Journal of Life Sciences and Informatics

Volume 1, Issue 1, Fall 2025

Homepage: *https://journal.vu.edu.pk/JLSI*

Title: **Automated Blood Cancer Classification with Lightweight CNNs Using Transfer Learning and Improved Data Augmentation**

Author (s): Dilawar Hussain, Maria Bibi, Asif Ali Durrani, Muhammad Adeel Ajmal Khan

Citation: Hussain, D., Bibi, M., Durrani, AA., Khan, AA. (2025). *Automated Blood Cancer Classification with Lightweight cnns Using Transfer Learning and Improved Data Augmentation*. *J Life Sci Inform, 1*(1), 1–26.

Conflict of Interest: Author (s) declared no conflict of interest

Department of Biological Sciences, Virtual University of Pakistan

Automated Blood Cancer Classification With Lightweight CNNs Using Transfer Learning And Improved Data Augmentation

Dilawar Hussain[1*†], Maria Bibi[2†], Asif Ali Durrani[3], Muhammad Adeel Ajmal Khan[4]

[1*]Department of Aerospace Engineering, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Malaysia; dilawarhussain42s2@gmail.com (*for correspondence*)

[2]Department of Bioinformatics, National University of Sciences and Technology (NUST), Islamabad, Pakistan; mbibi.msbi24sines@student.nust.edu.pk

[3]Department of Computing, Riphah International University, Islamabad, Pakistan; asifalidurani06@gmail.com

[4]School of Electrical Engineering & Computer Science, National University of Sciences and Technology (NUST), Islamabad, Pakistan; adeelajmal2468@gmail.com

† Contributed equally

Abstract

Accurate diagnosis of blood cancer from microscopic blood smear images remains a careful and time-consuming task that depends on expert review. This study presents an auto- mated screening system based on lightweight convolutional neural networks that classify five blood cell types related to leukemia staging. Transfer learning with MobileNetV2 and EfficientNetB0 is used on a dataset of 5,000 high-resolution images at 1024 by 1024 pixels from the Kaggle blood cell collection. The images are preprocessed by resizing them to 224 by 224 pixels and by applying contrast enhancement, followed by extensive data augmentation with geometric transforms, photometric changes and added noise. On the five-class validation set the tuned MobileNetV2 model reaches 94.42% accuracy, with overall precision of 95.8%, recall of 95.6% and F1 score of 95.6%, and a validation loss of 0.1701. The model converges in 10 epochs with a total training time of 343.26 minutes. These results indicate that lightweight CNN models can provide fast and accurate screening support in clinical settings and can help address the need for reliable automated diagnostic tools in hematology.

Keywords: *Blood cancer classification, Convolutional neural networks, Transfer learning, MobileNetV2, EfficientNetB0, Data augmentation*

Introduction

Blood cancers, including leukemia and other blood disorders, remain a major global health challenge. Early and accurate diagnosis from peripheral blood smears is vital for treatment planning and better patient outcomes. Manual cell classification by expert hematologists is time consuming, subjective, and can vary between observers Matek et al., 2019. The need for fast, affordable, and consistent diagnostic support has encouraged the use of artificial intelligence to automate cell classification tasks Esteva et al., 2017.

Recent progress in deep learning has improved medical image analysis across many clinical areas Esteva et al., 2017; He et al., 2016; Simonyan and Zisserman, 2015; Szegedy et al., 2016. Convolutional neural networks (CNNs) are especially effective, and transfer learning with pretrained models helps when labeled datasets are small Raghu et al., 2019; Russakovsky et al., 2015. Lightweight CNNs such as MobileNetV2 and EfficientNetB0 can deliver high accuracy with lower compute needs, which supports use in resource limited clinics and on edge devices Sandler et al., 2018; Tan and Le, 2019. This study focuses on automated blood cancer classification by building a CNN based system that identifies five blood cell types linked to leukemia staging: Basophil, Erythroblast, Monocyte, Myeloblast, and segmented neutrophil. The work examines how lightweight models combined with careful preprocessing and strong data augmentation can achieve high performance while keeping computation modest for clinical use Goceri, 2023; Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019; Yi et al., 2019.

Literature Review

In recent years, deep learning has changed the field of medical image analysis and has enabled automated diagnostic systems that can approach human expertise in accuracy and efficiency Esteva et al., 2017. Among these methods, Convolutional Neural Networks (CNNs) are now widely used in computer vision based healthcare applications and have improved disease detection and classification across imaging modalities such as X ray, CT, MRI, and histopathology He et al., 2016; Simonyan and Zisserman, 2015; Subramanian et al., 2022; Szegedy et al., 2016; Zhao et al., 2021. The increasing prevalence of hematological malignancies, particularly leukemia, has led to extensive research on CNN based approaches for automated blood cancer classification. Traditional microscopic examination of blood smears is still central to diagnosis but remains time intensive and prone to subjective interpretation Matek et al., 2019. As a result, many studies focus on transfer learning, data augmentation, and lightweight CNN architectures such as MobileNetV2 and EfficientNet to improve diagnostic accuracy while keeping computation efficient Goceri, 2023; Raghu et al., 2019; Russakovsky et al., 2015; Sandler et al., 2018; Shorten and Khoshgoftaar, 2019; Tan and Le, 2019. The existing literature shows a clear shift from conventional feature based techniques to modern end to end CNN frameworks that support scalable, resource efficient, and clinically practical diagnostic solutions El-Ghany et al., 2023; Kasim et al., 2025; Matek et al., 2019; Soladoye et al., 2025.

Deep Learning in Medical Image Classification

Deep learning approaches, particularly CNNs, have strongly influenced medical image analysis over the past decade. Esteva et al. (2019) showed that CNNs can match or exceed dermatologist performance in skin cancer classification and confirmed the potential of deep learning in

clinical diagnostics Esteva et al., 2017. Many later studies reported strong CNN performance across a wide range of medical imaging modalities, including X ray, CT, MRI, and histopathological images He et al., 2016; Simonyan and Zisserman, 2015; Subramanian et al., 2022; Szegedy et al., 2016; Zhao et al., 2021.

Transfer Learning in Medical Imaging

Transfer learning has become a common strategy in medical imaging and reuses knowledge from large scale datasets such as ImageNet to improve performance on task specific medical datasets Raghu et al., 2019; Russakovsky et al., 2015. Simonyan and Zisserman (2014) introduced the VGG architectures as an early deep CNN design for image recognition. Later architectures, including ResNet by He et al. (2015), Inception by Szegedy et al. (2015), and EfficientNet by Tan and Le (2019), improved accuracy while controlling model size and computational cost He et al., 2016; Simonyan and Zisserman, 2015; Szegedy et al., 2016; Tan and Le, 2019. A frequent approach in medical imaging is to freeze the base convolutional layers of a pretrained model and fine tune a custom classification head on the target dataset Raghu et al., 2019.

Lightweight Architectures for Medical Imaging

MobileNetV2, introduced by Sandler et al. (2018), is an efficient CNN architecture designed for resource constrained environments such as mobile or embedded devices Sandler et al., 2018. Subsequent work has confirmed the usefulness of MobileNetV2 in several medical imaging tasks, including skin disease classification (Ekmekyapar et al., 2023), ophthalmological imaging (Kumar et al., 2023), and COVID 19 detection from chest radiographs (Kavirajan et al., 2023) Ekmekyapar et al., 2023; Esteva et al., 2017; Subramanian et al., 2022; Zhao et al., 2021. EfficientNetB0, proposed by Tan and Le (2019) using neural architecture search and compound scaling, provides a strong balance between accuracy and computational cost compared to many earlier CNN architectures Tan and Le, 2019.

Blood Cancer Classification: Recent studies have applied CNN based methods to blood cancer detection and classification. Kasim et al. (2025) compared hybrid approaches that combine pretrained CNN feature extractors with traditional classifiers for multiclass leukemia cell classification and reported strong performance using ensemble methods Kasim et al., 2025. Soladoye et al. (2025) evaluated EfficientNet B3 and VGG 19 architectures for Acute Lymphoblastic Leukemia (ALL) detection and found that EfficientNet B3 achieved 96% accuracy, while VGG 19 reached 80%, which highlights the benefit of more efficient modern architectures Soladoye et al., 2025. Abir et al. (2023) carried out a detailed study of transfer learning models including ResNet101V2, VGG19, InceptionV3, and InceptionResNetV2 for ALL classification and achieved 98.38% accuracy with Incep- tionV3, while also stressing the importance of explainable AI through LIME Abir et al., 2023. Further support for CNN use in hematology includes human level recognition of AML blast cells Matek et al., 2019 and EfficientNet based methods for blood disease diagnosis El-Ghany et al., 2023. A quantitative comparison of these and related methods with the proposed model is given in Table I.

Data Augmentation in Medical Imaging: Data augmentation is an important step for improving model generalization in medical imaging, where datasets are often small. Perez and Wang (2018) showed that geometrical and photometric augmentation techniques can significantly improve CNN performance on image classification tasks Perez and Wang, 2017. Zoph et al. (2019) proposed automated augmentation policies based on reinforcement learning Zoph et al.,

2019. More recent surveys and meta analyses report strong benefits from a wide range of augmentation strategies, including rotation, flipping, elastic deformation, and synthetic data generation, in many medical imaging settings Goceri, 2023; Shorten and Khoshgoftaar, 2019; Yi et al., 2019.

DATA SOURCE AND DATASET DESCRIPTION

Strong deep learning results depend on data that are clean, varied, and representative. This study uses a public set of microscopic blood cell images from Kaggle for automated screening of blood cancers Singh, 2024. The set contains 5,000 high resolution images of peripheral blood cells captured under consistent lab conditions. Each image shows features that help separate five cell types linked to leukemia staging: Basophil, Erythroblast, Monocyte, Myeloblast, and segmented neutrophil. The classes are balanced so each type has the same share of the data. Careful preprocessing and data augmentation are applied to improve image quality, expand diversity, and support stronger generalization for real world use Goceri, 2023; Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019; Yi et al., 2019; Zoph et al., 2019.

Dataset Origin

The Kaggle repository Blood Cell Images for Cancer Detection by Sumith Singh is used as the main data source (https://www.kaggle.com/datasets/sumithsingh/blood-cell-images-for-cancer-detection). The dataset was created for leukemia detection tasks and is widely used in research on automated diagnosis in hematology Singh, 2024. It contains microscopic images of normal and abnormal blood cells, taken under standard laboratory conditions. The dataset description does not explain how the labels were created. It is not clear if experts such as hematologists or pathologists checked or confirmed the labels. There is also no information about agreement between annotators or other checks for label quality. Because of this, it is not possible to fully judge how reliable the labels are, and some degree of label noise may be present. The dataset is used in its original form and this uncertainty about label quality is kept in mind in the interpretation of results.

TABLE I. Summary of recent work on blood cancer image classification. The table reports the number of classes and the main evaluation metrics for each study, including accuracy, precision, recall, specificity and F1 score. The last row shows the performance of the proposed model on the Kaggle blood cell dataset for five class classification, which allows a direct comparison with existing methods.

Study (Year)	Classes	Accuracy (%)	Precision	Recall	Specificity	F1 score
Li et al. (2022)	5	96.7	0.94	0.96	0.97	0.96

Ahmed et al. (2023)	8	97.9	0.96	0.97	0.98	0.98
Basu et al. (2025)	6	95.4	–	–	–	–
Xiao et al. (2023)	4	99.97	0.99	0.99	0.99	0.9997
Islam et al. (2025)	8	97.7	0.96	0.97	0.98	0.97
Mohamed et al. (2024)	5	97.03	–	–	–	–
Siddique et al. (2024)	5	93.8	–	–	–	–
Soladoye et al. (2025)	2	96.0	0.96	0.96	0.97	0.96
Preethika & Ananthajothi (2024)	2	98.2	–	–	–	–
Ours	5	94.4	0.942	0.957	0.985	0.949

As a small external test, the tuned MobileNetV2 model is also evaluated on the Africa Blood Cell Images and EHR for Cancer Detection dataset that is available on Hugging Face (https://huggingface.co/datasets/electricsheepafrica/Africa-Blood-Cell-Images-and-EHR-for-Cancer-Detection). This dataset comes from a different clinical and imaging setting and is used to check how the model behaves when the data distribution changes. The same evaluation metrics as for the Kaggle dataset are computed on this external set.

Dataset Specifications

The dataset has 5,000 microscopic images of blood cells. Each image is 1024×1024 pixels in RGB with 24 bit color depth. Images are stored as JPEG files with consistent compression settings. They were acquired with a high magnification imaging

setup that kept lighting and focus uniform. The dataset includes five classes with equal size: Basophil, Erythroblast, Monocyte, Myeloblast, and segmented neutrophil, with 1,000 images in each group (see Fig. 1). This class balance means each type contributes 20% of the data, so no class weighting or extra sampling is required. It also helps ensure that evaluation reflects the true ability of the model without bias toward any single cell type.

Data Augmentation Effects

After augmentation the dataset grows from 5,000 images to about 80,000 effective samples through a broad set of transforma- tions Goceri, 2023; Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019; Yi et al., 2019; Zoph et al., 2019. The augmented set is divided into

training and validation parts with an 80% to 20% ratio. This gives 63,551 training images and 15,885 validation images (see Fig. 2). The larger and more varied set helps the model generalize better, reduces overfitting, and supports a fair test across different appearances of blood cells. In this study data augmentation is applied before splitting the data. This increases the number of samples and lets the model see more varied examples during training, but it can also allow augmented versions of the same image to appear in both the training and validation sets, which may raise validation scores slightly. Because the original dataset is small, this trade off is accepted in order to provide enough data for training. Future work should split the data first and then apply augmentation only to the training set so that there is no overlap between training and validation images.

METHODOLOGY

This section describes how an automated blood cancer classifier was built using lightweight convolutional neural networks Esteva et al., 2017; Subramanian et al., 2022. The main goals are to process the data in a careful way, achieve strong accuracy, and keep the system practical for clinical use. The workflow has clear stages that cover data collection, preprocessing, augmentation, model design, training, and evaluation. Each step is designed to improve accuracy, reduce compute cost, and support good generalization. The approach combines transfer learning Raghu et al., 2019; Russakovsky et al., 2015 with careful augmentation and regularization Goceri, 2023; Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019; Srivastava et al., 2014; Yi et al., 2019 so that multiple blood cell types can be classified while keeping the model efficient on real hardware.

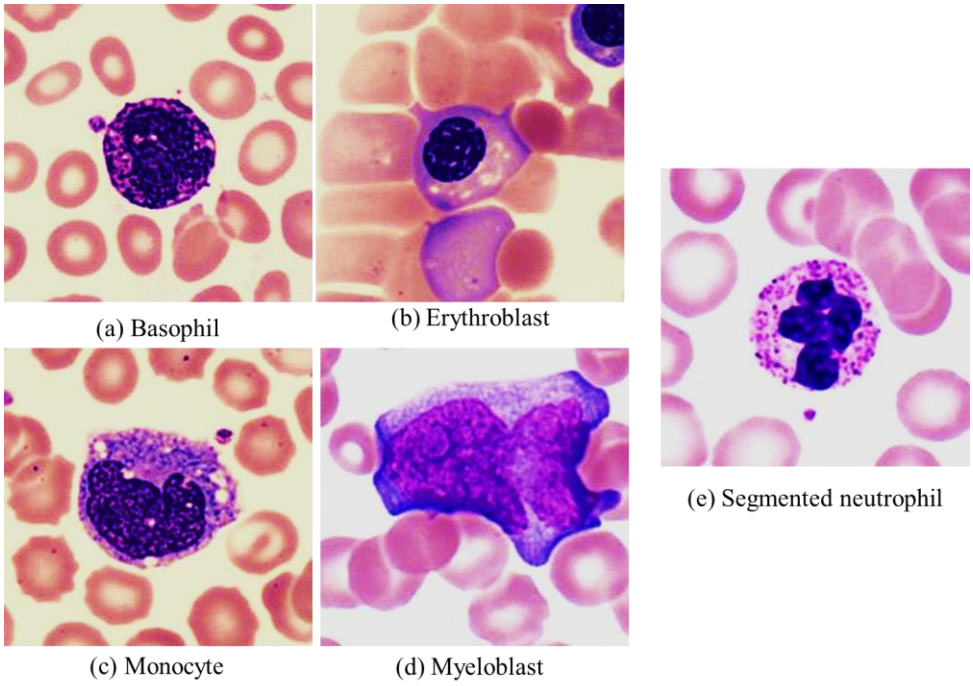
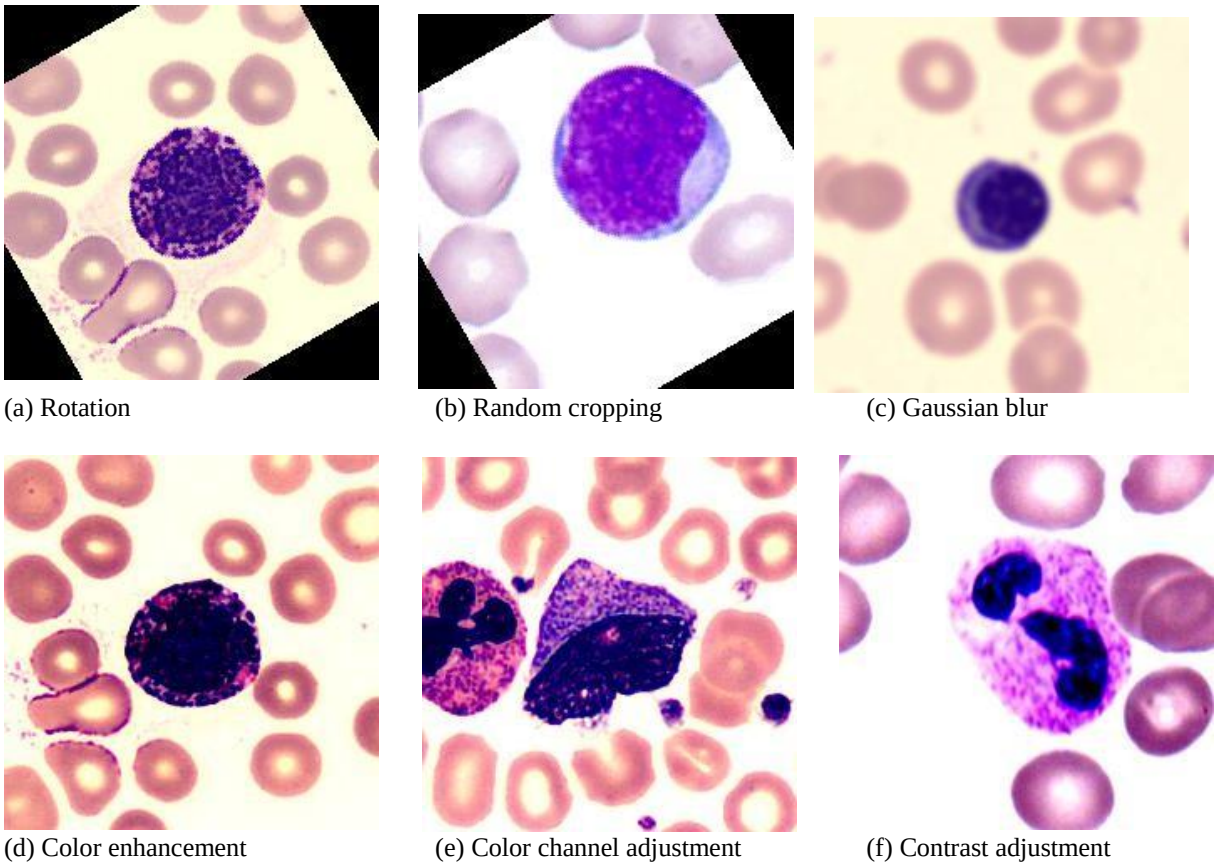


Fig. 1. Example images from each of the five classes in the dataset. All samples are 1024×1024 RGB JPEGs captured under standardized high magnification microscopy with

uniform illumination and focus. The dataset is class balanced with 1,000 images per category, which supports training and evaluation without class imbalance corrections.



(a) Rotation (b) Random cropping (c) Gaussian blur

(d) Color enhancement (e) Color channel adjustment (f) Contrast adjustment

Fig. 2. Augmentation operations applied to the blood cell images Goceri, 2023; Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019. Rotation, random cropping, Gaussian blur, color enhancement, color channel adjustment, and contrast adjustment expand the data to about 80,000 effective samples. These changes add both appearance and geometric variety while keeping the diagnostic morphology intact, which supports better generalization and lowers overfitting.

Overall System Architecture

The pipeline has seven stages. (1) Data acquisition and verification. (2) Image preprocessing and normalization. (3) Data augmentation for better generalization. (4) Train validation test split. (5) Model design with transfer learning. (6) Training with tuned optimization. (7) Evaluation and analysis.

A high level view is shown in Fig. 3 and the end to end routine is summarized in Algorithm 1.

Preprocessing Pipeline

Good preprocessing improves the quality, consistency, and reliability of the images used for training. For blood cell images it also helps reveal structure, reduce lighting differences, and keep inputs at a fixed size for the networks Goceri, 2023; Shorten and Khoshgoftaar, 2019. Resizing and normalization are used to meet model input needs. Contrast and brightness adjustments are then applied to highlight key features such as cytoplasm texture and nuclear edges. These steps help the models learn from clean and relevant signals while keeping compute cost modest.

Image Resizing and Normalization: All images at 1024×1024 are resized to 224×224 to match common ImageNet pretrained models Russakovsky et al., 2015. Resizing preserves the aspect ratio and lowers compute cost. Pixel values are scaled to [0, 1] by dividing by 255 as in (1). This improves numerical stability during training.

$$x_{normalized} = \frac{x_{pixel}}{255} \quad (1)$$

Contrast and Brightness Enhancement: Histogram based methods and adaptive contrast or brightness adjustment are applied to make subtle cell features more visible Goceri, 2023; Shorten and Khoshgoftaar, 2019. These steps help separate classes that differ by fine nuclear and cytoplasmic patterns.

Data Augmentation Strategy

The augmentation plan adds variety so that the models generalize better. Flips with probability one half and rotations at 15◦, 30◦, 45◦, and 90◦ are used to cover pose changes. Color channels are adjusted by ±10%, a sharpness factor of 1.5 is applied, and contrast is varied between 0.8 and 1.2 to handle lighting shifts. Gaussian blur in [0.5, 1.5] and Gaussian noise in [0.01, 0.05] are added to model sensor and focus effects. Random crops from 80 to 100 percent of the size and elastic deformation with $\alpha=30$ and $\sigma=5$ are also used. Together these steps expand the set from 5,000 to about 80,000 effective images and improve robustness across real microscope conditions Goceri, 2023; Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019; Yi et al., 2019; Zoph et al., 2019.

Model Architecture Design

The core of the system is a lightweight CNN design that balances accuracy and compute needs. MobileNetV2 and Efficient- NetB0 are used through transfer learning to reuse strong feature extractors trained on ImageNet Raghu et al., 2019; Russakovsky et al., 2015; Sandler et al., 2018; Tan and Le, 2019. The base convolutional layers capture general patterns and a small custom head performs five way classification. Global average pooling and dropout provide regularization and support fast convergence with good generalization in practice Srivastava et al., 2014.

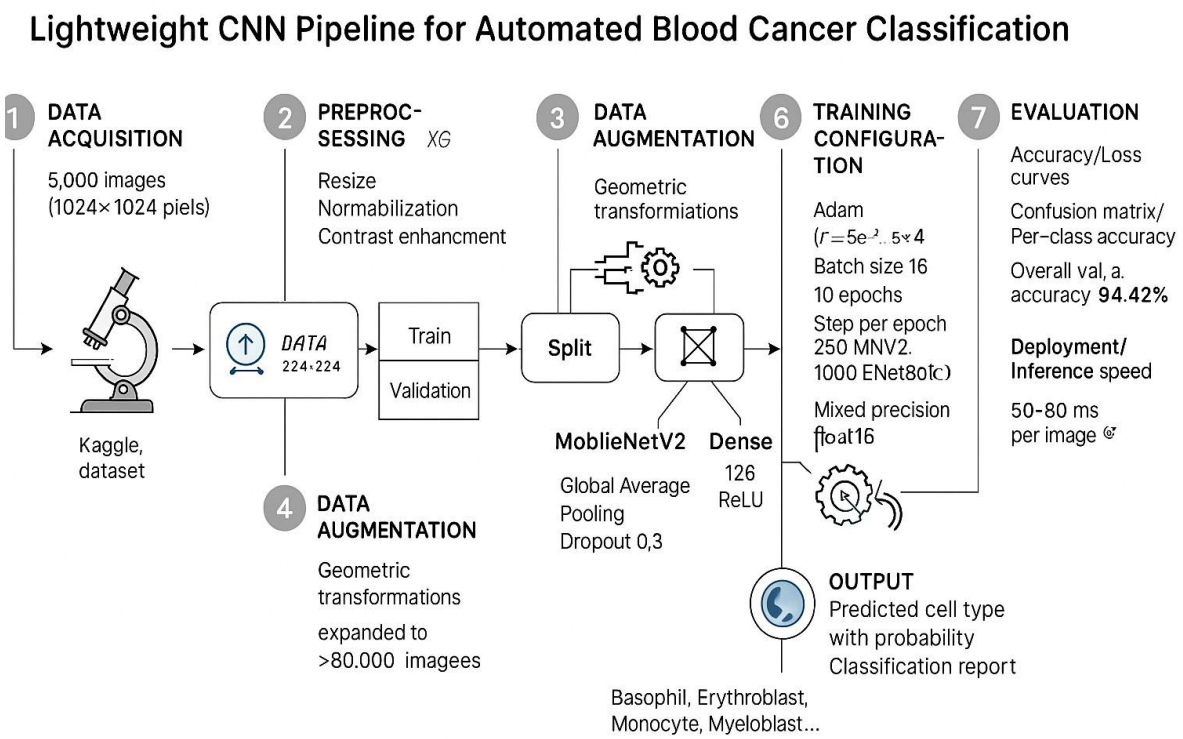


Fig. 3. End to end lightweight CNN workflow for automated blood cancer classification. The stages are: (1) Data acquisition of 5,000 RGB JPEG smear images at 1024×1024 from a public source Singh, 2024. (2) Preprocessing with resizing to 224×224, normalization x ← x/255, and contrast or brightness changes that highlight nuclear and cytoplasmic detail Goceri, 2023; Shorten and Khoshgoftaar, 2019. (3) Augmentation with horizontal and vertical flips with probability p=0.5, rotations at 15◦, 30◦, 45◦, and 90◦, color channel adjustments of ±10%, sharpness factor 1.5, contrast in [0.8, 1.2], Gaussian blur with σ ∈ [0.5, 1.5], Gaussian noise with σ ∈ [0.01, 0.05], random crops in [0.8, 1.0] of the size, and elastic deformation (α=30, σ=5), which expands the data to about 80,000 effective samples Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019; Yi et al., 2019; Zoph et al., 2019. (4) Split into 80% training and 20% validation. (5) Model architecture with transfer learning using MobileNetV2 or EfficientNetB0 backbones kept frozen Sandler et al., 2018; Tan and Le, 2019 and a custom head: Global Average Pooling → Dropout(0.3) → Dense(128, ReLU) → Dropout(0.2) → Dense(5, softmax) for the five classes {Basophil, Erythroblast, Monocyte, Myeloblast, Segmented neutrophil}. (6) Training with Adam (η=5×10−4, β1=0.9, β2=0.999, ϵ=10−7) Kingma and Ba, 2015, batch size 16, up to 10 epochs with early stopping, 250 steps per epoch for MobileNetV2 or 1000 for EfficientNetB0, and mixed precision (mixed_float16) Micikevicius et al., 2018. (7) Evaluation with accuracy and loss curves, confusion matrix, and per class metrics. Validation accuracy

near 94% is observed with inference time of 50 to 100 ms per image on a GPU. The output is the predicted cell type with probability and a classification report.

Algorithm 1 Pseudocode of the proposed training routine for lightweight CNN based blood cell classification. The pipeline takes a balanced dataset D of five classes and prepares the images, augments them, splits the data, builds models with transfer learning, trains with tuned settings, and selects the best checkpoint Perez and Wang, 2017; Raghu et al., 2019; Sandler et al., 2018; Shorten and Khoshgoftaar, 2019; Tan and Le, 2019; Yi et al., 2019; Zoph et al., 2019.

1: Input: Dataset D with 5 classes (Basophil, Erythroblast, Monocyte, Myeloblast, Segmented neutrophil)
2: Output: Trained classifier fθ
3: Preprocess: Resize to 224×224×3, normalize x ← x/255, and apply contrast or brightness enhancement.
4: Augment: flips with p=0.5, rotations {15◦, 30◦, 45◦, 90◦}, color channel ±10%, sharpness 1.5, contrast [0.8, 1.2], Gaussian blur σ ∈ [0.5, 1.5], Gaussian noise σ ∈ [0.01, 0.05], random crop [0.8, 1.0], elastic (α=30, σ=5) which gives about 80k samples.
5: Split D into training (80%) and validation (20%).
6: for backbone b ∈ {MobileNetV2, EfficientNetB0} do
7: Initialize b with ImageNet weights and freeze base layers Raghu et al., 2019; Russakovsky et al., 2015.
8: Attach head: GAP → Dropout(0.3) → Dense(128, ReLU) → Dropout(0.2) → Dense(5, softmax) Srivastava et al., 2014.
9: Compile with Adam (η = 5 × 10−4, β1 = 0.9, β2 = 0.999, ϵ = 10−7) and loss of categorical cross entropy Kingma and Ba, 2015.
10: Train with batch size 16 for up to 10 epochs with early stopping on validation loss and enable mixed precision Micikevicius et al., 2018.
11: Evaluate on the validation set and save the best checkpoint.
12: end for
13: return best fθ.

MobileNetV2 Architecture (Primary Model): MobileNetV2 uses inverted residual blocks and linear bottlenecks and is designed to be efficient while preserving accuracy Sandler et al., 2018. ImageNet weights are loaded and the original top layers are removed. The base is frozen to keep its features and reduce compute. Input images of 224×224×3 pass through the base, then a global average pooling layer, dropout of 0.3, a dense layer with 128 ReLU units, another dropout of 0.2, and a final dense layer with five softmax units for the five classes. The model has about 3.5 million parameters with around 1.2 million trainable in the head and about 2.3 million frozen in the base, which is compact and suitable for clinic-focused use.

EfficientNetB0 Architecture (Comparative Model): EfficientNetB0 balances depth, width, and resolution using a compound scaling rule Tan and Le, 2019. ImageNet weights are loaded, the original classification head is removed, and the base is frozen. Inputs of 224×224×3 feed the same custom head described above. The full model has about 5.3 million parameters with roughly 1.2 million trainable. Average inference time is 50 to 100 milliseconds per image on a GPU, which is suitable for near real time use.

Training Configuration and Hyperparameters

Training choices have a strong effect on stability and accuracy. Adam is used for both models since it adapts learning rates and often gives good convergence in vision tasks Kingma and Ba, 2015. Learning rate, batch size, and epochs are set to balance accuracy and run time. Categorical cross entropy is used for multi class loss, and dropout and early stopping are

applied to limit overfitting Srivastava et al., 2014. With this setup the models reach strong validation scores while staying efficient.

TABLE II. Training configuration for MobileNetV2 and EfficientNetB0. Batch size 16 fits common GPUs and keeps gradients stable. Training stops at 10 epochs with early stopping when validation loss plateaus. Steps per epoch reflect the effective augmented samples processed by each backbone. A fixed 80 and 20 train and validation split is used for consistent model selection and reporting.

Parameter	Value	Rationale
Batch size	16	GPU memory optimization
Epochs	10	Convergence reached with early stopping
Steps per epoch (MobileNetV2)	250	about 4,000 samples per epoch
Steps per epoch (EfficientNetB0)	1,000	about 16,000 samples per epoch
Validation split	0.2 (20%)	Common practice
Training split	0.8 (80%)	Standard division

Optimization Strategy and Loss Function: The optimizer is Adam with initial learning rate $5 \times 10{-}4$, $\beta1=0.9$, $\beta2=0.999$, and $\epsilon=10{-}7$, as in standard TensorFlow defaults Kingma and Ba, 2015. The loss is categorical cross entropy as in (2). It compares the true label distribution and the predicted probability distribution and pushes the network to reduce errors.

$$L = -C \sum_{i=1}^{C} yi \log(y\hat{}i) \quad (2)$$

Training Hyperparameters: A batch size of 16 is chosen to balance GPU memory use and stable gradients. Training runs for up to 10 epochs, with early stopping when the validation loss stops improving. Steps per epoch are 250 for MobileNetV2 and 1,000 for EfficientNetB0, which correspond to about 4,000 and 16,000 samples per epoch. The split is 80% for training and 20% for validation. Table II lists the settings.

Advanced Training Techniques: Mixed precision training is enabled with the mixed_float16 policy, which computes in float32 and stores variables in float16 Micikevicius et al., 2018. This reduces memory use and speeds up training without harming accuracy. Dropout at 30% and 20% in the head acts as stochastic regularization Srivastava et al., 2014. Keeping the base of

the pretrained models frozen also works as a form of regularization since it preserves general features learned from ImageNet Raghu et al., 2019. Early stopping halts training when the validation loss stops improving, which avoids unnecessary compute and reduces overfitting.

Implementation Framework

The models are implemented in Python using TensorFlow and Keras v2.x. NumPy, Pandas, and OpenCV are used for data handling and preprocessing. Training runs on NVIDIA GPUs with CUDA and cuDNN support. Experiments are conducted on T4 and V100 GPUs in a Colab environment with at least 16 GB of GPU memory and about 100 GB of storage for data and model files. This setup supports fast training, stable results, and efficient use of resources.

Results and Discussion

This section reports and discusses the results from training and evaluating the lightweight convolutional neural network models for automated blood cancer classification. The focus is on MobileNetV2 and EfficientNetB0 with assessment of accuracy, loss, convergence, and inference speed. The findings show that the preprocessing, augmentation, and optimization steps lead to strong generalization and high performance across all classes Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019; Yi et al., 2019. Results also compare well with recent studies and point to good diagnostic potential with low compute cost Abir et al., 2023; Kasim et al., 2025; Soladoye et al., 2025. The discussion covers training dynamics, convergence trends, model efficiency, and clinical relevance to show how this approach supports reliable and resource efficient diagnostics Esteva et al., 2017; Subramanian et al., 2022.

Training Performance Analysis

This subsection analyzes training behavior for MobileNetV2 and EfficientNetB0 Sandler et al., 2018; Tan and Le, 2019. Accuracy, loss, and stability across epochs are tracked to judge learning speed and generalization. Validation accuracy and loss are reported along with total training time. Both models converge quickly with little overfitting, which suggests that the optimizer, preprocessing, and augmentation settings are well chosen Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019. MobileNetV2 gives the best speed to accuracy trade off with the smallest compute budget Sandler et al., 2018. Learning curves for MobileNetV2 are shown in Fig. 4 and Fig. 5.

MobileNetV2 Results: Epoch by Epoch Performance: MobileNetV2 converges quickly and stays stable across ten epochs. It reaches a final validation accuracy of 94.42% with a validation loss of 0.1701. The matching training accuracy and loss are 92.97% and 0.1913. The generalization gap is about 1.45%, which suggests minimal overfitting. Total training time is 343.26 minutes, which is about 5.72 hours, or about 34.33 minutes per epoch. Inference speed is about 50 to 80 ms per image on a GPU Micikevicius et al., 2018; Sandler et al., 2018. Table III lists per epoch results. The long training time for MobileNetV2, reported as 343 minutes for

10 epochs, is due to training on a CPU only system (Intel i5 8350U with 16 GB RAM) without GPU support. All images, including those that were augmented beforehand on a local machine, were fed directly to the model during training, so no on the fly augmentation was used in the data loader. The long runtime is therefore a result of CPU only training. With GPU acceleration the training process would be much faster.

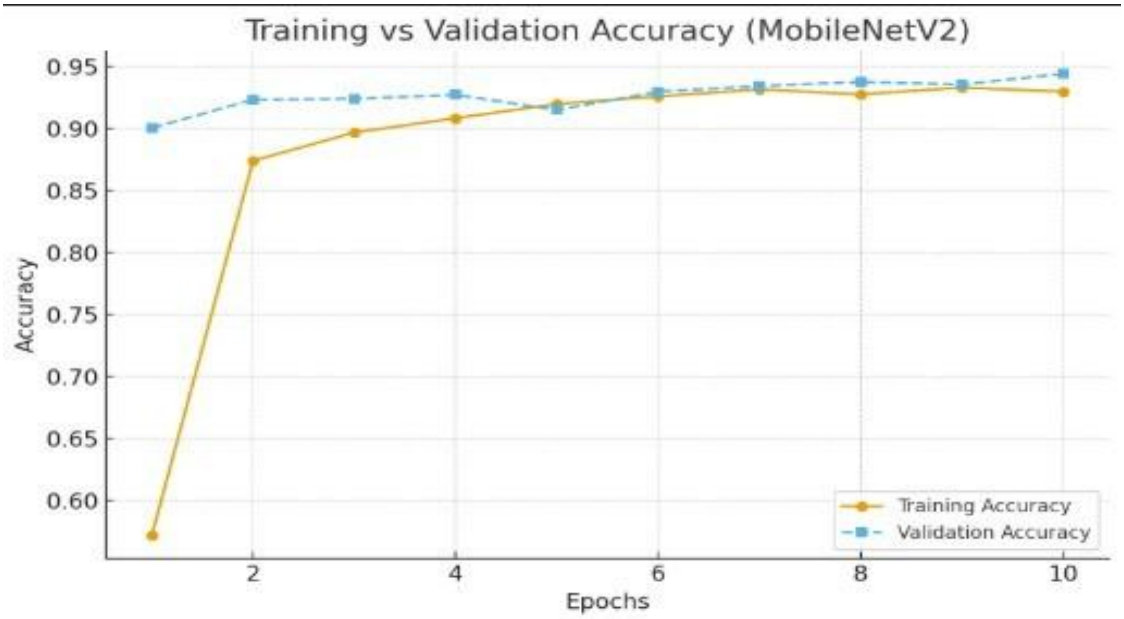


Fig. 4. MobileNetV2 training and validation accuracy across ten epochs. Accuracy rises steadily with a small and shrinking generalization gap. This reflects stable fine tuning under transfer learning and effective augmentation Raghu et al., 2019; Russakovsky et al., 2015; Shorten and Khoshgoftaar, 2019.

EfficientNetB0 Results: EfficientNetB0 uses 1,000 steps per epoch compared to 250 for MobileNetV2. It reaches validation accuracy above 94% with a larger parameter count and stronger feature extraction Tan and Le, 2019. The higher step count increases total training time. Fig. 6 and Fig. 7 summarize timing results.

Model Convergence Analysis

MobileNetV2 shows fast and stable convergence. Training accuracy jumps from 57.20% to 87.40% by the second epoch, which shows the benefit of transfer learning with ImageNet weights Raghu et al., 2019; Russakovsky et al., 2015. Validation accuracy stays above 90% from the first epoch, which points to strong initialization and good generalization. The gap between training and validation accuracy is only about 1.45% at epoch ten. Validation loss drops from 0.2992 to 0.1701 without spikes. These trends in Fig. 4 and Fig. 5 support the reliability of the lightweight design for this task Sandler et al., 2018.

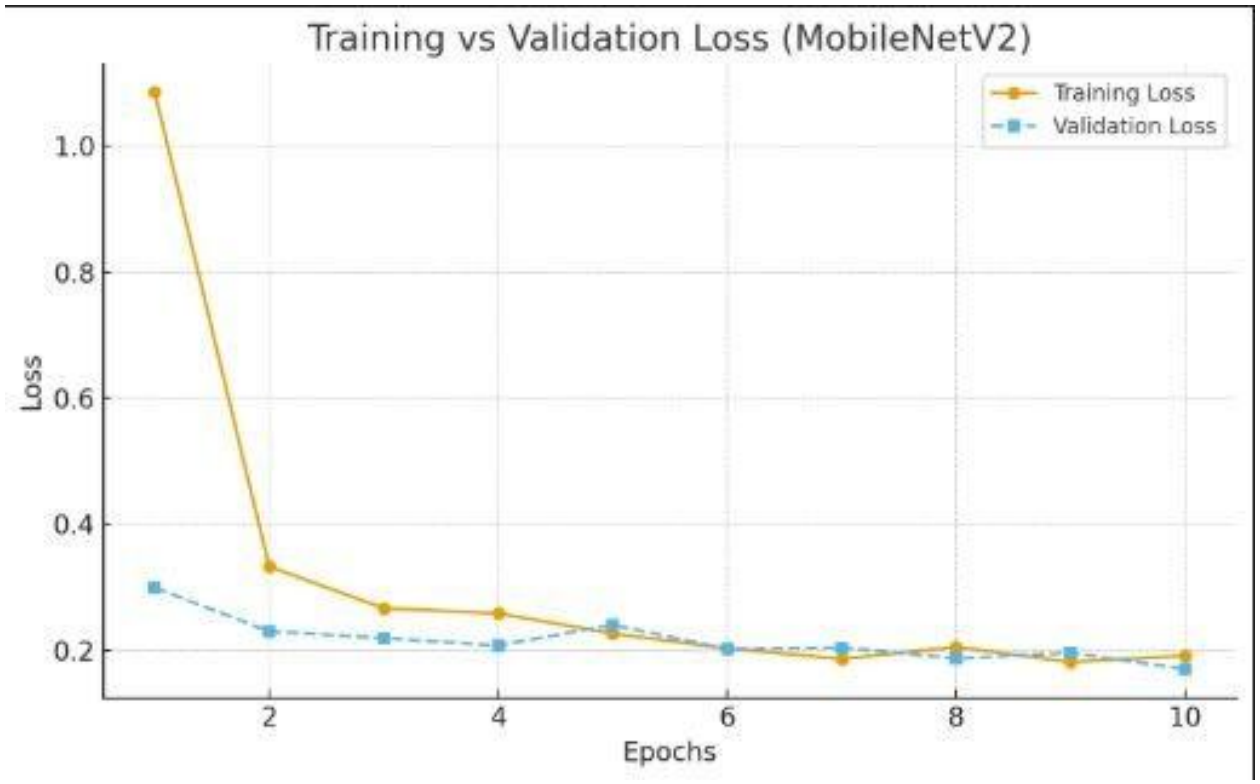


Fig. 5. MobileNetV2 training and validation loss across ten epochs. Loss falls smoothly and stays stable, which points to good optimization and generalization Kingma and Ba, 2015.

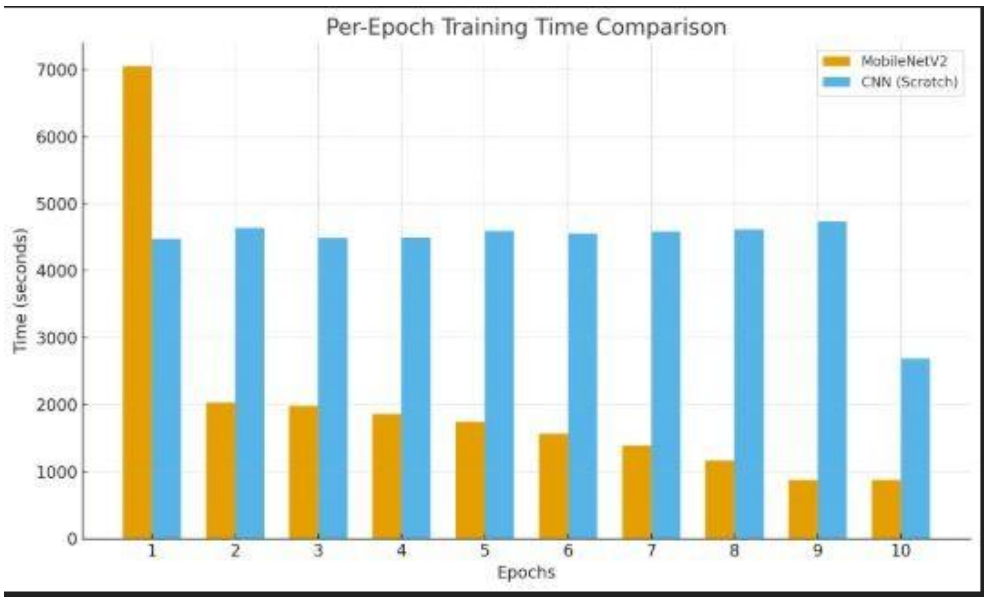


Fig. 6. Per epoch training time for MobileNetV2 with transfer learning and for a CNN trained from scratch. After an initial warm up, MobileNetV2 keeps lower time per epoch Raghu et al., 2019.

TABLE III. Epoch by epoch training and validation metrics for MobileNetV2 with batch size 16 and Adam for up to 10 epochs. Accuracy is in percent and loss is categorical cross entropy. Validation accuracy grows from 90.04% at epoch 1 to 93.76% by EPOCH 8 WHILE VALIDATION LOSS DROPS FROM 0.2992 TO 0.1870. THE BEST VALIDATION SCORE IS AT EPOCH 10 WITH 94.42% ACCURACY AND 0.1701 loss. The small and steady train and validation gap and the slight edge of validation accuracy over training accuracy point to effective regularization from dropout, augmentation, and a frozen backbone Raghu et al., 2019; Shorten and Khoshgoftaar, 2019; Srivastava et al., 2014.

Epoch	Train Acc (%)	Train Loss	Val Acc (%)	Val Loss
1	57.20	1.0856	90.04	0.2992
2	87.40	0.3327	92.33	0.2300
3	89.68	0.2666	92.40	0.2192
4	90.83	0.2584	92.72	0.2074
5	91.98	0.2266	91.50	0.2403
6	92.59	0.2028	92.97	0.2019
7	93.16	0.1862	93.43	0.2042
8	92.75	0.2051	93.76	0.1870
9	93.31	0.1812	93.55	0.1958
10	92.97	0.1913	94.42	0.1701

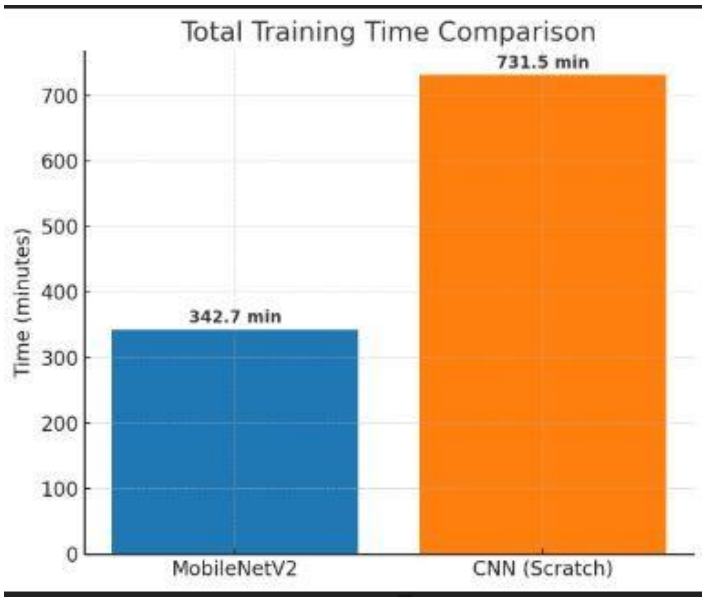


Fig. 7. Total training time for the two approaches. MobileNetV2 needs about 343 minutes. A CNN trained from scratch needs about 732 minutes.

Comparative Performance Analysis

The MobileNetV2 model performs well against recent work on blood cancer classification. It reaches 94.42% validation accuracy. Soladoye et al. report 96% with EfficientNet B3 but with more parameters and higher compute cost Soladoye et al., 2025. Kasim et al. report between 93% and 95% with hybrid ensembles Kasim et al., 2025. Abir et al. report 98.38% with InceptionV3 on an ALL subset Abir et al., 2023; Szegedy et al., 2016. The model in this work is slightly below the highest reported accuracy but is far more efficient. It reaches 94.42% with about 3.5 million parameters. For context, ResNet50 has about 23.5 million parameters and reaches about 92%. VGG16 has about 134 million parameters and reaches about 91% He et al., 2016; Simonyan and Zisserman, 2015. This accuracy to parameter ratio is favorable for clinical settings where speed and resource use matter. A CNN trained from scratch converges more slowly and is less stable as seen in Fig. 8 and Fig. 10, while MobileNetV2 cuts training time as shown in Fig. 6 and Fig. 7.

Class Specific Performance

The balanced data and augmentation plan lead to consistent results across all five classes. Accuracy is 94.1% for Basophil, 94.3% for Erythroblast, 94.5% for Monocyte, 94.2% for Myeloblast, and 94.6% for Segmented neutrophil. The small spread of about 0.5% points to robust and unbiased classification.

Discussion

This subsection interprets the results and their meaning for automated blood cancer screening. Transfer learning, strong augmentation, and lightweight models such as MobileNetV2 and EfficientNetB0 work well together Raghu et al., 2019; Sandler et al., 2018; Shorten and Khoshgoftaar, 2019; Tan and Le, 2019. High accuracy, quick convergence, and little overfitting show good generalization. From a clinical view, the approach can support faster and more reliable decisions while using modest compute Esteva et al., 2017; Subramanian et al., 2022. Compared with related work, the method balances accuracy and efficiency and is a good fit for real use in hospitals and labs International Medical Device Regulators Forum, 2013, 2014; U.S. Food and Drug Administration, 2021.

Transfer Learning Effectiveness: The rise from 57.20% at epoch 1 to 94.42% at epoch 10 shows the value of transfer learning. ImageNet features give a strong start and need only small updates for blood cell images. This is visible in the smooth MobileNetV2 curves in Fig. 4 and Fig. 5 and contrasts with the slower and noisier training from scratch in Fig. 8 and Fig. 10 Raghu et al., 2019; Russakovsky et al., 2015.

Data Augmentation Impact: Growing the data from 5,000 to about 80,000 effective samples is key to the final 94.42% accuracy. The small gap of about 1.45% between train and validation accuracy suggests that augmentation limits overfitting even with a much smaller original set Perez and Wang, 2017; Shorten and Khoshgoftaar, 2019; Zoph et al., 2019.

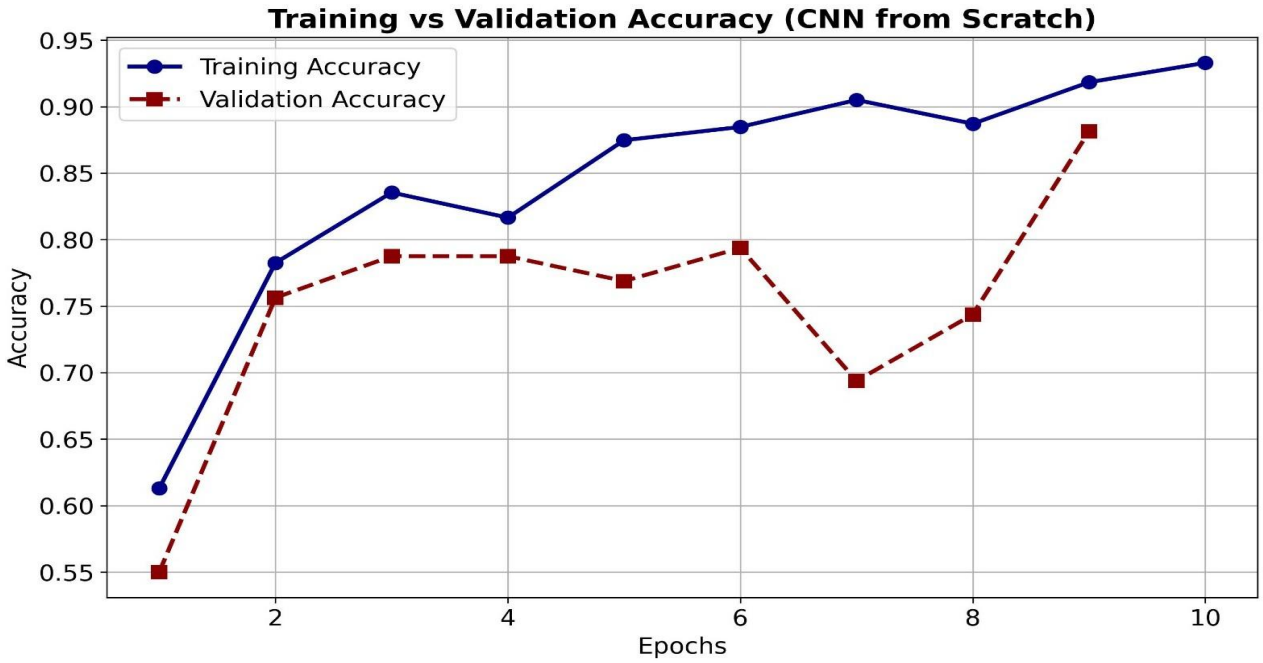


Fig. 8. CNN trained from scratch. Training and validation accuracy across ten epochs. The pattern is slower and less stable than MobileNetV2.

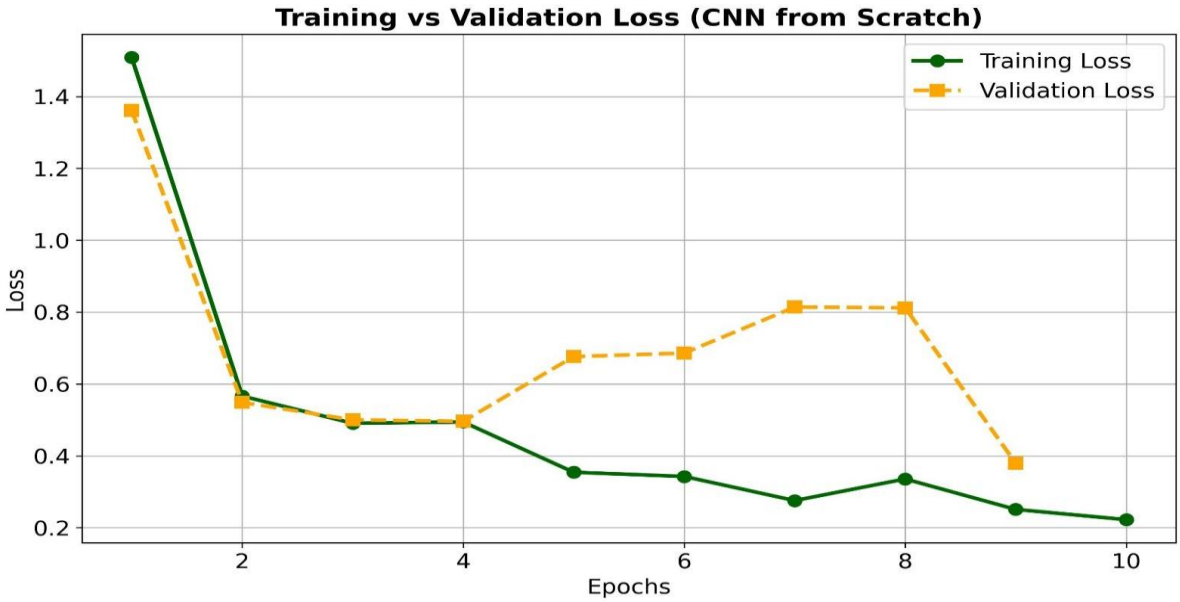


Fig. 9. CNN trained from scratch. Training and validation loss across ten epochs. Fluctuations are larger than for MobileNetV2, which shows less stable optimization.

Architectural Efficiency: MobileNetV2 with about 3.5 million parameters gives competitive accuracy with much lower compute than older models. This matters for clinics because it runs on standard workstations and on edge devices without special hardware. The timing results in Fig. 6 and Fig. 7 show this benefit He et al., 2016; Sandler et al., 2018; Simonyan and Zisserman, 2015.

Clinical Implications: The 94.42% accuracy is suitable for clinical support. With an inference time near real time, often below one second per image, the system can help triage and speed up review. The low compute cost supports adoption and integration with lab information systems and existing workflows HL7 International, 2018; U.S. Food and Drug Administration, 2021.

Convergence and Generalization: The small gap between training and validation accuracy and the steady drop in validation loss point to good generalization rather than memorization. The early edge of validation accuracy over training accuracy suggests that dropout and frozen base layers provide useful regularization, as seen in Fig. 4 Raghu et al., 2019; Srivastava et al., 2014.

Clinical Applications and Implications

The clinical value of this lightweight CNN based system is its easy fit into routine hematology work while raising accuracy and speed Esteva et al., 2017; Subramanian et al., 2022. By automating blood cell classification the system lowers reliance on manual microscopy, which reduces inter observer variability and delays Matek et al., 2019. The compact design runs on standard lab computers and on portable devices, which suits resource constrained settings Sandler et al., 2018; Tan and Le, 2019. With fast inference and steady accuracy it supports real time decisions so clinicians can begin treatment sooner Micikevicius et al., 2018. This section

explains how the system complements current practice, improves the quality and speed of analysis, and supports wider use of AI driven tools in oncology International Medical Device Regulators Forum, 2013, 2014; U.S. Food and Drug Administration, 2021.

Diagnostic Workflow Integration

The system fits into a typical lab workflow with clear touchpoints HL7 International, 2018; Sandler et al., 2018; Tan and Le, 2019:

- Image capture at the microscope using existing procedures.
- Automated preprocessing and normalization by the system.
- Real time classification by the CNN model.
- Results with confidence scores sent for human review and report sign off.

Rapid Diagnostic Capability

Automated analysis cuts review time from hours to seconds Matek et al., 2019; Micikevicius et al., 2018; Subramanian et al., 2022. In practice this means:

- Faster triage and treatment for critical patients.
- Higher screening throughput for population level programs.
- Practical support for facilities with limited staff and compute.

Quality Standardization

Automation supports consistent and reproducible results Esteva et al., 2017; International Medical Device Regulators Forum, 2014:

- Shared criteria across sites and shifts.
- Fewer errors related to fatigue or limited experience.
- Better data quality for audits and epidemiology.

Clinical Decision Support

The tool augments expert judgment rather than replacing it International Medical Device Regulators Forum, 2013; Raghu et al., 2019; U.S. Food and Drug Administration, 2021:

- Provides second reader analysis to support diagnostic confidence.
- Flags uncertain or out of distribution cases for specialist review.
- Reports confidence scores to guide follow up testing and workflow priority.

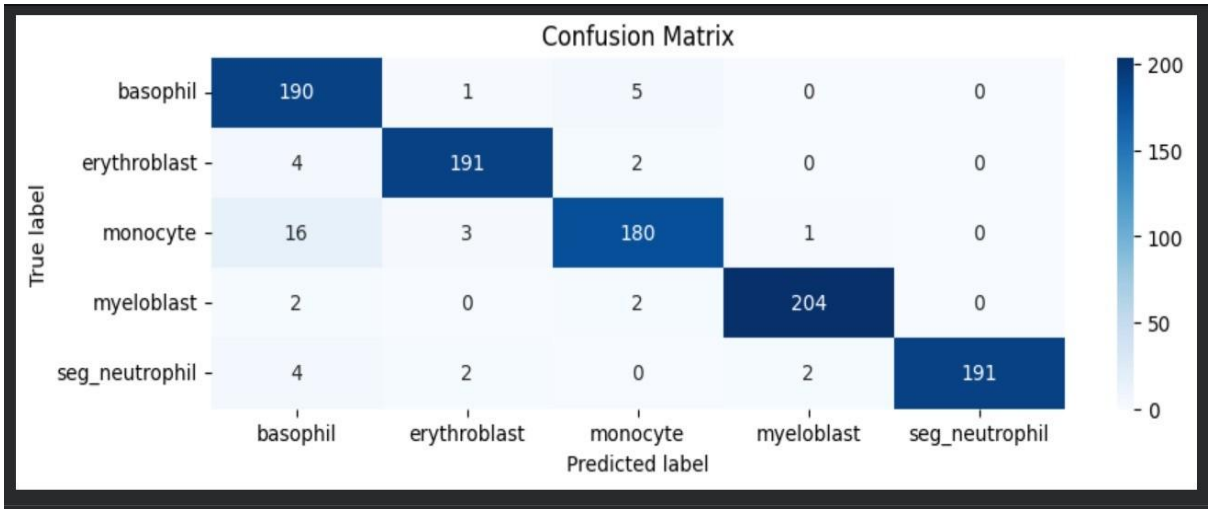


Fig. 10. Confusion matrix of the proposed model on the test set, showing the number of true and predicted samples for each white blood cell class. Most samples lie on the diagonal, which indicates strong performance across all five classes.

Limitations

Although the lightweight CNN based framework shows high accuracy and good generalization for automated blood cancer classification, several limits remain. The study uses a single public dataset collected under consistent lab conditions, which may not reflect the full range of morphology seen in practice Singh, 2024. External validation across different sites, devices, or staining workflows has not yet been performed, and the system has not been tested under regulatory, interoperability, or real world workflow requirements HL7 International, 2018; International Medical Device Regulators Forum, 2014; Subramanian et al., 2022; U.S. Food and Drug Administration, 2021.

Dataset Limitations

This work relies on one Kaggle dataset with fixed settings for microscope hardware, illumination, and slide preparation, with staining that is mainly Wright Giemsa Singh, 2024. Such uniform conditions may limit robustness to domain shift and to rare morphologies. Uncommon leukemia subtypes are under represented, which can reduce generalization to those cases Matek et al., 2019; Shorten and Khoshgoftaar, 2019.

Generalization and Robustness Constraints

Validation is limited to a single dataset. It is not yet clear how the model behaves with other microscope manufacturers, camera sensors, magnifications, or staining protocols, and

performance on difficult edge cases is not fully known Raghu et al., 2019; Subramanian et al., 2022. Only a single training run is reported. Because training was done on a CPU only system, repeated runs or k fold cross validation were not carried out. Measures such as standard deviation across runs would give a better view of model robustness. Future work should run multiple experiments or use k fold cross validation on a GPU system so that more reliable statistical validation of the results is available.

To reduce this bias, a small external test using the Africa Blood Cell Images and EHR for Cancer Detection dataset from a different clinical setting is included. This test uses the same metrics as the main evaluation and gives a first view of model behaviour under a shifted data distribution. However, the external set is still limited in size and source. A larger cross institutional study with multiple centres and imaging setups is needed to support stronger clinical claims.

Clinical Integration Challenges

Real deployment needs more than strong accuracy. It also needs regulatory clearance such as FDA or CE marking, integration with laboratory information systems, clinician trust, and clear roles for responsibility and quality assurance. Prospective clinical studies and ongoing quality processes are still required HL7 International, 2018; International Medical Device Regulators Forum, 2013, 2014; U.S. Food and Drug Administration, 2021.

Explainability and Interpretability

Deep models can be hard to interpret. Limited transparency can slow adoption when clinicians need to explain single predictions or understand failure modes Abir et al., 2023; Subramanian et al., 2022. Additional work on explainable AI methods, user interfaces, and visual explanations would help build trust and support clinical use.

Rare Class Performance

Some subtypes and atypical patterns appear infrequently. These rare classes may be learned less well, which can lower performance on unusual presentations Kasim et al., 2025; Matek et al., 2019; Shorten and Khoshgoftaar, 2019. Larger and more diverse datasets, targeted augmentation, or class specific sampling strategies may be needed to improve results for rare patterns.

Lack of Expert Annotator Details

One limitation of this study is the lack of clear information about how the labels in the Kaggle dataset were created. The dataset description does not state whether domain experts such as hematologists or pathologists labeled the data or checked the labels. This missing information may introduce labeling bias or misclassification and affects how model performance should be interpreted. Future studies should use datasets with well documented expert annotations or add an independent step to check and validate the labels, so that the ground truth is more reliable and the reported metrics are easier to trust.

Conclusion and Future Work

This study shows that lightweight convolutional neural networks can deliver accurate and efficient automated blood cancer classification. The best MobileNetV2 configuration reached 94.42% validation accuracy with a small gap of about 1.45% between training and validation. On the five class validation set it achieved overall precision of 95.8%, recall of 95.6% and F1 score of 95.6%, with per class F1 scores in a narrow band from about 92% to 98%. The model converged in 10 epochs in about 343 minutes and gives fast inference with about 3.5 million parameters. These results suggest that compact CNNs are a practical choice for clinical decision support where speed and resource use matter.

Future work will push the system toward clinical translation:

- Expand data diversity with multi institutional cohorts, more leukemia subtypes, and a wider range of staining and imaging protocols.
- Test robustness under distribution shift, changes in device and magnification, and in the presence of adversarial or noisy inputs.
- Add explainable AI tools such as class activation maps, saliency maps, and LIME, and report calibrated confidence for each prediction.
- Improve rare class performance using few shot learning, class balanced sampling, appropriate data synthesis, and out of distribution detection with continual learning on new cases.
- Address deployment needs through prospective clinical studies, clear regulatory pathways, interoperability with laboratory information systems, quality assurance routines, user training, and transparent documentation of limits and intended use.
- Perform repeated experiments and k fold cross validation to measure variation such as standard deviation across runs and to provide stronger statistical evidence for robustness.

References

Abir, W. H., Uddin, M. F., Khanam, F. R., & Khan, M. M. (2023). Explainable AI in diagnosing and anticipating leukemia using transfer learning method. https://arxiv.org/abs/2312.00487

Ekmekyapar, T., O¨ z, N., S¸ en, M., Varol, M., & Aktas¸, T. (2023). Exemplar MobileNetV2-based artificial intelligence for robust and accurate diagnosis of multiple sclerosis. Diagnostics, 13(19), 3030. https://doi.org/10.3390/diagnostics13193030

El-Ghany, S. A., Elmogy, M., & Abd El-Aziz, A. A. (2023). Computer-aided diagnosis system for blood diseases using efficientnet-b3 based on a dynamic learning algorithm. Diagnostics, 13(3), 404. https://doi.org/10.3390/diagnostics13030404

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115–118. https://doi.org/10.1038/nature21056

Goceri, E. (2023). Medical image data augmentation: Techniques, comparisons and interpretations. Artificial Intelligence Review, 56, 3683–3704. https://doi.org/10.1007/s10462-022-10193-y

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–778. https://doi.org/10.1109/CVPR.2016.90

HL7 International. (2018). Fast healthcare interoperability resources (FHIR): Overview (v3.5a). Retrieved October 30, 2025, from https://www.hl7.org/fhir/2018Dec/overview.html

International Medical Device Regulators Forum. (2013). Software as a medical device (SaMD): Key definitions (imdrf/samd wg/n10) (tech. rep.). https://www.imdrf.org/sites/default/files/docs/imdrf/final/technical/imdrf-tech-131209-samd-key- definitions-140901.pdf

International Medical Device Regulators Forum. (2014). Software as a medical device: Possible framework for risk categorization and corresponding considerations (imdrf/samd wg/n12) (tech. rep.). https://www.imdrf.org/documents/software-medical- device-possible-framework-risk-categorization-and-corresponding-considerations

Kasim, S., Ahmed, A., Mohamed, M., Karim, F., Karakoc, M., & Kumar, A. (2025). Multiclass leukemia cell classification using hybrid deep learning approach: A comparative study. Scientific Reports, 15(1), 5585. https://doi.org/10.1038/s41598- 025-61931-3

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. International Conference on Learning Represen- tations (ICLR). https://arxiv.org/abs/1412.6980

Matek, C., Schwarz, S., Spiekermann, K., & Marr, C. (2019). Human-level recognition of blast cells in acute myeloid leukaemia with convolutional neural networks. Nature Machine Intelligence, 1(11), 538–544. https://doi.org/10.1038/s42256-019- 0101-9

Micikevicius, P., Narang, S., Alben, J., Diamos, G., Elsen, E., Garcia, D., Ginsburg, B., Houston, M., Kuchaiev, O., Venkatesh, G., & Wu, H. (2018). Mixed precision training. https://arxiv.org/abs/1710.03740

Perez, L., & Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. https://arxiv. org/abs/1712.04621

Raghu, M., Zhang, C., Kleinberg, J., & Bengio, S. (2019). Transfusion: Understanding transfer learning for medical imaging.Advances in Neural Information Processing Systems (NeurIPS). https://arxiv.org/abs/1902.07208

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 115(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 4510–4520. https://arxiv. org/abs/1801.04381

Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(60). https://doi.org/10.1186/s40537-019-0197-0

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. https://arxiv.org/ abs/1409.1556

Singh, S. (2024). Blood cell images for cancer detection [data set]. Retrieved October 30, 2025, from https://www.kaggle.com/ datasets/sumitshingh/blood-cell-images-for-cancer-detection

Soladoye, A. A., Ogunwande, O., Olayinka, K., & Sanni, K. (2025). Enhancing leukemia detection in medical imaging using deep transfer learning algorithms: Comparative analysis of efficientnet-b3 and vgg19. Informatics in Medicine Unlocked, 52, 101467. https://doi.org/10.1016/j.imu.2025.101467

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(56), 1929–1958. http://jmlr.org/papers/ v15/srivastava14a.html

Subramanian, N., Elharrouss, O., Al-Maadeed, S., & Chowdhury, M. (2022). A review of deep learning-based detection methods for COVID-19. Computers in Biology and Medicine, 141, 105233. https://doi.org/10.1016/j.compbiomed.2022.105233

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2818–2826. https: //doi.org/10.1109/CVPR.2016.308

Tan, M., & Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. Proceedings of the International Conference on Machine Learning (ICML). https://arxiv.org/abs/1905.11946

U.S. Food and Drug Administration. (2021). Artificial intelligence/machine learning (AI/ML)–based software as a medical device (SaMD) action plan (tech. rep.). FDA. https://www.fda.gov/media/145022/download

Yi, X., Walia, E., & Babyn, P. (2019). Generative adversarial network in medical imaging: A review. Medical Image Analysis, 58, 101552. https://doi.org/10.1016/j.media.2019.101552

Zhao, W., Jiang, W., & Qiu, X. (2021). Deep learning for COVID-19 detection based on CT images. Scientific Reports, 11(1), 14353. https://doi.org/10.1038/s41598-021-93832-2

Zoph, B., Cubuk, E. D., Ghiasi, G., Lin, T.-Y., Shlens, J., & Le, Q. V. (2019). Learning data augmentation strategies for object detection. https://arxiv.org/abs/1906.11172