Enhancing Blood Cancer Recognition through Deep Learning Techniques: A Comparative Study of Deep Learning Models for Blood Cancer

[1]Jain Joseph, https://orcid.org/0009-0004-3007-3021
[2]Sherimon P.C., https://orcid.org/0000-0001-6912-1407
[3]Vinu Sherimon, https://orcid.org/0000-0003-4923-2841
[1]*Lincoln University College Malaysia, Marian College Kuttikanam (Autonomous), Kerala, India*
[2]*Faculty of Computer Studies, Arab Open University, Muscat, Sultanate of Oman*
[3]*College of Computing and Information Sciences, University of Technology and Applied Sciences, Muscat, Sultanate of Oman*

Corresponding Author: Jain Joseph (email: joseph.jain214@gmail.com)

***Abstract*— For efficient treatment and positive patient outcomes, early and precise diagnosis of blood malignancies is crucial in the field of medical diagnostics. PBSs, or peripheral blood smears, are important diagnostic tools for several blood-related conditions. However, manually reviewing these photographs can be time-consuming and prone to mistakes. So, this study proposes three deep learning models such as Convolutional Neural Network (CNN), the Modified U-Net, and VGG16 for recognition of blood cancer. The dataset consists of PBS pictures from patients with acute lymphoblastic leukemia (ALL) is used to train the models. Furthermore, metrics like accuracy, recall, and F1-score are used to assess how well the models perform in classifying various blood cancer subtypes. The CNN model achieved an accuracy of 42%, while the Modified U-Net model demonstrated an accuracy of 52%. Remarkably, the VGG16 model outshines its counterparts with an impressive accuracy rate of 99%, underscoring the potential of deep learning for intricate medical image analysis.**

***Keywords*- CNN, Modified U-Net, VGG16, Artificial intelligence**

I. INTRODUCTION

Traditional methods for diagnosing blood malignancies, which include leukemia, lymphoma, and myeloma, have relied on invasive procedures and complex laboratory investigations. There are typically substantial difficulties in relationships with time, money, and accuracy, even when using state-of-the-art methods. The development of AI has the potential to change the diagnostic environment for the better. "Convolutional Neural Networks (CNN)", a "***Modified U-Net architecture with Exponential Linear Unit Activation Framework (MU-EAF)***", and the "VGG-16 model" are only some of the ML models that are investigated in this paper and their potential use in the context of blood cancer diagnostics. To better diagnose diseases, the research analyses how well these AI models can learn from patterns in medical data, specifically in tiny blood pictures. This study aims to help advance blood cancer diagnoses and, by extension, improve patient outcomes via the use of AI's predictive abilities. Researchers want to choose the best model by making a comprehensive comparison of them and discuss how this novel method could change the face of diagnostics in the future.

II. MATERIALS AND METHODS

This research uses CNN, a "***Modified U-Net architecture with ELU Activation Framework (MU-EAF)***", and the VGG-16 model as its machine learning models for the detection of blood cancer. These models are geared to analyzing and categorizing pictures of blood cells into four distinct types of cancer: "Benign, Early, Pre, and Pro". To diagnose and categorize different forms of blood cancer using microscopic pictures for early identification and treatment is at the heart of the issue statement. The algorithm takes as input a set of annotated pictures of blood cells and outputs a classification of the cancer type that is most likely to be present in each picture. "***Accuracy, precision, recall, and F1-score***" are only few of the measures used to assess the models' performance after they have been trained on a collection of blood cell pictures with labels [4]. The goal is to create a reliable and precise method that can aid doctors in the diagnosis of blood cancer, leading to better patient outcomes and treatment options.

A. Dataset

"Acute lymphoblastic leukemia (ALL)" diagnostic "peripheral blood smear (PBS)" pictures were employed for this research. It takes a lot of time and effort to diagnose ALL, despite the fact that it is a common kind of cancer. There are 3,256 PBS pictures from 89 individuals with a possible case of ALL in the collection. Expert laboratory personnel at "Taleqani Hospital in Tehran, Iran", processed and stained these photographs [17]. There are two types of data in the set: benign and malignant. Hematogones belong to the benign class, whereas "***Early Pre-B, Pre-B, and Pro-B ALL***" make up the malignant class. The JPG files were taken from

photographs shot with a Zeiss camera at 100x magnification. Using flow cytometry, an expert performed the final identification of cell kinds and subtypes. After using a color threshold to segment in the HSV color space, we also gave the resulting segmented photos.

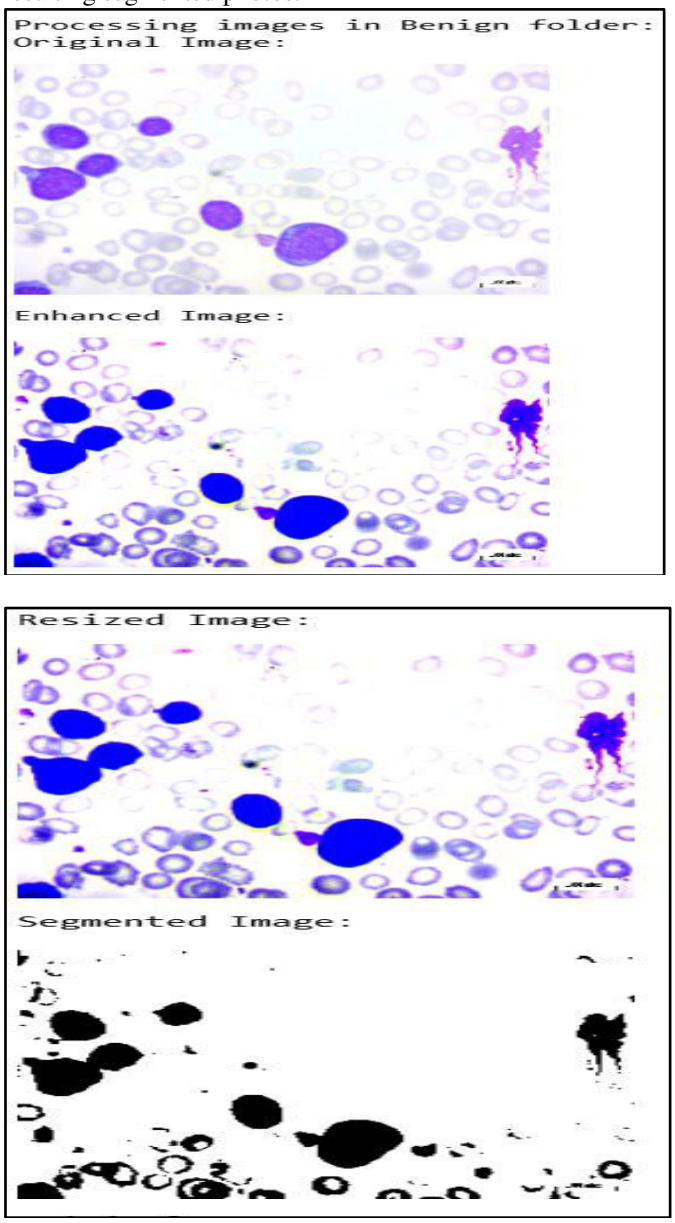

Fig. 1. Image segmentation process

The photos in the input collection are segmented using this code. As it loops over the directory tree, it applies Otsu's thresholding approach for segmentation, adjusts contrast, resizes, normalizes pixel values, converts to grayscale, and so on for each picture it encounters. The first two photos in each subfolder are processed by the code.

At each stage, the transformed picture is shown alongside the original, improved, scaled, and segmented versions. To facilitate further analysis and categorization of blood cancer cells, the images are segmented to isolate the leukemia stains inside the cells from the background.

B. Deep learning

The deep learning methods are used in this research to conduct classification and segmentation of malignant cells. To begin, investigators obtained and cleaned up a collection of snaps of blood cells. After that, researchers used a "***Convolutional Neural Network (CNN)***", a modified version of the "***U-Net using the ELU Activation Framework (MU-EAF)***", and the "VGG-16 model" for "Transfer Learning". The pre-processed dataset was used to train these models using the appropriate loss functions and optimization procedures. "***Accuracy, precision, recall, and F1-score***" were only few of the actions used to measure the trained models [10]. To make the models even more reliable and applicable, that used data augmentation and cross-validation. The strategy attempted to correctly categorize and separate cancer cells, paving the way for an efficient AI-assisted diagnosis system for blood cancer.

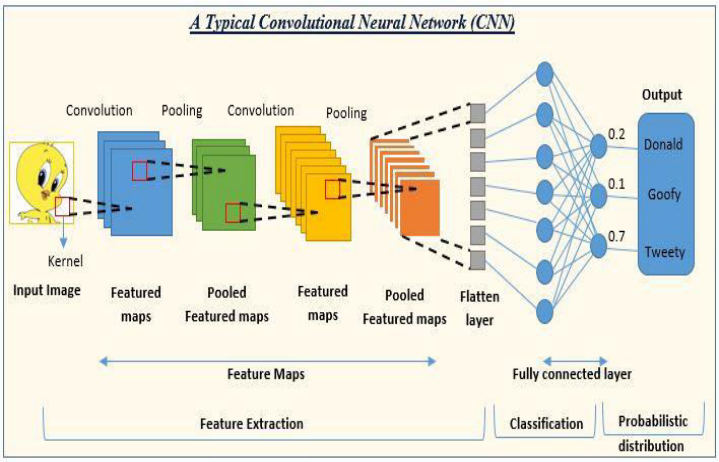

Fig. 2. CNN model architecture

Multiple layers of the CNN model architecture were utilized to extract characteristics from input photos for malignant cell detection and segmentation in this research. To begin, the model uses a convolutional layer to filter the input picture and identify significant features. Next, pooling layers are added, which bring the spatial dimensions down while keeping the essentials intact. Following this, fully linked layers aggregate the retrieved characteristics and carry out the final classification. Dropout layers are used to reduce the likelihood of overfitting. "***Rectified Linear Unit (ReLU) activation function***" is used to add non-linearity and boost the model's accuracy [14]. At the very end, a "***SoftMax activation function***" is used to generate a probability distribution over the various classes. The successive arrangement of these layers allows the CNN model to learn and recognize complicated patterns within the input pictures, allowing for precise categorization and segmentation of malignant cells.

Segmenting cancer cells in an image is only one use for the U-Net model architecture, a convolutional neural network developed for such purposes. A skip-connected encoder-decoder architecture is its primary component. In order to capture context, the input picture is down sampled, and features are extracted using many layers in the encoder component. The decoder up samples the feature maps to gradually put the fragmented picture back together. The layers of the decoder and encoder are connected by skip links so that the model may preserve and utilize fine-grained characteristics

during up sampling [15]. The U-Net model's ability to effectively collect both local and global information via its design makes it particularly well-suited for the accurate and exact segmentation of malignant cells in medical pictures.

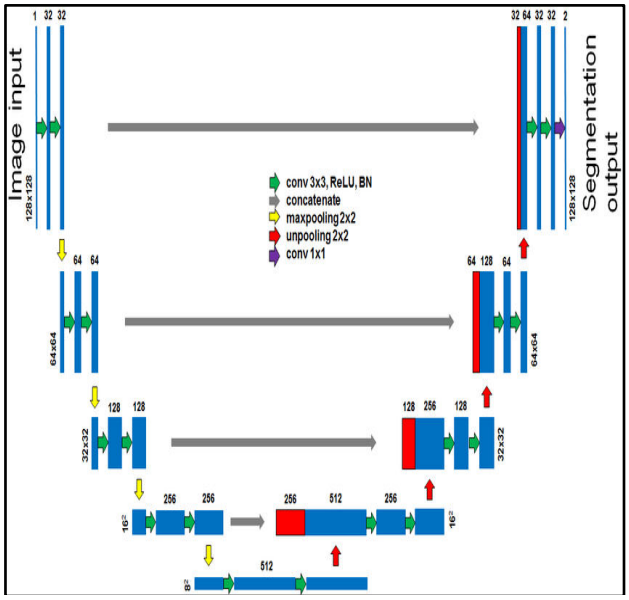

Fig. 3. U-Net model architecture

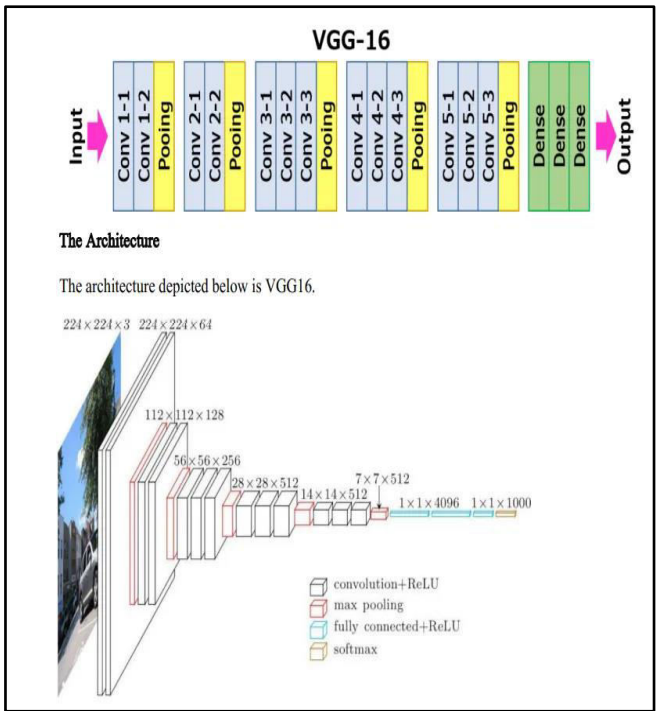

Fig. 4. VGG 16 model architecture

Built specifically for image classification, the VGG16 model architecture is a deep convolutional neural network. Multiple 3x3 convolutional filters precede a max-pooling layer in this network of 16 convolutional layers [16]. The network continuously increases the number of channels while decreasing the spatial dimensions. The last three layers are completely integrated and used for categorization. The consistent design and narrow receptive fields of the VGG16 model are what make it so effective at capturing both low-level and high-level elements in a picture. VGG16 is a popular basis model for transfer learning due to its remarkable performance on several picture classification benchmarks and its relative ease of implementation.

Table 1: Activation function comparison

Models	Activation Function	Description
CNN	ReLU	The segmentation and categorization of images is done using the "***Convolutional Neural Network (CNN)***" model. It starts with a few convolutional layers and then activates the model with the "***Rectified Linear Unit (ReLU)***" function, adding nonlinearity. The CNN classifies and segments blood cancer cells by removing characteristics from the input pictures.
MU-EAF (Modified U-Net)	ELU	The "***Modified U-Net model***" incorporates the ELU (Exponential Linear Unit) activation function. It consists of encoder and decoder pathways connected through skip connections. The ELU function introduces non-linearity to the model and helps in learning complex representations. The model is specifically designed for image segmentation tasks and is employed here for blood cancer cell segmentation in peripheral blood smear images.
VGG-16	ReLU	The VGG-16 model utilizes the ReLU activation function. It is a deep convolutional neural network architecture that consists of multiple convolutional and pooling layers. The VGG-16 model is widely known for its excellent performance in image classification tasks, and in this work, it is employed for blood cancer cell classification in peripheral blood smear images.

The CNN model used consists of multiple layers with different parameters. Here is a breakdown of the layers and their corresponding parameters:

1. Convolutional Layer 1:
   - Filter Size: 3x3
   - Number of Filters: 32
   - Activation Function: ReLU
2. Max Pooling Layer 1:
   - Pooling Size: 2x2
3. Convolutional Layer 2:
   - Filter Size: 3x3
   - Number of Filters: 64
   - Activation Function: ReLU
4. Max Pooling Layer 2:
   - Pooling Size: 2x2
5. Flatten Layer
6. Fully Connected Layer 1:
   - Number of Neurons: 128
   - Activation Function: ReLU
7. Output Layer:

- Number of Neurons: 4 (corresponding to the number of classes)
- Activation Function: SoftMax

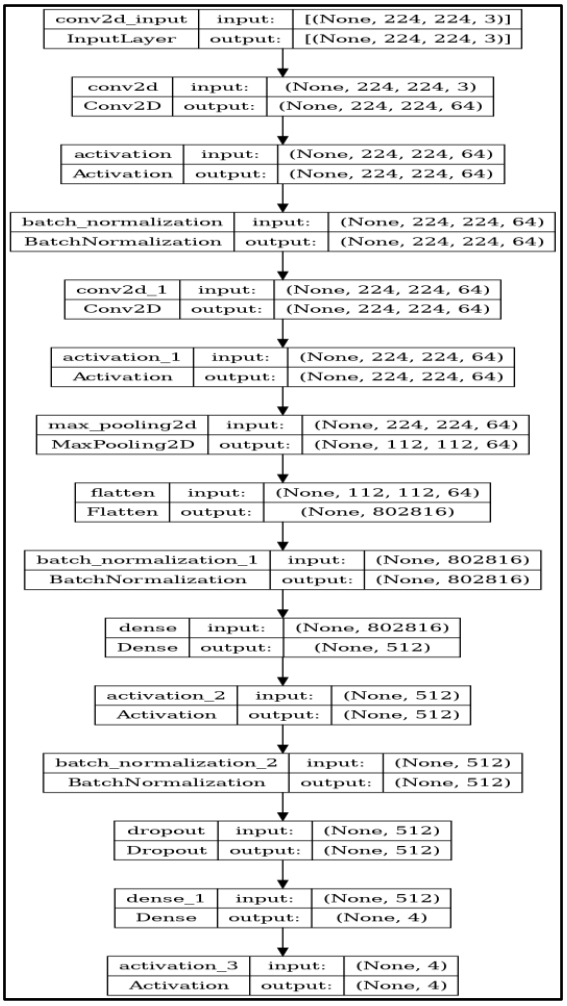

Fig. 5. Layers and its parameters for CNN model

The numerical formulas for the convolutional layer parameters are as follows:

- Total number of parameters in a convolutional layer

  = (filter size * number of input channels +1) * number of filters

  For example, the first convolutional layer has 32 filters, a filter size of 3x3, and 3 input channels:
- Total parameters in the first convolutional layer = (3x3x3 + 1) x 32 = 896

  Similarly, the second convolutional layer has 64 filters:
- Totals for the second convolutional layer's parameters = (3x3x32 + 1) x 64 = 18496

The formulas help us to compute the volume of parameters in every layer based on their formations.

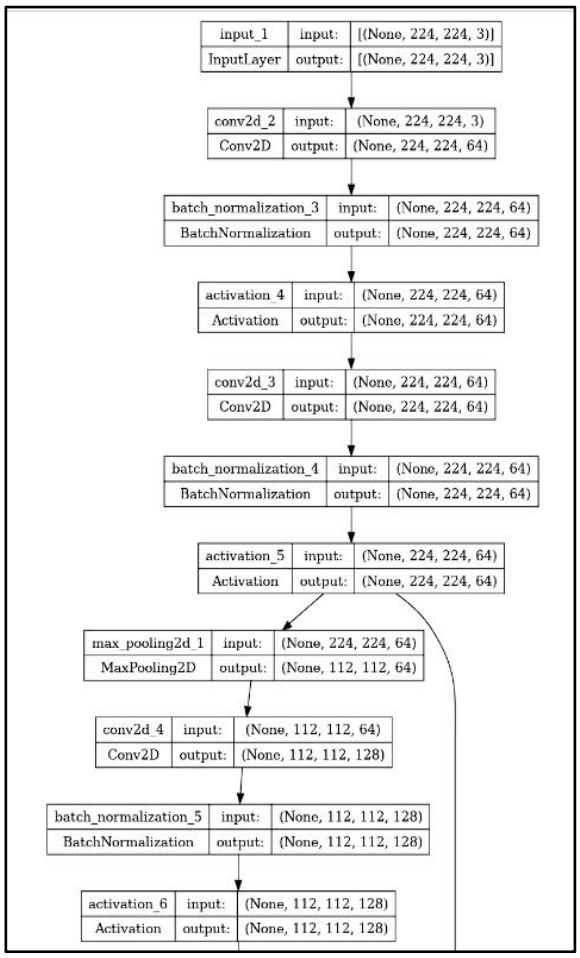

Fig. 6. Layers and its parameters for Modified U-Net model

The "Modified U-Net model "utilized in the code contains numerous layers with detailed parameters. Here is an outline of the layers and their consistent parameters:

1. Convolutional Block:
   - Number of filters: Varies
   - Filter/kernel size: 3x3
   - Padding: Same
   - Activation function: ELU
2. Max Pooling Layer:
   - Pool size: 2x2
3. Dropout Layer:
   - Dropout rate: 0.5
4. Up Sampling Layer:
   - Size: 2x2
5. Concatenate Layer: Combines the up sampled and residual feature maps
6. Output Layer (Conv2D):
   - Number of filters: 4 (corresponding to the number of classes)
   - Filter/kernel size: 1x1
   - Activation function: SoftMax

The mathematical formulas aimed at the convolutional block limitations are as follows:

a. Total number of parameters in a convolutional block = number of filters * (number of input channels * filter size + 1)

For instance, if a convolutional block has 3 input channels and 64 filters:

b. Total parameters in the convolutional block = (3x3x3 + 1) x 64 = 1792

Based on their configurations, these formulae may be used to determine the number of characteristics in every layer within the Modified U-Net model.

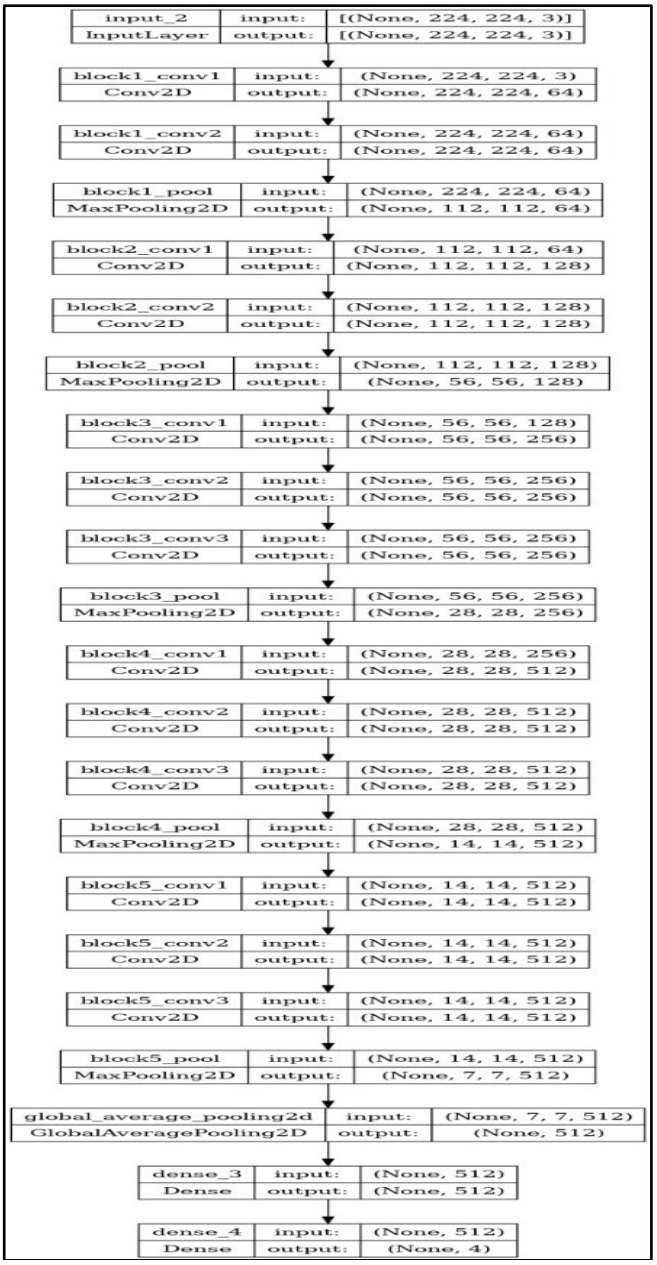

Fig. 7. Layers and its parameters for VGG 16 model

The VGG16 model essentially consists of numerous convolutional and entirely associated layers with specific limitations. Here is a summary of the layers and their dependable parameters:

1. Convolutional Layers:
   - Filter/kernel size: 3x3
   - Number of filters: Varies (64, 128, 256, 512, 512)
   - Padding: Same
   - Activation function: ReLU
2. MaxPooling Layers:
   - Pool size: 2x2
3. Fully Connected Layers:
   - Number of units: Varies (4096, 4096, 1000)
   - Activation function: ReLU
4. Output Layer:
   - Number of units: 4 (corresponding to the number of classes)
   - Activation function: SoftMax

The numerical formulas for the number of parameters in each layer can be calculated as follows:

- For convolutional layers:
  Total parameters =number of filters * (number of input channels * filter size + 1)
- For fully connected layers:
  Total parameters = number of output units * (1 +number of input units)

Based on the provided setup, it is possible to calculate the total amount of variables in all layers of the VGG16 model using these formulae.

III. Results and Discussion

A. Model Performance and testing

The CNN model established a test set accurateness of 42% following a training interval of 10 epochs. The model showed a decline in loss and a rise in accuracy through the training process. Nevertheless, the presentation of the model is relatively subpar, suggesting the requirement for additional improvements or alternative methodologies to achieve higher classification outcomes.

The CNN model has a 42% general accuracy on the test set, as stated by the classification report. The model's presentation is not uniform across classes; the "Early" category has the top accuracy and recall. Lower "***accuracy, recall, and F1-score***" values are attributed to the model's inability to appropriately classify the "Benign and Pre" classes.

Following a training period of 10 epochs, the accurateness of the "***Modified U-Net model***" on the test set is 52%. During training, the model showed a decrease in loss and an increase in accuracy. The model's performance is improved than the CNN model's, further improvements and substitute methods are still required to categorize blood cancers with additional precision.

The Modified U-Net model's classification statement demonstrates that it attained 52% accuracy on the test set. The model achieved quite well when categorizing the "Early and Pro" groups, through "***high precision, recall, and F1-score values***". The model had distress categorizing incidences that belonged to the "Benign" and "Pre" classifications, which caused in lower accuracy ratings for these specific categories.

The VGG16 model demonstrated extraordinary effectiveness in defining the type of blood cancer, with an extra ordinary accuracy rate of 99% on the test set. Through the training phase, the model continuously displayed a decline in loss and growth in accuracy. It is revealed that the VGG16 design is effective in accurately recognizing and identifying occurrences of blood cancer.

	precision	recall	f1-score	support
Benign	0.99	0.98	0.98	124
Early	0.98	1.00	0.99	196
Pre	1.00	1.00	1.00	182
Pro	1.00	0.99	1.00	150
accuracy			0.99	652
macro avg	0.99	0.99	0.99	652
weighted avg	0.99	0.99	0.99	652

Fig. 8. Classification report of VGG 16 model

The classification report of the VGG16 model showcases excellent presentation, reaching an outstanding accuracy of 99% on the test set. The model demonstrated exceptional performance in accurately categorizing all classes, such as "Benign, Early, Pre, and Pro", achieving "high precision, recall, and F1-scores". The model exhibited its proficiency in precisely identifying blood cancer, demonstrating its efficacy in aiding with the detection and diagnosis of cancer.

B. Different parameters of the model

Table 2: Activation function comparison

Model	**Layers**	**Parameters (Calculations)**	**Total Parameters**
CNN	"Conv2D, MaxPooling2D, Flatten, Dense"	3x3x3x64 + 64 + 64x4 + 4….	46,868
Modified U-Net	"Conv2D, MaxPooling2D, UpSampling2D, Concat"	3x3x3x64 + 64 + 3x3x64x128 + 128….	1,178,372
VGG16	"Conv2D, MaxPooling2D, GlobalAvgPooling2D"	(3x3x3x64 + 64) x 2 + 64x128 + ...	14,979,396

The CNN model utilized by the given code includes 46,868 parameters. Model layers are utilized to determine these characteristics. The number of output channels and the size of the kernel (filter) inform the parameters of the "convolutional layers (Conv2D)". No new parameters are introduced by the "pooling layers (MaxPooling2D)". The "Flatten layer flattens" the 2D array generated by the convolutional layers. The parameters of the Dense layer, which are set by the sizes of the input and output, are always the same. In this scenario, the Dense layer predicts the probability for the four classes using 64 neurons and a soft max activation function on the output layer.

In all, there are 8,880,132 parameters in the "***Modified U-Net model***" utilized by the given code. Model layers are utilized to determine these characteristics. Multiple convolutional blocks, each with two convolutional layers with "batch normalization" and "ELU activation functions", make up the model. The size of the kernel, the number of filters, and the widths of the input and output channels all have a role in determining the total number of convolutional layer parameters. Up-sampling blocks and skip connections, both of which add new parameters to the model, are also a part of it. Multi-class classification is handled by a final fully connected layer using SoftMax activation, and features are extracted in a final 1x1 convolutional layer.

There are 14,979,396 parameters in the VGG16 model utilized by the given code. Model layers are utilized to determine these characteristics. Multiple convolutional layers with varying filter sizes are the foundation of VGG16, which is followed by max pooling layers for down sampling. The size of the kernel, the number of filters, and the widths of the input and output channels all have a role in determining the total number of convolutional layer parameters. Adding to the overall number of parameters is the model's final set of completely linked layers. VGG16 can learn complex characteristics from pictures because to its large number of parameters, but this also makes it computational costly to train.

C. Cancerous cell Classification

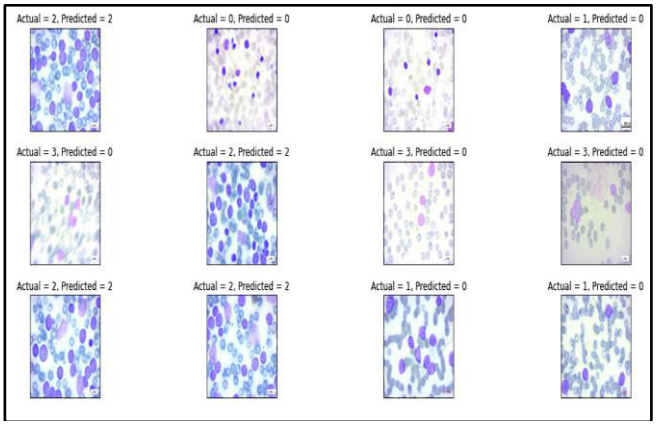

Fig. 9. Predicted results with CNN model

The CNN algorithm yields a mixed bag of right and wrong guesses in its predictions. While some forecasts are spot-on with the correct labels, others reveal disparities. Class 2 events were correctly predicted by the model, as were class 0 (benign) events. Class 0 was incorrectly predicted for classes 3

and 1, among other examples. The overall accuracy of the CNN model in predicting the classes was around average, with several areas showing significant potential for improvement.

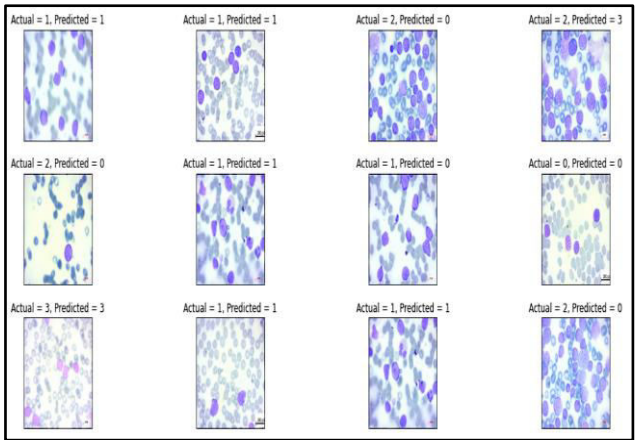

Fig. 10. Predicted results with U-Net model

The accuracy of the Modified U-Net model's predictions is greater than that of the CNN model. Class 1 was correctly predicted for class 1, while class 3 was correctly predicted for class 3. Some incorrect predictions were made, such as class 0 for class 2, and class I for class 1. Ultimately, the "Modified U-Net model" did a good job of classifying data, however there were some outliers.

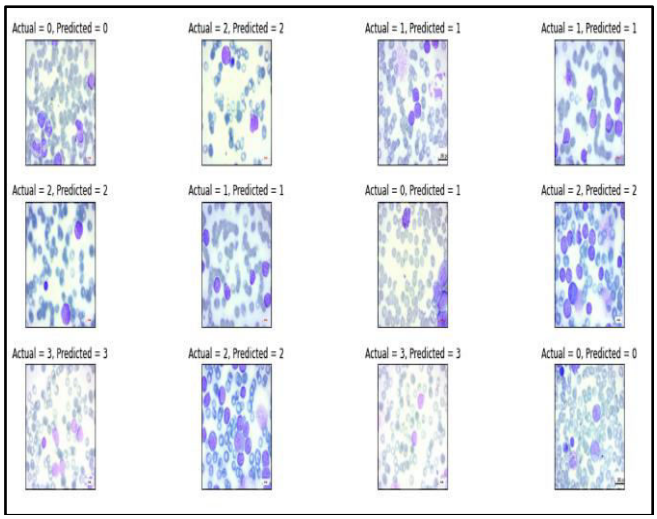

Fig. 11. Predicted results with VGG 16 model

The VGG16 model's predictions are very accurate, being right in many instances. It correctly predicted class 0 for real 0s, class 2 for real 2s, and class 3 for real 3s. However, there were a few instances of incorrect categorization, such as assuming class I when class 1 was the correct category. The VGG16 model did a good job predicting the classes generally, with just a few outliers.

D. Model comparison

	Model	Accuracy
0	CNN model	42
1	Modified U-Net	52
2	VGG-16	99

Fig. 12. Comparative Study of CNN, Modified U-Net, and VGG16 Models

The accuracies from every model are recorded and stored for the comparison process as highlighted in the figure above.

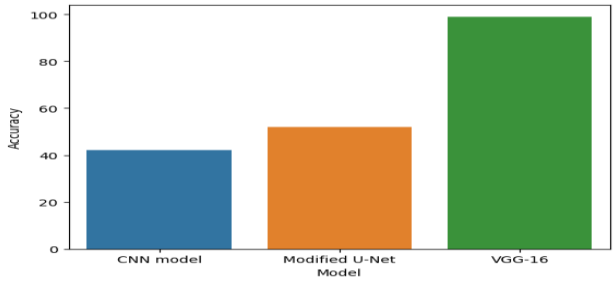

Fig. 13. Comparing model accuracies

The comparison of accuracies of every model is visually portrayed in the form of a bar plot. It is clearly understood that CNN model is the least performed model and VGG-16 is the most accurate model with an accuracy of 99%.

IV. Conclusion and Recommendations

A. Conclusion

This study summarizes the work toward a machine-learning-based artificial-intelligence system for the detection of blood cancer. This work's main contributions are the three models "(CNN, Modified U-Net, and VGG16)" that were implemented for blood cancer classification and segmentation. Images of peripheral blood smears taken from patients with "acute lymphoblastic leukemia (ALL)" were used to train and test these models. Overall, the CNN model showed a reasonable degree of accuracy, with a 42% success rate. The accuracy of the "Modified U-Net model" increased to 52%. The VGG16 model's 99% accuracy was much superior to that of the competition. With its strong performance in classification and segmentation, the VGG16 model shows great promise as a diagnostic aid for blood cancers. The findings underline the value of deep learning models in aiding the rapid and precise identification of blood cancer by medical experts.

Accurately detecting malignant cells is aided further using picture segmentation methods. The study's results pave the path for better patient care and management by adding to current attempts to use AI to boost cancer detection. However, further study and verification is required before these models may be used in actual clinical situations. Nonetheless, this discovery paves the way for the creation of AI-based solutions to aid doctors in making correct and timely diagnoses of blood cancer.

B. Recommendations

The following suggestions are based on the data and analysis offered in this study.

1. Experiment with new architectures, activation functions, and hyperparameters to further increase the performance of already-existing models like the "CNN, Modified U-Net", and "VGG16" [18].
2. The size of the datasets used to train and verify the models should be increased. The built AI-assisted diagnostic system will be more reliable and applicable if this is done.
3. To guarantee the system is compatible with current medical processes, it is important to encourage partnerships between medical specialists, researchers, and data scientists to collect domain-specific insights, verify the models using clinical data, and test the system's performance.
4. Extensive validation studies utilizing real-world clinical data should be conducted to evaluate the practical performance of the proposed models and their effect on clinical decision-making.

The area of AI-assisted diagnosis for blood cancer stands to benefit greatly from the implementation of these suggestions, which will lead to improved diagnostic accuracy and efficiency and, ultimately, better patient outcomes.

REFERENCES

[1]. Kumar, D., Jain, N., Khurana, A., Mittal, S., Satapathy, S.C., Senkerik, R. and Hemanth, J.D., (2020). Automatic detection of white blood cancer from bone marrow microscopic images using convolutional neural networks. *IEEE Access*, *8*, pp.142521-142531.DOI: 10.1109/ACCESS.2020.3012292

[2]. Wang, H., Dong, L., Song, W., Zhao, X., Xia, J., & Liu, T. (2022). Improved U-Net-Based Novel Segmentation Algorithm for Underwater Mineral Image. *Intelligent Automation & Soft Computing*, *32*(3). DOI:10.32604/iasc.2022.023994

[3]. Vogado, L., Veras, R., Aires, K., Araújo, F., Silva, R., Ponti, M., & Tavares, J. M. R. (2021). Diagnosis of leukemia in blood slides based on a fine-tuned and highly generalizable deep learning model. *Sensors*, *21*(9), 2989.https://doi.org/10.3390/s21092989

[4]. Drałus, G., Mazur, D., &Czmil, A. (2021). Automatic detection and counting of blood cells in smear images using retina net. *Entropy*, *23*(11), 1522. https://doi.org/10.3390/e23111522

[5]. Liu, J., Tripathi, S., Kurup, U., & Shah, M. (2020). Pruning algorithms to accelerate convolutional neural networks for edge applications: A survey. *arXiv preprint arXiv:2005. 04275.https://doi.org/10.48550/arXiv.2005.04275*

[6]. Patel, A., Degadwala, S., & Vyas, D. (2022, November). Lung Respiratory Audio Prediction using Transfer Learning Models. In *2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)* (pp. 1107-1114). IEEE. DOI: 10.1109/I-SMAC55078.2022.9986498

[7]. Devi, T. G., Patil, N., Rai, S., & Sarah, C. P. (2023). Real-time microscopy image-based segmentation and classification models for cancer cell detection. *Multimedia Tools and Applications*, 1-26.10. 1007/s11042-023-14898-w

[8]. Ahmed, K., Sachindra, D. A., Shahid, S., Iqbal, Z., Nawaz, N., & Khan, N. (2020). Multi-model ensemble predictions of precipitation and temperature using machine learning algorithms. *Atmospheric Research*, *236*, 104806. https://doi.org/10.1016/j.atmosres.2019.104806

[9]. Ackerson, J. M., Dave, R., &Seliya, N. (2021). Applications of recurrent neural network for biometric authentication & anomaly detection. Information, 12(7), 272. https://doi.org/10.3390/info12070272

[10]. Yacouby, R., & Axman, D. (2020, November). Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models. In *Proceedings of the first workshop on evaluation and comparison of NLP systems* (pp. 79-91). DOI: 10.18653/v1/2020.eval4nlp-1.9

[11]. Analyticsvidhya.com, (2023), *Convolutional Neural Network: An Overview*, Available at:10/07/2023, Acquired From: https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/

[12]. Researchgate.net, (2023), *U net Convolutional Neural Network model*, Available at:10/07/2023, Acquired From: https://www.researchgate.net/figure/U-net-Convolutional-Neural-Network-model-The-U-net-model-contains-two-parts_fig6_317493482

[13]. Medium.com, (2023), *VGG 16 Easiest Explanation*, Available at:10/07/2023, Acquired From: https://medium.com/nerd-for-tech/vgg-16-easiest-explanation-12453b599526

[14]. Bhatt, D., Patel, C., Talsania, H., Patel, J., Vaghela, R., Pandya, S., ... &Ghayvat, H. (2021). CNN variants for computer vision: History, architecture, application, challenges and future scope. *Electronics*, *10*(20), 2470.https://doi.org/10.3390/electronics10202470

[15]. Ibtehaz, N., & Rahman, M. S. (2020). MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. *Neural networks*, *121*, 74-87. https://doi.org/10.1016/j.neunet.2019.08.025

[16]. Mascarenhas, S., & Agarwal, M. (2021, November). A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification. In *2021 International conference on disruptive technologies for multi-disciplinary research and applications*