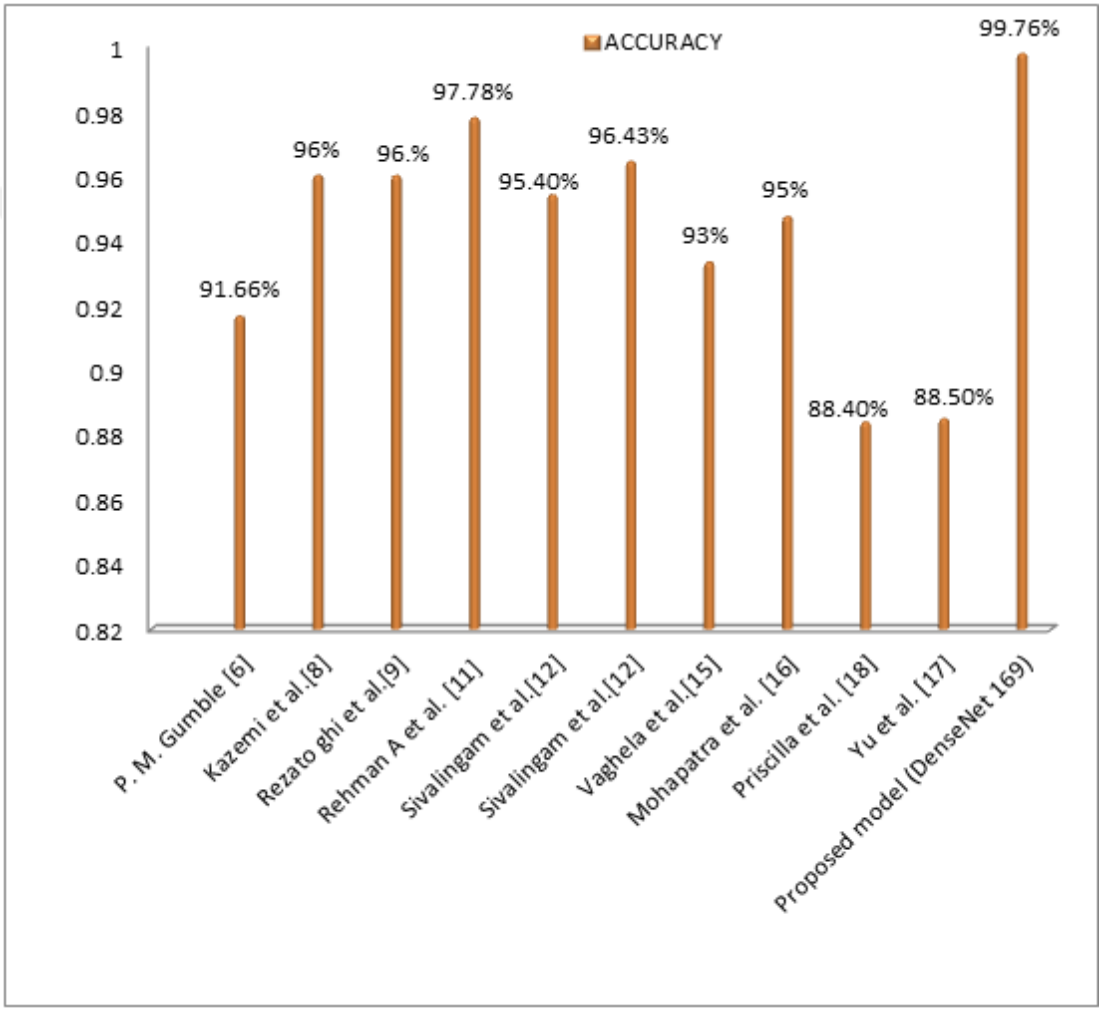

Figure 5.21. Comparing our suggested outcomes to others.

PART 6

CONCLUSION

This study compares the classification efforts of transfer learning and depth feature extraction to detect ALL based on images of pathological slides. Two deep CNN architectures (DenseNet 169 and VGG19) were used for deep feature extraction, and a basic CNN model was constructed for differentiation and comparison of results with transfer learning.

Because of the breakthroughs that have been made in technology, the overall level of complexity in our lives is progressively becoming less. One of the areas of technology that is progressing at the fastest pace in the medical sector is image processing and deep learning. This is one of the areas that has become one of the most critical applications of technology.

The primary objective of the study that we conducted was to develop a method to determine whether acute lymphoblastic leukemia is present in the blood cells of the human body. This was one of our primary goals in doing this research. This was the overarching objective of the job we were tasked with completing.

After looking at several studies that were done in the past, we came up with several different approaches to find the rapid rise of white blood cells (WBC), which is a sign of acute leukemia. A motion for acute leukemia is a fast growth of white blood cells (WBC).

First, we selected two well-known datasets, which both contain disease profiles for acute lymphoblastic leukemia (ALL), and the profiles for healthy cells that had not been infected with any pathogens. The disease profiles were present in each of these datasets. Next, we compared the information contained in these two datasets.

6.1. FUTURE WORK

Our study has a few minor drawbacks, even though we successfully implemented our suggested methodology for identifying aberrant blood cells in the human body that indicate acute lymphoblastic leukaemia (ALL). When we improve the model shortly, we wish to eliminate any restrictions. A hybrid neural network will be used to determine the patient's cancer type.

Due to a lack of data, we could not demonstrate the model's usefulness (micrograph of various types of leukaemia with blood sample). "Data Collection" is a portion of this study that provides information on how to gather data. Updates to our model are imminent. A product's simplicity of use is an important consideration when releasing a new one. We're currently testing a product prototype that involves several processes. To execute and document the outcomes of our proposed model, we'll need a desktop or laptop computer. We've tried every strategy and algorithm in Python and on the PC.

REFERENCES

1. Thanh, T. T. P., Vununu, C., Atoev, S., Lee, S.-H., and Kwon, K.-R., "Leukemia Blood Cell Image Classification Using Convolutional Neural Network", ***International Journal Of Computer Theory And Engineering***, 10 (2): 54–58 (2018).

2. Shafique, S. and Tehsin, S., "Computer-Aided Diagnosis of Acute Lymphoblastic Leukaemia", ***Computational And Mathematical Methods In Medicine***, 2018: (2018).

3. Wang, Q., Bi, S., Sun, M., Wang, Y., Wang, D., and Yang, S., "Deep learning approach to peripheral leukocyte recognition", ***PLoS ONE***, 14 (6): 1–18 (2018).

4. Liu, Y. and Long, F., "Acute lymphoblastic leukemia cells image analysis with deep bagging ensemble learning", ***Lecture Notes In Bioengineering***, (Cv): 113–121 (2019).

5. Lin, M., Jaitly, V., Wang, I., Hu, Z., Chen, L., Amer Wahed, M., Kanaan, Z., Rios, A., and Nguyen, A. N. D., "Application of deep learning on predicting prognosis of acute myeloid leukemia with cytogenetics, age, and mutations", ***ArXiv***, 77030 (713): 1–11 (2018).

6. Gumble, P. M. and Rode, S. V., "Analysis & Classification of Acute Lymphoblastic Leukemia using KNN Algorithm", ***International Journal On Recent And Innovation Trends In Computing And Communication***, 5 (2): 94–98 (2017).

7. Genovese, Angelo, et al. "Acute Lymphoblastic Leukemia detection based on adaptive unsharpening and Deep Learning." **ICASSP 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE**, (2021).

8. Kazemi, F., Najafabadi, T., and Araabi, B., "Automatic recognition of acute myelogenous leukemia in blood microscopic images using K-means clustering and support vector machine", ***Journal Of Medical Signals And Sensors***, 6 (3): 183–193 (2016).

9. Su, M. C., Cheng, C. Y., and Wang, P. C., "A neural-network-based approach to white blood cell classification", ***The Scientific World Journal***, 2014 (1): (2014).

10. Kasani, P. H., Park, S. W., and Jang, J. W., "An aggregated-based deep learning method for leukemic B-lymphoblast classification", ***Diagnostics***, 10 (12): (2020).

11. Rehman, A., Abbas, N., Saba, T., Rahman, S. I. ur, Mehmood, Z., and Kolivand, H., "Classification of acute lymphoblastic leukemia using deep learning", ***Microscopy Research And Technique***, 81 (11): 1310–1317 (2018).

12. Sivalingam, Nithya Priya and Chinnasamy, Sundar and Suruli Muniyandi, T., "An effective chronic lymphocytic leukemia detection method using hybrid optimization aware random multimodal deep learning", ***Concurrency And Computation: Practice And Experience, Wiley Online Library***, (2022).

13. TTP, Thanh, et al. "Acute leukemia classification using convolution neural network in clinical decision support system." ***CS & IT Conference Proceedings*.** Vol. 7. No. 13. CS & IT Conference Proceedings, (2017).

14. Jagadev, Preeti, and H. G. Virani. "Detection of leukemia and its types using image processing and machine learning." ***2017 International Conference on Trends in Electronics and Informatics (ICEI)*. IEEE**, (2017).

15. Vaghela, Himali, et al. "A novel approach to detect chronic leukemia using shape based feature extraction and identification with digital image processing." ***International Journal of Applied Information Systems (IJAIS)*** 11.5 (2016).

16. Mohapatra, S., Patra, D., and Satpathy, S., "An ensemble classifier system for early diagnosis of acute lymphoblastic leukemia in blood microscopic images", ***Neural Computing And Applications***, 24 (7–8): 1887–1904 (2014).

17. Yu, W., Chang, J., Yang, C., Zhang, L., Shen, H., Xia, Y., and Sha, J., "Automatic classification of leukocytes using deep neural network", ***Proceedings Of International Conference On ASIC***, 2017-Octob: 1041–1044 (2017).

18. Peng, C. Y. J., Lee, K. L., and Ingersoll, G. M., "An introduction to logistic regression analysis and reporting", ***Journal Of Educational Research***, 96 (1): 3–14 (2002).

19. Dey, U. K. and Islam, M. S., "Genetic Expression Analysis to Detect Type of Leukemia Using Machine Learning", ***1st International Conference On Advances In Science, Engineering And Robotics Technology 2019, ICASERT 2019***, 2019 (Icasert): 1–6 (2019).

20. Wang, J. L., Li, A. Y., Huang, M., Ibrahim, A. K., Zhuang, H., and Ali, A. M., "Classification of White Blood Cells with PatternNet-fused Ensemble of Convolutional Neural Networks (PECNN)", ***2018 IEEE International Symposium On Signal Processing And Information Technology, ISSPIT 2018***, 325–330 (2019).

21. Pansombut, Tatdow, et al. "Convolutional neural networks for recognition of lymphoblast cell images." ***Computational Intelligence and Neuroscience*** (2019).

22. Brameshuber, Mario, et al. "Understanding immune signaling using advanced imaging techniques." ***Biochemical Society Transactions*** 50.2 (2022).

23. Ishikawa, T., Takahashi, J., Takemura, H., Mizoguchi, H., and Kuwata, T., "Gastric lymph node cancer detection of multiple features classifier for pathology diagnosis support system", ***Proceedings - 2013 IEEE International Conference On Systems, Man, And Cybernetics, SMC 2013***, 2611–2616 (2013).

24. Demir, Önder, and Ali Yılmaz Çamurcu. "Computer-aided detection of lung nodules using outer surface features." ***Bio-medical materials and engineering*** 26.s1 (2015).

25. Selçuk, Osman, and Figen Özen. "Acute lymphoblastic leukemia diagnosis using image processing techniques." ***2015 23nd Signal Processing and Communications Applications Conference (SIU)*. IEEE**, (2015).

26. Selçuk, Osman, and Figen Özen. "Acute lymphoblastic leukemia diagnosis using image processing techniques." ***2015 23nd Signal Processing and Communications Applications Conference (SIU)*. IEEE**, (2015).

26. Abdoulaye, Issa Baban Chawai, and Önder Demir. "Mamografi Görüntülerinden Kitle Tespiti Amacıyla Öznitelik Çıkarımı." ***Fatih Sultan Mehmet Vak{\i}f {\"U}niversitesi Biyomedikal ElektronikTasar{\i}m~…}***(2017).

27. Pansombut, Tatdow, et al. "Convolutional neural networks for recognition of lymphoblast cell images." ***Computational Intelligence and Neuroscience*** (2019).

28. Wang, Chong, et al. "Efficient and Highly Accurate Diagnosis of Malignant Hematological Diseases Based on Whole-Slide Images Using Deep Learning." ***Frontiers in Oncology*** 12 (2022).

29. Kumar, Deepika, et al. "Automatic detection of white blood cancer from bone marrow microscopic images using convolutional neural networks." ***IEEE Access*** 8 (2020).

30. de Oliveira, José Elwyslan Maurício, and Daniel Oliveira Dantas. "Classification of Normal versus Leukemic Cells with Data Augmentation and Convolutional Neural Networks." ***VISIGRAPP (4: VISAPP)***. 2021.

31. Zhao, J., Zhang, M., Zhou, Z., Chu, J., and Cao, F., "Automatic detection and classification of leukocytes using convolutional neural networks", ***Medical And Biological Engineering And Computing***, 55 (8): 1287–1301 (2017).

32. Marzahl, Christian, et al. "Classification of leukemic b-lymphoblast cells from blood smear microscopic images with an attention-based deep learning method and advanced augmentation techniques." ***ISBI 2019 C-NMC challenge: classification in cancer cell imaging*. Springer**, Singapore, (2019).

33. Sipes, R. and Li, D., "Using convolutional neural networks for automated fine grained image classification of acute lymphoblastic leukemia", ***Proceedings - 3rd International Conference On Computational Intelligence And Applications, ICCIA 2018***, 157–161 (2018).

34. Vinayakumar, R., Alazab, M., Member, S., and Soman, K. P., "Deep Learning Approach for Intelligent Intrusion Detection System", ***IEEE Access***, 7: 41525–41550 (2019).

35. Macduff, Anne. "Deep learning, critical thinking and teaching for law reform." ***Legal Education Review*** 15.1 (2005): 125-135.

36. Shone, Nathan, et al. "A deep learning approach to network intrusion detection." ***IEEE transactions on emerging topics in computational intelligence*** 2.1 (2018).

37. Aftab, Muhammad Omer, et al. "Executing spark BigDL for leukemia detection from microscopic images using transfer learning." ***2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA)*. IEEE**,(2021).

38. Alajrami, Eman, et al. "Blood donation prediction using artificial neural network." ***International Journal of Academic Engineering Research (IJAER)*** 3.10, 1–7 (2019).

39. O'Shea, Keiron, and Ryan Nash. "An introduction to convolutional neural networks." ***arXiv preprint arXiv:****1511.08458*, 1–11 (2015).

40. Hidaka, Akinori, and Takio Kurita. "Consecutive dimensionality reduction by canonical correlation analysis for visualization of convolutional neural networks." ***Proceedings of the ISCIE international symposium on stochastic systems theory and its applications***, (2017).

41. Ma, Wei, and Jun Lu. "An equivalence of fully connected layer and convolutional layer." ***arXiv preprint arXiv****:1712.01252* (2017).

42. Alajrami, Eman, et al. "Handwritten signature verification using deep learning." ***International Journal of Academic Multidisciplinary Research (IJAMR)*** 3.12 (2020).

43. Christlein, Vincent, et al. "Deep generalized max pooling." ***2019 International conference on document analysis and recognition (ICDAR)*. IEEE**, (2019)

44. Gholamalinezhad, Hossein, and Hossein Khosravi. "Pooling methods in deep neural networks, a review." ***arXiv preprint arXiv:****2009.07485* (2020).

45. Jeczmionek, Ernest, and Piotr A. Kowalski. "Flattening Layer Pruning in Convolutional Neural Networks." ***Symmetry*** 13.7 (2021): 1147.

46. Alkronz, Eyad Sameh, et al. "Prediction of whether mushroom is edible or poisonous using back-propagation neural network.", ***IJARW***: (2019).

47. Molchanov, Dmitry, Arsenii Ashukha, and Dmitry Vetrov. "Variational dropout sparsifies deep neural networks." ***International Conference on Machine Learning*****. PMLR**, (2017).

48. Ba, Jimmy, and Brendan Frey. "Adaptive dropout for training deep neural networks." ***Advances in neural information processing systems*** 26 (2013).

49. Moolayil, Jojo, Jojo Moolayil, and Suresh John. *Learn Keras for deep neural networks*. Birmingham: Apress,***Springer***: (2019).

50. Park, Sungheon, and Nojun Kwak. "Analysis on the dropout effect in convolutional neural networks." **Asian conference on computer vision. Springe**r, Cham, 2016.

51. Bjorck, Nils, et al. "Understanding batch normalization." ***Advances in neural information processing systems*** 31 (2018).

52. Basha, SH Shabbeer, et al. "Impact of fully connected layers on performance of convolutional neural networks for image classification." ***Neurocomputing*** 378 (2020): 112-119.

53. Jais, Imran Khan Mohd, Amelia Ritahani Ismail, and Syed Qamrun Nisa. "Adam optimization algorithm for wide and deep neural network." ***Knowledge Engineering and Data Science*** 2.1 (2019): 41-46.

54. Dubey, Shiv Ram, et al. "diffGrad: an optimization method for convolutional neural networks." ***IEEE transactions on neural networks and learning systems*** 31.11 (2019).

55. Sommer, Christoph, and Daniel W. Gerlich. "Machine learning in cell biology–teaching computers to recognize phenotypes." ***Journal of cell science*** 126.24 (2013).

56. Ain, Qurat Ul, et al. "Diagnosis of Leukemia Disease through Deep Learning using Microscopic Images." ***2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2)*****.** IEEE, (2022).

57. Féraud, Raphael, and Fabrice Clérot. "A methodology to explain neural network classification." ***Neural networks*** 15.2 (2002).

58. Gulli, Antonio, Amita Kapoor, and Sujit Pal. Deep learning with TensorFlow 2 and Keras: regression, ConvNets, GANs, RNNs, NLP, and more with TensorFlow 2 and the Keras API. ***Packt Publishing Ltd***, (2019).

59. Media, O. R., "And Tensorflow , 2nd edition by Aurélien Géron", ***Physical And Engineering Sciences In Medicine***, 43 (3): 1135–1136 (2020).

60. Stevens, Eli and Antiga, Luca and Viehmann, T., "Deep Learning with PyTorch", ***Manning Publications***, (2020).

61. Garreta, Raul, and Guillermo Moncecchi. *Learning scikit-learn: machine learning in python*. ***Packt Publishing Ltd***, (2013).

62. Travis, O., "Numpybook", ***Trelgol Publishing USA***, 378 (2007).

63. Richert, Willi. Building machine learning systems with Python. ***Packt Publishing Ltd***, (2013).

64. Personal, E., "ALL-IDB : THE ACUTE LYMPHOBLASTIC LEUKEMIA IMAGE DATABASE FOR IMAGE PROCESSING Ruggero Donida Labati IEEE Member , Vincenzo Piuri IEEE Fellow , Fabio Scotti IEEE Member Universit ` a degli Studi di Milano , Department of Information Technologies ,", ***IEEE Journal Of Translational Engineering In Health And Medicine***, (2011).

65. Gupta, R., Gehlot, S., and Gupta, A., "C-NMC : B-Lineage Acute Lymphoblastic Leukaemia : A Blood Cancer Dataset C-NMC : B-lineage acute lymphoblastic leukaemia : A blood cancer dataset", ***Medical Engineering And Physics***, 103 (May): 103793 (2022).

66. López, R. R., Sánchez, L., Alazzam, A., Burnier, J. V, and Stiharu, I., "Numerical and Experimental Validation of Mixing Efficiency in Periodic Disturbance Mixers", ***MDPI***, 1–13 (2021).

67. Shorten, C. and Khoshgoftaar, T. M., "A survey on Image Data Augmentation for Deep Learning", ***Journal Of Big Data***, (2019).

68. Salamon, J. and Bello, J. P., "Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification", ***IEEE Transactions On Biomedical Engineering***, 1–5 (2017).

69. Li, W., Chen, C., Zhang, M., Li, H., and Du, Q., "Data Augmentation for Hyperspectral Image Classification With Deep CNN", ***IEEE Geoscience And Remote Sensing Letters***, 16 (4): 593–597 (2019).

70. Jogin, M., "Feature Extraction using Convolution Neural Networks (CNN) and Deep Learning", ***2018 3rd IEEE International Conference On Recent Trends In Electronics, Information & Communication Technology (RTEICT)***, (November): 2319–2323 (2020).

71. Shaheen, F. and Verma, B., "Impact of Automatic Feature Extraction in Deep Learning Architecture", ***IEEE Journal Of Translational Engineering In Health And Medicine***, (2016).

72. Hussain, M., Bird, J. J., and Faria, D. R., "A Study on CNN Transfer Learning for Image Classification A Study on CNN Transfer Learning for Image Classification", ***Springer***, (October 2018): (2022).

73. Bengio, Yoshua. "Deep learning of representations for unsupervised and transfer learning." ***Proceedings of ICML workshop on unsupervised and transfer learning. JMLR Workshop and Conference Proceedings***, (2012).

74. Modeling, L. M., Measurement, F., Snowrift, O. N., Environmental, A. R., Regional, S., Power, E., Limited, G. C., Influence, T. H. E., Snow, O. F., Title", ***Journal Of Wind Engineering And Industrial Aerodynamics***, 26 (3): 1–4 (2019).

75. X-ray, C., Rahman, T., Chowdhury, M. E. H., and Khandakar, A., "Applied sciences Transfer Learning with Deep Convolutional Neural Network (CNN) for Pneumonia Detection Using", ***MDPI***, .

76. Weiss, K., Khoshgoftaar, T. M., and Wang, D., "A Survey of Transfer Learning", Journal of Big Data, ***Springer International Publishing***, (2016).

77. Fan, H., Gao, S., Ma, H., and Liu, Q. I., "Intelligent Recognition of Ferrographic Images Combining Optimal CNN With Transfer Learning Introducing Virtual Images", ***IEEE Transactions On Systems, Man And Cybernetics Part C: Applications And Reviews***, 8: (2020).

78. Bhowmik, D. and Islam, M. T., "A Deep Face-Mask Detection Model using DenseNet169 and Image Processing techniques", ***Brac University***, (January): (2022).

79. Alafandy, K. A., Omara, H., Lazaar, M., and Achhab, M. Al, "Using Classic Networks for Classifying Remote Sensing Images : Comparative Study Using Classic Networks for Classifying Remote Sensing Images : Comparative Study", ***Advances In Science, Technology And Engineering Systems Journa***, (October): (2020).

80. Albardi, F., Kabir, H. M. D., and Bhuiyan, M. I., "A Comprehensive Study on Torchvision Pre-trained Models for Fine-grained Inter-species Classification", ***IEEE Transactions On Image Processing***, .

81. Vulli, A., Srinivasu, P. N., Sai, M., Sashank, K., Shafi, J., Choi, J., and Ijaz, M. F., "Fine-Tuned DenseNet-169 for Breast Cancer Metastasis Prediction Using FastAI and 1-Cycle Policy", ***MDPI***, 1–25 (2022).

82. Huang, Gao, et al. "Densely connected convolutional networks." ***Proceedings of the IEEE conference on computer vision and pattern recognition***. (2017).

83. Jaworek-Korjakowska, Joanna, Pawel Kleczek, and Marek Gorgon. "Melanoma thickness prediction based on convolutional neural network with VGG-19 model transfer learning." ***Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops***. (2019).

84. Karac, A., "VGGCOV19-NET : automatic detection of COVID-19 cases from X-ray images using modified VGG19 CNN architecture and YOLO algorithm", ***Springer***, 0123456789: 8253–8274 (2022).

85. Khan, M. A., Rajinikanth, V., Satapathy, S. C., Taniar, D., Mohanty, J. R., Tariq, U., and Damaševiˇ, R., "VGG19 Network Assisted Joint Segmentation and Classification of Lung Nodules in CT Images", ***IEEE Transactions On Medical Imaging***, 1–16 (2021).

86. Wen, L., Li, X., Li, X., and Gao, L., "A new transfer learning based on VGG-19 network for fault diagnosis", ***Proceedings Of The 2019 IEEE 23rd International Conference On Computer Supported Cooperative Work In Design, CSCWD 2019***, 205–209 (2019).

87. Bansal, M., Kumar, M., Sachdeva, M., and Mittal, A., "Transfer learning for image classification using VGG19 : Caltech ‐ 101 image data set", ***Journal Of Ambient Intelligence And Humanized Computing***, (0123456789): (2021).

88. Sun, Jing, et al. "Deep learning-based light scattering microfluidic cytometry for label-free acute lymphocytic leukemia classification." ***Biomedical Optics Express*** 11.11 (2020): 6674-6686.

89. Rezayi, Sorayya, et al. "Timely Diagnosis of Acute Lymphoblastic Leukemia Using Artificial Intelligence-Oriented Deep Learning Methods." ***Computational Intelligence and Neuroscience*** (2021).

RESUME

Hikmat Ibrahim ABED graduated first and elementary education in Anbar-Iraq. He completed high school education at (Al-Nahda High School) in Anbar Governorate , then, he obtained bachelor's degree from University of Al-Maaref/College of Computer sciences and Mathematics department of Computer sciences in 2012. After graduation he worked with UNDP as field monitors. To complete M.Sc. education, he moved to Karabuk/Turkey in 2019. He started his master education at the department of computer engineering in Karabuk University.