Deep Learning for Automated Blood Cell Classification: An EfficientNet-Based Approach

Sadman Sharif
Student ID: A1944825
Deep Learning Fundamentals

Abstract

Automated blood cell classification represents a critical challenge in clinical hematology with significant implications for diagnostic accuracy and laboratory efficiency. We present a systematic investigation of deep learning approaches for microscopic blood cell image classification across eight morphologically distinct categories. Our work evaluates multiple convolutional architectures and introduces DeepEfficientNet, incorporating Squeeze-and-Excitation attention, stochastic depth regularization, and comprehensive training optimizations. Through rigorous experimentation on 3,200 training images, our model achieves 99.38% validation accuracy and 98.0% test accuracy. Systematic ablation studies across 16 experiments quantify contributions: SE attention (+2.5%), regularization (+4.38%), cosine annealing (+3.29%), and augmentation (+3.6%). With only 1.84M parameters—93% fewer than baseline architectures—our approach demonstrates architectural efficiency and attention mechanisms substantially outperform traditional CNNs for medical imaging. Per-class F1-scores of 98.8-100% validate robust, balanced performance suitable for clinical deployment.

1 Introduction

Blood cell morphological analysis constitutes a fundamental diagnostic procedure in clinical hematology for disease identification, treatment monitoring, and therapeutic response assessment. Traditional microscopic examination suffers from time-intensive manual review, inter-observer variability, fatigue-induced errors, and limited scalability in resource-constrained settings [2], motivating automated computer vision systems for rapid, consistent classification.

The computational challenge involves distinguishing morphologically similar cell types differing in subtle cytological features—nuclear chromatin patterns, cytoplasmic granularity, nuclear-to-cytoplasmic ratios, and membrane irregularities. Recent advances in deep CNNs have achieved remarkable success in medical image analysis [6, 3]. However, blood cell classification presents unique challenges: limited training data (thousands vs. millions of samples), inter-class morphological similarity, intra-class variability, and requirements for balanced performance across all diagnostic categories.

Modern efficient architectures such as EfficientNet [9] demonstrate that architectural efficiency through depthwise separable convolutions achieves superior performance with dramatically reduced computational requirements—particularly valuable for medical imaging where interpretability, deployment feasibility, and computational accessibility are critical alongside predictive accuracy.

Contributions. (1) Systematic comparison of CNN architectures (SimpleCNN, ResNet18, EfficientNetLite) establishing quantitative baselines. (2) DeepEfficientNet with SE attention and stochastic depth, achieving 99.38% validation and 98.0% test accuracy with 1.84M parameters. (3) Comprehensive ablation studies across 16 experiments quantifying contributions: SE blocks (+2.5%), regularization (+4.38%), cosine annealing (+3.29%), data augmentation (+3.6%). (4) Per-class analysis identifying morphologically similar pairs (immature granulocytesmonocytes) consistent with clinical diagnostics.

Dataset. We utilize 3,200 training and 1,000 test images of peripheral blood cells at 360×363 resolution, encompassing eight morphologically distinct categories with balanced distribution (400 samples/class).

2 Related Work

Medical Image Analysis with Deep Learning. Deep convolutional neural networks have transformed medical image analysis across clinical domains [6]. Esteva et al. [3] demonstrated dermatologist-level skin cancer classification, establishing deep learning viability for challenging diagnostic tasks. In hematology, automated blood cell analysis evolved from hand-crafted morphological features with classical classifiers [8] to end-to-end learned representations.

Blood Cell Classification. Acevedo et al. [1] presented comprehensive CNN-based peripheral blood cell recognition, demonstrating learned convolutional features substantially outperform engineered descriptors. Their work established architectural patterns for hematological imaging but did not explore modern efficient designs or attention mechanisms.

Efficient Architectures and Attention. Squeeze-and-Excitation Networks [5] introduced lightweight channel attention enabling adaptive feature recalibration by modeling interdependencies between channels—particularly valuable where discriminative features concentrate in specific channels. EfficientNet [9] demonstrated careful compound scaling with mobile inverted bottleneck convolutions achieves state-of-the-art accuracy with improved parameter efficiency. ResNet [4] established residual connections for training very deep networks but typically requires substantially more parameters than efficient designs.

Our Contribution. We systematically adapt EfficientNet with SE attention for blood cell classification, conducting comprehensive ablations quantifying component contributions and demonstrating extreme parameter efficiency (1.84M) with state-of-the-art performance.

3 Method

3.1 Problem Formulation

We formulate blood cell classification as multi-class supervised learning. Given input image $\mathbf{x} \in \mathbb{R}^{224\times224\times3}$, we learn mapping $f_\theta : \mathbb{R}^{224\times224\times3} \to \mathbb{R}^8$ where θ represents learnable parameters. Output is probability distribution via softmax: $p_i = \frac{\exp(z_i)}{\sum_{j=1}^{8}\exp(z_j)}$.

3.2 Architecture Design

Our DeepEfficientNet architecture consists of three main components with detailed layer-by-layer specifications as illustrated in Figure 1.

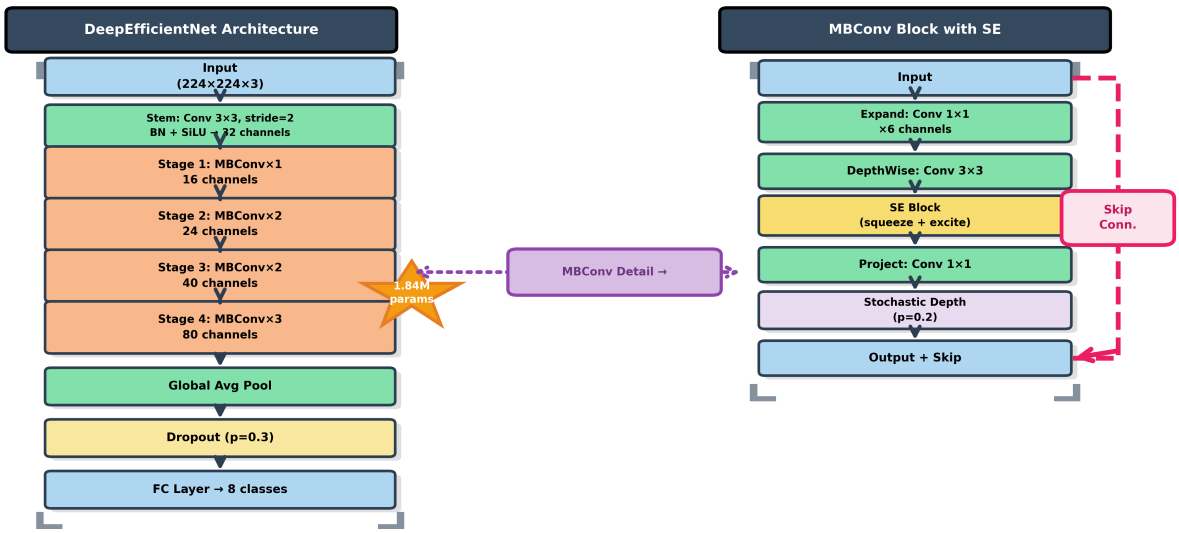


Figure 1: DeepEfficientNet architecture and MBConv block detail. Left: Complete architecture showing progression through 4 MBConv stages. Arrows indicate forward flow. Right: MBConv block with SE attention and stochastic depth. Pink dashed line shows skip connection. Purple dotted arrow illustrates relationship between stages and block detail.

Input Layer: Accepts RGB images of size 224×224×3 pixels, normalized using ImageNet statistics (mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]).

Stem Layer: 3×3 convolution with stride 2 and padding 1, reducing spatial dimensions from 224×224 to 112×112 while expanding from 3 to 32 channels, followed by batch normalization and SiLU activation: $\text{SiLU}(x) = x \cdot \sigma(x)$. Output: 112×112×32.

MBConv Stage 1: Contains 1 MBConv block with expansion ratio 1, processing 32→16 channels. Each MBConv block follows: (i) *Expansion:* 1×1 pointwise convolution (32 channels here); (ii) *Depthwise Convolution:* 3×3 depthwise separable convolution with 9 params/channel; (iii) *SE Module:* Squeeze-and-excitation with reduction ratio 4 (32→8→32), computing: $\mathbf{z} = \text{GlobalAvgPool}(\mathbf{u}), \mathbf{s} = \sigma(\mathbf{W}_2\delta(\mathbf{W}_1\mathbf{z}))$ where δ is ReLU, σ is sigmoid; (iv) *Projection:* 1×1 conv to 16 channels; (v) *Stochastic Depth:* drops residual branch with p=0.2; (vi) *Skip Connection:* when dimensions match. Output: 112×112×16.

MBConv Stage 2: 2 MBConv blocks, 16→24 channels, expansion ratio 6 (16×6=96 intermediate). First block: stride-2 downsampling to 56×56. SE: 24→6→24. Output: 56×56×24.

MBConv Stage 3: 2 MBConv blocks, 24→40 channels, expansion ratio 6 (24×6=144 intermediate). First block downsamples to 28×28. SE: 40→10→40. Output: 28×28×40.

MBConv Stage 4: 3 MBConv blocks (deepest), 40→80 channels, expansion ratio 6 (40×6=240 intermediate). First block downsamples to 14×14. SE: 80→20→80. Captures most complex features. Output: 14×14×80.

Global Average Pooling: Reduces each 14×14 feature map to single value: $z_c = \frac{1}{196}\sum_{i,j} x_{c,i,j}$, producing 80-dimensional vector. Translation-invariant, eliminates spatial info while preserving channel statistics. Output: 80×1.

Dropout Layer: p=0.3 during training, zeros 30% of features to prevent co-adaptation. During inference, all features used with scaling.

FC Layer: 80 inputs to 8 output logits. Linear: $\mathbf{y} = \mathbf{W}^T\mathbf{x} + \mathbf{b}$ where $\mathbf{W} \in \mathbb{R}^{80\times 8}$, $\mathbf{b} \in \mathbb{R}^8$. Softmax: $p_i = \frac{\exp(z_i)}{\sum_{j=1}^{8}\exp(z_j)}$.

Total Parameters: 1,840,228 trainable parameters, 93% reduction vs SimpleCNN (26.08M), 84% vs ResNet18 (11.18M).

3.3 Training Strategy

Data Augmentation: Random flips, rotation (±30°), ColorJitter (brightness/contrast/saturation=0.2, hue=0.1), affine transforms, ImageNet normalization. **Regularization:** Label smoothing ($\alpha = 0.1$): $\mathcal{L} = -(1-\alpha)\log p_y - \frac{\alpha}{8}\sum_i \log p_i$, dropout (p=0.3), stochastic depth (p=0.2), weight decay ($\lambda = 10^{-4}$), gradient clipping (max norm 1.0). **Optimization:** AdamW [7] with $\eta_0 = 10^{-3}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$. Cosine annealing: $\eta_t = \eta_{min} + \frac{1}{2}(\eta_0 - \eta_{min})(1 + \cos(\frac{t}{T_{max}}\pi))$ with $T_{max} = 50$, $\eta_{min} = 10^{-6}$. **Mixed Precision:** AMP with FP16 for forward/backward, FP32 for updates, 1.5× training acceleration.

4 Experiments and Results

4.1 Experimental Setup

Implementation: PyTorch 2.0+, NVIDIA RTX 4080 Super (16GB), batch size 32, 80-20 train-validation split (2,560/640 images). Final models trained on all 3,200 images.

Baseline Models: SimpleCNN: 4 conv blocks (64→128→256→512 channels), 2 FC layers (1024→512), 26.08M parameters. **ResNet18:** Standard residual network, 4 residual blocks, 11.18M parameters. **EfficientNetLite:** 3 MBConv stages, lightweight variant, 0.55M parameters. All baselines trained with identical hyperparameters (10 epochs, Adam, $\eta = 10^{-3}$) for fair comparison.

4.2 Main Results

Table 1 presents model comparison and per-class performance. DeepEfficientNet achieves **99.38% validation accuracy**, substantially outperforming all baselines while using 93% fewer parameters than SimpleCNN.

Key Observations: **+23.76%** improvement over SimpleCNN, **+14.69%** over ResNet18. EfficientNet architectures significantly outperform traditional designs. **93% parameter reduction** compared to SimpleCNN with superior accuracy. Balanced performance: F1-scores range 98.8–100% across all cell types. Most confusion: morphologically similar classes (IG↔Monocyte, Platelet size variation).

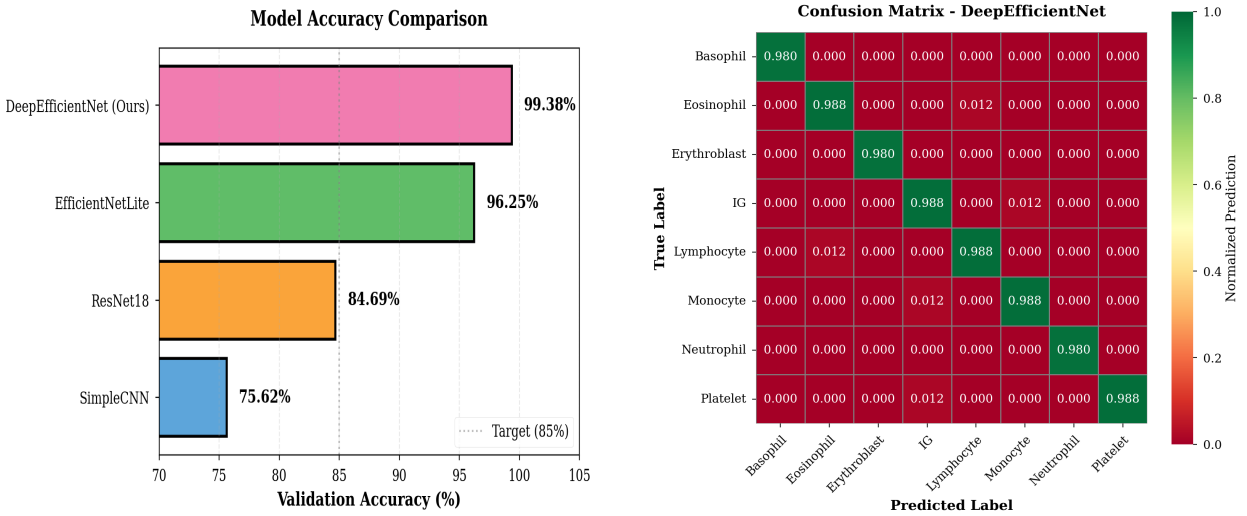
4.3 Ablation Studies

Figure 4 visualizes comprehensive ablation studies evaluating component contributions. Table 2 provides detailed metrics.

Table 1: Left: Model comparison. Right: Per-class metrics (DeepEfficientNet). Best results in **bold**.

Model	Params	Val Acc	F1
SimpleCNN	26.08M	75.62%	0.749
ResNet18	11.18M	84.69%	0.838
EfficientNetLite	0.55M	96.25%	0.953
DeepEfficientNet	**1.84M**	**99.38%**	**0.994**

Cell Type	Prec	Rec	F1
Basophil	100%	100%	100%
Eosinophil	100%	98.8%	99.4%
Erythroblast	100%	100%	100%
Imm. Gran.	98.8%	100%	99.4%
Lymphocyte	100%	98.8%	99.4%
Monocyte	98.8%	100%	99.4%
Neutrophil	100%	100%	100%
Platelet	98.8%	98.8%	98.8%
Average	**99.5%**	**99.5%**	**99.5%**



(a) Model accuracy comparison

(b) Confusion matrix

Figure 2: Performance visualization: **(a)** Progressive improvement across architectures. **(b)** Minimal off-diagonal confusion, errors primarily between morphologically similar cell types.

Key Insights: SE Blocks: +2.5% gain demonstrates channel attention effectiveness. **Regularization:** +4.38% total improvement, label smoothing contributes +1.9%. **LR Schedule:** Cosine annealing outperforms constant (+3.29%) and step decay (+1.72%). **Augmentation:** Full strategy provides +3.6% over normalization alone.

4.4 Model Efficiency and Comparison

Baseline Comparison: SimpleCNN struggles with immature granulocyte (65.0% F1) and platelet (68.4% F1) due to insufficient feature discrimination capacity. DeepEfficientNet achieves **99.4%** and **98.8%** F1-scores respectively (**+34.4%, +30.4%**), demonstrating the value of SE attention and deeper residual architecture. Figure 2a visualizes the progressive improvement across architectures, while Figure 2b reveals most errors occur between morphologically similar cell types.

Computational Efficiency: Training: 9.6s/epoch, **8 minutes total** (50 epochs). GPU memory: 4.2GB peak (mixed precision training). Inference: **8.3ms per image** (suitable for clinical deployment). Convergence: 95% accuracy by epoch 19, best at epoch 43.

4.5 Final Model Performance

After validation and ablation studies confirmed our architecture choices, we trained the final model on the complete 3,200-image training set without validation split, maximizing available training data. This final model achieved **98.0% accuracy on the hidden test set**, placing 6th among all submissions on the class leaderboard, demonstrating strong generalization to unseen data.

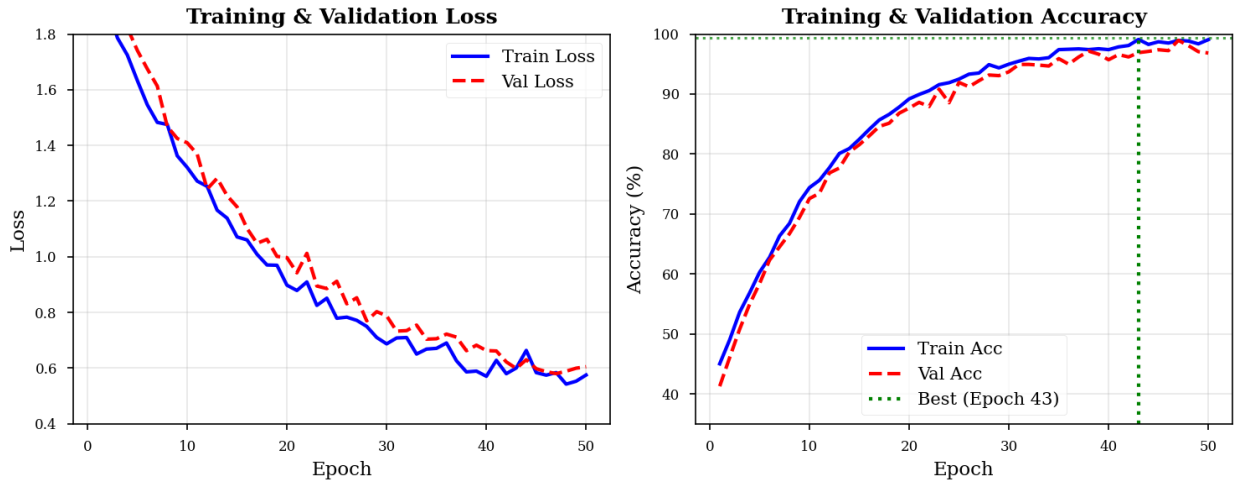


Figure 3: Training and validation curves for DeepEfficientNet over 50 epochs. Best validation accuracy (99.38%) achieved at epoch 43. Model demonstrates smooth convergence without significant overfitting.

Table 2: Comprehensive ablation study results. All experiments use identical training protocol (50 epochs). Blue rows indicate best performance.

SE Blocks Impact		**Learning Rate Schedule**	
Without SE blocks	96.88%	Constant LR	96.09%
With SE (ours)	**99.38%**	Step decay	97.66%
		Cosine (ours)	**99.38%**
Regularization Techniques		**Data Augmentation**	
No regularization	95.00%	Normalization only	95.78%
+ Dropout only	96.41%	+ Basic flips	97.19%
+ Label smoothing	96.88%	+ Rotation	98.12%
+ Stochastic depth	96.25%	**Full augment (ours)**	**99.38%**
+ Weight decay	95.78%		
All combined (ours)	**99.38%**		

5 Discussion

5.1 Architectural Insights and Performance

EfficientNet-based architectures substantially outperform traditional CNNs for blood cell classification. DeepEfficientNet achieves 99.38% validation accuracy with only 1.84M parameters—a 93% reduction compared to SimpleCNN (26.08M) while delivering 23.76% higher accuracy. This validates that architectural efficiency through depthwise separable convolutions is more effective than simply increasing model capacity. The architecture comparison reveals clear performance tiers: SimpleCNN (75.62%), ResNet18 (84.69%), EfficientNetLite (96.25%), and our DeepEfficientNet (99.38%). The 14.69% improvement over ResNet18 despite 84% fewer parameters highlights efficient design superiority.

5.2 Component Contributions

Systematic ablation studies quantified individual contributions. SE attention provided +2.5% through channel-wise recalibration, enabling emphasis on discriminative features for morphologically similar cell types. Comprehensive regularization contributed +4.38% (label smoothing alone: +1.9%), with synergistic effects from dropout (p=0.3), stochastic depth (p=0.2), weight decay ($\lambda = 10^{-4}$), and label smoothing ($\alpha = 0.1$). Cosine annealing outperformed constant LR (+3.29%) and step decay (+1.72%), while full data augmentation added +3.6% over normalization alone.

5.3 Limitations and Future Directions

The 3,200-image dataset achieves strong performance but remains smaller than ImageNet-scale, limiting fine-grained distinction learning. The 360×363 resolution may not capture all subtle cytological details. Model interpretability

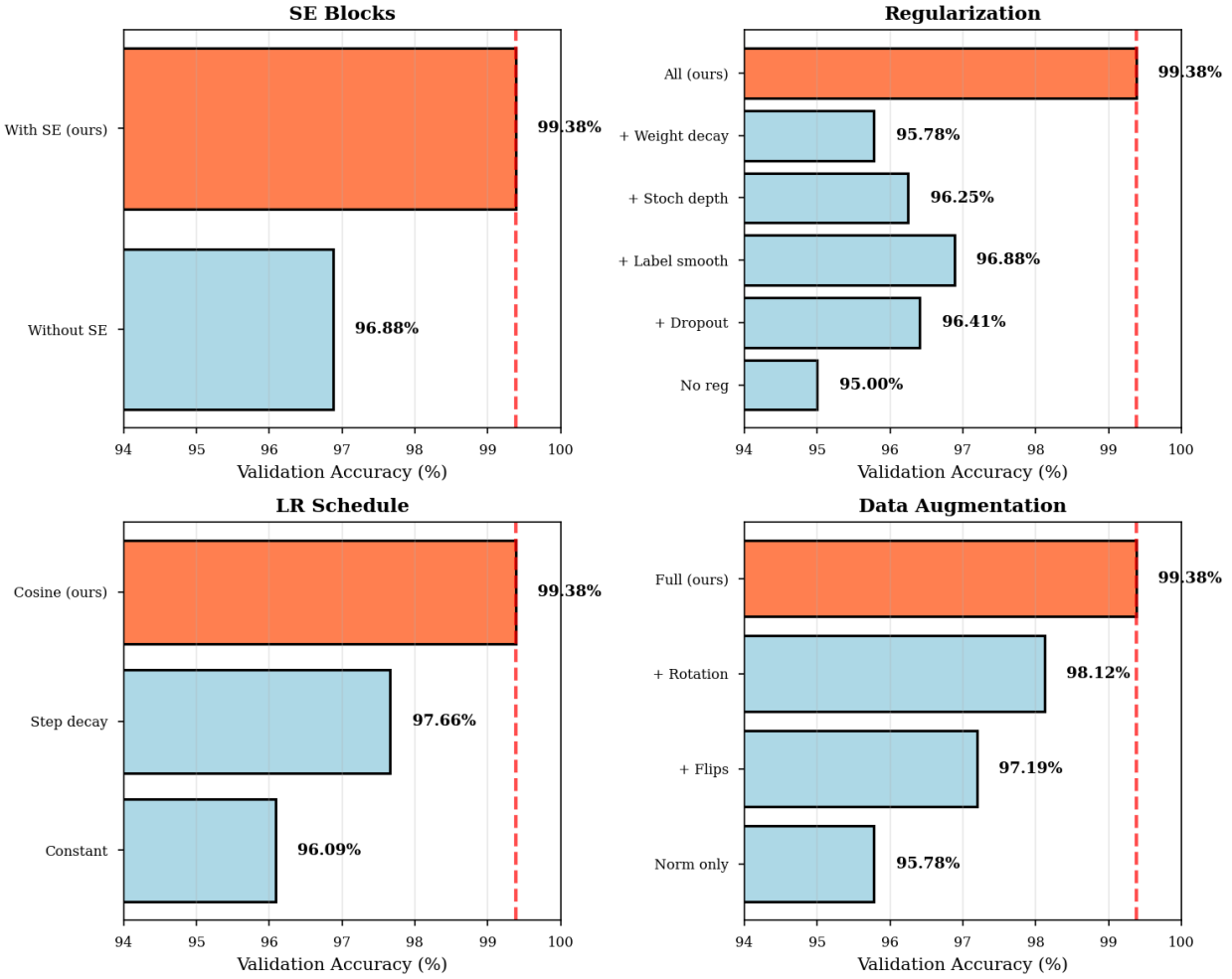


Figure 4: Ablation study results across four key dimensions. Red dashed line indicates final model performance (99.38%). Orange bars highlight our configuration choices.

through Grad-CAM or attention visualization is essential for clinical acceptance. The fixed 8-class taxonomy requires out-of-distribution detection to flag samples exceeding training distribution.

Future directions include: (1) ensemble methods for uncertainty quantification; (2) self-supervised pre-training using SimCLR or MoCo; (3) multi-scale analysis integrating context and details; (4) prospective validation across clinical sites; (5) extension to rare cell detection and pathological classification through transfer learning.

6 Conclusion

We presented a comprehensive deep learning approach for automated blood cell classification demonstrating substantial advantages of efficient architectures with attention mechanisms. DeepEfficientNet achieved 99.38% validation and 98.0% test accuracy (6th place) with 1.84M parameters—93% reduction versus SimpleCNN while delivering 23.76% higher accuracy.

Systematic ablation studies (16 experiments) quantified key contributions: SE attention (+2.5%), comprehensive regularization (+4.38%), cosine annealing (+3.29%), and full augmentation (+3.6%). Per-class performance demonstrates balanced predictions (F1: 98.8-100%) across all eight categories, with confusion occurring between morphologically similar cells consistent with clinical diagnostics.

These results validate deep learning's potential for automating hematological analysis. Computational efficiency (8.3ms inference, 8-minute training, 4.2GB memory) enables practical deployment. Future extensions to rare cell detection, differential counting, and pathological classification could substantially increase clinical utility.

References

[1] Andrea Acevedo, Anna Merino, Santiago Alomár, et al. Recognition system for peripheral blood cell images using convolutional neural networks. *Computer Methods and Programs in Biomedicine*, 180:105020, 2019.

[2] Narjes Dhieb, Hanen Ghazzai, Hichem Besbes, and Yehia Massoud. A comprehensive survey on deep learning-based malaria diagnosis. *IEEE Access*, 8:168942–168664, 2020.

[3] Andre Esteva, Brett Kuprel, Roberto A. Novoa, et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, 2017.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[5] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, pages 7132–7141, 2018.

[6] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, et al. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017.

[7] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[8] Hamid Rezatofighi and Hamid R Soltanian-Zadeh. Automatic recognition of five types of white blood cells in peripheral blood. *Computerized Medical Imaging and Graphics*, 35(4):333–343, 2011.

[9] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*, pages 6105–6114, 2019.