LGLD-Net: A Lightweight Global and Local feature-based Dual-stream Neural Network for Microscopic Medical Image Classification

Arko Dasgupta[1,a][0009−0000−5675−6120], Arjeesh Palai[1,a][0009−0007−7774−8075], Asfak Ali[2,a][0000−0002−4034−1956], Dmitrii Kaplun[3,*], and Ram Sarkar[1]

[1] Dept. of Computer Science & Engineering, Jadavpur University, Kolkata, India
arkodasgupta05@gmail.com, arjeeshpalai@gmail.com, ramjucse@gmail.com
[2] Dept. of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata, India
asfakali.etce@gmail.com
[3] St. Petersburg Electrotechnical University "LETI", St. Petersburg, Russian Federation
dikaplun@etu.ru

Abstract. Microscopic image analysis provides vital information at the cellular level. However, most deep learning models struggle to jointly capture fine-grained and global features and often remain computationally expensive. To this end, we propose a compactly designed lightweight model (161K parameters), called Lightweight Global and Local feature-based Dual-stream Neural Network (LGLD-Net). Here, we use involution-based patch stem to capture fine-grained local textures and a Fourier-based global stream to model long-range dependencies and contextual features. Then, an adaptive fusion module is added that dynamically balances local and global features for optimal feature representations. The proposed LGLD-Net yields state-of-the-art performance on two microscopic image datasets, Acute lymphoblastic leukemia (ALL) and Raabin-WBC, achieving classification accuracies of 99.69% and 96.50%, respectively. The proposed model is evaluated on multiple hardware platforms, including the Raspberry Pi 5 for CAD applications, achieving 17.56 FPS with a computational cost of only 1.88 GFLOPs. The codes are publicly available at https://github.com/arkodasgupta0412/LGLD-Net.

Keywords: Microscopic Image · Blood-cell Classification · Lightweight CNN · Local and Global Feature · Dual-branch Neural Network

1 Introduction

Computer-based medical image analysis has, in recent years, proven to be an influential tool in disease prediction, detection, treatment planning, faster and

⋆ [a] Authors contributed equally to this research
⋆⋆ [*] Corresponding author

more accurate interpretation of advanced visual data, early diagnosis, and minimized costs for healthcare. In this regard, microscopic image analysis provides key information at the cellular level. The classification of white blood cells (WBC) and classification of benign hematogones from subtypes of malignant lymphoblasts: early Pre-B, Pre-B, and Pro-B is a must for the diagnosis of hematologic disorders. Acute lymphoblastic leukemia (ALL), the most common childhood malignancy, accounts for approximately 25% of all cancers in children[†] with more than 6,000 new diagnoses each year in the U.S.[‡] ; prompt detection of lymphoblasts by blood smears is crucial, as delays in diagnosis negatively affect survival. Sepsis, a systemic inflammatory state associated with immune cell imbalances, kills 11 million people each year due to 49 million cases worldwide[§] , many caused by neglected neutrophilic markers. As such, the identification of atypical lymphocytes is important for the diagnosis of infectious mononucleosis and differentiation from leukemia subtypes. Conditions such as aplastic anemia, which is characterized by pancytopenia, also require the proper evaluation of blood cells to inform immediate treatment. Automating these categorizations using microscopy-based analysis can significantly mitigate diagnostic errors and improve clinical pathways in hematological emergencies.

Recent advances in deep learning further enhance this potential by addressing the limitations of manual interpretation. Diseases such as ALL, sepsis, and related blood disorders require a precise classification of bone marrow cells. These tasks are increasingly performed by convolutional neural networks (CNNs). Numerous models have been developed across datasets such as ALL and Raabin-WBC. For example, Singh et al. [1] and Gokulkrishnan et al. [2] achieved 99.38% and 99.39% accuracy in the ALL dataset, while Jiang et al. [3] and Chen et al. [4] obtained F1 scores above 94% on Raabin-WBC, demonstrating the efficacy of CNN pipelines. However, generalization remains challenging; many models falter under variations in staining, resolution, or class distribution. Most CNNs struggle to jointly capture fine-grained cytological features and global morphology and often remain computationally expensive for real-time or resource-limited deployment. These issues highlight the need for robust, interpretable, and efficient models with strong performance.

To overcome these limitations, we introduce **LGLD-Net** (Lightweight Global and Local feature-based Dual-stream Neural Network), a lightweight deep learning architecture that integrates local and global feature streams through a dynamic fusion module. Specifically, our **contributions** are:

- An involution-based patch stem to capture fine-grained local textures.
- A Fourier-based global stream to model long-range dependencies and contextual features.

[†] https://www.cancer.gov/types/leukemia/patient/child-all-treatment-pdq
[‡] https://pmc.ncbi.nlm.nih.gov/articles/PMC8124693
[§] https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(19)32989-7

– An adaptive fusion module that dynamically balances local and global features for optimal representation.
– A compactly designed lightweight model (161K parameters) that achieves notable performances across two standard datasets, namely ALL and Raabin-WBC.
– Achieves real-time inference on edge devices like Raspberry Pi 5 with only 1.88 GFLOPs, enabling efficient deployment in resource-constrained environments.

2 Proposed Methodology

We propose **LGLD-Net (Lightweight Global and Local feature-based Dual-stream Neural Network)**, a lightweight but effective dual-branch neural network that integrates local and global feature representations for blood cell classification tasks. The model is designed to effectively capture minute local texture variations along with global structural context using specialized streams, followed by a local-global feature mixer. The pipeline is visualized in Fig. 1.

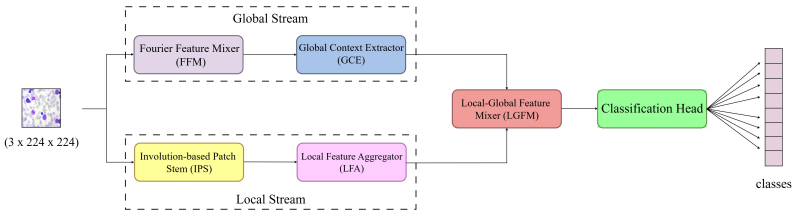

Fig. 1. Overall block diagram of the proposed LGLD-Net.

2.1 Overall Architecture

LGLD-Net consists of a local feature stream, a global feature stream, a learnable feature mixer, and a classification head. The local and global streams capture the fine-grained textures and general contextual patterns of the input image, respectively. These are distinct but complementary features of the image. The Local-Global Feature Mixer (LGFM) adaptively balances the contributions of these features and combines them together. Finally, the fused output is passed through a compact classification head to achieve the final output.

2.2 Local Feature Stream

To capture localized patterns often found in medical images, we use a local feature stream composed of two parts: an involution-based patch stem (IPS) and a local feature aggregator block (LFA).

Involution-based Patch Stem (IPS): For initial downsampling, the IPS uses a standard convolution. Then, the feature map is passed through an **involution** layer (Fig. 2). Involution dynamically creates **spatially adaptive, channel-agnostic** filters based on the input itself, as opposed to predefined filters. As a result, the model can effectively capture minute variations in nearby areas.

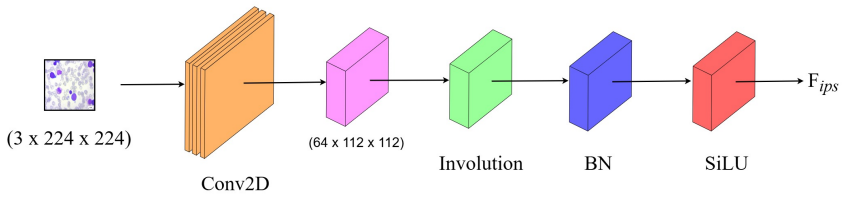

Fig. 2. Illustration of the IPS block: extracts low-level features via spatially-adaptive involution with channel-agnostic filters.

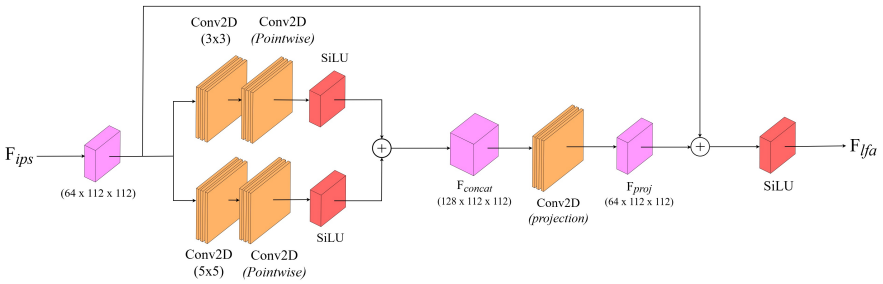

Fig. 3. Illustration of the LFA block: aggregates local context using parallel convolutions and uses residual fusion to preserve both local and contextual richness.

Local Feature Aggregator (LFA): The LFA module is used to capture fine structures at different scales, which enhances local representations. It uses parallel depth-wise paths to extract spatial features with different receptive fields (Fig. 3). The features are then integrated and passed through a non-linear SiLU activation (Sigmoid Linear Unit). The final output is residually combined with the input to retain context while enriching local detail.

2.3 Global Feature Stream

The global stream captures long-range dependencies and larger spatial contexts that are often overlooked by local processes. Its two main modules are the Fourier Feature Mixer (FFM) and the Global Context Extractor (GCE).

Fourier Feature Mixer (FFM): To incorporate spatial structure and relative positional information, the FFM first enriches the input tensor by embedding sinusoidal positional encoding along the spatial axes (Fig. 4). Specifically, the sine and cosine functions are applied across the height and width dimensions and concatenated with the original image tensor.

$$\tilde{X} = \text{Concat}(X, \sin(2\pi i/H), \cos(2\pi i/H), \sin(2\pi j/W), \cos(2\pi j/W)) \quad (1)$$

where $X \in \mathbb{R}^{B\times C\times H\times W}$ is the input feature map and i, j denote spatial indices across height and width, respectively. The enriched representation $\tilde{X}$ is then processed through a 1×1 convolution to reduce dimensionality and produce a frequency-aware embedding.

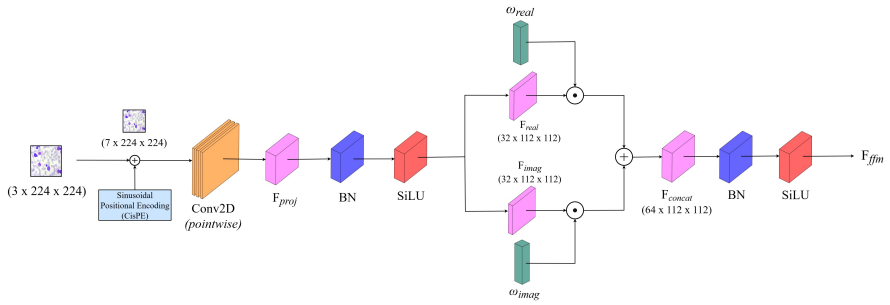

Fig. 4. Illustration of the FFM block: enhances spatial representation by combining sinusoidal Positional Encoding with learnable frequency modulation, simulating complex-valued responses via real-imaginary decomposition to capture long-range dependencies.

$$F = \text{Conv}_{1\times1}(\tilde{X}) \tag{2}$$

This output is split into real and imaginary components to simulate complex-valued responses:

$$F_{\text{real}},\ F_{\text{imag}} = \text{Split}(F) \tag{3}$$

Each component is modulated using learnable parameters that control the contribution of the respective frequency part:

$$F_{\text{ffm}} = \text{SiLU}(\text{BN}(\omega_{\text{real}} \cdot F_{\text{real}} \,\|\, \omega_{\text{imag}} \cdot F_{\text{imag}})) \tag{4}$$

Here, ω_{real} and ω_{imag} are channel-wise learnable scaling factors, BN denotes batch normalization, and $\|$ denotes channel concatenation.

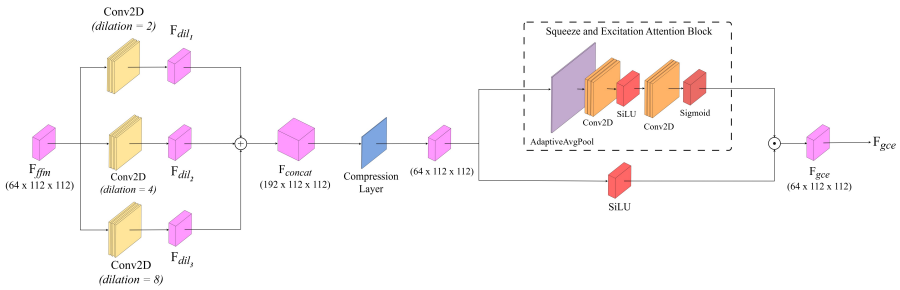

Fig. 5. Illustration of the GCE block. It selectively enriches fused features by aggregating multi-scale spatial information and refining them through dynamic gating, enabling enhanced global representations with minimal computational overhead.

Global Context Extractor (GCE): The GCE pulls data from various spatial scales. It broadens the receptive field of the FFM block output. This is done using dilated convolutions, which are used to capture long-range dependencies (Fig. 5). It is followed by a squeeze-and-excitation attention mechanism, which refines the aggregated features by emphasizing globally important channels and suppressing less informative ones.

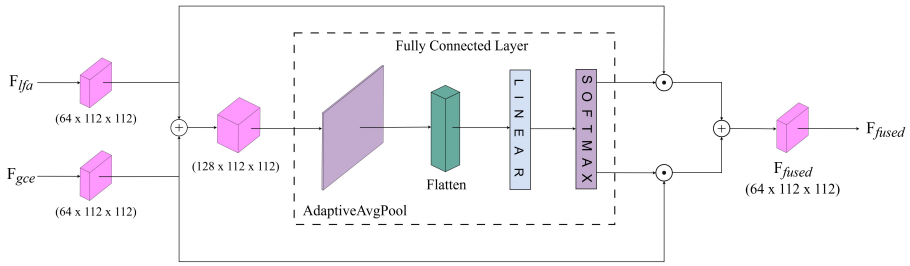

Fig. 6. Illustration of the LGFM block. It adaptively merges fine-grained local features and global contextual information through attention-guided re-weighing, enhancing the semantic richness of the fused representation.

2.4 Local-Global Feature-Mixer (LGFM)

The complementary features from the local and global streams are effectively merged using the LGFM. It captures the summary statistics of the fused features by applying adaptive average pooling. In this adaptive pooling mechanism, the number of output channels is specified. The dimensions of the filter and the stride are automatically selected (Fig. 6). A compact fully connected layer learns the relative importance of each stream, producing weights that are normalized using a softmax function. These weights are then used to adaptively blend the two branches in a soft attention-like manner, yielding the final fused representation as:

$$F_{\text{fused}} = w_l \cdot F_{\text{local}} + w_g \cdot F_{\text{global}} \quad ; \text{where } w_l + w_g = 1 \tag{5}$$

The weights (w_l, w_g) are calculated using a softmax operation on the global descriptors of $F_{\text{local}}(= F_{\text{lfa}})$ and $F_{\text{global}}(= F_{\text{gce}})$. This formulation ensures that the network can prioritize spatial detail or global context as needed, depending on the input.

2.5 Classification Head

To generate the final prediction, the fused features are passed through a compact classification block that distills the learned representation and maps it to the output classes. It has been visualized in Fig. 7.

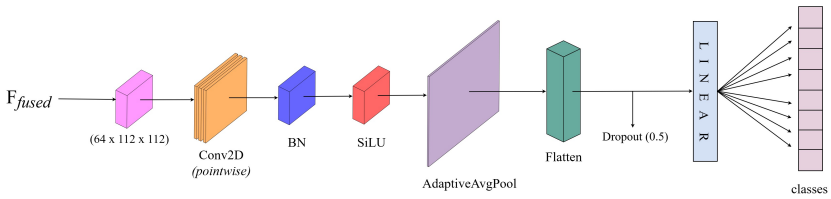

Fig. 7. Illustration of the classification head. It maps the fused features to class scores using a lightweight design that preserves discrimination and reduces overfitting.

3 Experimental Setup and Datasets

3.1 Experimental Setup

The experiments were carried out on a Linux system running kernel version 6.6.56+, with Python 3.11.11 as the primary programming environment. The machine is equipped with a CPU consisting of 2 physical cores and 4 threads, based on a 64-bit `x86_64` architecture, operating at a base clock speed of 2.00 GHz. The system has 31.35 GB RAM, with approximately 29.83 GB available for computations. An NVIDIA Tesla P100 GPU (featuring 16 GB dedicated memory and 56 streaming multiprocessors) was used to accelerate deep learning tasks. All model training and evaluation were performed using the PyTorch framework in this single-GPU setup.

3.2 Datasets

Table 1. Dataset summary with split ratios and image counts. Ratio A:B:C denote splits from a unified set; Ratio A:B indicates a separately provided test set. Further details: https://github.com/arkodasgupta0412/LGLD-Net/blob/main/DATASET.md.

Dataset	Split Ratio	Train / Validation / Test Images
ALL [5]	80:10:10	2605 / 325 / 326
Raabin-WBC [6]	80:20	8140/ 2035 / 4339

4 Results and Analysis

We conducted a comprehensive study to evaluate the performance of our proposed **LGLD-Net** against several state-of-the-art models in two benchmark datasets: **ALL** and **Raabin-WBC**. The findings show that **LGLD-Net** is a **very lightweight solution** that **exceeds** the existing models used for classification tasks.

LGLD-Net achieves an accuracy of **99.69%** (Table 2) in the ALL dataset. It outperforms previous best models such as Singh et al. [1], Gokulkrishnan et al. [2], and Islam et al.[7]. Our model reaches **96.50%** accuracy on the Raabin-WBC dataset and shows strong scores in other performance metrics (Table 2). It surpasses the reported baselines, including Chen et al. [4] and Jiang et al. [3].

Table 2. Comparison of the proposed model and SOTA models on ALL and Raabin-WBC dataset. All scores are in %. Best scores are in bold.

ALL Dataset		Raabin-WBC Dataset				
Model	**Accuracy**	**Model**	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Basymeleh et al. [8]	97.42	Yu et al. [9]	89.56	78.39	88.31	82.47
Tusar et al. [10]	97.50	Jiang et al. [3]	96.13	91.69	97.00	93.97
Gokulkrishnan et al. [2]	99.39	Sharma et al. [11]	95.99	92.62	95.08	97.00
Nalluru et al. [12]	91.50	Tavakoli et al. [13]	94.65	89.78	92.47	91.10
Singh et al. [1]	99.38	Chen et al. [4]	96.36	92.87	96.15	94.28
AlexNet	96.96	ResNet18	87.18	87.46	86.26	86.75
Islam et al. [7]	99.12	ResNet50	94.25	94.34	93.53	93.90
ResNet50	97.85	InceptionV3	93.36	93.06	92.57	92.78
InceptionV3	96.93	Densenet201	96.13	96.20	95.72	95.94
LGLD-Net	**99.69**	**LGLD-Net**	**96.50**	**93.04**	**95.91**	**94.38**

Despite its significantly smaller size, LGLD-Net outperforms larger models in both datasets. LGLD-Net shows strong generalization across datasets due to its modular architecture. Each block complements the others. The components enhance feature representation and collectively increase semantic expressiveness, without added complexity.

The ablation study (Table 3) highlights the importance of each core component in LGLD-Net. The LFA block alone contributes to substantial improvements in all datasets, highlighting the importance of fine-grained local information. Similarly, the **GCE** captures long-range dependencies, although its individual impact is slightly lower on the Raabin-WBC dataset. When combined **(LFA + GCE)**, performance consistently improves, demonstrating their complementary nature. In particular, the complete architecture **(LGLD-Net + Involution + LFA + GCE)** outperforms the SOTA models in all datasets. This shows that incorporating the **IPS** further boosts contextual understanding and discriminative power. Overall, the results validate that each module contributes meaningfully and their synergy leads to state-of-the-art performance.

Table 3. Results of ablation studies demonstrating the contribution of each component in the LGLD-Net architecture.

Dataset	**ALL Dataset**				**Raabin -WBC Dataset**			
Model	**Acc**	**Pre**	**Rec**	**F1**	**Acc**	**Pre**	**Rec**	**F1**
LGLD-Net (LFA)	98.98	98.67	98.83	98.75	96.66	93.37	96.17	94.58
LGLD-Net (GCE)	97.96	97.27	97.81	97.52	96.27	96.16	95.78	94.79
LGLD-Net (LFA+GCE)	98.57	98.07	98.65	98.34	96.47	92.75	96.02	94.23
LGLD-Net (Invo+LFA+GCE)	99.69	99.51	99.75	99.63	96.50	93.04	95.91	94.38

Table 4 shows the efficiency of our model. LGLD-Net has about **161K trainable parameters**, which is much smaller than traditional backbone models such as AlexNet (62.3M), ResNet50 (25.6M) or DenseNet201 (20.2M). Further plots are available at: https://github.com/arkodasgupta0412/LGLD-Net/tree/main/plots.

Table 4. Comparison of the proposed LGLD-Net and state-of-the-art (SOTA) models in terms of trainable parameters, computational complexity (FLOPs), and inference latency on various hardware platforms.

Model	# Trainable Params	GFLOPs	Runtime (ms) on		
			Tesla P100 (GPU)	Xeon CPU@2.20GHz	ARM Cortex-A76 CPU (RPi5)
AlexNet	62.3M	0.71	1.42	20.40	23.75
ResNet18	11.4M	1.82	2.31	33.36	36.81
ResNet50	25.6M	4.11	5.58	79.05	88.96
InceptionV3	23.9M	5.73	11.89	132.19	175.23
DenseNet201	20.2M	4.34	27.98	145.15	452.47
LGLD-Net	**161K**	**1.88**	**3.54**	**81.71**	**56.93**

The proposed model was evaluated across various hardware platforms, including an NVIDIA Tesla P100 GPU, an Intel Xeon CPU @ 2.20GHz, and an ARM Cortex-A76 CPU on a Raspberry Pi 5. The model demonstrates efficient inference capabilities, achieving a latency of just **3.54 ms on the Tesla P100, 81.71 ms on the Xeon CPU, and 56.93 ms on the Raspberry Pi 5**. This corresponds to approximately **17.56 frames per second (FPS) on the Raspberry Pi**, making the model suitable for real-time edge deployment. Moreover, the model maintains a **low computational complexity of only 1.88 GFLOPs**, highlighting its lightweight design and suitability for resource-constrained environments.

5 Conclusion

In this paper, we introduced a novel lightweight neural network architecture, LGLD-Net, which employs a dual-stream design to effectively integrate global and local feature learning within a unified framework. LGLD-Net achieves strong performance across two benchmark microscopic image classification datasets while being significantly more parameter-efficient than several existing models. The model was successfully deployed on edge devices such as the Raspberry Pi 5, achieving an inference speed of approximately 17.56 FPS, demonstrating its potential for real-time applications. However, it exhibits slightly higher computational complexity and inference time compared to simpler models such as AlexNet.

Future work will focus on extending LGLD-Net to handle 3D and temporal microscopic data, as well as incorporating transformer-based modules to enhance contextual representation learning. In addition, its robustness under staining variability and acquisition noise should be further investigated. Moreover, optimizing the model for ultralow power devices and exploring neural architecture search (NAS) techniques presents a potential direction to advance the efficiency and adaptability of LGLD-Net.

References

1. Retinderdeep Singh, Neha Sharma, Priyanshi Aggarwal, Mukesh Singh, and Kanegonda Ravi Chythanya. Inceptionv3 in medical imaging: Enhancing precision in

acute lymphoblastic leukaemia diagnosis. In *2024 2nd International Conference on Computer, Communication and Control (IC4)*, pages 1–6. IEEE, 2024.
2. Nitla Gokulkrishnan, Tushar Nayak, and Niranjana Sampathila. Deep learning-based analysis of blood smear images for detection of acute lymphoblastic leukemia. In *2023 IEEE international conference on electronics, computing and communication technologies (CONECCT)*, pages 1–5. IEEE, 2023.
3. Ming Jiang, Liu Cheng, Feiwei Qin, Lian Du, and Min Zhang. White blood cells classification with deep convolutional neural networks. *International Journal of Pattern Recognition and Artificial Intelligence*, 32(09):1857006, 2018.
4. Hua Chen, Juan Liu, Chunbing Hua, Jing Feng, Baochuan Pang, Dehua Cao, and Cheng Li. Accurate classification of white blood cells by coupling pre-trained resnet and densenet with scam mechanism. *BMC bioinformatics*, 23(1):282, 2022.
5. Mehrad Aria, Zohreh Javanmard, Donia Pishdad, Vahid Jannesari, Maryam Keshvari, Mahshid Arastonejad, Reza Safdari, and Mohammad Esmaeil Akbari. Towards diagnostic intelligent systems in leukemia detection and classification: A systematic review and meta-analysis. *Journal of Evidence-Based Medicine*, 18(1), 2025.
6. Zahra M. Kouzehkanan, Somayeh Saghari, Saeed Tavakoli, et al. A large dataset of white blood cells containing cell locations and types, along with segmented nuclei and cytoplasm. *Scientific Reports*, 12:1123, 2022.
7. Oahidul Islam, Md Assaduzzaman, and Md Zahid Hasan. An explainable ai-based blood cell classification using optimized convolutional neural network. *Journal of Pathology Informatics*, page 100389, 2024.
8. Aiman Muhamad Basymeleh, Bagus Esa Pramudya, and Reinato Teguh Santoso. Acute lymphoblastic leukemia image classification performance with transfer learning using cnn architecture. In *2022 4th international conference on biomedical engineering (IBIOMED)*, pages 30–35. IEEE, 2022.
9. Wei Yu, Jing Chang, Cheng Yang, Limin Zhang, Han Shen, Yongquan Xia, and Jin Sha. Automatic classification of leukocytes using deep neural network. In *2017 IEEE 12th international conference on ASIC (ASICON)*, pages 1041–1044. IEEE, 2017.
10. Md Taufiqul Haque Khan Tusar and Roban Khan Anik. Automated detection of acute lymphoblastic leukemia subtypes from microscopic blood smear images using deep neural networks. *arXiv preprint arXiv:2208.08992*, 2022.
11. Mayank Sharma, Aishwarya Bhave, and Rekh Ram Janghel. White blood cell classification using convolutional neural network. In *Soft Computing and Signal Processing: Proceedings of ICSCSP 2018, Volume 1*, pages 135–143. Springer, 2019.
12. Manohar Nalluru, Naga Prudhvi Raj Vattikuti, Naga Tejaswini Nakirikanti, Phanivarma Mudunuri, and Jahnavi Gaddam. Acute lymphoblastic leukemia classification with jetson nano. In *2024 2nd International Conference on Device Intelligence, Computing and Communication Technologies (DICCT)*, pages 249–253. IEEE, 2024.
13. Sajad Tavakoli, Ali Ghaffari, Zahra Mousavi Kouzehkanan, and Reshad Hosseini. New segmentation and feature extraction algorithm for classification of white blood cells in peripheral smear images. *Scientific Reports*, 11(1):19428, 2021.