National College of Ireland

Leukemia Cell Classification with using DC-GAN versus 3 Traditional Techniques

MSc Research Project
Data Analytics

Urun Gungor
Student ID: X20246404

School of Computing
National College of Ireland

Supervisor: Dr. Catherine Mulwa

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Urun Gungor
Student ID:	X20246404
Programme:	Data Analytics
Year:	2023
Module:	MSc Research Project
Supervisor:	Dr. Catherine Mulwa
Submission Due Date:	14/08/2023
Project Title:	Leukemia Cell Classification with using DC-GAN versus 3 Traditional Techniques
Word Count:	XXX
Page Count:	33

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	17th September 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Leukemia Cell Classification with using DC-GAN versus 3 Traditional Techniques

Urun Gungor
X20246404

Abstract

Leukemia is one of the deadliest cancer types. Therefore, early diagnosis is one of the most critical processes of cancer treatment. Traditional diagnosis of leukemia is a long, costly, and complex process involving the possibility of expert error. Therefore, many researchers focused on this issue with machine learning. But the problem is that the datasets available are imbalanced because of patients' privacy therefore the classification models are inclined to overfit.

This research focuses on increasing classification model accuracy by solving the imbalanced data problem by using Deep Convolutional Generative Adversarial Network and 3 traditional techniques which are Adaptive Synthetic (ADASYN) sampling approach, Weighted random sampling, and data augmentation. The results of the classification models are compared with their accuracy, data generation speed, and cost.

According to the results fast and high-quality images were generated using DC-GAN and the classification model was developed by adding those images. On the other hand, traditional techniques produced low-quality images and did not solve the overfitting problem on the classification model.

1 Introduction

Leukemia is a malignancy process of the transformation of normal blood cells into white blood cells. Myeloid leukemia begins in the marrow cell that produces red blood cells, platelets, and white blood cells, while lymphoblastic leukemia begins in the white blood cells in the bone marrow. Fast-spreading leukemia is called acute leukemia, and slow-spreading leukemia is called chronic leukemia. There are four main types of leukemia which are acute lymphoblastic leukemia (ALL), Acute Myeloid Leukemia (AML), Chronic Lymphocytic Leukemia (CLL), and Chronic Myeloid Leukemia (CML) (Ramaneswaran et al.; 2021).

White Blood Cells (WBCs) produced in the bone marrow join the blood circulation and fight infections by attacking bacteria, viruses, and microbes that enter the body within the immune system. There are five main types of white blood cells neutrophils, lymphocytes, eosinophils, monocytes, and basophils. Neutrophils, basophils, and eosinophils have small granules (Alharbi et al.; 2022).

In acute lymphoblastic leukemia, the B-lymphoblast cell becomes cancerous and proliferates uncontrollably, making it fatal. The most important sign of the diagnosis of ALL is the presence of high numbers of immature leukocytes or lymphoid cells in the bone

marrow, and rapid lymphoid cell production. Leukemia symptoms are sudden weight loss, infections, recurrent nosebleeds, swollen lymph nodes, night sweats, bone pain, and anemia (Gupta et al.; 2022).

The traditional diagnosis is made by experts through a blood count test or bone marrow examination, but this is a long and complex process and takes risks of being misunderstood or misinterpreted by experts. On the other hand, flow cytometry and Polymerase Chain Reaction (PCR) amplification of antigen-receptor genes is another classical diagnostic method but requires expensive equipment that cannot be found easily. Therefore, diagnosis of ALL with artificial intelligence is cheap, fast, and highly functional because it does not require expert opinion.

1.1 Motivation and Project Background

Fast, cheap, and accurate leukemia classification is vital for patients in the biomedical field. Artificial intelligence has eliminated the need for experts, and in recent years, deep learning models have been proven to provide accurate and rapid diagnosis. Resnet-50 and CNN (Convolutional Neural Networks) architectures have the highest and most effective results in the field of leukemia image classification in the literature. On the other hand, finding a large, publicly available dataset is difficult, as personal data is not shared by patients or ethical procedures. Therefore many datasets are imbalanced in the medical field.

In this research C- NMC 2019 dataset [1] is chosen due to being very common in the literature but it is imbalanced. As a solution to this problem, the Gan Network has been used very effectively in the medical field in recent years because it generated fast and realistic data.

The most powerful area for using Gan Network is to balance imbalanced datasets by generating synthetic data in the literature. Furthermore, DC-GAN is chosen as a technique due to becoming just recently popular and having some gaps in the literature. On the other hand, traditional methods do not generate synthetic data but reproduce data. Even though they give fast results, they cannot produce high-quality images and the model has the risk of overfitting.

This research presents classification with the original dataset, classification adding the data generated by DC-GAN into the dataset, and classification adding augmented new data to the dataset produced by Adaptive Synthetic (ADASYN), data augmentation, and weighted random sampling. The contributions of DC-GAN and traditional methods to the results in terms of speed and cost have been examined in detail. In addition, ADASYN is based on the Synthetic Minority Over-sampling Technique (SMOTE) and is new to the literature since it has been introduced in recent years. On the other hand, data augmentation is preferred in sub-processes such as data preprocessing, as it produces similar data and makes biased estimates. It was also questioned whether data augmentation alone is sufficient in early diagnosis, whether it is effective, and to what extent it is effective.

Patients have privacy right to keep their personal data. Therefore, many datasets in the literature are imbalanced for acute lymfobic leukemia. This research was prepared to contribute to the automated diagnosis of leukemia using a neural network that focused on 3 main parts. In the first part, Deep Convolutional Generative Adversarial Network

[1]C- NMC 2019 Dataset:`https://wiki.cancerimagingarchive.net/pages/viewpage.action?pageId=52758223`

was chosen to balance the C-NMC 2019 dataset by generating data and discussing the contributions of the classification model. In the second part, 3 traditional techniques are selected (ADASYN, Weighted Random Sampling, and data augmentation) based on oversampling to produce data and discuss contributions to the classification model. In the last step, the contributions of 3 traditional techniques and the DC-Gan model to classification model results were compared.

1.2 Research Question

RQ:*"How effective are GAN Networks in a high-speed sampling of the on the imbalanced C-NMC 2019 Dataset to improve classification models?"*

In recent years, GAN Network has been effectively used in the biomedical field to produce synthetic data due to its high quality and speed.

Sub RQ: *"How effective are ADASYN, weighted random sampling and data augmentation at oversampling data to improve classification models using the C-NMC 2019 Dataset?"*

To solve the research question the following objectives are specified and implemented.

1.3 Research Objectives

Obj1: A critical review of literature Image Leukemia Classification (2002-2023)

Obj2(a): Prepocessing of Leukemia Dataset

Obj2(b): Implementation, evaluation, and results of Classification

Obj2(c): Implementation, evaluation, and results of DC-Gan based model classification

Obj2(d): Implementation, evaluation, and results of ADASYN based model classification

Obj2(e): Implementation, evaluation, and results of Weighted Random Sampling based model classification

Obj2(f): Implementation, evaluation, and results of Data augmentation classification

Obj2(h): Comparison of used techniques.

Obj2(i): Comparison of the proposed model with existing models.

The rest of the technical report is structured as follows, chapter 2 presents existing literature on image Classification, Acute lymphoid leukemia Classification, GAN Network, investigation about DC-GAN, traditional techniques, dataset challenges, and comparison. Chapter 3 presents the scientific methodology of ALL detection, and finally chapter 4 presents implementation evaluation and results.

2 Related Work

2.1 Introduction

This literature review investigates the detection of image classification in the medical field. This section is divided into many subsections, (i) Literature Review on Image Classification and Identified Gaps, (ii) Acute Lymphobic Leukemia Classification and identified gaps (2015 -2023), (iii) GAN Network and identified gaps (2014-2023), (iv)Investigation of DCGAN, (v) Traditional Techniques and identified gaps (2002-2023), (vi) Critique of

Techniques, Models, Metrics, (vii) Dataset challenges of Acute Lymphobic Leukemia and (viii) Reviewed Acute Lymphobic Leukemia Comparison respectively.

2.2 Literature Review on Image Classification and Identified Gaps

In the perspective of leukemia classification (Mittal et al.; 2022) summarized 149 papers for defining literature gaps and containing the best methods among machine learning, deep learning, and hybrid methods and result comparisons. Furthermore, the paper also contains information about the preprocessing techniques detected such as noise filtering, color space conversion, noise filtering, contrast enhancement, and data augmentation.

The most common segmentation techniques are intensity thresholding, edge detection-based methods, region-based methods, and deformable model-based methods. Deformable model-based methods are classified as parametric model-based methods, geometric model-based methods, machine learning–based methods and hybrid methods. On the other hand, the most common classification techniques are KNN, support vector machine, artificial neural networks, and ensemble classifiers.

For overlapping and aggregating cells, machine learning performed better compared to deep learning.

In addition, (Wang et al.; n.d.) defined machine learning techniques have multi-step processes which are geometric and statistical features on preprocessing and classification of images in the medical field. Deep learning methods are more effective compared to machine learning methods on image classification due to the use of feature engineering.

In the literature (Mondal et al.; 2021b), different deep learning-based methods provided effective results in areas of detection and segmentation such as skin lesions, breast cancer, brain tumors, diabetic retinopathy, the COVID-19 pandemic, and minimally invasive surgery.

In the literature, the biggest challenges were that there are colour similarities between the background and cell image itself and the cells overlapping each other which they caused difficult for the algorithms to classify. This situation is solved by feature extraction due to deep learning being very effective in image classification.

2.3 Acute Lymphobic Leukemia Classification and Identified Gaps (2015 -2023)

(Amin et al.; 2015) proposed a 5-step method based on image preprocessing, kernel segmentation, image postprocessing, feature extraction, and classification. The main goal is to achieve better image classification results through methods such as image preprocessing, color space conversion, and histogram equalization. Furthermore, cell nucleus segmentation is based on K-means clustering and morphological processing which separates each nucleus from clusters and finally, the geometric and statistical properties of the cell nucleus are classified.

As an example of the difficulty of classifying cell similarities (Gupta et al.; 2023) works with the largest publicly available dataset including B-ALL and healthy cancer cells which includes the cytoplasm, cell nucleus, and plasma cells that are captured from two different cameras, with different pixel sizes, and could not be separated with a clear border due to the absence of color difference between the cell background and the cytoplasm, which were classified using five different models. Additionally, by using the feature extraction

method, CNN , and statistics-based Salp Swarm Algorithm (SESSA) together 95.2% F-1 score was obtained.

Future extraction is the most important and distinctive step in cell classification. When used effectively as an intermediate process, it significantly increases the success of the techniques, for instance

(Negm et al.; 2018) classified leukemia cells by blending feature extraction, decision tree, and MLP (Multi Layer Perceptron) techniques with K-means clustering, and segmentation and achieved 99.5% accuracy.

The structural similarity of leukemia cells makes the data preprocessing step vital to achieve better classification results. Therefore, the selected techniques are supported by statistical or morphological feature extraction.

2.4 GAN Network and Identified Gaps (2014-2023)

The GAN Network has recently been introduced to medical fields as an effective method for early and accurate diagnosis of diseases.

GAN Network is an unsupervised and semi-supervised machine learning algorithm introduced in 2014 to develop decision processes in the biomedical field. To avoid overfitting the training set is never included in the generator. During training, the dataset must have enough data in order not to raise the computational cost and to generate new data (Goodfellow et al.; 2020).

There are many sub-types of GAN Network and its usage areas are diverse. In order to classify the usage areas and sub-techniques in the literature (Ali et al.; 2022) summarized 57 articles about COVID-19 diagnosis in 2020 and beyond. GAN was used at most with 74% of the studies to work on the data imbalance problem with data augmentation. CNN is the most successful architecture for the biomedical field decision-making process which is used for diagnosis (9%) representing 5 articles, superresolution (5%), segmentation (5%), and feature extraction (5%) representing 5 articles respectively. Additionally, GAN was implemented in 17 articles (30%), CycleGAN in 9 articles (16%), Conditional GAN in 9 articles (16%), and deep convolutional GAN in 4 articles (7%) respectively.

It is not possible to achieve good results with datasets having unbalanced data to yield effective results. Therefore, it is inconvenient to use them while diagnosing a disease in the medical field. In the perspective of balancing unbalanced datasets by producing synthetic data, (Li et al.; 2022) proposed EID-GAN, supported by a data augmentation technique to balance highly imbalanced datasets, has a new algorithm based on outliers. The goal is to optimize the generator's parameters using K-fold cross-validation while improving the quality of the generated images to combine the two outlier detectors. F1 score was increased from 94% to 98% by using synthetic images with the current dataset.

The use of GAN Network in the medical field is getting more and more popular and the use of DC-GAN is increasing. Balancing unbalanced data sets by generating synthetic data with CNN architecture is the area where the GAN Network gives the most effective results.

2.5 Investigation of DC-GAN

DC-GAN is one of the most effective Gan Network networks in the literature (Radford et al.; 2015). The architecture of Deep Convolutional GANs has a discriminator and a generator that use collective norm rather than putting the layers together; this makes

training of neural networks faster and more stable to the generator and the discriminator by providing normalization of the input layers by re-centering and re-scaling.

To create deeper architectures, connected hidden layers are deleted and ReLU activation is used in the generator for all layers, and LeakyReLU activation in the discriminator for all layers.

$min_G max_G V(G, D) = E_x \sim p_r(x) [\log D(x)] + E_z\ p \sim_z (z) [1 - \log(D(G(z)))]$

The aforementioned (Tong et al.; 2022) formula During training, a discriminator (D) and a generator (G) always optimize to a Nash equilibrium where D is unable to distinguish between actual and artificial data. The generator obtains data from a noise distribution (pz(z)) in order to produce artificial data (G(z)) that is comparable to actual data from an original data distribution (pr(x)). By including class labels in the generator and discriminator, it may be transformed into a conditional model.

2.6 Traditional Techniques and Identified Gaps (2002-2023)

Since high data quality is vital in the correct diagnosis of the disease, traditional techniques are generally used in the literature for intermediate processes and for comparison purposes (Chawla et al.; 2002). They are generally divided into 2 namely oversampling and undersampling. In addition, the increase in data processing time means loss of time and energy consumption. Traditional methods resample the data and therefore produce fast data. SMOTE(Sampling Algorithm is a Synthetic Minority Over-sampling) is the most popular oversampling method as it produces fast data for imbalanced datasets. However, overfitting may occur in the model, and data quality may be poor. Undersampling causes data loss and is not preferred in the literature because it does not give effective results.

ADASYN is a SMOTE-based algorithm to improve model performance from overfitting caused by class inequality (Gosain and Sardana; 2017). ADASYN is applied to minority classes based on class weights, and by augmenting data, it reduces the performance error caused by the imbalance data amount between classes, and the more data it trains, the more effective it is (Mohanty et al.; 2019). On the other hand, random sampling is a type of oversampling and generates data with statistical theories by choosing a random sample from the population (Cook et al.; 2002).

The class weight method is a type of random sampling that is created by assigning a higher class weight to minority classes and a lower class weight to majority classes (Depto et al.; 2023).

$Weight = NTNC * NCn(1)$

Here C represents a particular class for which the class weight is calculated (ALL or Healthy), NT is the total number of samples in the dataset, NC is the number of classes, and NC is the number of samples in a particular class to calculate the class weight.

Working with high-quality images is critical for accurate diagnosis in the medical field. For this reason, data augmentation is often used as an intermediate process in the literature due to producing low-quality data.

(de Sant'Anna et al.; 2022) used Gaussian Naive Bayes, KNN, a linear vector machine classifier combined with ANN architecture and support with data augmentation was used to balance the training and validation sets and was not used to the test images. Data augmentation methods are rotation, blurring, mirroring, shearing transformation, and addition of salt-and-pepper noise, 60 degrees clockwise rotation.

SMOTE is the most preferred oversampling method. ADASYN, although is a Smote-based oversampling method, has been used in the literature in recent years. While data augmentation is preferred in the preprocessing step instead of a stand-alone technique, weighted random sapling is one of the most popular random sampling methods. These techniques give fast results but do not produce new data.

2.7 Critique of Techniques, Models, Metrics

In this section, the architectures used in the literature with the perspective of CNN networks and the preferred Resnet-50 architecture during this research are examined with the preferred metrics.

CNNs are very popular. CNN uses end-to-end learning and can extract task-specific features from the data itself. It works better with large training datasets. However, this requirement can be achieved by transfer learning, in which a pre-trained network is used for direct feature extraction and fine-tuning on another dataset (Gupta et al.; 2022).

In order to clarify the cell differences between tissues, (Duggal et al.; 2017) achieved 93.2% accuracy in leukemia detection by adding the SD Layer (Stain Deconvolutional Layer), which can be converted to optical density colour space to microscopic images, to the CNN architecture.

(Ahmed et al.; 2019) created a CNN architecture that can classify all subtypes of leukemia that combined techniques such as Naive Bayes, Support Vector Machine, K-Nearest Neighbor, and Decision Tree working on two publicly available leukemia datasets.

CNNs are widely and effectively used for image classification. The most famous architectures are VGG16, which is the first deep convolutional neural network, while ResNet-50 is the first CNN with residual learning modules that allow networks to have any number of deep layers. On the other hand, DenseNet121 introduced feature reuse through dense connections. ResNet-50 is a deep convolutional network with includes 48 convolution layers, with 1 maximum pool and 1 average pool layer. Deep neural networks have the gradient problem, which is lost due to their depth. ResNet-50 comprises overlapping redundant blocks to form its network, learning residual functions according to the layer inputs. Each block has a convolutional layer, a Batch Normalization layer, and a ReLU activation function that is connected to the previous and next blocks. Additionally, the maximum pooling layer and global average pooling layer are put below and on top of the general architecture (He et al.; 2016).

On the other hand (Liu et al.; 2022) developed AIMIC (artificial intelligence-based microscopy image classifier), a software that does not require deep learning methods classification code for microscopic images.

Additionally, (Alharbi et al.; 2022) built a model, which detected leukocytes from blood samples using three datasets of three different WBC types with Resnet and UNet architecture and achieved 96% accuracy to separate nuclei and cytosols.

Transfer learning architectures reduce the time required to train. Within this context, (Honnalgere and Nayak; 2019) used VGG16 and transfer learning to adjust parameters, normalized activations, and limited activation values to between 1 and -1 without batch normalization.

CNN is the most powerful architecture for image classification, and it gives effective results when used with Resnet-50, which is a deep convolutional neural network. Additionally, mobile Unet and Alexnet are also is also very common and effective in this field. Within the scope of performance evaluation, speed and operation time are the two most

important criteria.

2.8 Dataset Challenges of Acute Lymphocytic Leukemia

The steps followed while preparing the dataset are slide preparation, microscopic image capture, stain normalization, cell annotation, and cell classification. However, there are many complex steps involved in the preparation of this processed dataset. Preparation of medical data in diseases such as leukemia classification is a difficult and long process.

Standard protocols are followed, but images prepared by different technicians may have different color contrasts. Images have different pixels as they are created using different devices such as a microscope or camera. Additional explanations are needed by the experts in the references, as any wrong decision to be made while diagnosing the patients may result in death. The structure and size of the nucleus and cytoplasm of different cells differ. When the cells are in conjoint position, the cytoplasm of the two cells comes into contact, the cytoplasm of one cell comes into contact with the nucleus of the other, or the nuclei of the cells interact with each other. Therefore, different models need to be used to divide clusters into individual cells. The segmentation of cells may be close to the background color of the entire image, making it difficult to distinguish cell borders. The datasets are generally imbalanced, the classifier will be biased towards the majority class.

2.9 Acute Lymphobic Leukemia Literature Comparison

Researcher Name	Architecture	Technics	Metrics	Type	Cell Type
Das et al 2022	Resnet -18	Combine Logistic Regression and SVM	93.5% specificity 100% sensitivity 91.3% precision 96.% accuracy 0.95 f1 score	ALL Classification	Acute Lymphoblastic Cell
	Resnet -18	Deep learning with Hybrid models	98.4% specificity 96% sensitivity 98.5% precision 97% accuracy 0.97 f1 score	ALL Classification	Acute Lymphoblastic Cell
Ghaderzadeh et al. 2022	8 popular CNN networks	Feature Extraction from B-ALL Ensemble Classifier	99.4% sensitivity 96.7% specificity 98.5% accuracy 98.3% AUC	ALL Classification	B-ALL lymphoblast cells
de Sant'Anna et al. 2022	ANN + SVM + NB ANN + SVM + KNN ANN + KNN + NB SVM + KNN + NB	Ensemble Classifier (Based on Naive Bayes, K-Nearest Neighbor, Support Vector Machine, and Neural Network	F1-Score 89.87% 91.20% 93.70% Accuracy 83.19% 86.82% 88.13% Sensitivity 86.60% 88.11% 95.47%	ALL Classification	B-ALL lymphoblast cells
Jamakayala & Gorthi 2021	Proposed a new architecture Multi-channel Cytopathology Analysis Net	-	96.3% accuracy	ALL Classification	Eosinophil, Lymphocyte, Monocyte and Neutrophil)
Liu et al. 2022	Resnet-50	AIMIC (artificial intelligence-based microscopy image classifier),	97 % accuracy, 0.97 % F1 score	ALL Classification	4 microscobic dataset
	MobileNetV2	Latest Deep Learning Technics	95.7% accuracy	ALL Classification	4 microscobic dataset
Mondal et al. 2021	VGG-16 Xception MobileNet InceptionResNet-V DenseNet-121 Weighted Ensemble of Network	CNN	Weighted Precision 0.84% 0.86% 0.84% 0.83 % 0.82% 0.9% Weighted Recall 0.85% 0.86% 0.84% 0.83% 0.82% 0.9% F1-score 0.85% 0.86% 0.84% 0.84% 0.83% 0.9% Weighted FS 0.85% 0.86% 0.84% 0.84% 0.83% 0.9% Balanced Accuracy 0.83% 0.84% 0.83% 0.83% 0.83% 0.9%	ALL Classification	C-NMC-2019

Figure 1: Comparison of ALL Detection

Figure 1 illustrates ALL classification literature reviewed for this research.

Mondal et al. (2021b) worked with the most popular neural networks and the CNM-C 2019 dataset. DenseNet-121 achieved the most effective result with 82% weighted precision, 82% weighted recall, 83% F1 Score, and 83% balanced accuracy.

Furthermore, (Das et al.; 2022) achieved more effective results than the hybrid models, with 93.5% specificity, 100% sensitivity, 91.3% precision, 96% accuracy, 0.95 f1 score.

On the other hand, (Ghaderzadeh et al.; 2022) made an effective classification by getting 99.4% sensitivity, 96.7% specificity, and 98.5% accuracy results with feature ex-

traction. Additionally, (de Sant'Anna et al.; 2022) achieved 95.47% sensitivity with the architecture created by combining SVM, KNN, and NB. Finally, (Jamakayala and Gorthi; 2021) proposed a new network, achieving 96.3% accuracy.

It can be seen that hybrid models were more effective in leukemia classification.

Reseacher Name	GAN Type	Architecture	Technics	Metrics	Aim
Ahmad et al. 2022	Conditinonal Gan	Resnet-50	CNN	Accuracy was increased from 72% to 96% (with synthetic data) recall 0.76 specificity 0.83 precision 0.83 F1 score 0.80	Balance dataset
Li et al. 2022)	EID- Gan (new model)	CNN	two outlier detectors with k-fold cross-validation	F1 score increased from 94% to 98%	Balanced extremely imbalanced dataset
Zulkifley et al. 2020	DCGAN	LightCovidNet	five-fold cross-correction	Accuracy increased from 94% to 97%.	Balance dataset
Tong et al. 2022	ACGAN-SN	CNN		Accuracy 100 % F1 99.7 % AUC 99.8 %	Balance dataset
	ACGAN-SN	LSTM	–	Accuracy 99.2 % F1 100 % AUC 100 %	Balance dataset
	ACGAN-SN	Resnet	–	Accuracy 100 % F1 99.9 % AUC 99.9 %	Balance dataset
	ACGAN-SN	Alexnet	–	Accuracy 100 % F1 99.3 % AUC 99.6 %	Balance dataset
Saleem et al. 2021	Cycle Gan	DarkNet 53	Segmentation	Accuracy 100%	Balance dataset
	Cycle Gan	ShufeNet	Segmentation	Accuracy 99.7 %	

Figure 2: Comparison of GAN Usage for Image Classification

Figure 2 illustrates image classification models using GAN to balance datasets.

As part of the use of the GAN network for dataset balancing, (Ahmad et al.; 2022) achieved the most significant improvement by increasing the accuracy from 72% to 96% with the Conditional Gan and Resnet-50 architecture.

On the other hand, (Tong et al.; 2022) used the most effective architectures (such as LSTM, Alexnet, Resnet) with ACGAN-SN and achieved the most effective result with ResNet with 99.9% F1 score and 100% accuracy.

Apart from that, (Saleem et al.; 2021) achieved 100% accuracy with Cycle Gan and DarkNet-53.

In the literature, studies on the use of GAN and dataset balancing have yielded to very effective results.

(Li et al.; 2022) improved their F1 score from 94% to 98% with EID-GAN, which they recommended to balance extremely imbalanced datasets. In addition, (Zulkifley et al.; 2020) increased accuracy from 94% to 97% with DC-GAN.

When the data produced with the GAN network was included in the original dataset for classification, it helped to get very high and effective results by resolving data imbalance.

Researcher Name	Architecture	Technics	Metrics	Type	Cell Type	Aim
De Sant'Anna et al. (2021)	CNN	Data Augmentation (lipping, rotation, cropping, blurring, and adding salt-and-pepper noise)	0.9 F1 score.	Morphological Features Extraction (colors as red, green and blue, hue, saturation, value)	Lymphocyte	Produced Synthetic Data
Shafique & Tehsin 2018	AlexNet	Data Augmentation lipping, rotation, cropping, blurring, and adding salt-and-pepper noise	99.5% highest accuracy 0.9 f1 score	Morphological Features Extraction (colors as red, green and blue, hue, saturation, value)	Lymphocyte	Produced Synthtetic Data
Kaur et al. 2022	Combined SVM, and CNN algorithms	with SMOTE	94% recall and 94% F1 score	--	--	Produced Synthtetic Data
Kaur et al. 2022	Combined SVM, and CNN algorithms	with ADASYN	94% precision, 93% recall 93%F1 score	Imbalanced dataset	UCI datasets	Produced Synthtetic Data
Kurniawati et al. 2018	KNN	with ADASYN	highest accuracy, 95.3% precision, 95.5% recall 95.4% f-score 95.3%.	Imbalanced dataset	Laboratory cancer data	Produced Synthtetic Data

Figure 3: Comparison of Traditional Techniques for Image Classification

(Shafique and Tehsin; 2018) achieved highest accuracy with 99.5% and 0.9 F1 score by combining Alexnet and data augmentation. Kurniawati et al. (2018) combined KNN and ADASYN to achieve 95.3% highest accuracy, 95.5% precision and 95.3% F1 score. (De Sant'Anna et al.; 2021) achieved 0.9 F1 score with data augmentation and CNN. Combining SVM and CNN with ADASYN and SMOTE, (Kaur et al.; 2022) achieved 94% precision, 93% recall, 93% F1 score, and 94% recall and 94% F1 score, respectively.

Traditional techniques, when combined with different architecture, achieve much more effective results than used alone.

3 Scientific Methodology Approach Used

3.1 Introduction

There is no business layer in this project. Since the main research motivation is the discovery and optimal use of data, the KDD methodology was preferred.KDD begins with the identification of targets and consists of cyclical steps that include the optimal application of the discovered data to the selected area. The steps are, combining data from multiple data sources, data selection from database related to the project aim, data transformation for best performance parallel with selecting techniques, data mining and feature extraction, selecting techniques, performance measure, and presentation of extracted information to users.

3.2 Modified Knowledge Discovery and Data Mining Methodology

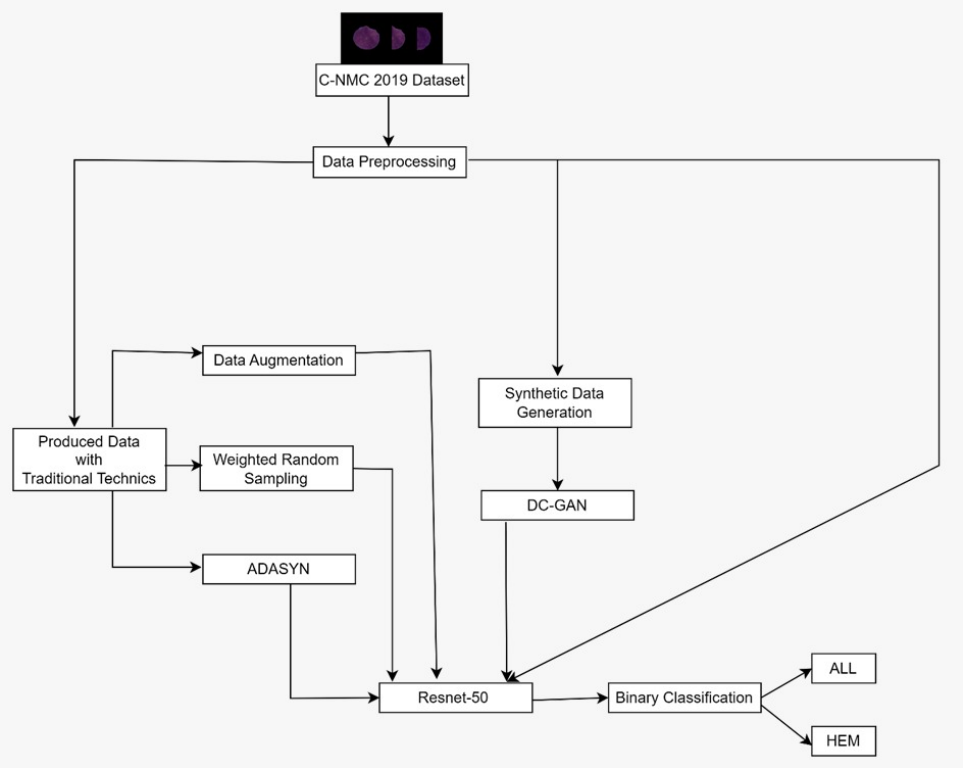

Figure 4: Methodology of leukemia image classification with selected techniques

Modified KDD methodology (Figure 4) for leukemia image processing consists of (i) Data selection from the cancer archive database is collected in .bmp format (ii) Image preprocessing for most effective results (iii) All the data in .bmp format image are extracted to arrays based on Python libraries (iv) Choosing Resnet-50 architecture to get the most effective results (v) Determining the techniques to be applied, for the classification model with the available data, (vi)Determining the techniques to be applied, in order to balance the data set, a) classification model with used traditional methods: ADASYN, data augmentation, and weighted random sampling selected for new data produced b) Classification model with DC-Gan Network for synthetic data generation (vii) Models are evaluated and interpreted by using selected metrics (accuracy, sensitivity, specificity, and precision). (viii) Results and information presented.

3.3 Design Specification

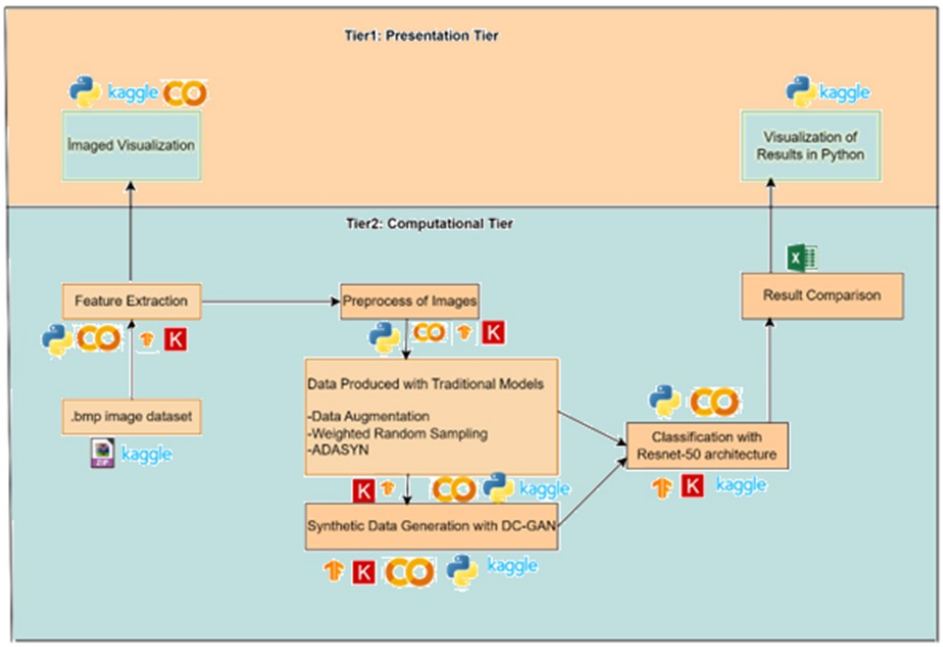

Figure 5: Project Design process of leukemia image classification

The project design process (Figure 4) of leukemia image classification consists of

(i) Presentation tier includes image visualization with Python to present information of results to users

(ii) computational tier includes data selection, exploratory analysis, feature extraction, transformation and training of classification models, evaluation, and the interpretations of 5 classification models.

3.4 Conclusion

The methodology of KDD is modified to the main aim of balancing the CNMC- 2019 dataset for effective classification models. The project design process includes 2 tiers which can be classified as presentational and computational. The computational tier includes all data collection, data preprocessing, architecture selection, and all classification models training and evaluations.The presentation tier includes result visualization to present information to users.

The implementation, evaluation, and results of models to leukemia image classification are interpreted in the next section.

4 Implementation, Evaluation, and Results of Leukemia Image Classification Models

4.1 Introduction

The implementation, evaluation, and results of classification models used in leukemia image classification are discussed in this section. Data preprocessing steps are also explained in detail in this section. To evaluate the models, accuracy, sensitivity, specificity, and precision are used as metrics, and a confusion matrix is used to study the class-wise true or false rate prediction model performance. In the last part of this section implemented models are compared and the model with the highest contribution is selected. It includes the steps of leukemia classification with the imbalanced CNMC-2019 dataset, leukemia classification with balanced dataset by generating synthetic data with DC-GAN and leukemia classification with balanced dataset by data produced by selected 3 traditional techniques which are ADASYN, data augmentation and weighted random sampling. Comparison of results with selected metrics and interpretation of results.

4.1.1 Metrics

In this research sensitivity, specificity, precision, and accuracy metrics are interpreted as performance measures.

$$Sensitivity = \frac{TP}{TP + FN} \tag{1}$$

$$Specificity = \frac{TN}{TN + FP} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Accuracy = \frac{TP + TN}{TotalSamples} \tag{4}$$

The formulas for the evaluation metrics are illustrated above. TP denotes true positive, FP denotes false positive, TN denotes true negative and FN denotes false negative.

4.2 Image Data Preprocessing

4.2.1 Dataset

To avoid ethical dilemmas, the publicly available C-NMC 2019 dataset was selected for this study. The C-NMC 2019 dataset (Clark et al.; 2013) is widely preferred in biomedical research due to having high image quality and large amount of data, and open source.

C-NMC 2019 dataset was selected which includes three parts: a train set with 10,661 total cell images, a preliminary test set with 1,867 total cell images, and a final test set with 2,586 total cell images. The train set has 7,272 images of cancerous cells and 3,389

images of normal cells, while the preliminary test set has 1,219 cancerous and 648 normal cell images. To summarize, there are 7272 cancerous samples representing ALL (0), and 3389 samples representing HEM (1) No cancer, in hypothesis testing. Figure 6 illustrates a sample of the cell images in the dataset.

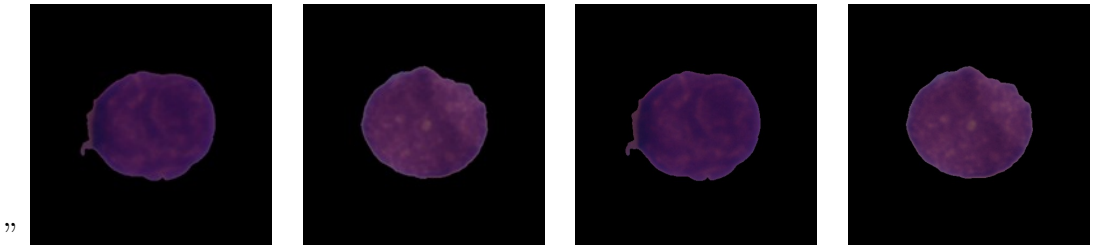

Figure 6: Lukemia Cell Images from C-NMC 2019 Dataset

4.2.2 Data Preprocessing

Firstly, the dataset was uploaded to the cloud services to get better hardware performance. As seen in Figure 6 the image contains a large black background. To get better results while applying classification algorithms the images were cropped from size 410,410 to 210,210 so that the new images contained less background.

Secondly, the image data was converted to a numpy array format. The RGB channel was added as 3rd dimension so the new image size became 210,210,3. The RGB values of the images were between [0, 255]. With using rescaling layer images were standardized and the values were set to be in the [0, 1] range. In the dataset, there is a test folder containing images with labels that are not defined. Therefore that folder was not able to be used.

Finally, the train data was split into 20% test set and 80% train set. The train set consisted of 9572 images, while the test set consisted of 1867 images with two classes HEM and ALL.

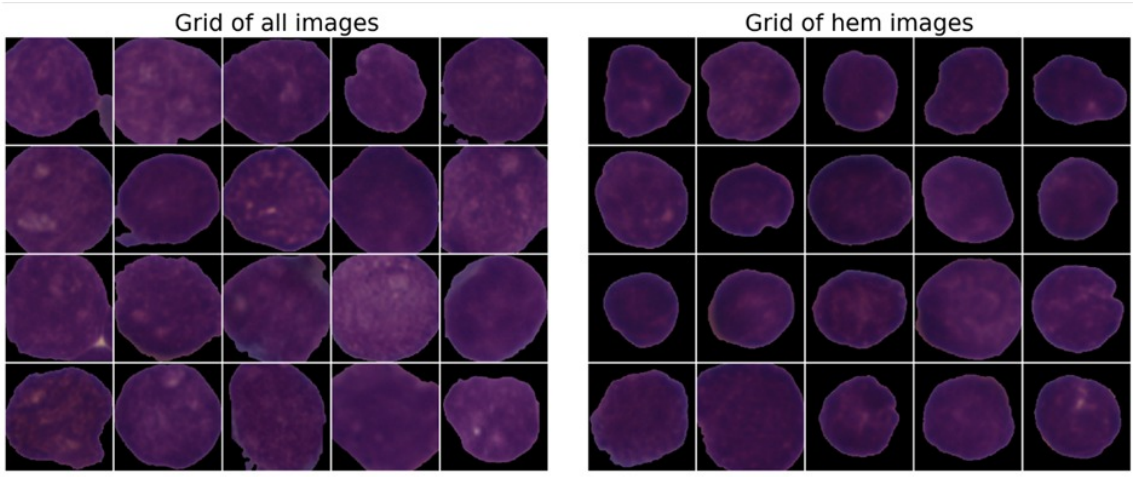

Figure 7: Cropped data after preprocessing

4.3 Implementation,Evaluation and Results of Classification with ResNet-50

The classification model was trained with CNN and ResNet-50 architecture. The deep neural network model was built with 5 layers. Every layer has 5 blocks. For the block-out part, Activation, for the rest blocks Conv2D, and batch normalization were used. The total parameters are 23,587,712.

After that, the first step of transfer learning parameters is defined. Binary cross entropy was chosen as the loss function. The optimizer was chosen as Adam with 0.001 learning rate. Epoch number 5, and batch size was set to 32. Following that, the second step of transfer learning parameters is defined. Binary cross entropy was chosen as the loss function. The optimizer was chosen as Adam with 0.0001 learning rate. Epoch number 10, and batch size was set to 32.

4.3.1 Implementation

The code is written using Python. The tensorflow library was used for model implementation. Kaggle Notebook with GPU as accelerator was used as a Cloud service to obtain results faster. The model was trained with Keras, Tensorflow, Numpy, and Pandas libraries and many sub-libraries in Python.

4.3.2 Evaluation and Results

As seen in Figure 10 at the 4th epoch train loss and validation loss are equal to the same value. But the model was overfit because while train loss decreased validation loss started to increase after the 4th epoch. On the other hand, while train set accuracy increased validation accuracy decreased dramatically. This situation is another evidence of model overfitting. Therefore these results are not effective for classification.

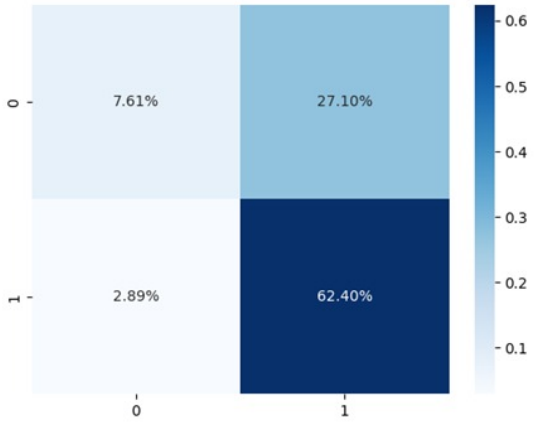

Figure 8: Confusion Matrix

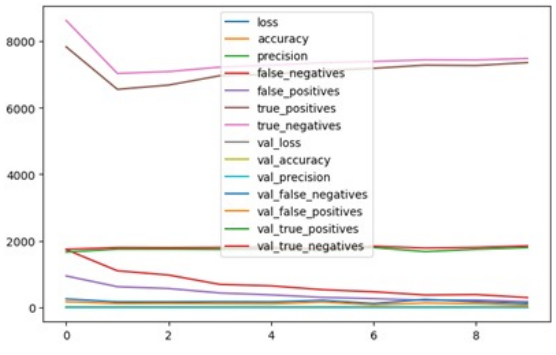

Figure 9: All Metrics Line Graphic

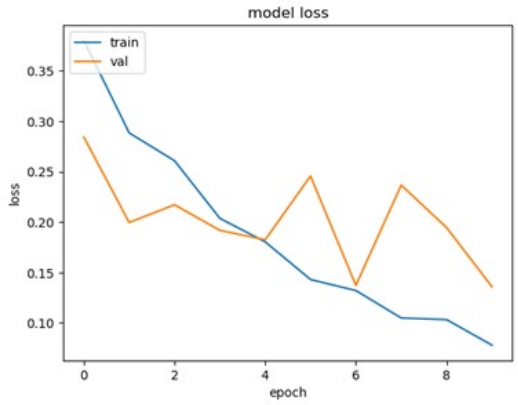

Figure 10: Loss Graphic

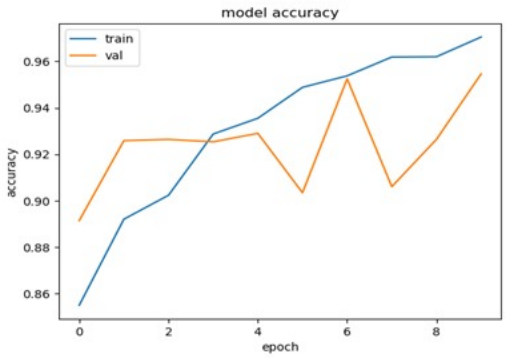

Figure 11: Accuracy Graphic

4.4 Implementation,Evaluation and Results of Classification with DC-GAN Generated Data

The first step is generating new data using DC-GAN.

DC-Gan architecture is complex, therefore the images were resized to 128,128,3 to achieve effective results in the cloud. Parameters were optimized after every experiment.

Firstly, general model parameters were defined.The parameters are; 'generator, discriminator, gan, X_hem, latent_dim, n_epochs=n_epoch, n_batch, batch_size'. The number of training epochs was assigned to 10000, 'batch size' was set to 128, 'latent dimensional vector' was set to 100, 'learning rate' was set to 0.0001, 'beta1' was set to 0.5, the optimizer was set to 'Adam'. and padding was assigned to 'same'.The dataset has 2 classes called 'HEM'(healthy) and 'ALL'(leukemia).For this reason 'binary crossentropy' was selected as the loss function.

Secondly, the generator parameters were tuned to generate data with the most similarity to the original dataset. The model has five layers. Output layer size is 128,128,3. 'Conv2D' was used for the out layer and 'LeakyReLU' was used in the inner layers. The activation function was 'tanh' and the alpha value was 0.2.

Thirdly, the discriminator model parameter is defined to check the similarity of generated and original data. The model has 5 layers.Every layer has 2 blocks. Conv2D and LeakyRelu were used in 2 blocks of each layer respectively. Padding was assigned to 'same' and the 'alpha' value was assigned to 0.2. Classifier parameters were assigned as 'dropout value' 0.4, the Activation function was set to 'sigmoid', the optimizer was set to 'Adam', 'learning rate was set to ' 0.0001', and beta1 was set to 0.5.The loss function was set to 'binary crossentropy'.

4.4.1 Implementation

There are 7272 all and 2300 hem images in the train set, and 1219 all and 648 hem images in the test set. For this reason, nearly 5000 hems(healthy) image data were generated. The code was implemented with Python. Tensorflow and Keras libraries were used. Kaggle GPU is used as a Cloud service to get fast results.5028 generated images were saved as .zip files to be added to the dataset. Finally, a classification model was applied to check the result of DC-Gan generated data to classification results.

4.4.2 Experiments

Many experiments were applied and learning rate 0.0001 was chosen for the best result and binary cross-entropy was chosen as the loss function. But DC-GAN generated different images and even used totally the same parameters. For this reason, 3 experiments were applied with the same parameters. Model trained with 1000 epoch and it nearly took 8 and half hours for every experiment.

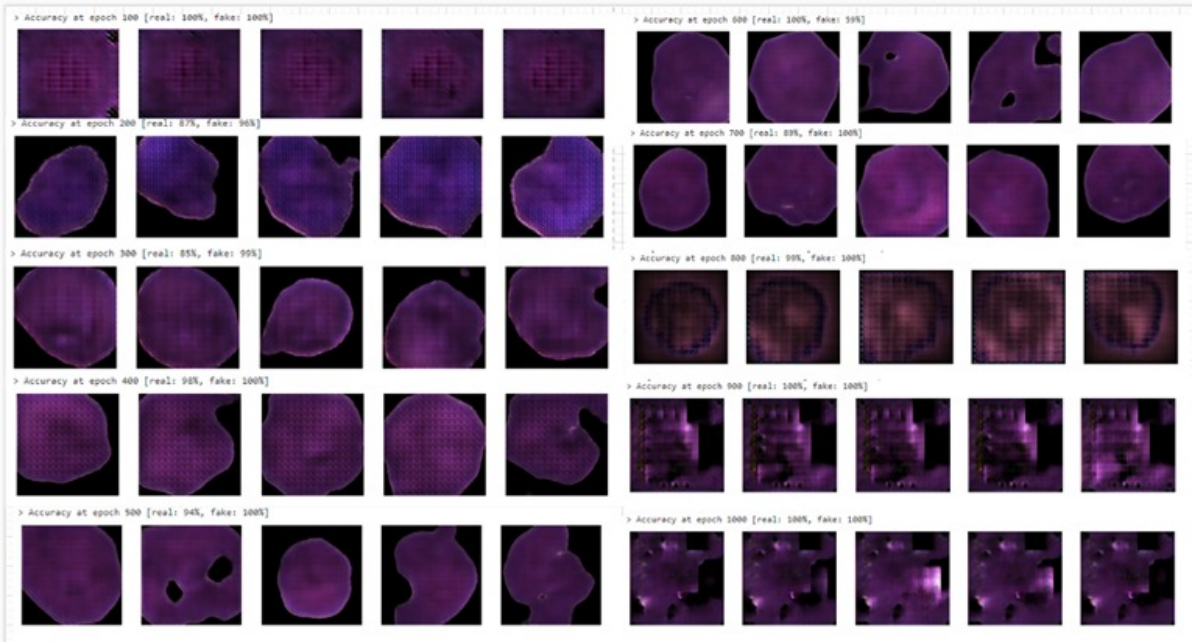

Figure 12: Experiment 1

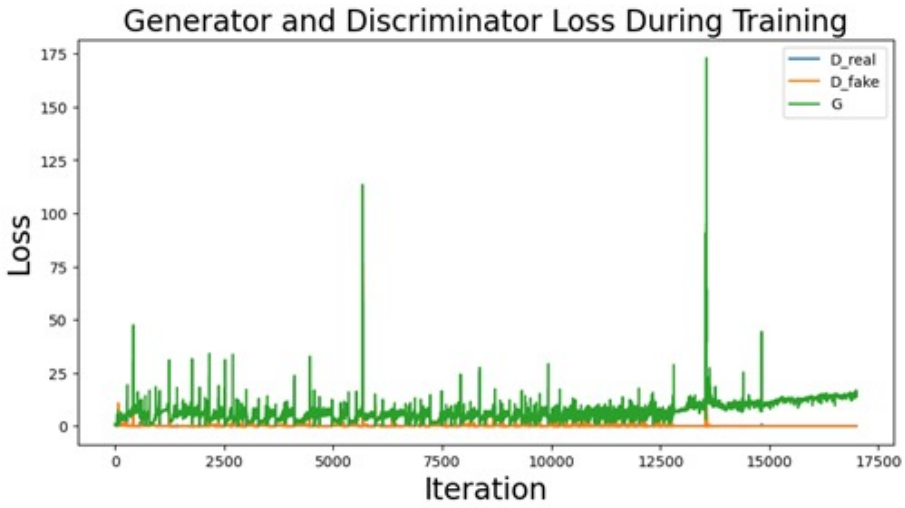

Figure 13: Generator and Discriminator Loss Graphic for Experiment 1

The Figure 12 illustrates the cell images generated by DC-GAN and Figure 13 depicts generator and discriminator loss graphic during the first experiment. As seen in Figure 13 at 700th epoch, and 13750th loop DC-GAN training achieved best performance.

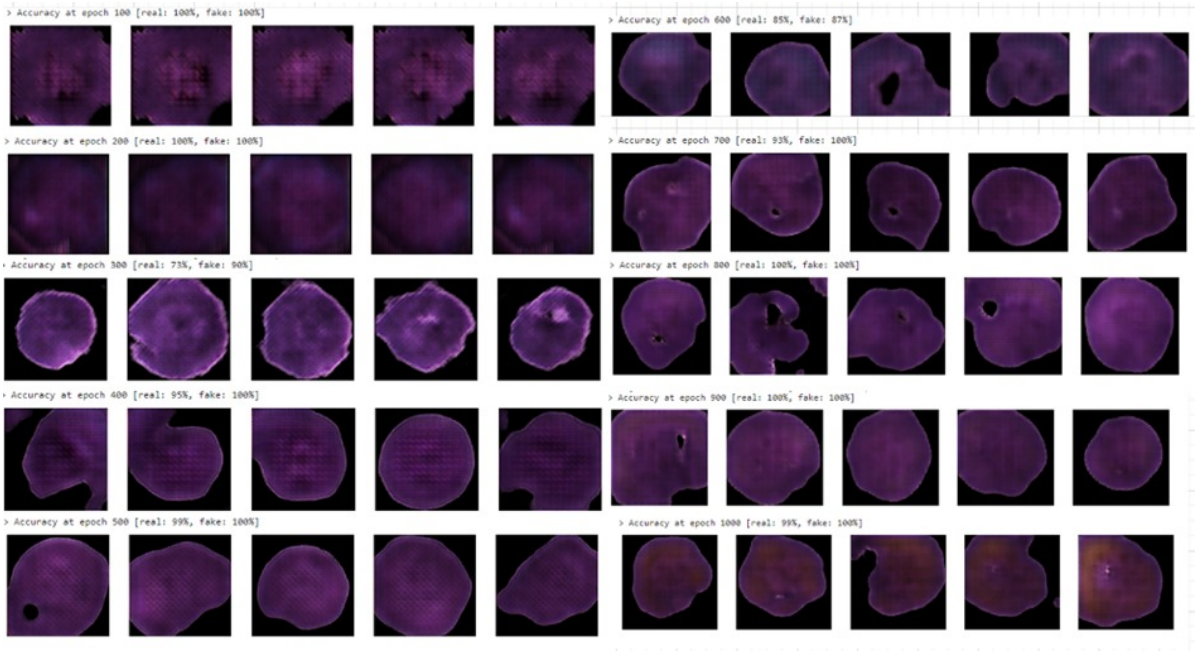

Figure 14: Experiment 2

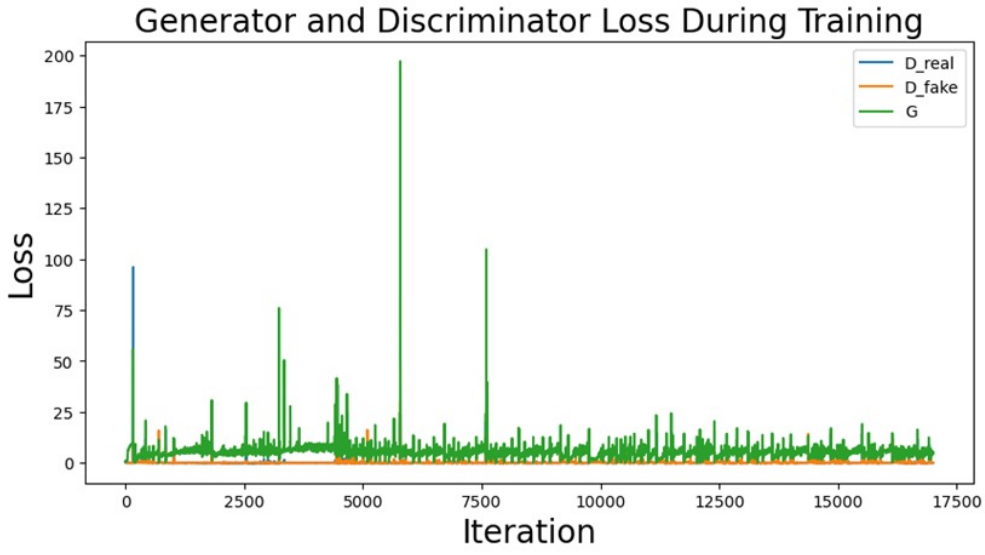

Figure 15: Generator and Discriminator Loss Graphic for Experiment 2

The Figure 14 illustrates the cell images generated by DC-GAN and Figure 15 depicts generator and discriminator loss graphic during the second experiment. As seen in Figure 15 DC-Gan trained better after the 600th epoch, between the 5000th and 7500th iteration.

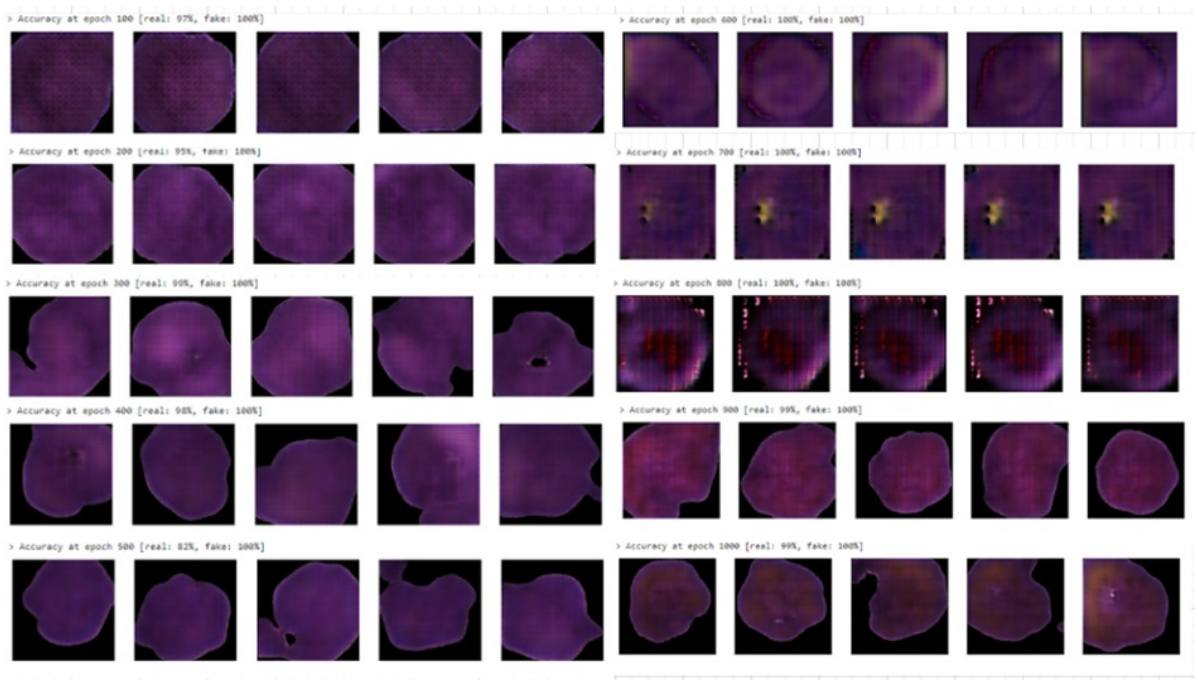

Figure 16: Experiment 3

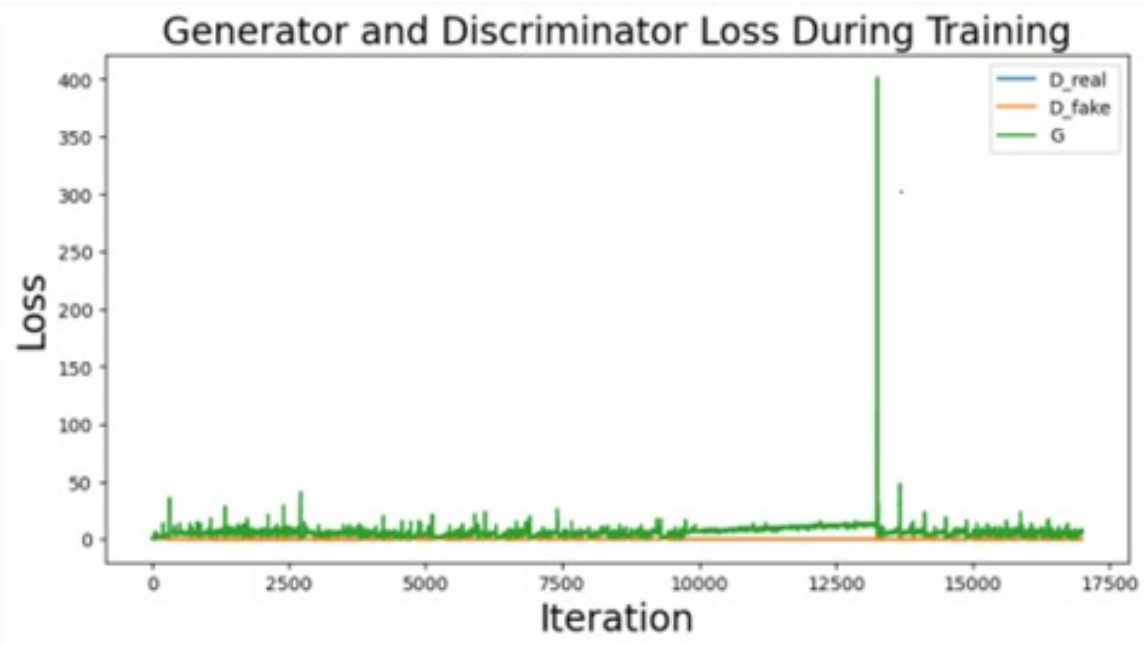

Figure 17: Generator and Discriminator Loss Graphic for Experiment 3

The Figure 16 illustrates the cell images generated by DC-GAN and Figure 17 depicts generator and discriminator loss graphic during the second experiment. As seen in Figure 17 DC-GAN achieved the best performance after the 12500th iteration.

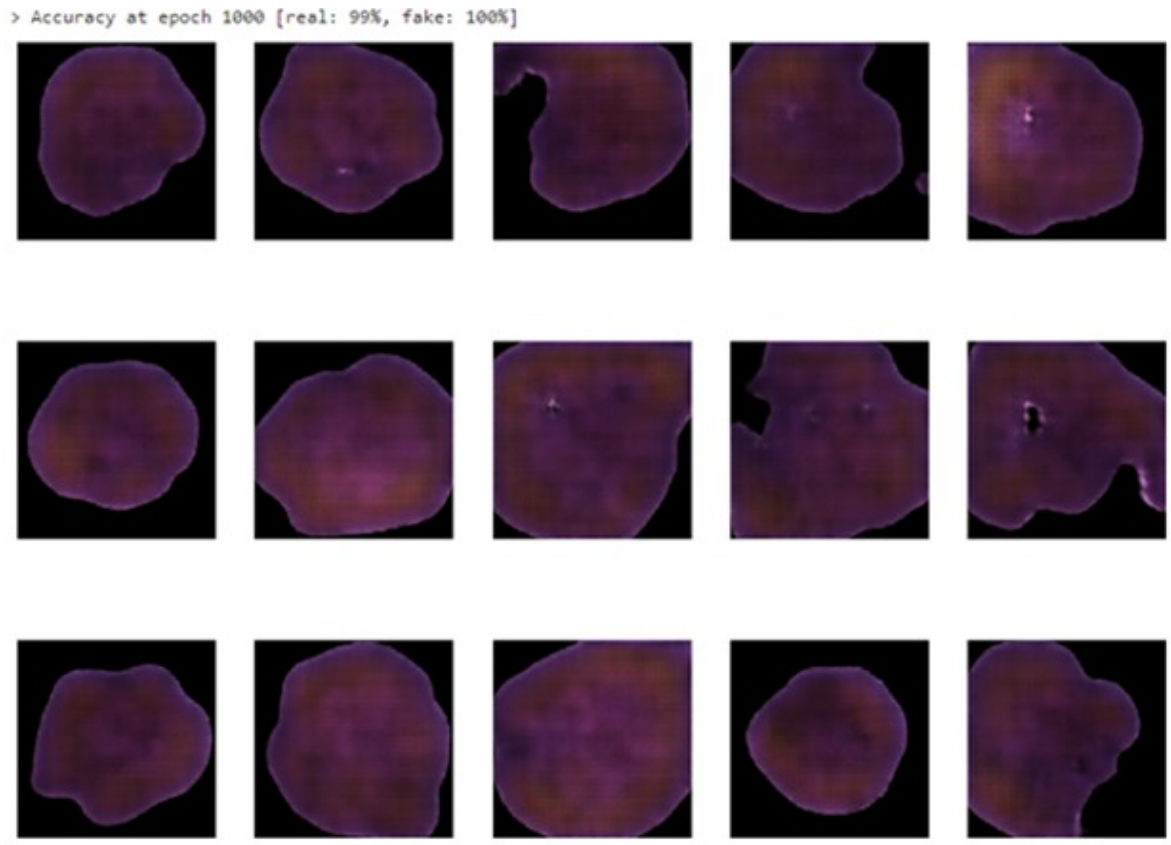

Figure 18: Results After 1000th Epoch

The Figure 18 illustrates the DC-GAN generated cell images after the 1000th epoch training. As can be seen from the image DC-GAN generated maximum similarity image data with the original dataset.

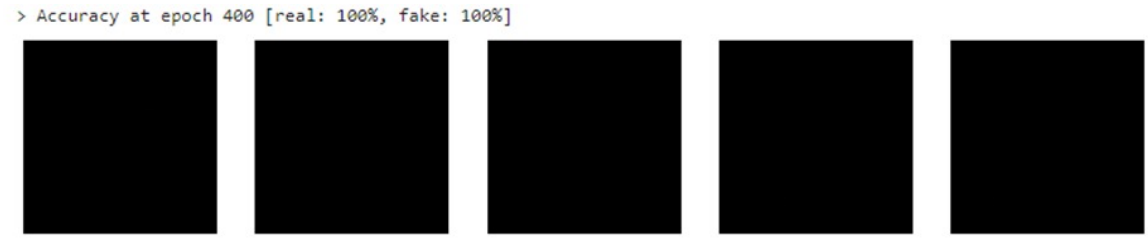

Figure 19: Additional experience

The Figure 19 illustrates an additional experiment applied for data generation with DC-GAN. The aim of this experiment was to check the effects of the loss function. Turing this experiment model was trained with categorical cross entropy was chosen as loss function, the same parameters as the other experiment were kept and 0.0001 learning rate was used.

As seen in Figure 19, categorical cross entropy is an unfunctional loss function for binary classification datasets and DC-GAN was unable to generate similar data.

After completing this step, the second step was applying classification with DC-GAN generated data by adding the generated images to the original dataset.

Firstly, 5028 HEM (healthy) images were generated with DC-GAN. These new images were added to the original dataset. Thus, the two classes were able to be balanced. Furthermore, the classification model was trained with ResNet-50.

After that, the first step of transfer learning parameters were defined. 'binary crossentropy' was chosen as the loss function. The optimizer was chosen as 'Adam' with '0.0001' learning rate. Epoch number 10, and batch size was set to 32. Following that, the second step of transfer learning parameters were defined. 'binary crossentropy' was chosen as the loss function. The optimizer was chosen as 'Adam' with '0.0001' learning rate. Epoch number 20, and batch size was set to 32.

The results were obtained as accuracy 0.99, precision 0.99, validation accuracy 0.91, false negative 34, false positive 7, true positives 8142, and true negatives 8169. Based on these values sensitivity was 0.99 and specificity was 0.99.

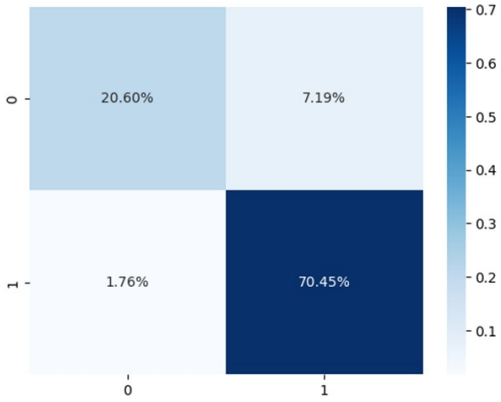

Figure 20: Confusion Matrix

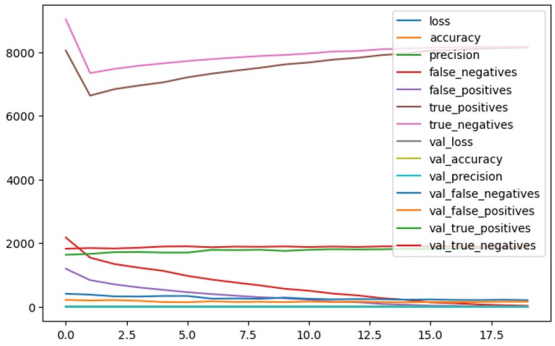

Figure 21: All Metrics Line Graphic

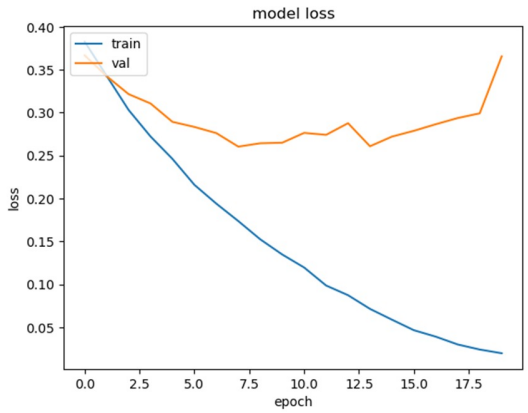

Figure 22: Loss Graphic

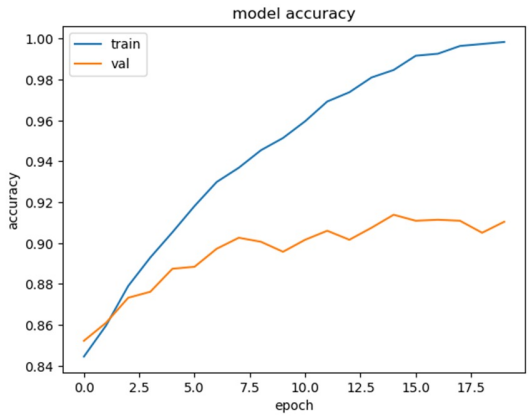

Figure 23: Accuracy Graphic

While the training loss decreases if the validation loss increases, the model is overfit. As can be seen in Figure 22 after the 20th epoch the model started to overfit. For this reason, 20 epochs is the optimal fit for the model. As seen Figure 22 the number of epochs increased while the value of train and validation loss decreased. The model showed more effective performance as the number of epochs increased.

As seen in Figure 23 when epoch number increased train set accuracy and validation set accuracy increased proportionally. The validation accuracy and the performance of the model are directly proportional. This graph confirms that the model makes more effective classification as the number of epochs increases.

As a result, the classification model which used generated images and dataset images together performed highest. This is an indication that the validation of the table objective Obj2(c) has been achieved and answered the research question fully. DC-Gan generated minor class data in the unbalanced dataset, to balance the dataset so the trained model was classified as high-performance and very efficient.

4.5 Implementation, Evaluation, and Results of Classification with ADASYN

ADASYN is a traditional oversampling technique based on the SMOTE algorithm. Hem and All classes in the training set were balanced with 11470 images separately. Parameters are defined as sampling_strategy equal to 'auto', which means only applied to the minority class.

The model trained with Resnet-50 has 5 layers. Every layer has 5 blocks. The last block of every layer has an 'activation' layer. Conv2D, batch normalization, and activation were used in inner layers based on Resnet-50 architecture. After that, the first step of transfer learning parameters were defined. 'binary crossentropy' was chosen as the loss function.The optimizer was chosen as 'Adam' with '0.000001' learning rate. The epoch number was set to 3, and the batch size was set to 32.

Following that, the second step of transfer learning parameters are defined. 'binary crossentropy' was chosen as the loss function. The optimizer was chosen as 'Adam' with '0.0001' learning rate. Epoch number 9, and batch size was set to 32.

4.5.1 Implementation

The code was written using Python. The tensorflow library was used for model implementation. Kaggle Notebook with TPU as an accelerator was used as a Cloud service to obtain results faster. The model was trained with Keras, Tensorflow, Numpy, and Pandas libraries and many sub-libraries in addition to imblearn.over_sampling to import ADASYN in Python.

4.5.2 Evaluation and Results

The results were obtained as accuracy 0.96, precision 0.96, validation accuracy 0.94, false negative 1377, false positive 124, true positives 10093, and true negatives 11346. Based on these values sensitivity was 0.89 and specificity was 0.84.

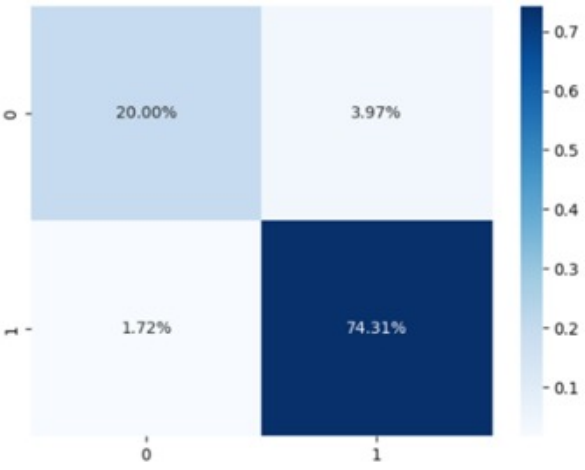

Figure 24: Confusion Matrix

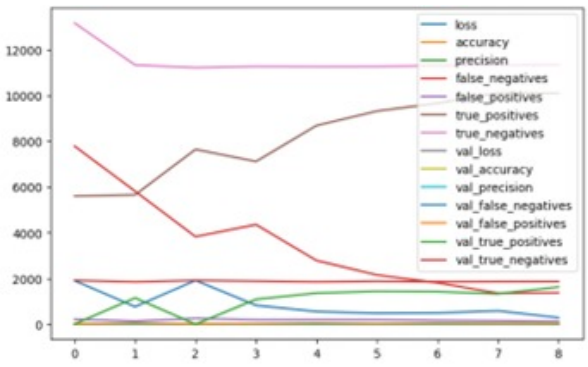

Figure 25: All Metrics Line Graphic

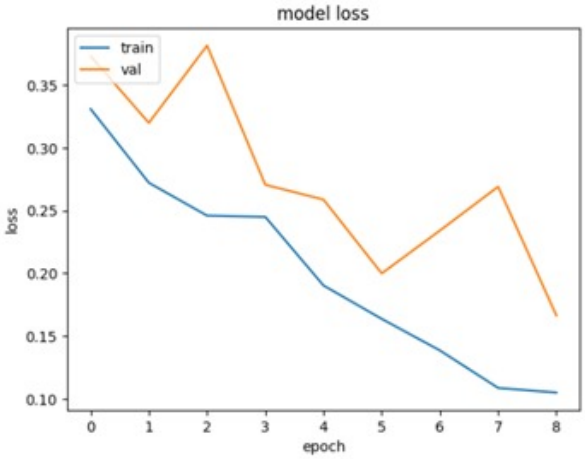

Figure 26: Loss Graphic

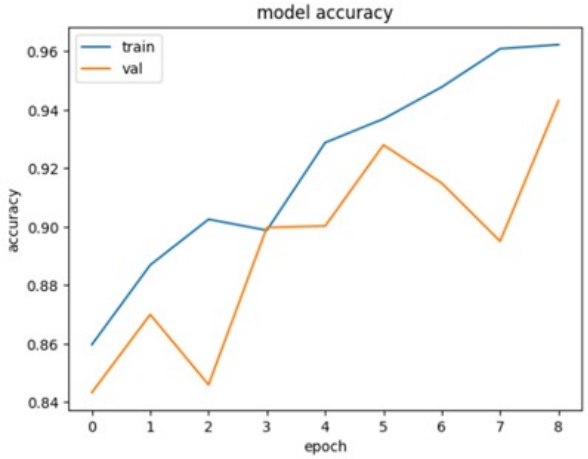

Figure 27: Accuracy Graphic

As seen in Figure 27 validation loss started to increase at the 5th epoch while training loss decreased consistently. This indicates that the model is overfitting. This is an indication that the validation of the table objective Obj2(d) has been achieved and answered the sub-research question fully.

4.6 Implementation, Evaluation, and Results of Classification with Weighted Random Sampling

Weighted random sampling is a method of oversampling that uses class weights. In this dataset, there are 7657 ALL images and 1915 HEM images in the training set. Therefore ALL class weight was assigned to 2, and HEM class weighted was assigned to 7 to balance each class.

The same model was used during the implementation of ADASYN, but different parameters were preferred to obtain effective results. The model trained on the Resnet-50 architecture has 5 layers. Every layer has 5 blocks. The last block of every layer has an 'activation' layer. Conv2D, batch normalization, and activation used to inner layers based on Resnet-50 architecture. After that, the first step of transfer learning parameters

were defined.'binary crossentropy' was chosen as the loss function. Optimizer was chosen as 'Adam' with '0.001' learning rate. The epoch number was set to 5, and the batch size was set to 32.

Following that, the second step of transfer learning parameters are defined. 'binary crossentropy' was chosen as the loss function. The optimizer was chosen as 'Adam' with '0.00001' learning rate. Epoch number 10, and batch size was set to 32.

4.6.1 Implementation

The code was written using Python. The tensorflow library was used for model implementation. Kaggle Notebook with TPU as an accelerator was used as a Cloud service to obtain results faster. The model was trained with Keras, Tensorflow, Numpy, and Pandas libraries and many sub-libraries in Python.

4.6.2 Evaluation and Results

The results were obtained as accuracy 0.99, precision 1.0, validation accuracy 0.95, false negative 1377, false positive 124, true positives 10093 and true negatives 11346. Based on these values sensitivity was 0.89 and specificity was 0.84.

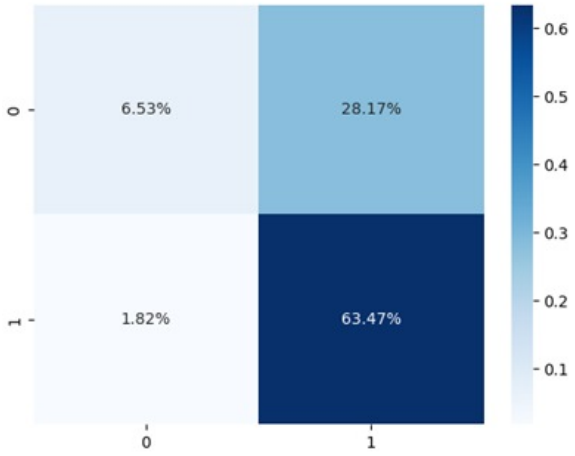

Figure 28: Confusion Matrix

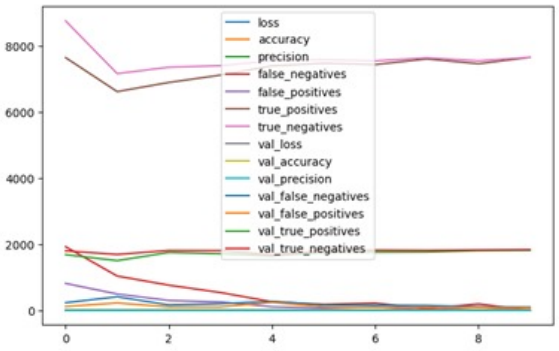

Figure 29: All Metrics Line Graphic

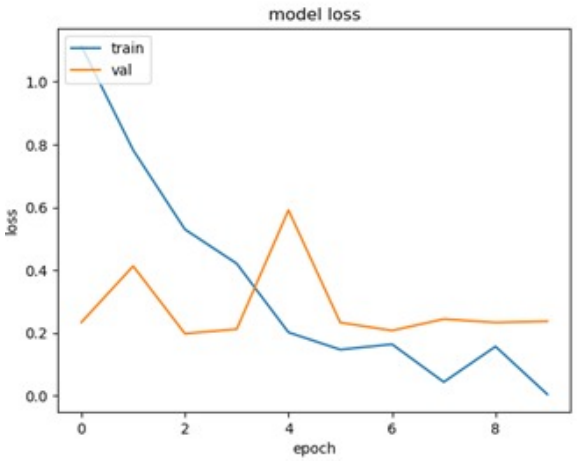

Figure 30: Loss Graphic

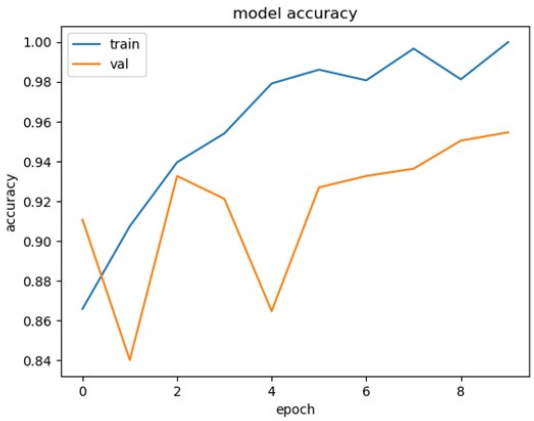

Figure 31: Accuracy Graphic

As seen in Figure 30 even at the 3rd epoch the model achieves optimal fit, while train loss decreased, validation loss increased after the 4th epoch. Therefore, the model overfits. At the 4th epoch, validation accuracy is too low even though train accuracy is high. This is another indication that the model is overfitting. This is an indication that the validation of the table objective Obj2(e) has been achieved and answered the sub-research question fully.

4.7 Implementation, Evaluation, and Results of Classification with Data Augmentation

Data augmentation is an oversampling technique that is generally preferred as a step of preprocessing. The image generator is only applied to the minority class. Image data generator parameters are rescaled by multiplying with '1./ 255', the rotation range was set to 10, the width shift range was set to 0.1, the height shift range was set to 0.1, the shear range was set to 0.2, the zoom range was set to 0.2, the horizontal flip was set to 'True', batch size was set to 'max_class_samples-min_class_samples' and shuffle was set to 'True'.

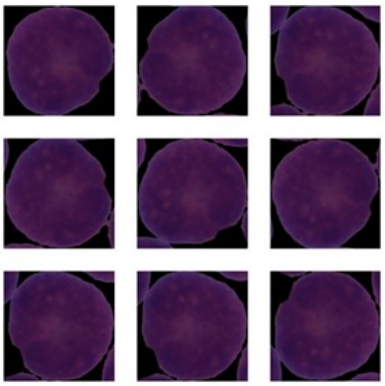

Figure 32: HEM images produced with data augmentation

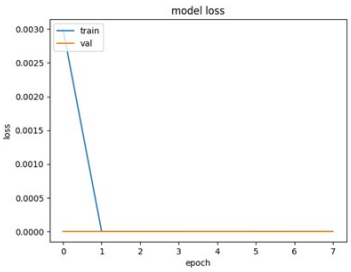

Figure 33: Loss Graphic

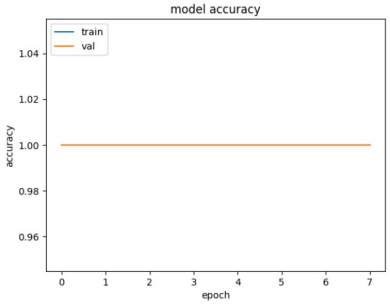

Figure 34: Accuracy Graphic

The same model was used during the implementation of ADASYN, but different parameters were preferred to obtain effective results. The model trained on the Resnet-50 architecture has 5 layers. Every layer has 5 blocks. The last block of every layer has an 'activation' layer. Conv2D, batch normalization, and activation are used to inner layers based on Resnet-50 architecture. After that, the first step of transfer learning parameters were defined. 'binary crossentropy' was chosen as the loss function. The optimizer was chosen as 'Adam' with '0.000001' learning rate. The epoch number was set to 5, and the batch size was set to 32.

Following that, the second step of transfer learning parameters are defined. 'binary crossentropy' was chosen as the loss function. The optimizer was chosen as 'Adam' with '0.0001' learning rate. Epoch number 8, and batch size was set to 32.

4.7.1 Implementation

The code was written using Python. The tensorflow library was used for model implementation. Kaggle Notebook with TPU as an accelerator was used as a Cloud service to obtain results faster. The model was trained with Keras, Tensorflow, Numpy, and Pandas libraries and many sub-libraries in Python. Additionally, to generate images with data augmentation ImageDataGenerator was imported from Keras Preprocessing library.

4.7.2 Evaluation and Results

The results were obtained as accuracy 100% and precision 1.0.

As seen in Figure 33 model is overfitting during training. The model was retrained using categorical cross entropy as a loss function and different learning rates and batch sizes but the result did not change. As can be seen, this method is the worst one because the model is too prone to overfit. This is an indication that the validation of the table objective Obj2(f) has been achieved and answered the sub-research question fully.

4.8 Comparison of Techniques Leukemia Classification

All techniques were implemented with Resnet-50 architecture. DC-GAN-based model, ADASYN, weighted random sampling, and data augmentation techniques are compared below that balanced dataset in leukemia classification.

The DC-GAN based model performed best with the according results: accuracy 0.99, precision 0.99, validation accuracy 0.91, sensitivity 0.99, and specificity 0.99. The model did not overfit while training and classified hem and all classes very effectively. On the other hand, 3 traditional techniques models overfit while training. Therefore there is no evidence that these techniques have an effect on balancing the dataset. But comparing all 3 techniques with each other ADASYN performed better than the other 2 techniques because it is the easiest one to recover from the overfitting problem among the 3 traditional techniques that have an advantage based on SMOTE. Weighted random sampling is the second most effective technique. Data augmentation was the worst technique because the model always overfit even though all parameters were tuned during the training period. On the other hand DC-GAN based model worked on Kaggle GPU and took 8 and half hours. Traditional technics worked on Kaggle TPU and took nearly 3 hours each. TPU works harder than GPU. Gan network was trained quickly and used ram more effectively at GPU. Traditional models were trained in less time, but they were more prone to over-

fitting. In terms of GPU and TPU performance, DC-GAN based model was much more complex but provided an advantage by using less ram.

4.9 Comparision of Developed Models with Existing Models

In this section DC-Gan based model selected was compared to the existing ALL classification models. Traditional models were not included in the comparison due to the overfitting problem on results.

The effects of the wrong treatment process for leukemia can be fatal. Therefore, accuracy is not enough alone. Sensitivity is the probability of a positive test result for a person with the disease, while specificity is the probability of a negative test result for a healthy individual. Therefore, these two metrics have the most critical value.

Year	Researcher	Sensitivity	Specificity	Accuracy
2013	Mohapatra S et al.	94.9%	95.0%	94.7%
2014	Singhal et al.	80.4%	95.1%	89.7%
2018	Shafique S et al.	96.8%	99.3%	96.1%
2017	Yu et al.	88.5%	77.8%	98.1%
2019	Pansombut T et al.	81.5%	Not mentioned	81.7%
2019	Hosseinzadeh S et al.	95.2%	98.6%	96.2%
2021	Mondal et al.	99.4%	96.7%	98.5%
2023	**Proposed model**	**99.5%**	**99.5%**	**99.5%**

Figure 35: Performance comparison of different models for ALL detection and Proposed model based on DC-GAN.

Figure 35 illustrates results from other researches and the results of DC-GAN based model in this research.

(Mondal et al.; 2021a) achieved the best classification performance and got 99.4% sensitivity, 96.7% specificity and 98.5% accuracy. Furthermore, (Shafique and Tehsin; 2018) got the highest specificity with 99.3%. (Mohapatra et al.; 2011) achieved efficient classification performance and got 94.9% sensitivity, 95% specificity and 94.7% accuracy.Additionally, (Singhal and Singh; 2014) achvieved good classification result 95.1% specificity and 89.7% accuracy however, (Pansombut et al.; 2019) did not achieved good result compared with other researchs. On the other hand, the proposed model performed the highest classification result with 99.5% sensitivity, 99.5% specificity, and 99.5% accuracy. As a result, the contribution of the data produced with DC-Gan to the classification results and model performance is quite high.

Year	Researcher	Optimizer	Learning Rate	Loss Fuction	Batch Size
2019	Xiao et al.	SGD	0.003	Cross-entropy	32
2019	Xie et al.	Adam	0.0001	Cross-entropy	64
2019	Prellberg et al.	Adam	1	Weighted cross-entropy	16
2019	Marzahl et al	Adam	0.01	Cross-entropy + L1	–
2019	Verma and Singh	Adam	0.001	Weighted cross-entropy	–
2019	Ding et al.	Adam	0.0001	Cross-entropy	48
2019	Kulhalli et al.	Adam	7.00E-05	Cross-entropy + JSD	–
2019	Liu et al.	RMSprop	0.001	Weighted cross-entropy	–
2019	Khan and Choo	Adam	0.0001	Cross-entropy	64
2021	Mondal et al.	Adamax	0.0002	Cross-entropy	8
2023	**Proposed model**	**Adam**	**0.000001**	**Binary Cross-entropy**	**32**

Figure 36: The comparison of ALL detection methods with different parameters performing with C-NMC 2019 dataset

Additionally, during the research, the effects of parameter tuning on the model performance was observed. Figure 36 shows parameters for existing research for ALL classification. Adam is the most commonly chosen optimizer type. (Xie et al.; 2019), (Prellberg and Kramer; 2019)., (Marzahl et al.; 2019),(Verma and Singh; 2019) , (Ding et al.; 2019), (Kulhalli et al.; 2019), (Khan and Choo; 2019) used Adam as optimizers. In addition, (Xiao et al.; 2019) preferred SGD, (Liu and Long; 2019) preferred RMSPROP, and (Mondal et al.; 2021a) preferred Adamax as the optimizer. On the other hand cross entropy and its subtypes are commonly preferred as loss function. Batch size and learning rate are unique depending on the characteristics of the research. The proposed model is applied to choose the loss function as 'binary cross entropy', for optimizer 'Adam' used with a very low learning rate as '0.000001', and batch size 32.

4.10 Conclusion

Based on the techniques applied and the results produced, the research question and the sub-research question were fully answered (in section 1.2). Also, all research objectives (in section 1.3) were discussed.

Among the selected techniques, DC-Gan performed best. In addition, using DC-GAN can accelerate data generation. The model did not overfit and it made contributions to the classification results. Classification results improved with the balanced dataset. DC-GAN can be used in the medical area to solve imbalanced dataset problems.

The classification model was overfitted while using 3 traditional techniques. However, ADASYN was the technique that showed the least susceptibility to overfitting due to its SMOTE-based algorithm. On the other hand, weighted random sampling and data augmentation techniques are insufficient for leukemia classification because the model is very susceptible to overfitting even if parameter optimization is applied. The epoch numbers used in the selected techniques are different from each other. There are two main reasons for choosing different epoch numbers. The first reason is to prevent overfitting of the model for each selected technique, and the second reason is the ram capacity of the selected model due to hardware limitations.

The findings obtained in this research are; that generating new data in order to balance the dataset in leukemia classification makes a lot of contributions to the classification model, but producing from existing data using traditional techniques has a tendency to overfit the classification model. Fast and high-quality images were produced

with DC-GAN and the classification model was developed. On the other hand, traditional techniques produced low-quality images because of resampling the data could not contribute to the classification model and not solve the overfitting problem in the model.

5 Conclusion and Future Work

The proposed DC-Gan based model performed with classification results 99.5% sensitivity, 99.5% specificity, and 99.5% accuracy. The model performed on Kaggle GPU, took 8 hours 30 minutes to generate images. On the other hand, the model was overfit to produce data with traditional techniques. The images produced were low quality because these techniques were based on oversampling. The model performed on Kaggle TPU, took 3 hours to produce data separately.

High-quality images were generated with DC-GAN and the classification model was developed. On the other hand, traditional techniques produced low-quality images and did not solve the overfitting problem on the classification model. Furthermore, Kaggle GPU used RAM less than Kaggle TPU. DC-Gan model is more effective than traditional models because it used GPU for RAM. The datasets that are available for research are insufficient and imbalanced due to patient data privacy. With fast, high-quality data generation, Gan Network has eliminated the complex and expensive traditional leukemia diagnosis which has a tendency to have expert error and issues of patient privacy while saving more lives and increasing the quality of patients.

For future work, with the usage of higher GPU, RAM, and developed parameter optimization Gan Network algorithm can be designed more optimally and effectively used for another medical field. Additionally, it can reduce the cost of treatment by adapting it to rural areas.

6 Acknowledgement

I would like to thank Dr. Catherine Mulwa for her supervision and guidance during this research.

References

Ahmad, B., Sun, J., You, Q., Palade, V. and Mao, Z. (2022). Brain tumor classification using a combination of variational autoencoders and generative adversarial networks, *Biomedicines* **10**(2): 223.

Ahmed, N., Yigit, A., Isik, Z. and Alpkocak, A. (2019). Identification of leukemia subtypes from microscopic images using convolutional neural network, *Diagnostics* **9**(3): 104.

Alharbi, A. H., Aravinda, C., Lin, M., Venugopala, P., Reddicherla, P. and Shah, M. A. (2022). Segmentation and classification of white blood cells using the unet, *Contrast Media & Molecular Imaging* **2022**.

Ali, H., Shah, Z. et al. (2022). Combating covid-19 using generative adversarial networks and artificial intelligence for medical images: Scoping review, *JMIR Medical Informatics* **10**(6): e37365.

Amin, M. M., Kermani, S., Talebi, A. and Oghli, M. G. (2015). Recognition of acute lymphoblastic leukemia cells in microscopic images using k-means clustering and support vector machine classifier, *Journal of medical signals and sensors* **5**(1): 49.

Chawla, N. V., Bowyer, K. W., Hall, L. O. and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique, *Journal of artificial intelligence research* **16**: 321–357.

Clark, K., Vendt, B., Smith, K., Freymann, J., Kirby, J., Koppel, P., Moore, S., Phillips, S., Maffitt, D., Pringle, M. et al. (2013). The cancer imaging archive (tcia): maintaining and operating a public information repository, *Journal of digital imaging* **26**: 1045–1057.

Cook, T. D., Campbell, D. T. and Shadish, W. (2002). *Experimental and quasi-experimental designs for generalized causal inference*, Vol. 1195, Houghton Mifflin Boston, MA.

Das, P. K., Diya, V., Meher, S., Panda, R. and Abraham, A. (2022). A systematic review on recent advancements in deep and machine learning based detection and classification of acute lymphoblastic leukemia, *IEEE access* .

De Sant'Anna, Y. F. D., De Oliveira, J. E. M. and Dantas, D. O. (2021). Lightweight classification of normal versus leukemic cells using feature extraction, *2021 IEEE Symposium on Computers and Communications (ISCC)*, IEEE, pp. 1–7.

de Sant'Anna, Y. F. D., de Oliveira, J. E. M. and Dantas, D. O. (2022). Interpretable lightweight ensemble classification of normal versus leukemic cells, *Computers* **11**(8): 125.

Depto, D. S., Rizvee, M. M., Rahman, A., Zunair, H., Rahman, M. S. and Mahdy, M. (2023). Quantifying imbalanced classification methods for leukemia detection, *Computers in Biology and Medicine* **152**: 106372.

Ding, Y., Yang, Y. and Cui, Y. (2019). Deep learning for classifying of white blood cancer, *in* A. Gupta and R. Gupta (eds), *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging*, Springer Singapore, Singapore, pp. 33–41.

Duggal, R., Gupta, A., Gupta, R. and Mallick, P. (2017). Sd-layer: stain deconvolutional layer for cnns in medical microscopic imaging, *Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part III 20*, Springer, pp. 435–443.

Ghaderzadeh, M., Hosseini, A., Asadi, F., Abolghasemi, H., Bashash, D. and Roshanpoor, A. (2022). Automated detection model in classification of b-lymphoblast cells from normal b-lymphoid precursors in blood smear microscopic images based on the majority voting technique, *Scientific Programming* **2022**: 1–8.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2020). Generative adversarial networks, *Communications of the ACM* **63**(11): 139–144.

Gosain, A. and Sardana, S. (2017). Handling class imbalance problem using oversampling techniques: A review, *2017 international conference on advances in computing, communications and informatics (ICACCI)*, IEEE, pp. 79–85.

Gupta, A., Gehlot, S., Goswami, S., Motwani, S., Gupta, R., Faura, Á. G., Štepec, D., Martinčič, T., Azad, R., Merhof, D. et al. (2023). Segpc-2021: A challenge & dataset on segmentation of multiple myeloma plasma cells from microscopic images, *Medical Image Analysis* **83**: 102677.

Gupta, R., Gehlot, S. and Gupta, A. (2022). C-nmc: B-lineage acute lymphoblastic leukaemia: A blood cancer dataset, *Medical Engineering & Physics* **103**: 103793.

He, K., Zhang, X., Ren, S. and Sun, J. (2016). Deep residual learning for image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Honnalgere, A. and Nayak, G. (2019). Classification of normal versus malignant cells in b-all white blood cancer microscopic images, *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging: Select Proceedings*, Springer, pp. 1–12.

Jamakayala, J. and Gorthi, R. K. S. (2021). Feature fusion ensemble architecture with active learning for microscopic blood smear analysis, *2021 IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 3767–3771.

Kaur, G., Kaur, V., Sharma, Y., Bansal, V. et al. (2022). Analyzing various machine learning algorithms with smote and adasyn for image classification having imbalanced data, *2022 IEEE International Conference on Current Development in Engineering and Technology (CCET)*, IEEE, pp. 1–7.

Khan, M. A. and Choo, J. (2019). Classification of cancer microscopic images via convolutional neural networks, *in* A. Gupta and R. Gupta (eds), *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging*, Springer Singapore, Singapore, pp. 141–147.

Kulhalli, R., Savadikar, C. and Garware, B. (2019). Toward automated classification of b-acute lymphoblastic leukemia, *in* A. Gupta and R. Gupta (eds), *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging*, Springer Singapore, Singapore, pp. 63–72.

Kurniawati, Y. E., Permanasari, A. E. and Fauziati, S. (2018). Adaptive synthetic-nominal (adasyn-n) and adaptive synthetic-knn (adasyn-knn) for multiclass imbalance learning on laboratory test data, *2018 4th International Conference on Science and Technology (ICST)*, IEEE, pp. 1–6.

Li, W., Chen, J., Cao, J., Ma, C., Wang, J., Cui, X. and Chen, P. (2022). Eid-gan: Generative adversarial nets for extremely imbalanced data augmentation, *IEEE Transactions on Industrial Informatics* **19**(3): 3208–3218.

Liu, R., Dai, W., Wu, T., Wang, M., Wan, S. and Liu, J. (2022). Aimic: Deep learning for microscopic image classification, *Computer Methods and Programs in Biomedicine* **226**: 107162.

Liu, Y. and Long, F. (2019). Acute lymphoblastic leukemia cells image analysis with deep bagging ensemble learning, *in* A. Gupta and R. Gupta (eds), *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging*, Springer Singapore, Singapore, pp. 113–121.

Marzahl, C., Aubreville, M., Voigt, J. and Maier, A. (2019). Classification of leukemic b-lymphoblast cells from blood smear microscopic images with an attention-based deep learning method and advanced augmentation techniques, *in* A. Gupta and R. Gupta (eds), *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging*, Springer Singapore, Singapore, pp. 13–22.

Mittal, A., Dhalla, S., Gupta, S. and Gupta, A. (2022). Automated analysis of blood smear images for leukemia detection: a comprehensive review, *ACM Computing Surveys (CSUR)* **54**(11s): 1–37.

Mohanty, F., Rup, S., Dash, B., Majhi, B. and Swamy, M. (2019). Mammogram classification using contourlet features with forest optimization-based feature selection approach, *Multimedia Tools and Applications* **78**: 12805–12834.

Mohapatra, S., Samanta, S. S., Patra, D. and Satpathi, S. (2011). Fuzzy based blood image segmentation for automated leukemia detection, *2011 International Conference on Devices and Communications (ICDeCom)*, IEEE, pp. 1–5.

Mondal, C., Hasan, M. K., Ahmad, M., Awal, M. A., Jawad, M. T., Dutta, A., Islam, M. R. and Moni, M. A. (2021a). Ensemble of convolutional neural networks to diagnose acute lymphoblastic leukemia from microscopic images, *Informatics in Medicine Unlocked* **27**: 100794.

Mondal, C., Hasan, M. K., Ahmad, M., Awal, M., Jawad, M. T., Dutta, A., Islam, M. and Moni, M. A. (2021b). Ensemble of convolutional neural networks to diagnose acute lymphoblastic leukemia from microscopic images, *Informatics in Medicine Unlocked* **27**: 100794.

Negm, A. S., Hassan, O. A. and Kandil, A. H. (2018). A decision support system for acute leukaemia classification based on digital microscopic images, *Alexandria engineering journal* **57**(4): 2319–2332.

Pansombut, T., Wikaisuksakul, S., Khongkraphan, K., Phon-On, A. et al. (2019). Convolutional neural networks for recognition of lymphoblast cell images, *Computational Intelligence and Neuroscience* **2019**.

Prellberg, J. and Kramer, O. (2019). Acute lymphoblastic leukemia classification from microscopic images using convolutional neural networks, *in* A. Gupta and R. Gupta (eds), *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging*, Springer Singapore, Singapore, pp. 53–61.

Radford, A., Metz, L. and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks, *arXiv preprint arXiv:1511.06434* .

Ramaneswaran, S., Srinivasan, K., Vincent, P. D. R. and Chang, C.-Y. (2021). Hybrid inception v3 xgboost model for acute lymphoblastic leukemia classification, *Computational and Mathematical Methods in Medicine* **2021**: 1–10.

Saleem, S., Amin, J., Sharif, M., Anjum, M. A., Iqbal, M. and Wang, S.-H. (2021). A deep network designed for segmentation and classification of leukemia using fusion of the transfer learning models, *Complex & Intelligent Systems* pp. 1–16.

Shafique, S. and Tehsin, S. (2018). Acute lymphoblastic leukemia detection and classification of its subtypes using pretrained deep convolutional neural networks, *Technology in cancer research & treatment* **17**: 1533033818802789.

Singhal, V. and Singh, P. (2014). Local binary pattern for automatic detection of acute lymphoblastic leukemia, *2014 Twentieth National Conference on Communications (NCC)*, IEEE, pp. 1–5.

Tong, Q., Lu, F., Feng, Z., Wan, Q., An, G., Cao, J. and Guo, T. (2022). A novel method for fault diagnosis of bearings with small and imbalanced data based on generative adversarial networks, *Applied Sciences* **12**(14): 7346.

Verma, E. and Singh, V. (2019). Isbi challenge 2019: Convolution neural networks for b-all cell classification, *in* A. Gupta and R. Gupta (eds), *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging*, Springer Singapore, Singapore, pp. 131–139.

Wang, D., Khosla, A., Gargeya, R., Irshad, H. and Beck, A. H. (n.d.). Deep learning for identifying metastatic breast cancer. arxiv 2016, *arXiv preprint arXiv:1606.05718* .

Xiao, F., Kuang, R., Ou, Z. and Xiong, B. (2019). Deepmen: Multi-model ensemble network for b-lymphoblast cell classification, *in* A. Gupta and R. Gupta (eds), *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging*, Springer Singapore, Singapore, pp. 83–93.

Xie, X., Li, Y., Zhang, M., Wu, Y. and Shen, L. (2019). Multi-streams and multi-features for cell classification, *ISBI 2019 C-NMC Challenge: Classification in Cancer Cell Imaging: Select Proceedings*, Springer, pp. 95–102.

Zulkifley, M. A., Abdani, S. R. and Zulkifley, N. H. (2020). Covid-19 screening using a lightweight convolutional neural network with generative adversarial network data augmentation, *Symmetry* **12**(9): 1530.