Received: October 19, 2024. Revised: November 5, 2024.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

EfficientNetV2-S Enhancement to Classify Acute Lymphoblastic Leukemia: Integrating Pre-Trained Models and Grid Search for Optimal Performance

Arif Muntasa[1]* Rima Tri Wahyuningrum[1] Husni[1] Alisa Sugiarti[1] Muhammad Yusuf[1]
Abdelwahed Motwakel[2] Deshinta Arrova Dewi[3] Yuli Panca Asmara[4]
Zabrina Tuzzahra[5] Wayan Firdaus Mahmudi[6]

[1]Informatics Engineering Department, University of Trunojoyo Madura,
Raya. Telang Po. Box 2, Kamal, Bangkalan, East Java, Indonesia
[2]Department of Management Information Systems, College of Business Administration in Hawtat Bani Tamim,
Prince Sattam bin Abdulaziz University, AlKharj, Saudi Arabia
[3]INTI International Universiti, Faculty of Data Science & Information Technology (FDSIT),
Nilai 71800 Negeri Sembilan, Malaysia
[4]INTI International Universiti, Faculty of Engineering and Quantity Surveying,
Nilai 71800 Negeri Sembilan, Malaysia
[5]Information System Department Institut Teknologi Sepuluh Nopember, Surabaya, East Java, Indonesia
[6]Computer Science Department, University of Brawijaya,
Veteran No.10-11, Lowokwaru, Malang, East Java, Indonesia
* Corresponding author's Email: arifmuntasa@trunojoyo.ac.id

Abstract: Leukemia is one of the deadliest types of cancer. The hospitals have been working on several occasions to conduct early screenings of the preventable death. Unfortunately, leukemia detection is very expensive. This study tries to classify Leukemia images to find the best value of the hyperparameter using the Grid Search method processed using Pre-trained EfficientNetV2-S. We have modified the activation function ReLu6 (Variant of the Rectified Linear Unit has value less than 6) to SeLu6 (Variant of the Scaled Exponential Linear Unit has value less than 6) to help the network maintain stable statistical properties during training. Our proposed model has five major stages: Input layer, Stem Layer, Mobile Inverted Bottleneck Convolution (MBConv), Fused Mobile Inverted Bottleneck Convolution (Fused-MBConv) and Head layer. We use a total of five different Fused-MBConv layers. The depth-wise convolutional architecture and expansion operation are fused into one single unified step. The process can be applied to improve the general performance to be more reliable and precise in getting responses. Our proposed way adopts a comprehensive scaling approach to adjust the depth, width, and image resolution proportionally. Besides, MBConv and Fused-MBConv are applied to enhance the performance of the model. We utilize Grid Search to perform hyperparameter tuning and obtained $\alpha = 0.001$, and $E = 10$ as the optimal hyperparameter for our proposed architecture model. Our proposed model has been tested on the C-NMC-2019 Leukemia image dataset. Experimental in the training process have achieved accuracies 98.89% to 99.80%. The validation results give an accuracy within the range of 96.65% to 98.31%, while the testing results produce accuracy within the range of 98.01% to 99.85%. The AUC values for all folds have constantly generated an area of 0.97. We compare our proposed results with other methods, and the comparison results have shown that our performance results are better than EfficienNetB0, CNN-based ECA Module, Vision Transformer, Majority Voting Technique, CNN Model-based Tversky Loss Function, and Lightweight EfficientNet-B3. This research results can be followed up as product innovation in medical fields.

Keywords: Grid search, Leukemia image, Product innovation, Preventable death, EfficientNetV2-S.

1. Introduction

The computer vision has been implemented in several fields, especially biomedical imaging. Automatic detection, prediction, and classification processing have supported the research results. Some researchers carried out and developed research associated with biomedical imaging like leukemia detection and classification. Early leukemia diagnosis needs to be done early because early action can increase the chance of recovery. They developed an aggregated deep-learning model for classifying Acute Lymphoblastic Leukemia. They have applied augmentation to overcome the limitations of the dataset and transfer learning strategies to accelerate the learning process. Their research results demonstrate that our method can combine features extracted from the best deep-learning models, achieving a test accuracy of 96.58% in diagnosing Acute Lymphoblastic Leukemia [1]. One limitation of the research is that large discrepancies between the source and target domains may result in a loss of model performance.

One of the complete studies, starting from the morphological analysis of blood cells, is done manually by experienced operators. The paper proposes an automated approach for recognizing and classifying WBCs from microscopic images, which initiates with WBCs identification and proceeds with the extraction of morphological features demanded for the ultimate stage of classification. The results were supplied with 92% accuracy [2]. One of the limitations of this study is the quality of the classification results depends on the image pre-processing stage. If the pre-processing is not properly done, it will directly lead to classification errors. Other studies have also been performed with another approach—namely, by first performing fuzzy-based color segmentation in order to separate leukocytes from the rest of the blood components.

Discriminative features like the shape and texture of nuclei are used to identify leukemia with the help of new shape features, namely Hausdorff Dimension and contour marks followed by classification by a Support Vector Machine (SVM) [3]. Such an approach may be less practical in distinguishing between different subtypes of leukemia, which require further analysis. Inconsistent results may be obtained due to variability in image acquisition and analysis processes.

In addition, a different approach was used, which was an optimized Dense Convolutional Neural Network (DCNN). This model achieved a high accuracy of 97.2%, with correct cancer predictions 94 times out of 100 trials, outperforming conventional machine learning methods such as SVM and Decision Tree [4]. It shows that the DCNN model is adequate for determining cancer types in bone marrow with fewer parameters. However, small changes in the training data can result in significantly different tree structures, affecting the results' consistency.

Another approach researchers use in leukemia image classification is to segment white blood cells with pre-processing, conversion from RGB to CMYK, histogram equalization, thresholding with the Zack technique, and image background removal [5]. Furthermore, color, texture, and shape features are extracted and normalized using z-score, min-max, and gray-scaling. In the study, the dataset used for the experiment was ALL-IDB2, consisting of 260 cell images (130 normal and 130 leukemia). Various classifiers were tested, and the final results showed that the K-Nearest Neighbors (K-NN) algorithm achieved the highest classification accuracy. Nonetheless, the study has a limitation in that the color model may be less robust to changes in scale or rotation, which can affect classification accuracy.

Researchers used a few features, such as shape, color, and texture, and extracted by using background removal techniques [6]. The proposed method yields 92% accuracy in identifying 245 of 267 leukocytes from 33 ALL-IDB1 images and with Support vector Machine and RBF kernel can classify ALL-IDB1 images up to 93% accuracy and 98% sensitivity. Another researcher has produced a similar approach. It depends on grey level co-occurrence matrix (GLCM) based feature extraction and probabilistic principal component analysis (PPCA) based feature reduction. The related features are then utilized in a random forest (RF) based classifier [7]. A considerable number of experiments is carried out on ALL-IDB1 dataset, and comparative analysis is also executed with the existing schemes in terms of sensitivity, specificity, and classification accuracy. The segmentation accuracy was 96.29%, while the classification accuracy was 99.004% for the nucleus and 96% for the cytoplasm. The GLCM is reliable for small datasets like ALL-IDB1 or ALL-IDB2, but for the bigger datasets, it focuses more on local information, which makes it not able to grab broader patterns or more detailed textures in the images.

In the recent past, a majority of researchers have used Convolutional Neural Networks in the detection and classification of tasks for Leukemia images. This study proposed a method that utilizes Convolutional Neural Network (CNN) for the classification of normal and abnormal blood cell images. The proposed method achieved 96.6% accuracy using the ALL-IDB1 dataset of 1,188 blood cell images [8].

Others also reported the application of transfer learning of DenseNet201, in order to reduce the number of training steps and evaluate the effect of dataset pre-processing [9]. Results demonstrated that the DenseNet201 model produced the best performance at an 0.87 AUC, which utilized histogram equalization and reduced the performance. Among the disadvantages of DenseNet201 is that this model is developed for more general image recognition, so it may not be optimal considering the specific characteristics in the ALL-IDB1 dataset. Another very important factor affecting the results is inappropriate hyper-parameter selection during training.

New results have been reported that using a computer-based system to diagnose leukemia, utilizing Convolutional Neural Networks (CNN) and transfer learning has been tested with a Support Vector Machine as a classifier, without a segmentation process first. The test results using ALL-IDB1 showed that feature extraction using CNN and similarity measurement using SVM produced more than 99% accuracy [10, 11]. Choosing the proper kernel and hyperparameters is crucial; an incorrect choice can reduce the model's performance.

The use of convolutional neural networks (CNN) has displaced convolutional methods for feature extraction, such as grey-level co-occurrence matrix (GLCM), colour co-occurrence matrix (CCM) [12], Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Linear Preserving Projection (LPP), Independent Component Analysis, and appearance-based kernels. One of the strengths of CNN is that it combines the processes of feature extraction and classification. CNN is becoming a viral method in medical image classification, including Leukemia image diagnosis. Since the introduction of CNNs, many studies have demonstrated their superiority in various image classification tasks. However, the CNN method is also not free from shortcomings, such as the need for large training data and high computation time to perform the training and learning process.

Several researchers have been conducting studies to classify the Leukemia images, among them is the classification of Leukemia based on pattern extraction using GLCM and its derivatives [7, 12]. GLCM and its derivatives are two-order-based statistical approaches in analyzing the texture of an object. The GLCM process starts with the forming of a matrix equal to the number of grey levels of the image by considering its nearest neighbors. Second-order statistics are subsequently applied to provide the object pattern characteristics in terms of homogeneity, energy, entropy, and contrast, depending on the outcome of the matrix formation. In this manner, several drawbacks are associated with this approach: "limiting object texture information [13, 14]. GLCM and its derivatives prove to be much sensitive in respect to position, scale, and rotation alterations, on which classification results are solely dependent. The GLCM cannot do optimally if an image of reasonably low contrast level is used. The last disadvantage is that GLCM cannot extract multi-scale features. These defects make GLCM less precise in modeling positive and negative Leukemia images.

Because of these deficiencies, more sophisticated feature extraction techniques, such as Leukemia images, have been commonly applied in the classification of medical images. Application of Convolutional Neural Networks (CNNs): CNNs can automatically extract significant features from images without any explicit steps in defining texture features [15, 16]. Deep Learning-Based Models: More complex deep learning models, such as ResNet, DenseNet, and EfficientNet, can provide better results in medical image classification due to their ability to handle complexity and variation in data. Considering these drawbacks, it is important to choose a feature extraction method that suits the characteristics of the data and the purpose of the analysis, especially in critical applications such as medical diagnosis.

AlexNet was an important milestone in introducing image patterns before LeNet, but this method has some significant weaknesses. AlexNet has about 60 million parameters, which requires large storage memory and long computing time. Such architectures would lead to overfitting conditions, especially on smaller data sets, despite the use of data augmentation techniques, hits, and drop-outs [17, 18]. Some researchers have conducted Leukemia image classification using Pre-trained AlexNet Architecture, where they employed Acute Lymphoblastic Leukemia - Database (ALL-IDB dataset, two classes with 130 images each class). The results show that 98.05% [17], 96.1% [18], and 98% [19] accuracies. However, the method is not efficient, as the method requires high costs for pre-processing, data augmentation, segmentation, and classification. Even though the researchers only used small data as a dataset in the experiment. Meanwhile, other researchers have also conducted image classification for Acute Myeloid Leukemia (AML) [20]. However, they used the same method, which is called AlexNet. They delivered 98.58% accuracy for classification. Unfortunately, manual microscopic examination is tedious and time-consuming.

On the other hand, VGGNet also introduces its architecture with many convoluted layers of small size. However, this condition also creates problems, including large memory requirements, where VGGNet requires 140 million parameters, which is also huge. The impact of this memory requirement will cause the speed to slow the training process and decision-making to slow. This architecture is inefficient for use in mobile applications. Some researchers have employed pre-trained VGGNet to classify Leukemia images [21, 22]. They have delivered 93.01% accuracy using fine-tuning of pre-trained VGG16 [22]. The limitation of the research is the unavailability of the huge and well-annotated dataset. The available data is not extensive and well-annotated, so there are dependencies. In addition, other researchers have produced accuracy.

The GoogleNet architecture has tried to reduce the number of parameters using the Inception module. Still, the complexity of the architecture applied affects the implementation, and tuning of the model becomes more complex, especially in hyperparameter adjustment [23]. They employed Contrast-Limited Adaptive Histogram Equalization (CLAHE) to segment the main object and a new hybrid technique to classify the Leukemia image, which combines CNN feature extraction with an Efficient Salp Swarm Algorithm (ESSA) to optimize the extracted features. They delivered 98.1% and 98.8% accuracies for the dataset 1 and 2 [23]. However, the GoogleNet architecture has some weaknesses. First, GoogleNet will be more difficult to understand and implement when using many layers and Inception modules. Secondly, it requires much training time and memory because of the number of parameters and computational operations used. Third, the complexity of GoogleNet is relatively high, so it can lead to overfitting, especially when using small or less varied training data.

The research conducted by Amreen Batool and Young-Cheol Byun focuses on developing an algorithm for more accurate classification of acute lymphoblastic leukemia (ALL) by utilizing deep learning techniques through the lightweight EfficientNet-B3 model [24], their proposed model has delivered 99.31% accuracy. Given the difficulty in distinguishing between ALL cancer cells and normal cells through microscopic analysis, their study introduces a model that uses deep separable convolution to improve the classification performance of leukemia cells. This model is evaluated using the C-NMC-19 dataset and is measured by confusion matrix such as accuracy, precision, recall, and f1-score. The results show that the model is superior to other deep learning models, and is more efficient with fewer parameters. In addition, the proposed algorithm is compared with other classification algorithms that also use the same dataset: C-NMC-2019. Gao et al. [25] proposed the EfficientNetB0 architecture and compared it with various CNN models, including ResNet and Inception V3, where the presented method achieved the highest classification accuracy of 95.18%. Meanwhile, Ullah et al. applied CNN-based ECA module to improve hyper-parameters and obtained 91.10% accuracy [26]. Another study recommended a vision transfer model with 88.20% accuracy [27]. In addition, the study conducted by Ghaderzadeh et al. introduced a deep learning-based model by applying eight convolutional neural network (CNN) models to extract features and classify lymphoblast and normal cells. Of the eight models, four CNN models with the best performance were selected to form an ensemble classifier using the majority voting method. The proposed architecture was evaluated using C-NMC-2019, the accuracy results reached 98.5% [28]. Voting technique was also applied to compare the model performance in accuracy. Experimental findings showed that Ansari et al. [29] used Tversky loss function based on CNN model, their proposed architecture have delivered 99.01% accuracy.

Reflecting on the limitations highlighted in earlier research conducted by previous scholars, we introduce EfficientNetV2-S, a new class of convolution networks to enhance the scale efficiency curves of parameters and drive state-of-the-art performance on various tasks. We optimized the parameter values of models and frequencies in neural architecture (NAS) searches and scales. We employ Search Grid as a tuning hyperparameter to find the best hyperparameter during training. We also modified an activation function Relu6 into SeLu6 to support the network in maintaining stable statistical properties during the training process. Our results indicate that the EfficientNetV2 model can practice more architecture than before. Increasing image size can speed up the training process. In this work, we also presented a new progressive learning procedure to balance the regularization and image size techniques, so it is also suitable for classifying objects at low resolution.

2. Material and method

2.1 Material

We employed the ImageNet-trained to evaluate our proposed architecture using the C-NMC-2019 dataset. The dataset splitted into two components,

which are a training set and testing set. The training set consists of 26 persons (with 3389 images) and 47 people diagnosed with acute lymphoblastic Leukemia (ALL, totaling 7272 images). Furthermore, the preliminary testing set consists of 648 ALL images from 15 individuals and 13 individuals with ALL (1219 images). Overall, the training and testing sets contain 10,661 and 1867 images.

In this study, we use a pre-trained model that was previously trained on the ImageNet dataset. ImageNet is one of the largest and most famous datasets in the computer vision domain. ImageNet has 1,281,167 training images, 50,000 validation images, and 100,000 test images. ImageNet images have a large variety that allows the trained model to learn various general features of images, such as patterns, shapes, textures, and colours. A pre-trained model is a model that has undergone training. ImageNet allows its weights or parameters to be optimized to identify common visual features. The training results of the model already have knowledge, which can be helpful to adapt when used for different object classifications.

2.2 Method

Our proposed method comprises five layers, with several sub-layers for each layer. The main stages are Input, Stem, Mobile Inverted Bottleneck Convolution Layer (MBConv), Fused- MBConv, and Head layers, as shown in Fig. 1.

2.2.1. Input layer

The architecture of EfficientNetV2s has mentioned that it allows the input image to be 224x24 with three channels. It indicates that every entered image has to be uniformly re-sized to an already known scale of 224 p by 224 pixels. At the same time, input resolution and the skipped volume can also be transmitted as colouring images, which are twenty-four bits-for instance, eight bits for each red, green, and blue channel.

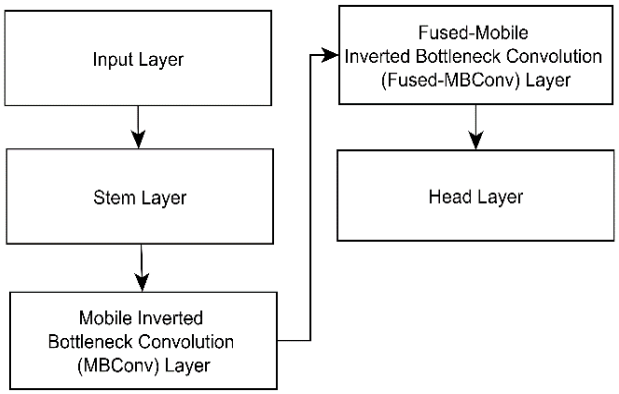

Figure. 1 Our Proposed Method

2.2.2. Stem layer

The stem layer that forms an essential element of the architecture of EfficientNetV2-S then has a tremendous impact on it. It is a starting place for input images and conducts preliminary work before higher layers are used to perform more complex feature extraction and classification operations, included in multiple operations in a series to convey more information with a relatively large kernel size, followed by batch normalization and the Sigmoid Liner Unit activation function. Specifically, the stem layer usually involves the 3x3 convolution with 32 Kernels and stride=2 on the input image to capture local features and reduce the dimensions. Furthermore, batch normalization is used to normalize the feature maps to improve the convergence of the training phases. The next step is to pass the result to the activation function, which makes the function and allows the network to learn more layers of abstraction using the Sigmoid Liner Unit activation function, as shown in Fig. 2.

2.2.3. Mobile inverted bottleneck convolution layer (MBConv) layer

MBConv process aims to maximize EfficientNetV2-S performance by expanding an input channel, applying depth-wise convolution, and projecting back to a lower-dimensional space. In addition, MBConv also ensures the network can learn complex representations at a minimum cost. In this stage, there are several processes: Expansion phase, Depth-wise convolution, Squeeze-and-excitation, and Projection phase. We provide a list of variables, as shown in Table 1, to facilitate the understanding of the equations we use.

Expansion phase

The Expansion phase goal is to increase the input channels using point-wise convolution, followed by an activation function.

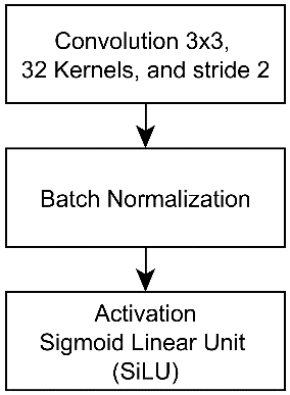

Figure. 2 Stem Layer

Table 1. The Notation Employed in the Proposed Method

No	Symbols	Description
1	$\mathcal{X}, \mathcal{W}_e, t, Activation$ and $\mathcal{X}_e$	Input tensor, convolution weight, expansion factor, activation function, and output tensor
2	$(H, W, C_{in}), (1, 1, C_{in}, t \times C_{in})$, and $(H, W, t \times C_{in})$,	Dimension of input tensor, convolution weight, and output tensor
3	$W_{e,} \odot, X,$ and X_1	Depth-wise convolution weight, a Hadamard operation, an input, constant value, and an expanded feature map.
4	X_2	output depth-wise convolution
5	$\otimes$	Convolution operation
6	$GAP(X)_c$	Global average pooling for (X) specific c channel
7	H	The height of the feature map or the input tensor.
8	W	The width of the feature map or the input tensor.
9	i, j	Index of height and width of the feature map
10	Z_{ReLU}, and b	Output of Rectified linear unit and bias
11	Z_{se}	Sigmoid Output of Global average pooling
12	W_p	Projection convolution weight with the shape dimension $(1, 1, t \times C_{in}, C_{out})$, where C_{in} and C_{out} show the number of channels used for input and output.
13	W_{fused} and W_{proj}	weight of the combined convolution filter and projection weight
14	k	Kernel size
15	$GAP(Y)_c$	Global average pooling for (Y) specific c channel
16	$Y_{ijc}, W_{fc}, b_{fc},$ and Z_h	Fused-MBConv result, fully connected weight, fully connected bias, and fully connected result
17	$W_{fc}^{(t)}$	Initial value can be generated from the randomly normal distribution ($\mathcal{N}$) using mean 0 and variance $\frac{1}{n_{in}}$
18	*Tp, Tn, Fp, and Fn*	True positive, true negative, false positive, and false negative
19	α, K	Learning rate and the value of fold cross validation

Suppose the following variables $\mathcal{X}, \mathcal{W}_e, t, Activation$ and $\mathcal{X}_e$ represent an input tensor, convolution weight, expansion factor, activation function, and output tensor. In addition, input tensor, convolution weight, and output tensor have dimension $(H, W, C_{in}), (1, 1, C_{in}, t \times C_{in})$, and $(H, W, t \times C_{in})$, respectively. The symbols of C_{in} and C_{out} state number of channel input and output at the tensor. In this case, the model using the SeLU6 to replace ReLU6 activation functions is as follows.

$$X_1 = min\left(6, \lambda \begin{cases} W_e \odot X \ \ if\ W_e \odot X > 0 \\ \alpha\left(e^{W_e \odot X} - 1\right) \ \ if\ W_e \odot X \leq 0 \end{cases}\right) \quad (1)$$

$W_{e,} \odot, X, \lambda,$ and X_1 represent depth-wise convolution weight, a Hadamard operation, an input, constant value, and an expanded feature map.

Depth-Wise Convolution

Depth-wise convolution carries out the task of independent filter. It is conducted on each input to reduce memory usage and computational complexity as shown in Eq. (2).

$$X_2 = min(max(0, W_d \otimes X_1), 6) \quad (2)$$

X_2 is output depth-wise convolution

Squeeze-and-excitation

It is required to help recalibrate the feature map by global average pooling followed by fully connected layers using ReLU and Sigmoid activation functions.

$$GAP(X)_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X_{ijc} \quad (3)$$

$$Z_{ReLU} = Max(0, GAP(X) \times W + b) \quad (4)$$

$$Z_{se} = \frac{1}{1 + e^{-(GAP(X) \times W + b)}} \quad (5)$$

$GAP(X)_c, H, W, i,$ and j represent Global average pooling for specific channel, the height of the feature map or the input tensor, the width of the feature map or the input tensor, index of height, and index of width.

Projection Phase

This stage is used to prevent the channels from going back to output. It employs point-wise convolution as follows.

$$X_3 = X_2 \otimes W_p \quad (6)$$

W_p describes projection convolution weight with the shape dimension $(1, 1, t \times C_{in}, C_{out})$, where C_{in} and C_{out} show the number of channels used for input and output.

2.2.4. Fused-mobile inverted bottleneck convolution layer (Fused-MBConv)

One of the variations of the Mobile Inverted Bottleneck Convolution (MBConv) standard used in efficient neural network architectures is Fused-Mobile Inverted Bottleneck Convolution (Fused-MBConv). In this case, we involve five different Fused-MBConv, as shown in Fig. 3. This process has integrated the expansion and depth-wise convolutional model into a single operation. This process improves an architecture's performance for accurate results and reduces the computational time required for training. Fused-MBConv employs Fused Expansion and Depth-wise Convolution, as shown in Eq. (7). Suppose X is the input tensor with (H, W, C_{in}) dimension. We can write the operation in the Fused-MBConv as follows.

$$X_{fused} = max\left(0, X \otimes W_{fused}\right) \quad (7)$$

$$Y = X_{fused} \otimes W_{proj} \quad (8)$$

Regarding Eq. (7), the W_{fused} has the shape size $(k, k, C_{in}, t \times C_{in})$ dimension, the symbol of k shows the kernel size.

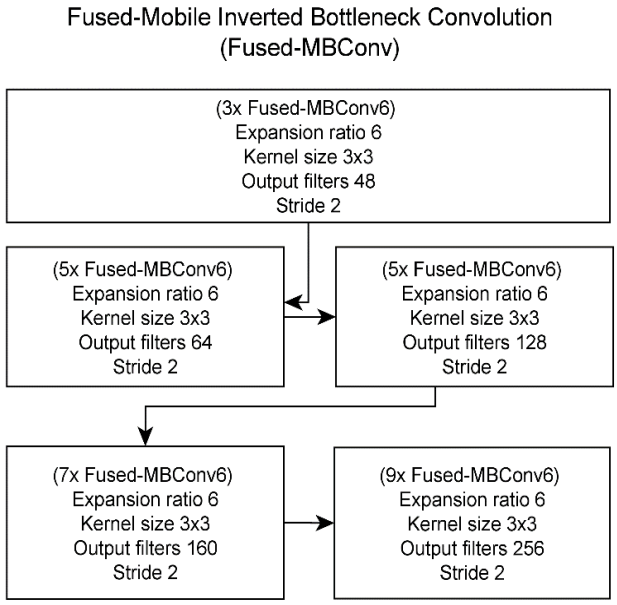

Figure. 3 Fused-MBConv Model at Our Proposed Method

Furthermore, the variables W_{fused} and W_{proj} represent the weight of the combined convolution filter and projection weight. Based on Fig. 3, we obtained information that Fused-MBConv operations were performed as many as five times with the same stride and a different number of Fused-MBConv, i.e., 3, 5, 5, 7, and 9 times.

Head Layer

In general, the last operation of the neural network is to change high-level features into classification output. In addition, the head layer involves some operations, i.e., global average pooling as shown in Eq. (9), fully-connected activation function. Fully-connected transforms the global average pooling results into the classification result as the following Equations

$$GAP(Y)_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} Y_{ijc} \quad (9)$$

$$Z_h = GAP(Y) \times W_{fc} + b_{fc} \quad (10)$$

$$ReLU(Z_h) = \max\left(0, Z_h\right) \quad (11)$$

$$Softmax = \left(\frac{e^{Z_i}}{\sum_j e^{Z_i}}\right) \quad (12)$$

The symbols of $Y_{ijc}, W_{fc}, b_{fc},$ and Z_h describe the Fused-MBConv result as shown in Eq. (8), fully connected weight, fully connected bias, and fully connected result, respectively.

$$W_{fc}^{(t+1)} = W_{fc}^{(t)} - \eta \times \frac{\vartheta \mathcal{L}}{\vartheta W_{fc}} \quad (13)$$

The initial value of $W_{fc}^{(t)}$ can be generated using the Xavier Initialization

$$W_{fc}^{(t)} \sim \mathcal{N}\left(0, \frac{1}{n_{in}}\right) \quad (14)$$

Eq. (14) shows that the initial values of $W_{fc}^{(t)}$ can be generated from the randomly normal distribution $(\mathcal{N})$ using mean 0 and variance $\frac{1}{n_{in}}$.

2.2.5. Performance measurement

We also measure the classification results based on the Eq. (12). We employ confusion matrix to measure accuracy, precision, recall, and f1-score of our experimental results as follows

$$Accuracy = \frac{Tp+Tn}{Tp+Tn+Fp+Fn} \quad (15)$$

$$Precision = \frac{Tp}{Tp+Fp} \quad (16)$$

$$Recall = \frac{Tp}{Tp+Fn} \quad (17)$$

$$F1 - Score = \frac{2\times Precision\times Recall}{Precision+Recall} \quad (18)$$

True Positive (Tp) describes that the actual target and prediction are positive. If the actual target and prediction are negative, then it shows True Negative (Tn). False Positive (Fp) shows the actual target is negative, where the prediction delivers positive. If the actual target is positive and prediction are negative, then it is called False Negative (Tn).

3. Results and discussion

In this present paper, we utilized two important hyperparameters to obtain the best model: $learning\ rate = \{\alpha|(0.001, 0.0001, 0.00001) \subseteq \alpha\}$, *K-Fold*= $\{K|(1,2,3,4,5) \subseteq K\}$ and $epoch = \{E|(5,10,15) \subseteq E\}$. In addition, we employ the *Grid Search* method to select the best hyperparameter value. Regarding the above three hyperparameters, we have composed experimental scenarios based on combination of two hyperparameter: α and E.

3.1 Training results

The experimental results using our proposed architecture based on the *Grid Search* results with the best hyperparameter values are $\alpha = 0.001$, and $E = 10$ as the best hyperparameter. We have represented the best experimental results, as shown in Figs. 4 to 7. The model has performed reasonably well on the C-NMC-2019 image dataset, as reflected in the accuracy and average accuracy. The accuracies range from 98.89% to 99.80%, where the highest achieved by this model is around 99.80%.

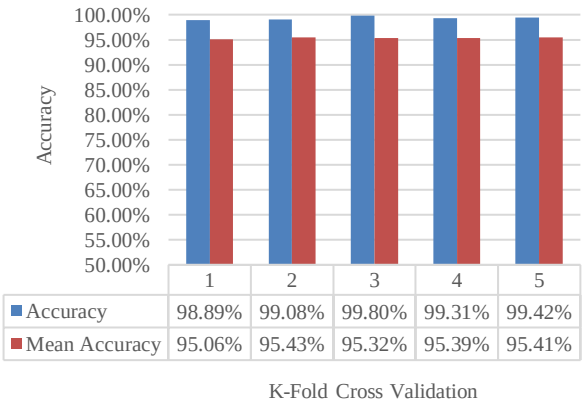

Figure. 4 Proposed Method Training Accuracy

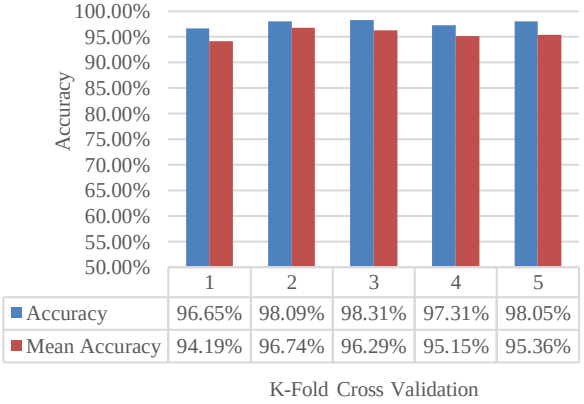

Figure. 5 Our Proposed Method Validation Accuracy

The mean accuracies are between 95.06% and 95.43%, as described in Fig. 4. A minimal variation between the models' analytical values implies that it has provided an almost consistent performance, while a small range under mean accuracy indicates stable experimental results. Thus, the ideal combination would be a model having an accuracy of 99.08% with a mean accuracy of 95.43%.

3.2 Validation results

Our proposed method has passed the validation results. This is the best model score for 98.31% of accuracy, and the average test will be as high as 96.74%. This shows no variation in the accuracy and mean of accuracies outcomes implies a consistent model performance for all training set types. From the above analysis, we can observe that models are well-trained, generalize to new data, and can be deployed reliably. Further steps may be fine-tuning hyper-parameters and maybe making use of an ensemble method for an even more robust and more accurate model.

Based on Figs. 4 and 5, we can state that models have good results in the training and validation phases. It is a typical pattern: the accuracy should slightly drop from training to validation, showing that your model could generalize well. This model which demonstrated the best training performance with 99.08% accuracy and 95.43% mean. The model is also consistent concerning validation- at 98.09% accuracy and mean, it becomes an excellent generalized model for deployment. Additional validation and even ensemble techniques may be adopted to improve stability and accuracy.

3.3 Model evaluation using training and validation datasets

Over the training and validation phases, the best performance model illustrated eminent accuracy, as

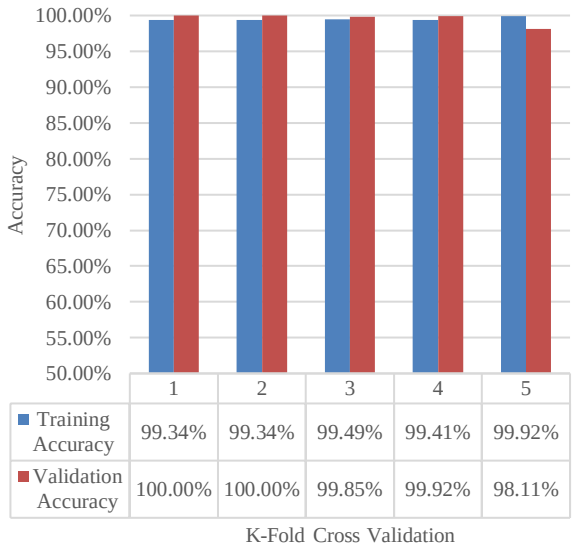

Figure. 6 Our Proposed Method Validation Accuracy

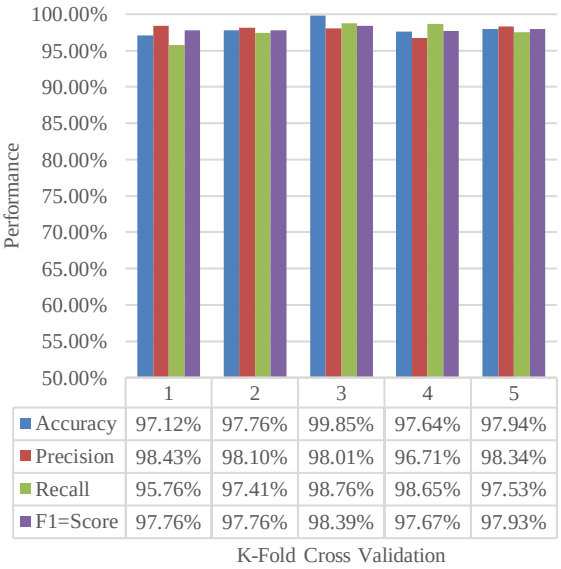

Figure. 7 Our Proposed Method: The Best Performance of the Testing Results Using 0.001 Learning Rate and Ten Epochs

always shown by its validation results. The high training accuracies (99.34% to 99.92%) and near perfect validation accuracies (98.11% to ~100%, only the third model has one sample wrong) suggest an ideal generalization of the model on processes new data, unseen by it during its development phase, as seen in Fig. 6. The close-to-perfect validation accuracies make us realize that a model is robust and reliable for practical applications. The other validation fell 98.11%, which is significantly less, and we can ignore it as the combined performance of the model on both datasets was top-notch.

3.4 Testing results

We produced the testing experimental results: accuracy, precision, recall, and f1-score, as shown in Fig. 7. These results confirm that the model performs well during training, validation, and testing. On the first fold, the results show a high recall. It indicates that the model captures the positive cases with precision slightly lower, suggesting some false positives. We can also strengthen all performance on the second fold with few false positives and negatives. We also identify high recall but lower precision on the third fold. Similar with the second fold, we demonstrated well-balanced precision and recall, indicating effective identification of true positives with minimal errors in the fourth fold. The last fold displays High precision and recall. It shows a solid ability to identify positive cases and maintain overall accuracy correctly.

The classification model performed very well, with accuracy ranging between 97.12% and 99.85% in each fold, indicating high accuracy. The 3rd fold achieved the highest accuracy of 99.85%, indicating that the model could correctly recognize most samples. This performance shows that setting a learning rate of 0.001 and using ten epochs is optimal, as shown in Fig. 7.

The precision value is high in all folds, reaching 98.43% in the first fold, indicating the model's ability to minimize false positive predictions. In addition, the recall was highest in the 3rd fold at 98.76%, indicating that the model could identify almost all positive samples, although there were some minor variations between the other folds.

The F1-score was also high and consistent, with a range of 97.67% to 98.39%, indicating a good balance between precision and recall. It shows that the model has stable performance in detecting positive classes and maintaining overall accuracy in each fold. Overall, these results show that the model is highly reliable and has a high potential to be applied to similar data.

3.5 Receiver operating characteristic of the testing result

We have plotted the experimental results in the form of ROC and calculated the AUC value to identify the performance of our proposed method, as shown in Fig. 8. The plotted results show that our proposed method has produced an AUC of 0.97 for all folds, as shown in Fig. 8. The high and stable AUC value indicates that the proposed method performs well, as our proposed method can distinguish positive and negative classes.

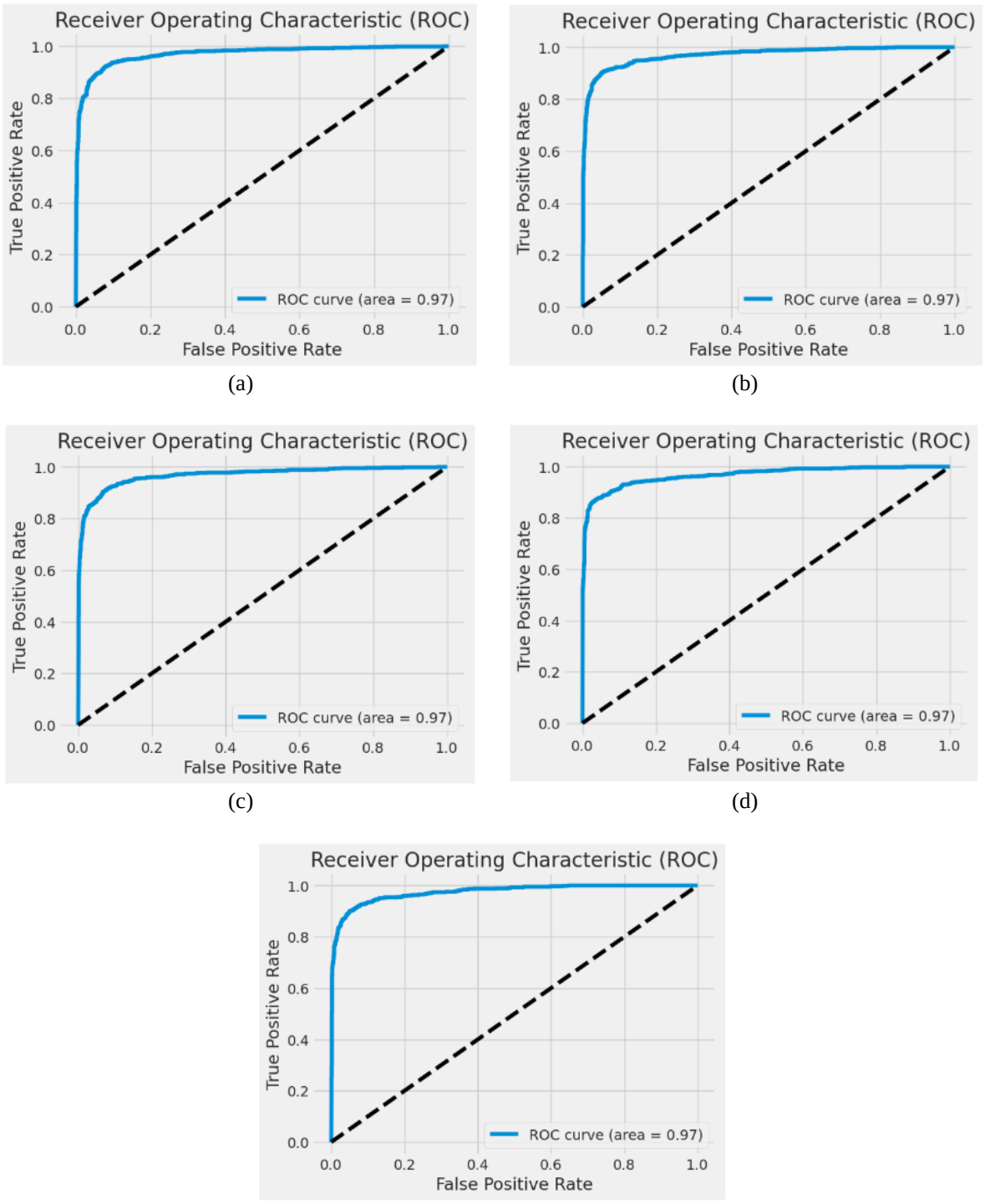

Figure. 8 The ROC and AUC Graphic of Our Experimental Results: (a) $K = 1$, (b) $K = 2$, (c) $K = 3$, (d) $K = 4$, and (e) $K = 5$

3.6 Comparison of results

To measure achievement level to other methods, we have compared our accuracy results with other techniques that use the same leukemia image dataset: C-NMC-2019, i.e., Lightweight EfficientNet-B3 [24], EfficienNetB0 [25], CNN Based ECA Module [26], Vision Transformer [27], Majority Voting Technique [28], and CNN Model based Tversky Loss Function [29]. Our proposed method has produced the best result with 99.85% accuracy. Its achievement outperformed the others, i.e., Lightweight EfficientNet-B3 [24], EfficienNetB0 [25], CNN-Based ECA Module [26], Vision Transformer [27], Majority Voting Technique [28], CNN Model-based Tversky Loss Function [29], as shown Table 2.

Table 2. Comparison Result Between Our Accuracy Results with the Other Methods

Methods	Accuracy (%)
Lightweight EfficientNet-B3 [24]	99.31
EfficienNetB0 [25]	95.18
CNN-Based ECA Module [26]	91.1
Vision Transformer [27]	88.2
Majority Voting Technique [28]	98.5
CNN Model-based Tversky Loss Function [29]	99.01
Our Proposed Method	99.85

Our results have delivered 99.85% accuracy. The best accuracy of our experimental results occurred on the 3-fold cross-validation, E=10, and $\alpha = 0.001$. Our complete results can be seen in Fig. 7. Our best achievement occurred on the 3rd fold, which are 99.85%, 98.01%, 98.76%, and 98.39% for accuracy, precision, recall, and F1-score, respectively. It indicates that our proposed method is visible to implement on the real word, such as the biomedical field to help a leukemia specialist doctor.

4. Conclusion

The results of hyperparameter tuning using the Grid Search method from learning rate and epoch, which are $(0.001,0.0001,0.0000 \subseteq \alpha\}$ and $\{E|(5,10,15) \subseteq E\}$ for the C-NMC-2019 dataset have obtained the best hyperparameters, namely learning rate (α=0.001), and epoch (E=10).

The optimal hyperparameters are realized on our proposed model. Summary: The following is the experimental result for the training and validation phases on the C-NMC-2019 dataset. During the training phase, the obtained accuracy fluctuated between 99.34% and 99.92%. In the validation phase, the achieved accuracy was between 98.11% and ~100%. The highest accuracy of 99.80% was obtained with an average accuracy of 95.43%. The best average validation accuracy was 96.74%, with the highest validation accuracy being 98.31%.

A small decrease in the next model, to 98.11% in validation, is very reasonable. From the results of validation, one can understand that the model shows good generalization ability by the tiny decrease in accuracy from the training phase to the validation phase. Results like these prove the model has reliable ability to be applied to new data outside of the data used during the process of training.

As indicated by the testing results, the proposed method performs very well in terms of classification for leukemia, with accuracy falling between 97.12% and 99.85% for each fold. On the third fold, the model reached the highest accuracy of 99.85%, which manifests that the model can classify most of the samples rightly. From the results, it could be figured out that the learning rate setting at 0.001 with 10 epochs is the most optimal configuration of the proposed model.

Precision and recall values are also consistently high on all folds, reaching 98.43% on the first fold and the highest recall of 98.76% on the third fold. The presented data demonstrates that the model can keep false positive predictions low and correctly classify most of the positive samples. The F1-score is between 97.67% and 98.39%, which shows that there is a good balance between precision and recall and stable performance in the detection of positive classes.

The ROC curve is a plotted curve; it is, therefore, shown in the following that the value of AUC in all folds is 0.97, indicating the model has great power in separating positive and negative samples of the class. The accuracy results obtained are higher than other methods applied to the same dataset, C-NMC-2019: Lightweight EfficientNet-B3, EfficienNetB0, CNN-Based ECA Module, Vision Transformer, Majority Voting Technique, and CNN Model-based Tversky Loss Function.

Conflicts of Interest

We would like to declare no conflicts of interest.

Author Contributions

We have finished our fundamental research in 2024. Our research has been conducted some persons. Each person of our team has conducted their work: Conceptualization: Arif Muntasa, Rima Tri Wahyuningrum, and Husni. Methodology: Arif Muntasa, Rima Tri Wahyuningrum, and Husni. Literature review: Arif Muntasa, Muhammad Yusuf, Yuli Panca Asmara, and Deshinta Arrova Dewi. Image Data preparation: Zabrina Tuzzahra and Alisa Sugiarti. Architecture model: Muhammad Yusuf, Deshinta Arrova Dewi and Alisa Sugiarti. Algorithm Designer: Arif Muntasa, Rima Tri Wahyuningrum, Wayan Firdaus Mahmudi, Yuli Panca Asmara, and Deshinta Arrova Dewi, and Husni. Software Implementation: Zabrina Tuzzahra and Alisa Sugiarti. Experimental designer: Arif Muntasa, Rima Tri Wahyuningrum, and Husni. Result analysis: Arif Muntasa, Muhammad Yusuf, and Rima Tri Wahyuningrum. Investigation: Muhammad Yusuf, Wayan Firdaus Mahmudi, Yuli Panca Asmara, and Abdelwahed Motwakel. Writing original draft preparation: Arif Muntasa, Wayan Firdaus Mahmudi, and Abdelwahed Motwakel. Writing review and editing: Muhammad Yusuf, Yuli Panca Asmara, and Deshinta Arrova Dewi. Supervision: Arif Muntasa,

Rima Tri Wahyuningrum, and Husni. Project administration: Rima Tri Wahyuningrum.

Acknowledgments

We would like to thank to Ministry of Education, Culture, Research, And Technology-Directorate General-Higher Education, Research; Technology and Prince Sattam bin Abdulaziz University; and Inti International University

1 This research is funded by Ministry of Education, Culture, Research, And Technology-Directorate General-Higher Education, Research, And Technology: fundamental research grants, project numbers 008/UN46.4.1/PT.01.03/2024, and 101/E5/PG.02.00. PL/2024.

2 This study is supported via funding from Prince Sattam bin Abdulaziz University project number (PSAU/2024/R/1445)

3 This study is also supported by Inti International University, Nilai 71800 Negeri Sembilan, Malaysia

References

[1] P. H. Kasani, S. W. Park, and J. W. Jang, "An aggregated-based deep learning method for leukemic B-lymphoblast classification", *Diagnostics*, Vol. 10, No. 12, 2020.

[2] L. Putzu and C. Di Ruberto, "White Blood Cells Identification and Classification from Leukemic Blood Image", In: *Proc. of International Work-Conference on Bioinformatics and Biomedical Engineering*, pp. 18-20. 2013.

[3] S. Mohapatra, S. S. Samanta, D. Patra, and S. Satpathi, "Fuzzy based blood image segmentation for automated leukemia detection", In: *Proc. of 2011 International Conference on Devices and Communications, ICDeCom 2011 - Proceedings*, pp. 1-5, 2011.

[4] D. KUMAR, N. JAIN, A. KHURANA, S. MITTAL, S. C. SATAPATHY, and R. SENKERIK, "Automatic Detection of White Blood Cancer From Bone Marrow Microscopic Images Using Convolutional Neural Networks", *Spec. Sect. Emerg. Deep Learn. Theor. METHODS Biomed. Eng.*, Vol. 8, No. August, pp. 142521-142531, 2020.

[5] A. M. Abdeldaim, A. T. Sahlol, M. Elhoseny, and A. E. Hassanien, "Computer-Aided Acute Lymphoblastic Leukemia Diagnosis System Based on Image Analysis", In: *Proc. of Advances in Soft Computing and Machine Learning in Image Processing*, A. E. Hassanien and D. A. Oliva, Eds. Cham: Springer International Publishing, pp. 131-147, 2018.

[6] L. Putzu, G. Caocci, and C. Di, "Artificial Intelligence in Medicine Leucocyte classification for leukaemia detection using image processing techniques", *Artif. Intell. Med.*, Vol. 62, No. 3, pp. 179-191, 2014, doi: 10.1016/j.artmed.2014.09.002.

[7] S. Mishra, B. Majhi, P. K. Sa, and L. Sharma, "Gray level co-occurrence matrix and random forest based acute lymphoblastic leukemia detection", *Biomed. Signal Process. Control*, Vol. 33, pp. 272-280, 2017, doi: 10.1016/j.bspc.2016.11.021.

[8] S. Gehlot, A. Gupta, and R. Gupta, "SDCT-AuxNetθ: DCT augmented stain deconvolutional CNN with auxiliary classifier for cancer diagnosis", *Med. Image Anal.*, Vol. 61, p. 101661, 2020, doi: 10.1016/j.media.2020.101661.

[9] D. R. Putri, A. Jamal, and A. A. Septiandri, "Acute Lymphoblastic Leukemia Classification in Nucleus Microscopic Images using Convolutional Neural Networks and Transfer Learning", In: *Proc. of 2021 2nd International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, pp. 1-6, 2021, doi: 10.1109/AiDAS53897.2021.9574176.

[10] L. H. S. Vogado, R. M. S. Veras, F. H. D. Araujo, R. R. V Silva, and R. T. Aires, "Engineering Applications of Artificial Intelligence Leukemia diagnosis in blood slides using transfer learning in CNNs and SVM for classification", *Eng. Appl. Artif. Intell.*, Vol. 72, No. 2017, pp. 415-422, 2018, doi: 10.1016/j.engappai.2018.04.024.

[11] L. H. S. Vogado, R. M. S. Veras, F. H. D. Araujo, R. R. V. Silva, and K. R. T. Aires, "Leukemia diagnosis in blood slides using transfer learning in CNNs and SVM for classification", *Eng. Appl. Artif. Intell.*, Vol. 72, pp. 415-422, 2018.

[12] A. Muntasa and M. Yusuf, "Multi Distance And Angle Models Of The Gray Level Co-Occurrence Matrix(Glcm) To Extract The Acute Lymphoblastic Leukemia (All) Images", *Int. J. Intell. Eng. Syst.*, Vol. 14, No. 6, pp. 357-368, 2021, doi: 10.22266/ijies2021.1231.32.

[13] A. Muntasa, A. Motwakel, R. T. Wahyuningrum, Z. Tuzzahra, M. Yusuf, and W. F. Mahmudi, "A Pyramid Model of Convolutional Neural Network to Classify Acute Lymphoblastic Leukemia Images", *Int. J. Intell. Eng. Syst.*, Vol. 15, No. 6, pp. 576-588, 2022, doi: 10.22266/ijies2022.1231.51.

[14] A. Muntasa., R. T. Wahyuningrum, and D. Nafisah, "A New Model : Commutative Hypercomplex - Convolutional Neural Network

to Classify Acute Lymphoblastic Leukemia Images", *Int. J. Intell. Eng. Syst.*, Vol. 16, No. 5, pp. 208-225, 2023, doi: 10.22266/ijies2023.1031.19.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks", *Commun. ACM*, Vol. 60, No. 6, pp. 84-90, 2017, doi: 10.1145/3065386.

[16] S. (2014). Deepika, Jaswal., Sowmya., K., P., "Image Classification Using Convolutional Neural Networks", *Int. J. Sci. Eng. Res.*, Vol. 5, No. 6, pp. 1661-1668, 2014.

[17] R. Arif, S. Akbar, A. B. Farooq, S. Ale Hassan, and S. Gull, "Automatic Detection of Leukemia through Convolutional Neural Network", In: *Proc. of 2022 International Conference on Frontiers of Information Technology (FIT)*, 2022, pp. 195-200, doi: 10.1109/FIT57066.2022.00044.

[18] Q. Ul Ain, S. Akbar, S. A. Hassan, and Z. Naaqvi, "Diagnosis of Leukemia Disease through Deep Learning using Microscopic Images", In: *Proc. of 2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2)*, 2022, pp. 1-6, doi: 10.1109/ICoDT255437.2022.9787449.

[19] T. Shawly, A. A., and Alsheikhy, "Biomedical Diagnosis of Leukemia Using a Deep Learner Classifier", *Comput. Intell. Neurosci.*, Vol. 2022, No. 1, pp. 1-9, 2022.

[20] J. Su, S. Liu, and J. Song, "A segmentation method based on HMRF for the aided diagnosis of Acute Myeloid Leukemia", *Comput. Methods Programs Biomed.*, Vol. 152, pp. 115-123., 2017, doi: 10.1016/j.cmpb.2017.09.011.

[21] K. (2023). Ibrahim, H., Al-Kharsan., Zaid, Nidhal, "Leukemia Classification using a Convolutional Neural Network of AML Images", *Malaysian J. Fundam. Appl. Sci.*, Vol. 19, No. 3, 2023, doi: 10.11113/mjfas.v19n3.2901.

[22] A. Abhishek, S. D. Deb, R. K. Jha, R. Sinha, and K. Jha, "Classification of Leukemia using Fine Tuned VGG16", In: *Proc. of 2023 International Conference on Signal Processing, Computation, Electronics, Power and Telecommunication (IConSCEPT)*, pp. 1-5, 2023, doi: 10.1109/IConSCEPT57958.2023.10170285.

[23] S. Amutha, "VGGNet-Cnn based classification of white blood cell leukemia with efficient salp swarm optimization algorithm", *J. Intell. Fuzzy Syst.*, Vol. 44, No. 3, pp. 6973-6989, 2023.

[24] A. Batool and Y.-C. Byun, "Lightweight EfficientNetB3 Model Based on Depthwise Separable Convolutions for Enhancing Classification of Leukemia White Blood Cell Images", *IEEE Access*, Vol. 11, pp. 37203-37215, 2023, doi: 10.1109/ACCESS.2023.3266511.

[25] et al. Z. Gao, J. Chung, M. Abdelrazek, S. Leung, W. K. Hau, Z. Xian, "Privileged modality distillation for vessel border detection in intracoronary imaging", *IEEE Trans. Med. Imag*, Vol. 39, No. 5, pp. 1524-1534, 2020.

[26] M. Z. Ullah *et al.*, "An Attention-Based Convolutional Neural Network for Acute Lymphoblastic Leukemia Classification", *Appl. Sci.*, Vol. 11, No. 22, p. 10662, 2021.

[27] A. T. and H.-J. Y. P. Cho, S. Dash, "Image transformers for classifying acute lymphoblastic leukemia", In: *Proc. of SPIE*, pp. 633-639, 2022.

[28] D. B. and A. R. M. Ghaderzadeh, A. Hosseini, F. Asadi, H. Abolghasemi, "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", *Sci. Progr.*, Vol. 2022, No. January, pp. 1-8, 2022.

[29] J. V. G. and S. D. S. Ansari, A. H. Navin, A. B. Sangar, "A customized efficient deep learning model for the diagnosis of acute leukemia cells based on lymphocyte and monocyte images", *Electronics*, Vol. 12, No. 2, pp. 322-330, 2023.