A Hybrid Deep Convolutional Neural Network for Efficient Diabetic Retinopathy Classification

Sandeep Madarapu[a], Samit Ari[a] and Kamalakanta Mahapatra[a]

[a]*Department of Electronics and Communication, National Institute of Technology Rourkela, Rourkela, Odisha-769008, India*

ARTICLE INFO

Keywords:
CAD
CNN
Diabetic retinopathy
ResNet50V2
Channel attention mechanism,
SVM
Xception

ABSTRACT

White blood cells (WBC) are important parts of our immune system and they protect our body against infections by eliminating viruses, bacteria, parasites, and fungi. There are five types of WBC. These are called Lymphocytes, Monocytes, Eosinophils, Basophils, and Neutrophils. The number of WBC types and total WBCs provide important information about our health status. Diseases such as leukemia, AIDS, autoimmune diseases, immune deficiencies, and blood diseases can be diagnosed based on the number of WBCs. In this case, a CNN-based system performs better in recognizing partially visible cells for reasons such as overlap or only partial visibility of the image. Therefore, it has been the motivation of this study to increase the performance of existing blood test devices with deep learning method. Blood cells have been identified and classified by Regional Based Convolutional Neural Networks. Designed architectures have been trained and tested on PBC data set and Rabbin data set.Convolutional Neural Networks (CNN) has been used as a methodology. In this way, different cell types within the same image have been classified simultaneously with a detector. While training CNN which is the basis of CNN architecture; MobileNet_V2, Xception, DenseNet, ResNet50 architectures have been tested with full learning and transfer learning. At the end of the study, the system has showed 100% success in determining WBC cells. MobileNet_V2, one of the CNN architectures, has showed the best performance with transfer learning. With This Model we got an accuracy of 99.57% for PBC Dataset and 98.54% for Rabbin Dataset.

1. Introduction

Normal human blood has three different types of cells. WBCs are the larger cells with the darker material in the nucleus, platelets are the smaller scattered cells, and RBCs are the smaller solid pieces.[1] WBCs, also known as immune cells, play a vital role in the immune system of the human body when present in blood. WBC protects the body from external invaders and infectious illnesses.[2] Haematologists typically divide white blood cells into two categories using their basic understanding of medical applications: (a) granular cells and (b) non-granular [3]. White blood cells are part of the immune system and are produced in bone marrow and lymphoid tissues. They protect the body against infections such as bacteria, viruses and fungi. There are five types of white blood cells. They are called Lymphocytes, Monocytes, Eosinophils, Basophils and Neutrophils [4][5]. Numerous disorders can result from either an excess or a deficiency of white blood cells. Blood tests are used to make these disease diagnoses. [6]

The study of white blood cell classification has emerged as a crucial and vital area of study for medical analysis. [7] The necessary step in the diagnosis of leukaemia by the attending physician is to classify the white blood cells in the bone marrow, which requires the attending physician to have a wealth of clinical experience [8]. When the number of WBC is less than the reference value, it is known as leukopenia.[9] Neutrophils show an increase in blood in cases of hormonal causes, metabolic disorders, hemolysis and bleeding.[10] The process of manually identifying and classifying WBCs is slow and labor-intensive, requiring significant time from skilled technicians.

Convolutional neural networks (CNNs) have been a major breakthrough in a number of industries, including image and video recognition, medical image analysis[11], and other domains where automated feature extraction and classification are needed. When using Convolutional Neural Networks (CNNs) to classify white blood cells (WBCs), several dataset-related challenges can arise, impacting model performance and generalizability. Class imbalance, while data quality issues like noisy or blurry images degrade learning. Annotation errors due to human subjectivity, variability in imaging conditions, and limited data availability further complicate training. High intra-class variability and class overlap make distinguishing between cell types difficult, and data augmentation challenges can introduce unrealistic artifacts. High dimensionality of images demands significant computational resources, and class distribution drift over time or across datasets can affect model accuracy. Addressing these challenges

Sandeep Madarapu (S. Madarapu); Samit Ari (S. Ari)
madarapusandeep@gmail.com (S. Madarapu); samit.ari@gmail.com (S. Ari)
ORCID(s): 0009-0009-4413-9833 (S. Madarapu); 000-0002-5599-081X (S. Ari)

requires careful dataset curation, preprocessing, augmentation, and robust modeling techniques, including transfer learning for improved generalization.

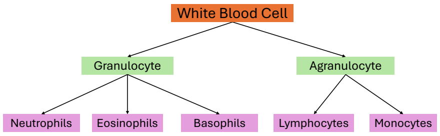

Figure 1: types of white blood cell.

The contributions of the proposed work can be summarized as follows [12]:

- The suggested architecture combines two commonly used convolutional neural network (CNN) architectures, Xception and MobileNetV2, to use their complementing strengths. The model captures different information from both architectures by concatenating their feature maps, boosting representative capacity.

- The suggested method improves the accuracy of image classification by making use of the DKCAB and MSFFB block attention mechanisms. The DKCAB improves classification accuracy by locating and emphasizing important characteristics in a picture.

- Compared to existing CNNs, the suggested model creates a more computationally efficient network with a significantly lower average inference time and fewer trainable parameters.

- Extensive tests conducted on two publicly accessible WBC datasets—the PBC and Rabbin datasets—show that our suggested approach performs better than current state-of-the-art approaches in terms of computational time and accuracy. Furthermore, a thorough quantitative study confirms the method's effectiveness in classifying WBC and offers insightful information about its possibilities.

- The contributions of this work lie in the effective integration of multiple architectures, the utilization of a powerful attention mechanism. These contributions collectively improve the accuracy of the proposed model in the classification of white blood cell.

The remaining sections are organized as follows: Section 3 describes the classification methodology proposed for fundus images, Section 4 contains experimental results and discussion, and Section 5 provides a conclusion.

2. Related Works

The many approaches that are currently being used for the segmentation, classification, and counting of white blood cells will be illustrated in this chapter. These approaches use diverse image processing techniques. Additionally, we will talk about a few of the newest CNN types—more precisely, the ones we used for our suggested methodology.

A multitude of techniques have recently been introduced for WBC classification from optical wbc images, and these may be broadly classified into two groups: deep-learning techniques and machine-learning techniques.

a) Machine-Learning Method

b) Deep-Learning Method

3. Methodology

This part includes a discussion of the suggested CNN model architecture, a thorough set of preprocessing techniques used to increase the dependability of fundus images, and the database used to evaluate the suggested method are all covered in this part.

3.1. Dataset

Two public WBC datasets Rabbin dataset and PBC dataset are utilized for this article to validate the performance of the proposed CNN model, as will be explained below.

3.1.1. Rabbin dataset

: Raabin White Blood Cell (WBC) dataset which consisted of 14514 WBC images across five classes 301 basophils, 795 monocytes, 1066 eosinophils, 8891 neutrophils, and 3461 lymphocytes at resolutions of 575 x 575. The data set mainly focuses on the classification of white blood cells. We selected each type of white blood cells and introduced them into our new data set. Rabbin dataset is utilized for the classification of WBC, which offers a comprehensive collection of images specifically designed for WBC categorization. This dataset is pivotal for distinguishing between different types of white blood cells, aiding in the accurate diagnosis and treatment of various hematological conditions. The Rabbin dataset ensures a thorough and nuanced understanding of WBCs. It includes images of different types of WBCs such as neutrophils, lymphocytes, monocytes, eosinophils, and basophils, which are critical for diagnosing diseases and monitoring immune responses. The meticulous curation and extensive volume of the Rabbin dataset enable robust training and validation of machine learning models, facilitating high-accuracy classification and contributing significantly to advancements in medical diagnostics.

PBC dataset: The Peripheral Blood Cell (PBC) dataset consists of 17,092 images. These images are further organized into the following eight groups: neutrophils, eosinophils, basophils, lymphocytes, monocytes, immature granulocytes (including promyelocytes, myelocytes, and metamyelocytes), erythroblasts, and platelets or thrombocytes. Each image is 360 x 363 pixels in size and is in JPG format, annotated by expert clinical pathologists. This dataset focuses on images of peripheral blood cells. For our newly introduced dataset, we have selected five types of white blood cells from this dataset The PBC (Peripheral Blood Cells) dataset is a collection of high-resolution images used primarily for the classification and analysis of white blood cells (WBCs). This dataset is crucial for developing and testing machine learning models aimed at automating the identification and classification of various types of WBCs, including neutrophils, lymphocytes, monocytes, eosinophils, and basophils. Each image in the dataset typically includes detailed annotations and labels, allowing researchers and developers to train models to recognize subtle morphological differences and abnormalities in WBCs, which are vital indicators in medical diagnostics and disease monitoring. The PBC dataset serves as a valuable resource in advancing automated systems for hematological analysis, contributing to improved accuracy, efficiency, and accessibility in healthcare diagnostics.

3.2. Pre-processing

For the white blood cell (WBC) classification task , To ensure high accuracy and reliability of the model the data is meticulously prepared . The dataset, comprising images of various WBC types such as neutrophils, lymphocytes, monocytes, eosinophils, and basophils, underwent thorough data cleaning to remove duplicates and correct labeling errors. Image preprocessing involved resizing, normalizing pixel values, and applying contrast enhancement and noise reduction techniques to improve quality. Data augmentation, including rotations, flips, shifts, and zooms, was employed to expand the dataset and prevent overfitting. The dataset was then split into training, validation, and test sets, ensuring a balanced distribution of WBC types across each set. Labels were encoded into a numerical format suitable for machine learning algorithms, and feature extraction techniques were applied to highlight significant image features. These steps ensured that the Rabbin dataset was well-organized and high-quality, enabling the training of robust and accurate WBC classification models.

3.3. Proposed methodology

Figure 2 shows that the proposed architecture consists of a base model, DKCAB block, CAB block, and multi-scale feature fusion. Each component plays a critical role in ensuring high classification accuracy.The base model consists of a MobilVnet backbone, which it utilizes as a base feature extractor. The DKCAB block generates attention maps, which enhances feature learning by adaptively selecting different receptive fields for feature extraction. To learn rich, hierarchical representations that capture the fine details and the overall context of the input images, multi-scale feature fusion combines feature maps from various stages of the network and blocks aggregate features from multiple scales. The CAB is utilized to improve feature refinement by focusing on important areas within the feature maps, enhancing discriminative feature learning and reducing noise . This combination extracts informative features by utilizing the attention mechanism provided by DKCAB and CAB, which are subsequently aggregated using MSFF. After training, the features are taken from the base model. They are typically upsampled and maxpooled except one, which passed to DKCAB for training. Then all features are provided to Multi-Scale Fusion for training. the features are taken from it and then provided to CAB for training . The Dense output layer with the softmax activation function for classification was reached at the end. the layer wise description of the proposed network in table:

3.3.1. backbone

MobileNetV2 is a lightweight and efficient convolutional neural network designed specifically for mobile and edge devices. It was introduced by Google in 2018mention about the paper of it and is known for its depthwise separable convolutions, which significantly reduce the number of parameters and computation costs compared to traditional convolutions.it is built using a series of bottleneck residual blocks with shortcut connections between them . expansion layer expands the input channels by a factor of 't'. It uses a 1x1 convolution to increase the number of channels. The Depthwise convolution layer performs a spatial convolution separately over each output channel from the expansion layer . this operation retains the spatial structure while reducing computational costs . projection layer reduces the number of channels back to a smaller dimension . It preserves the feature information from the depthwise convolution. mobileNet_V2 incorporates skip connections in its bottleneck blocks when input and output channels are the same . These connections help to preserve the input information, enabling the network to learn more effectively. We have used only three three significant layer from the MobileNetV2 . These layers are chosen to capture low-level, mid-level, and high-level features, respectively. We have frozen the layers of mobileNetV2, ensuring that the pre-trained weight remains intact and only the DKCAB, CAB, and multi-fusion scale blocks are trained. This transfer learning approach helps to leverage the rich features learned from the large ImageNet dataset, thus contributing to better generalization and higher accuracy.

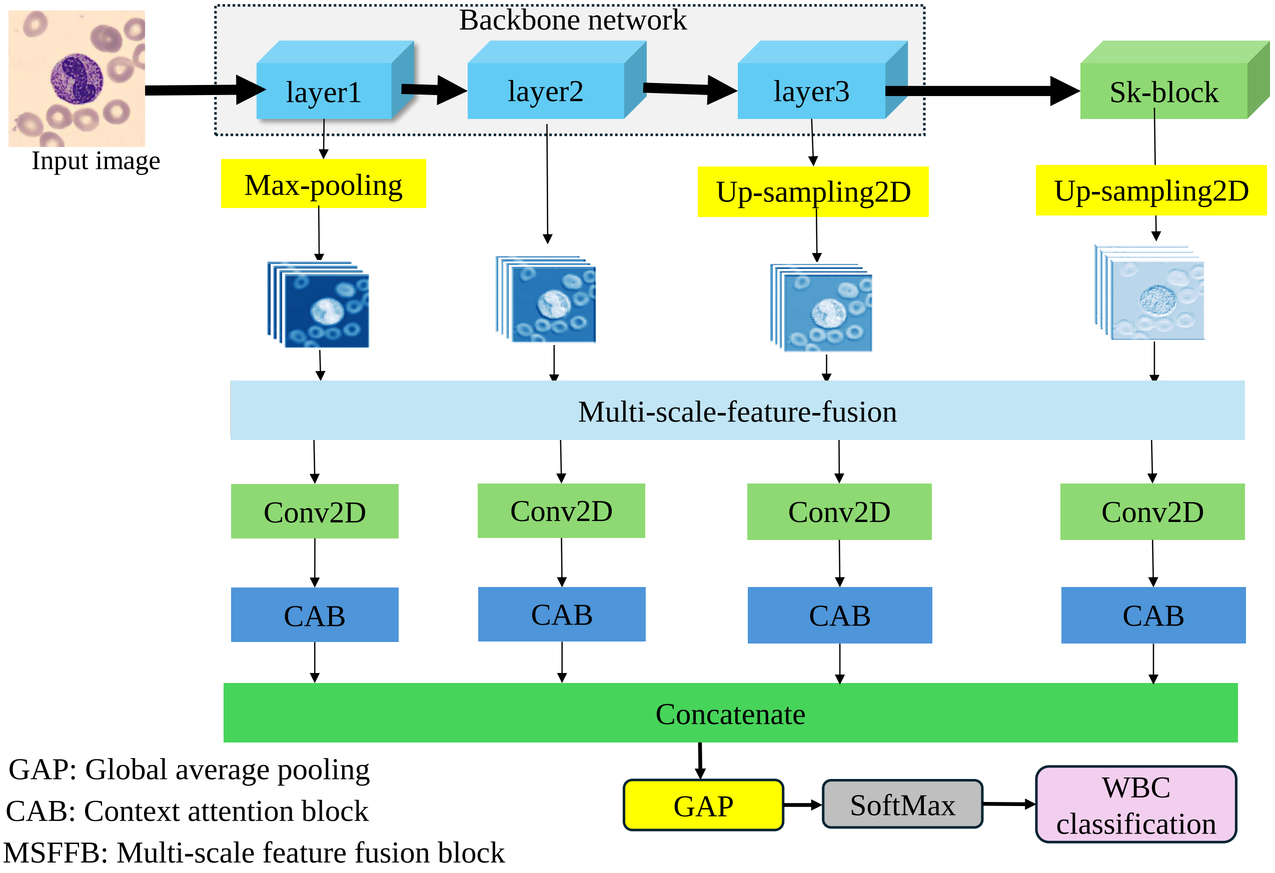

Figure 2: Classification of a white blood cell.

3.3.2. CAB block

The Context Attention Block is designed to capture important features by focusing on spatial context and feature relationships. It uses a combination of convolutions (point-wise, depth-wise, and dilated) to capture both fine-grained details and border context features from the input image. CAB generates an attention map through a combination of convolutional layers and element-element- wise multiplication. By reweighting the features based on their relevance. By applying attention to the input features, this effectively enhances the most discriminative parts of the input image, such as edges, textures, and regions with high contrast. This process helps the model focus on the most important parts of the image, such as the nucleus and cytoplasm of WBC, which are crucial for differentiating between different types of cells.

consider the input feature map

$$F_{in} \in R^{H\times W\times C}$$

from the previous block that have the dimension _____ where h and w represent the spatial dimension , precisely height and width , whereas C denotes the channel count.The extracted features are subjected to point-wise convolution with 1x1 kernel size along with the gelu activation function. this operation is crucial for reducing the dimensionality of the input feature and transforming the input channel space to new space .

$$X_Z \in f_{conv}^{1\times 1}\,(F_{in}/$$

X_Z has spatial dimensions $h\times w\times c$ and undergoes $f_{conv}^{1\times 1}$

represents the convolution operation with 1x1 kernel size.this operation is crucial for reducing the dimensionality of the input feature and transforming the input channel space to new space. XZ undergoes DW operations with a series of depthwise convolutions with different dilation rate. It is subjected to two depthwise convolution of kernel size 5x5 and kernel size 7x7 with different dilation rate along with point wise convolution of 1x1 kernel size .

$$X_{Att} \in f_{conv}^{1\times 1}(f_{Dconv}^{5\times 5,dr=3}(f_{Dconv}^{7\times 7,dr=5}(X_Z)))$$

$$X_{Att}$$

has the dimension $h \times w \times c$, and $f_{Dconv}^{5\times 5,dr=3}$ represents the depthwise convolutions with kernel size 5×5 and dilation rate $= 3$. Similarly, $f_{Dconv}^{7\times 7,dr=5}$ represents the

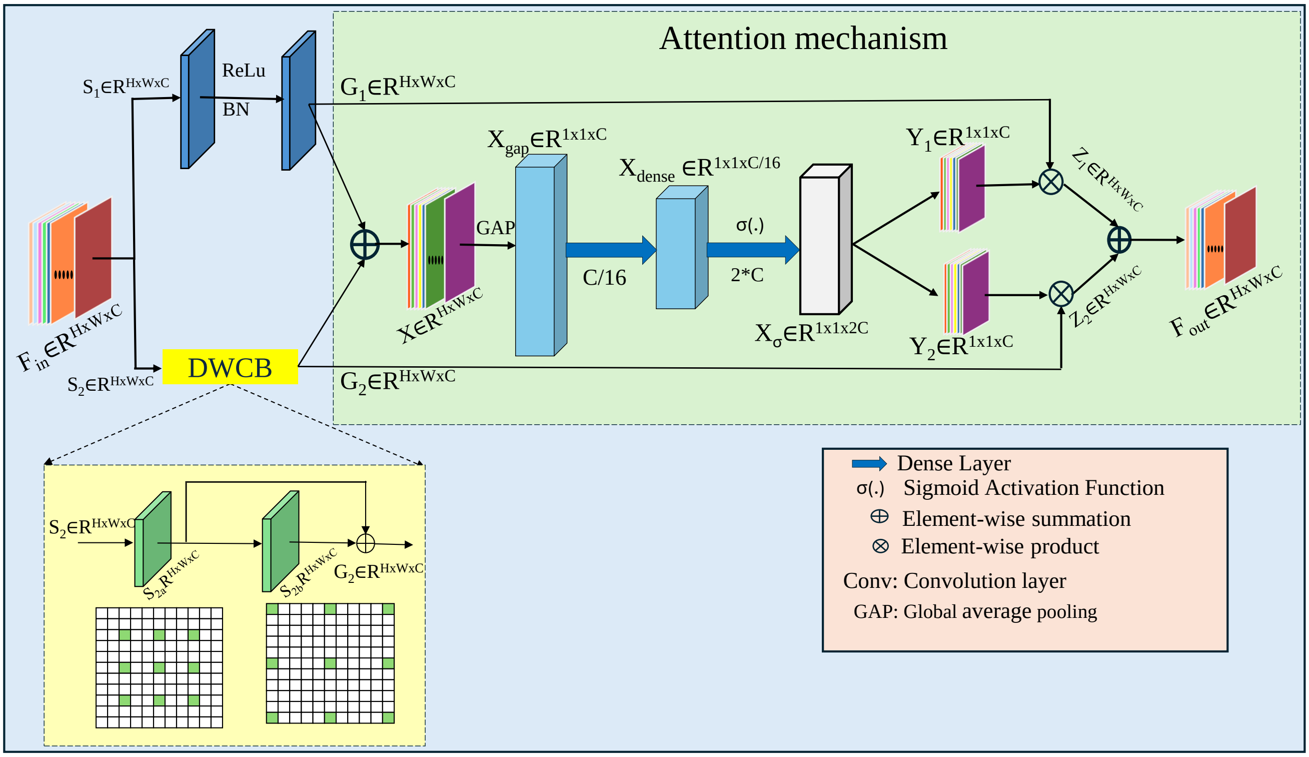

Figure 3: types of white blood cell.

next layer. These convolutions introduce gaps between kernel elements, allowing the network to have a large receptive field without losing resolution.

$$X_u \in (F_{in} \otimes (X_{Att} \otimes X_Z))$$

$$X_u \in h \times w \times c$$

,similarly

$$X_{Att}, F_{in}$$

have the same dimension .

$$\otimes$$

represents the multiplication of all these . this intergerates attention mechanism to focus on the most informative features. XU undergoes point wise convolution to reduce the dimensionality of the feature map and transforming the input channel to new space.Xu undergoes point wise convolutions with depthwise separable convolution with kernel 3x3 for depthwise convolution and 1x1 for convolutions.

$$X_{XL} \in f^{1\times1}_{conv}(f^{3\times3}_{conv}(f^{1\times1}_{conv}(X_U)))$$

have the dimension

$$h \times w \times c$$

. it first extracts spatial features per channel and then merges them across channels , maintaining efficiency while preserving rich feature representation .

$$X_{XL}$$

undergoes multiplication with the

$$X_U$$

.

$$X_O \in X_U \otimes X_{XL}$$

The final output is obtained by aggregating the original input features and the depthwise -seperable and feature fusion.

3.3.3. *MultiScale feature fusion block*

The Multi-Scale Feature Fusion Block is designed to concatenate features extracted at different scales, combining both fine and coarse details. This fusion allows the model to make more informed decisions by leveraging diverse feature information from the multiple levels. The input features F1, F2, F3, and F4 from the base model and DKCAB block have the shapes HxWxC,

The concatenated feature map Fconcat will have the shape HxWx(4C). after concatenation , a 1x1 convolution is applied to the concatenated features to mix the information across channels while maintaining the spatial dimensions HxW.

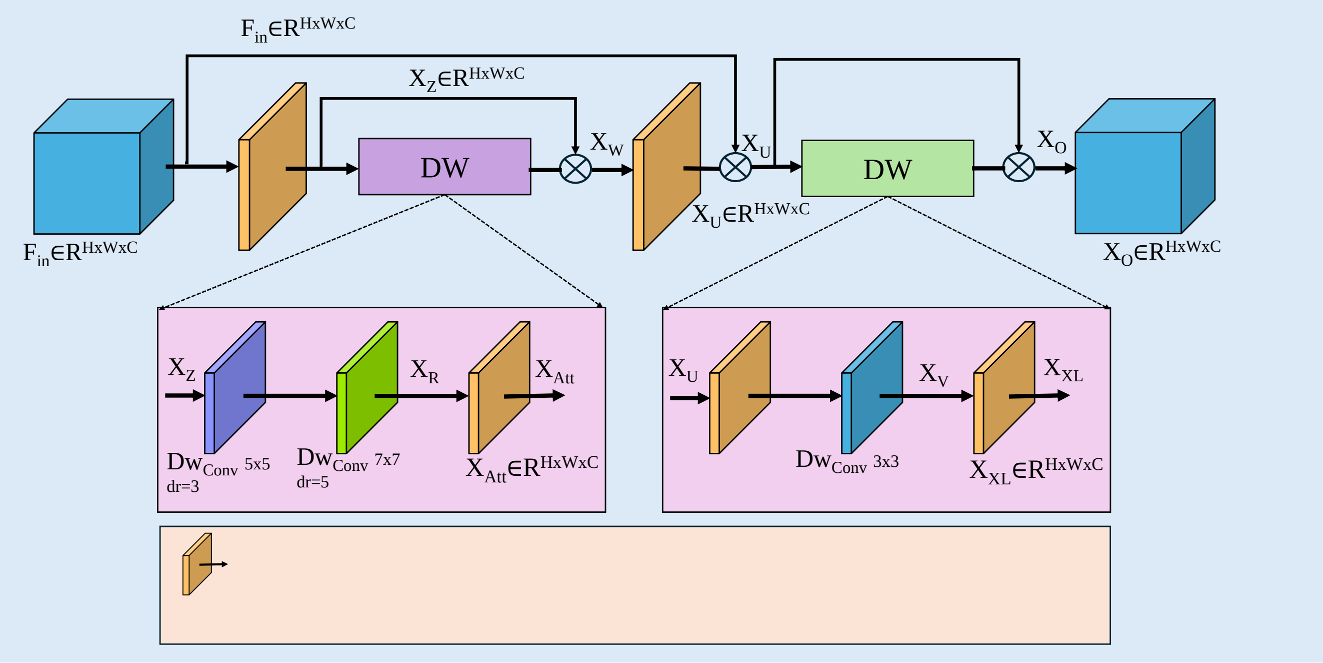

Figure 4: confusion matrix for multi-class classification of the proposed model on APTOS-2019 dataset.

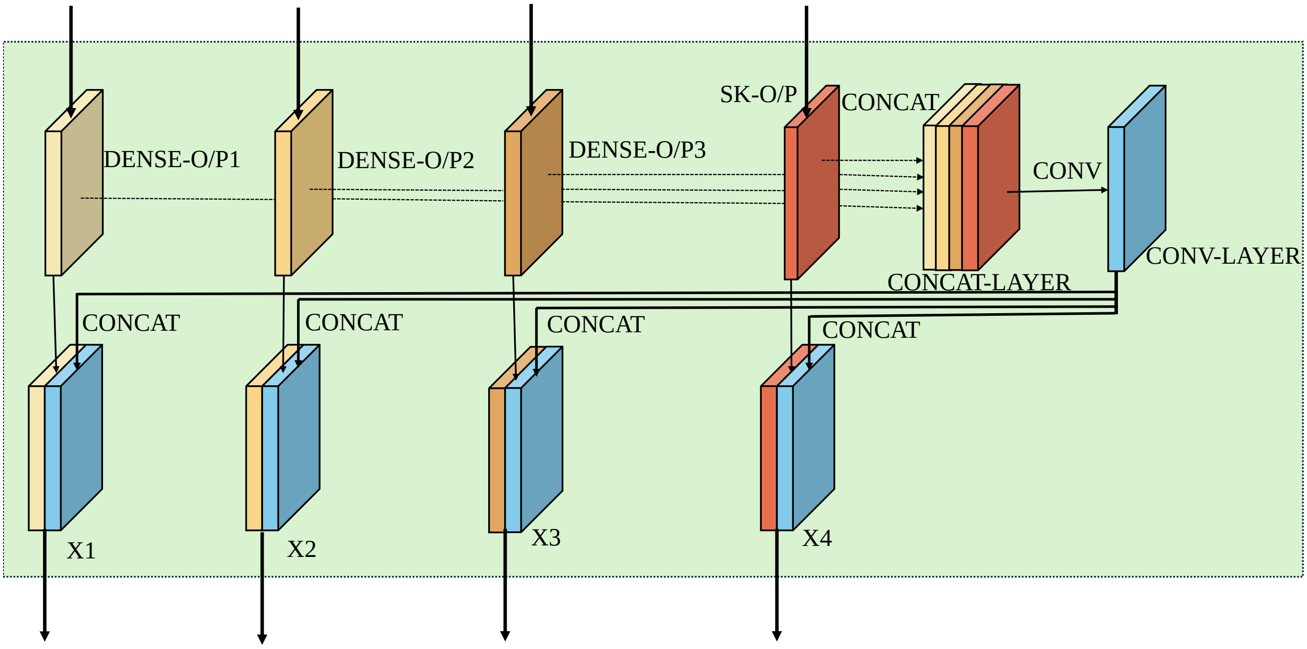

Figure 5: confusion matrix for multi-class classification of the proposed model on APTOS-2019 dataset.

3.3.4. DKCAB Block

Dilated kernel Convolution Attention block is designed to adaptively select features from different Convolution branches,allowing model to focus on contextually relevant features at multiple scales. it consist of convolutional branches , feature fusion , global pooling and attention mechanism using fully connected (FC) layers. consider that the features extracted from the previous layer have the dimensions Fin belongs to define the dimension of h and w the extracted feature are subjected to series of convolution in branch 1 with layerwise structure of proposed network

Layer Name	Output Size	Layer
M1	28×28×144	INPUT
M2	14×14×192	INPUT
M3	7×7×576	INPUT
M4(SK)	7×7×64	M3
F1	14×14×144	M1
F3	14×14×576	M3
SK(o/p)	14×14×64	M4
MSB	14×14×128	[F1, M2, F3, SK(op)]
Conv1	14×14×128	MSB[0]
Conv2	14×14×128	MSB[1]
Conv3	14×14×128	MSB[2]
Conv4	14×14×128	MSB[3]
Cs1	14×14×128	Conv1
Cs2	14×14×128	Conv2
Cs3	14×14×128	Conv3
Cs4	14×14×128	Conv4
CONCAT	14×14×128	[Cs1, Cs2, Cs3, Cs4]
GAP	14×14×128	CONCAT
SOFTMAX	14×14×128	GAP
FINAL O/P	1×1×5	SOFTMAX

different kernel size 3x3 and 5x5 along with relu and batch normalization

then dimension of

in parallel the extracted feature Fin are also subjected to dilated convolution in branch 2 with different kernel size 3x3 , 5x5 and dilation rate =3,5 *Split:* For every given feature map:$F : X \to U \in \mathbb{R}^{H\times W\times C}$,we perform two transformations: $f_1 = \left(\left(f_{\text{conv}}^{3\times3}\{F\}\right)\right)$, $f_2 = \left(f_{\text{DConv}}^{5\times5}\left(f_{\text{DConv}}^{5\times5}\{F\}\right)\right)$ with kernel sizes 3x3 and 5x5, respectively. In f_1, we perform groupwise convolution (Conv2D) using a kernel size of 3x3, followed by batch normalization and the ReLU function. Within the pictures of the white blood cells, the 3x3 convolutions layer pick up minute patterns, edges, and textures. They are crucial for spotting minute details.In f_2, we perform series depthwise convolution (DConv2D) using the same kernel size of 5x5 with different dilation rates of 3 and 5, respectively. As a result, the network can have a bigger Receptive Field (RF) without adding more parameters by using different dilation rates. Differentiating between features at different sizes is crucial for identifying distinct structural aspects of white blood cells, and each DConv2D layer has the ability to "see" more of the input picture and record more information. Again split and reshaped: $f_{s1} = f_x\left\lfloor :, :, : \frac{c}{2}\right\rfloor$, $f_{s2} = f_x\left\lfloor :, :, \frac{c}{2} :\right\rfloor$

The feature maps' form is matched by splitting and reshaping the attention weights. The output of each branch can now have its attention mechanism applied independently.

Fuse: f3=con[f1,f2 By fusing the outputs of branches (f1) and (f2), the network can incorporate both contextual and local features (f3). This fusion improves the model's comprehension and interpretation of intricate details in the white blood cell images, resulting in better classification performance. another fusion in end of block

The attention weights for f1 and f2 are multiplied by the feature maps f(s1) and f(s2). The next step then two dense layer: f(X)=f^(Dense, 2*C, activation=softmax f^(Dense ,C/16,activation= relu f(gap)

4. Experimental Results

4.1. Experimental setup

Our framework is implemented using Keras, with TensorFlow functioning as the backend. During training, the cross-entropy loss function and the Adam optimization technique are employed. The CNN is trained with the following hyperparameters set: a learning rate of 0.001 and a maximum of 50 epochs. To reduce overfitting, real-time data augmentation techniques including random rotation, vertical flipping, and horizontal flipping are applied.

4.2. Evaluation metrics

4.3. Ablation Study

The impact of the CSCA module, multiscale feature fusion block, ACSConv-based ALA module, and various backbones on the recognition outcomes is examined in this part through a series of ablation experiments.

1) Effectiveness of ACSConv-Based ALA Module:

2) Effectiveness of CSCA Module:

3) Effectiveness of Multiscale Feature Fusion Block:

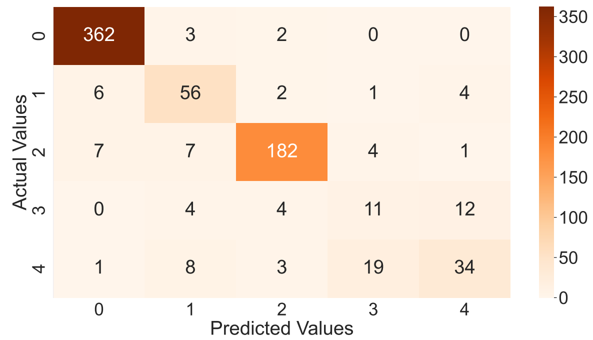

Figure 6: confusion matrix for multi-class classification of the proposed model on APTOS-2019 dataset.

layerwise structure of proposed network for PBC Dataset

Model	Accuracy	QWK
Mobilenet_v2	89.57	87.15
Mobilenet_v2+DKCAB	98.56	97.91
Mobilenet_v2+CAB	99.47	99.51
Mobilenet_v2+DKCAB+CAB	99.58	99.31
Xception	90.90	87.63
Xception+DKCAB	96.96	96.65
Xception+CAB	99.36	99.14
Xception+DKCAB+CAB	99.41	99.39
ResNet	91.48	87.46
ResNet+DKCAB	97.39	95.42
ResNet+CAB	99.25	98.86
ResNet+DKCAB+CAB	98.77	98.34
DenseNet	96.86	96.68
DenseNet+DKCAB	98.82	98.85
DenseNet+CAB	99.46	99.45
DenseNet+DKCAB+CAB	98.88	98.84

layerwise structure of proposed network for Rabbin Dataset

Model	Accuracy	QWK
Mobilenet_v2	86.05	65.50
Mobilenet_v2+DKCAB	96.70	91.20
Mobilenet_v2+CAB	98.11	95.98
Mobilenet_v2+DKCAB+CAB	98.54	96.62
Xception	9	8
Xception+DKCAB	9	9
Xception+CAB	9	9
Xception+DKCAB+CAB	97.85	95.49
ResNet	9	8
ResNet+DKCAB	9	9
ResNet+CAB	9	9
ResNet+DKCAB+CAB	97.28	93.05
DenseNet	9	9
DenseNet+DKCAB	9	9
DenseNet+CAB	9	9
DenseNet+DKCAB+CAB	98.15	95.81

4.4. Results for binary class classification on the APTOS-2019 dataset.

5. Conclusion

This work presents a comprehensive method for DR classification that

References

[1] N. D. Jambhekar, Red blood cells classification using image processing, Science Research Reporter 1 (3) (2011) 151–154.

[2] Y. Y. Baydilli, Ü. Atila, Classification of white blood cells using capsule networks, Computerized Medical Imaging and Graphics 80 (2020) 101699.

[3] C. Cheuque, M. Querales, R. León, R. Salas, R. Torres, An efficient multi-level convolutional neural network approach for white blood cells classification, Diagnostics 12 (2) (2022) 248.

[4] H. Kutlu, E. Avci, F. Özyurt, White blood cells detection and classification based on regional convolutional neural networks, Medical hypotheses 135 (2020) 109472.

[5] M. J. Macawile, V. V. Quiñones, A. Ballado, J. D. Cruz, M. V. Caya, White blood cell classification and counting using convolutional neural network, in: 2018 3rd International conference on control and robotics engineering (ICCRE), IEEE, 2018, pp. 259–263.

[6] A. Veronelli, M. Laneri, R. Ranieri, D. Koprivec, D. Vardaro, M. Paganelli, F. Folli, A. E. Pontiroli, White blood cells in obesity and diabetes: effects of weight loss and normalization of glucose metabolism, Diabetes care 27 (10) (2004) 2501–2502.

[7] A. Patil, M. Patil, G. Birajdar, White blood cells image classification using deep learning with canonical correlation analysis, Irbm 42 (5) (2021) 378–389.

[8] M. Jiang, L. Cheng, F. Qin, L. Du, M. Zhang, White blood cells classification with deep convolutional neural networks, International Journal of Pattern Recognition and Artificial Intelligence 32 (09) (2018) 1857006.

[9] L. Carli, C. Tani, S. Vagnani, V. Signorini, M. Mosca, Leukopenia, lymphopenia, and neutropenia in systemic lupus erythematosus: prevalence and clinical impact—a systematic literature review, in: Seminars in arthritis and rheumatism, Vol. 45, Elsevier, 2015, pp. 190–194.

[10] B. Amulic, C. Cazalet, G. L. Hayes, K. D. Metzler, A. Zychlinsky, Neutrophil function: from mechanisms to disease, Annual review of immunology 30 (2012) 459–489.

[11] I. Bankman, Handbook of medical image processing and analysis, Elsevier, 2008.

[12] S. Madarapu, S. Ari, K. Mahapatra, A multi-resolution convolutional attention network for efficient diabetic retinopathy classification, Computers and Electrical Engineering 117 (2024) 109243.

$$F : X \rightarrow U \in \mathrm{R}^{H\times W\times C}$$

$$F : X \rightarrow U \in \mathrm{R}^{H\times W\times C}$$

$$F : X \rightarrow U \in R^{H\times W\times C}$$

$$f_1 = \left\lfloor \left\lfloor f_{conv}^{3\times 3}\left\{F\right\} \right\rfloor \right\rfloor$$

$$f_2 = \left\lfloor f_{DConv}^{5\times 5} \left\lfloor f_{DConv}^{5\times 5}\left\{F\right\} \right\rfloor \right\rfloor$$

$$f_3 = conc.\left[f_1, f_2\right]$$

$$f_{GAP} = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W}\left\{f_3\right\}$$

$$f(x) = Soft\max\left(Dense_{2C}\left(\mathrm{Re}LU\left(Dense_{\frac{C}{16}}\left(f_{GAP}(x)\right)\right)\right)\right)$$

$$f_{s1} = f_x\left\lfloor :,:,:\frac{c}{2}\right\rfloor$$

$$f_{s2} = f_x\left\lfloor :,:,\frac{c}{2}:\right\rfloor$$

$$f_{f1} = \left[f \otimes f_{s1}\right]$$

$$f_{f2} = \left[f \otimes f_{s2}\right]$$

$$f_f = f_{f1} \oplus f_{f2}$$