

Received March 24, 2022, accepted April 1, 2022, date of publication April 6, 2022, date of current version April 14, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3165193

Automatic Severity Classification of Diabetic Retinopathy Based on DenseNet and Convolutional Block Attention Module

MOHAMED M. FARAG¹, MARIAM FOUAD^{1,2}, AND AMR T. ABDEL-HAMID¹

¹Department of Electronics Engineering, German University in Cairo, New Cairo 11835, Egypt

²Chair of Medical Engineering, Ruhr Universitat Bochum 102148, Germany

Corresponding author: Mohamed M. Farag (mohamedfarag2000@icloud.com)

ABSTRACT Diabetic Retinopathy (DR) - a complication developed due to heightened blood glucose levels - is deemed one of the most sight-threatening diseases. Unfortunately, DR screening is manually acquired by an ophthalmologist, a process that can be considered erroneous and time-consuming. Accordingly, automated DR diagnostics have become a focus of research in recent years due to the tremendous increase in diabetic patients. Moreover, the recent accomplishments demonstrated by Convolutional Neural Networks (CNN) settle them as state-of-the-art for DR stage identification. This paper proposes a new automatic deep-learning-based approach for severity detection by utilizing a single Color Fundus photograph (CFP). The proposed technique employs DenseNet169's encoder to construct a visual embedding. Furthermore, Convolutional Block Attention Module (CBAM) is introduced on top of the encoder to reinforce its discriminative power. Finally, the model is trained using cross-entropy loss on the Kaggle Asia Pacific Tele-Ophthalmology Society's (APTOS) dataset. On the binary classification task, we accomplished (97% accuracy - 97% sensitivity - 98.3% specificity - 0.9455, Quadratic Weighted Kappa score (QWK)) compared to the state-of-the-art. Moreover, Our network showed high competency (82% accuracy - 0.888 (QWK)) for severity grading. The significant contribution of the proposed framework is that it efficiently grades the severity level of diabetic retinopathy while reducing the time and space complexity required, which demonstrates it as a promising candidate for autonomous diagnosis.


INDEX TERMS Diabetic retinopathy, convolutional neural networks (CNN), attention mechanism, deep learning.

I. INTRODUCTION

Diabetes Mellitus is a chronic metabolic disease characterized by elevated blood glucose levels or (Hyperglycemia), which over time affects the blood vessels in the human body on both micro and macro scales. According to the World Health Organization (WHO), the number of diabetic people hiked to 422 million in 2014, with an expectation to reach 700 million by 2045 [1], [2]. One of the long-term diabetic micro-vascular effects is diabetic retinopathy, a progressive abnormality revealed and detected through ocular pathologies, which leads to blocking and bleeding of the retinal capillaries. Fortunately, early detection can prevent vision impairment. However, without frequent screening, it may induce irreversible damage. International Diabetes Federation (IDF) affirmed that 93 million diabetics suffer from

eye damage, yet only 200,000 ophthalmologists are available worldwide [3]. Grading inconsistency, critical deficiency in the available number of ophthalmologists as well as the laborious process remains hindering factors for diabetic retinopathy detection. Therefore, automating retinopathy diagnostics is desired to reduce the high strain on health care systems. Motivated by this, significant efforts have been directed to enhance Computer-aided medical diagnosis (CAMD) systems.

DR grading systems can be categorized into two clusters: segregation of diabetic retinas from healthy ones (binary-classification task) and severity estimation (multi-class classification task) of affected retinas from class 0 (healthy) to class 4 proliferative DR (PDR). Traditional Machine Learning (ML) algorithms are Artificial Intelligence (AI) techniques that learn through experience by being exposed to data. They were employed for detecting diabetes type based on patient attributes by Nagaraj *et al.* [4], they utilized the

The associate editor coordinating the review of this manuscript and approving it for publication was Chulhong Kim .

Artificial Flora Algorithm (AFA) [5] for feature selection in addition to using Gradient Boosted Trees (GBT) [6] as a classification model. Furthermore, exploited by Gharaibeh *et al.* in [7] and [8] by employing feature engineering process, then applying Support Vector Machines (SVM) as a classifier for DR detection [9]. ML algorithms need personalized experience and domain knowledge to find the most informative representation despite its effectiveness.

Deep Learning (DL) has gained a foothold in various fields by representing the world as a nested hierarchy of concepts, with each concept defined through its relation to simpler concepts [10]. Convolutional Neural Networks was the standout DL architecture in the late nineties. Since then, it has been used extensively for processing data such as images and time series. Moreover, it has demonstrated outstanding performance in practical applications such as Natural Language Processing (NLP) [11], [12] and Computer Vision (CV) problems [13]–[15].

Exploiting convolutional neural networks' power for a medical domain has developed more robust solutions, specifically in the DR domain. [16] and [17] demonstrated the effectiveness of such a technique for retinal vessel segmentation. Similarly, by leveraging Generative Adversarial Networks (GANs), Zhao *et al.* [18] could synthesize fundus images. Dai *et al.* [19] utilized multi-sieving convolutional neural network and image to text mapping for Micro-aneurysms (MA) early detection. [20] evaluated the performance of three recognized CNN architectures: VGG16, VGG19, and InceptionV3 [21], [22] by employing transfer learning and fine-tuning for binary and multi-class classification. Zeng *et al.* [23] introduced Siamese-like architecture [24] trained with transfer learning to classify fundus images into two grades. Kassani *et al.* [25] used a Multi-Layer Perceptron (MLP) as a classification head on top of the modified Xception network [26] by concatenating different feature maps from different convolutional layers. Four Inception models were utilized [27] for multi-class classification, each fundus image was sliced into four quadrants, and each quadrant will be classified by one of the four models. [28] exploited blended models to enhance data representation, Gangwar *et al.* [29] investigated a new hybrid model inherited from Inception and ResNet architectures. Al Antary *et al.* [30] designed ResNet architecture integrated with a Multi-Scale Attention mechanism (MSA) to enhance the representational power of the encoder. Moreover, they employed a multi-level approach for feature reuse for more improvements. Since our focus in this paper is to enhance the grading system both on binary and multi-class classification tasks, we observed drawbacks related to the aforementioned algorithms despite their success ranging from high time and space complexity to drop out mitigating the severe data imbalance inherited.

DR severity grading remains a challenging task due to three factors: (i) *Data rarity*. Acquiring massive labeled data is a crucial issue for DL and more significant in the medical domain due to the data privacy issues or/and having costly

devices to get high-quality images. (ii) *Implicit stochasticity*. Retinal fundus images experience large variations caused by different devices and environmental conditions regarding color, contrast, illumination, and size. As a result, the model's decision may be distorted. (iii) *Fading classes' disparity*. The threshold chosen for image classification between two closely distributed classes (e.g., mild and moderate in the APTOS dataset) is blurry, as will be shown in Section III.C, due to the dependence on microscale ocular pathologies. To solve the problem of fading disparity, large CNN architectures were employed in the literature to extract more informative features, data augmentation and preprocessing were used to enhance CNNs' generalizability. Finally, transfer learning was exploited to overcome data shortage.

In this paper, we investigate the efficacy of light-weight deep learning architecture for fast and robust severity grading of diabetic retinopathy. Our framework is based on a modified version of DenseNet [31] with integrating an attention mechanism with the former architecture for more feature refinement. Furthermore, we observe the effect of data imbalance on the model performance and mitigate such an effect by using an imbalanced learning technique. As shown in Fig.1, we first pass and preprocess the retinal image for quality enhancement, afterward, the images were passed to the DenseNet encoder C for feature extraction, then the features are sent to the attention module A for more improved representation. We train our model by freezing Densenet's encoder, trained on the ImageNet [32] dataset for the model's convergence acceleration by using the pre-trained weights θ_C and training only the attention module and the classification head using APTOS data in a supervised approach to update θ_A & θ_M . Our main contributions are as follows:

- 1) We developed a modified architecture to reduce the time needed for training and inference while enhancing DR severity grading by using a relatively small model with 8.5 million parameters compared to 10.8 million in the previous work.
- 2) We exploited the effect of using an attention mechanism as a supplementary module for feature refinement which led to an increase in accuracy while preserving low model complexity.
- 3) We tested the effect of using an imbalanced learning approach to alleviate the impact of data imbalance on the model's performance and proved its efficiency in enhancing the overall metrics.
- 4) We utilized transfer learning only by freezing the convolutional encoder without extra fine-tuning which led to relatively low number of learnable parameters (150K).

The paper is divided as follows. The related work is presented in Section II. In Section III, the methodology is presented. In Section IV, the results and discussions are demonstrated. Finally, conclusions are provided in Section V.

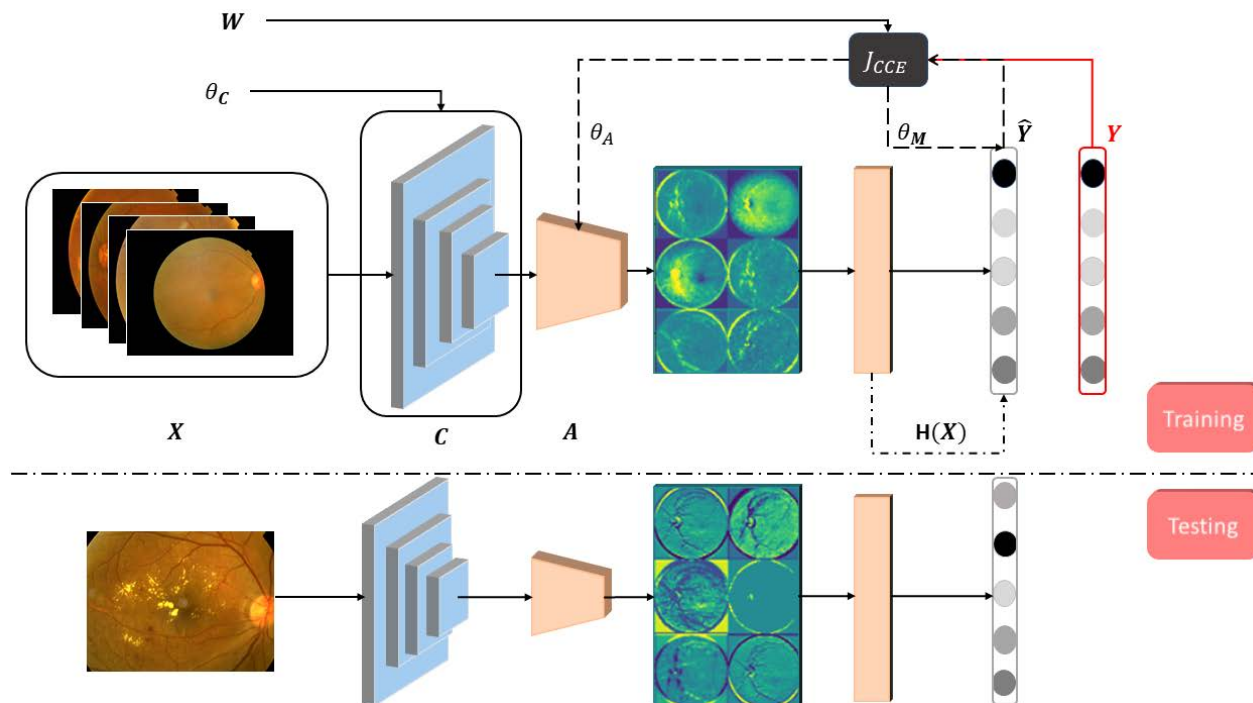


FIGURE 1. In the scheme of our proposed approach, In the network training step (upper), we pass a batch of labeled preprocessed images X to our convolutional encoder C for feature extraction, then an attention mechanism A for feature refinement. Finally, in the testing phase (lower), we directly pass the data to the network to predict the image class.

II. RELATED WORK

Deep learning has been deployed extensively in DR due to the rising of the transfer learning paradigm that offers fast convergence and performance enhancement while reducing the need for massive data and computational resources. This has opened the door for more robust algorithms in the medical domain. Wang *et al.* [33] developed Lesion-Net; the main aim of the network was to aim was to add lesion detection to severity grading to reinforce the representational power of the encoder. The architecture was built on InceptionV3, which was trained and validated using a private dataset. An ensemble stacking approach was investigated by Qummar *et al.* [34] by using five reputable architectures (Resnet50, InceptionV3, Xception, DenseNet121, DenseNet169) in order to improve produced feature maps. Furthermore, they used the Kaggle EyePACS dataset to assess the model. A hybrid deep learning model introduced by Cortes *et al.* [35] was built using InceptionV3 encoder for feature extraction and then training Gaussian Process (GP) regressor to get uncertainty of the prediction using EyePACS and Messidor-2 datasets, for DR binary classification task. The EfficientNet-B3 architecture was deployed by Sugeno *et al.* [36] for both binary and severity classification using APTOS dataset. Furthermore, they developed a method for lesion detection and validated with ground truth exploiting DIARETDB1¹ dataset. Meta-Plasticity, a bio-inspired phenomenon, was artificially implemented at CNN’s

¹<https://www.it.lut.fi/project/imageret/diaretdb1/>

back-propagation path to reinforce less common occurrences during the learning process by Boix *et al.* [37] for performance enhancement. Moreover, they deployed this technique in different deep learning architectures, using APTOS data for binary and severity grading tasks. Zhang *et al.* deployed a Source-Free Transfer Learning (SFTL) [38] model for referable DR, which utilized the unlabelled retinal images to alleviate the challenges of medical data annotation and privacy. They applied their algorithm to APTOS dataset for binary and multi-class classification tasks.

III. METHODOLOGY

In this section, we present the details of our framework. First, we introduce APTOS data, followed by data preprocessing, then data augmentation, balancing, and analysis. Finally, we introduce our architecture, training settings, and evaluation metrics.

A. DATASETS

In 2019 (APTOS) dataset² was released on the Kaggle website³ as a part of public competition for DR detection. The main aim of using fundus imaging was to classify disease severity by producing a probability that an image located in one of five clusters: No DR, Mild, Moderate, Severe, and Proliferative DR. This data was collected by Aravind Eye Hospital in India, 13,000 (approximately) images were

²<https://www.kaggle.com/c/aptos2019-blindness-detection>

³<https://www.kaggle.com/>

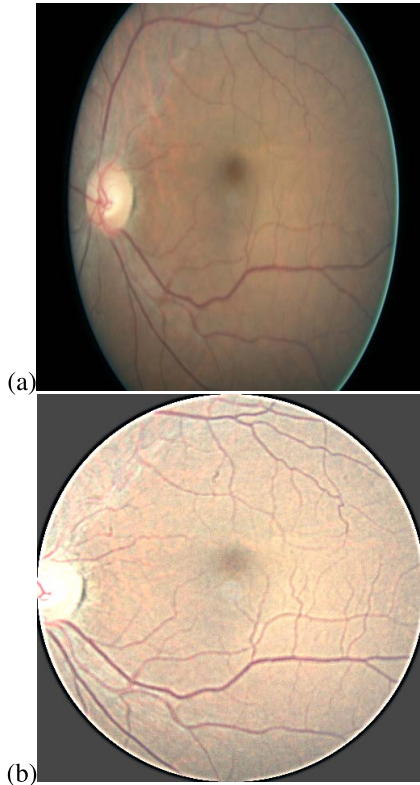


FIGURE 2. In visual comparison between (a) Raw fundus image and (b) Pre-processed fundus image, we observe the removal of the black side borders, by removing the black pixels and applying a Gaussian filter, the clarity of blood vessels and other bio-markers enhanced significantly.

provided at this competition; however, we had only access to the ground truth labels of 3662 images.

B. DATA PRE-PROCESSING

The uninformative black areas on the sides of the images were first trimmed then a circular crop was applied to have a centered retinal image. Moreover, a filtering technique was exploited [39] to enhance the clarity of visual bio-markers, and described by the following equations:

$$X'' = \alpha \times X + \beta \times X' + \gamma \tag{1}$$

$$X' = G(\sigma_x) * X \tag{2}$$

X indicates the input data, $G(\sigma_x)$ is a 2D Gaussian kernel with a standard deviation of $\sigma_x = 15$ in x-direction and $*$ is the convolution operation. α , β , and γ were chosen empirically to be 5, -4, and 70, respectively. Finally, each image was normalized to be in the range of [0, 1], resized to (256×256) using bilinear interpolation, and decoded to a 32-bit floating-point. Fig.2 represents the input and output from the pre-processing step.

C. DATA AUGMENTATION, BALANCING & ANALYSIS

Investigating APTOS data revealed severe class imbalance, i.e., 49.29%, 10.1%, 27.28%, 5.27%, 8.05% belonging to normal, mild, moderate, severe, and proliferative DR grades.

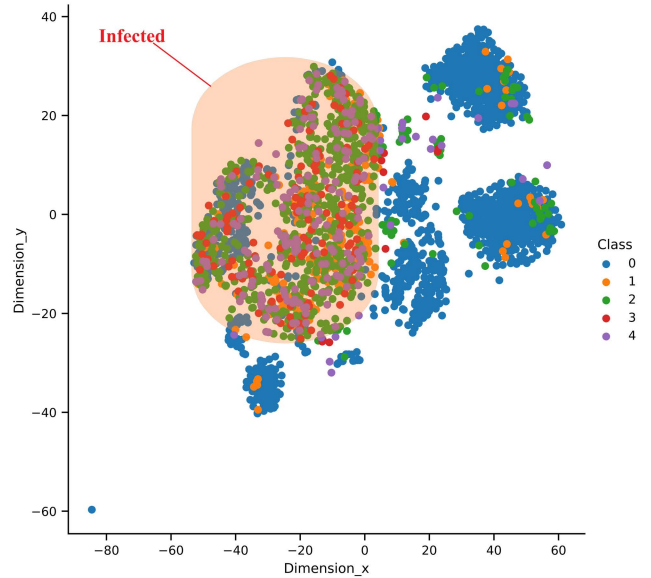


FIGURE 3. 2D representation for APTOS data, class 0 forms dense clouds in low dimensional feature space while having scattered representation for other classes due to data shortage.

Furthermore, by its projection in lower-dimensional feature space, using Principle Component Analysis (PCA) to lower the data dimensionality to 500-D followed by applying the t-Distributed Stochastic Neighbor Embedding (t-SNE) algorithm to analyze data distribution across different classes [40], Intuitions were developed by exploiting Fig.3:

- Class 0 forms feature clusters all over the 2-D space, making it one of the easiest classes to be detected.
- Classes (1-4) have acute overlapping, which generates a challenging task for the algorithm to fit a proper hyperplane.
- We artificially clustered the data to form only two regions (infected and healthy), and we observed that DL, based on our understanding, is robust enough to solve the binary classification problem.

Thus, to mitigate such effect, we used an Inverse Number of Samples (INS) learning approach where each class is weighted inversely proportional to its distribution in the original dataset as described in (3) and (4):

$$W = \frac{1}{S_i} \tag{3}$$

$$W = \frac{W}{(\sum_{i=1}^N W_i) \times N} \tag{4}$$

W , S_i are 1-D array that contains weights for each class and the total number of samples per class. N is the total number of classes and i is the class index. As a consequence, we used the updated version of the Categorical Cross-Entropy loss function (CCE):

$$J_{cce} = -\frac{1}{M} \sum_{i=1}^N \sum_{m=1}^M w_i \times y_m^i \times \log(h_\theta(x_m, i)) \tag{5}$$

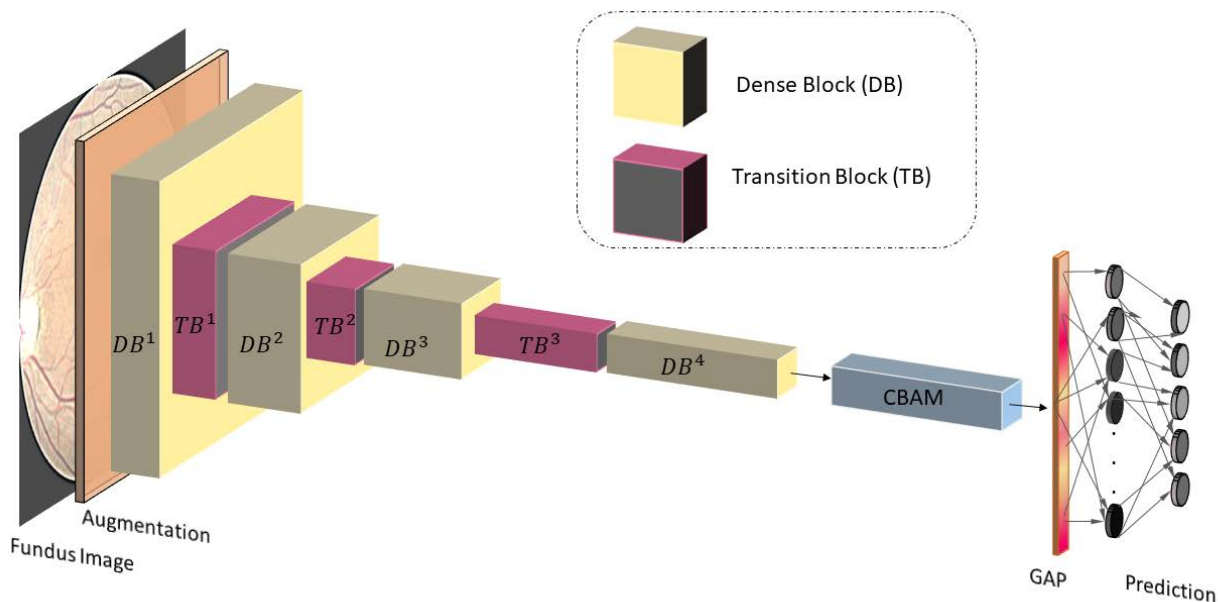


FIGURE 4. Proposed network architecture for DR severity grading.

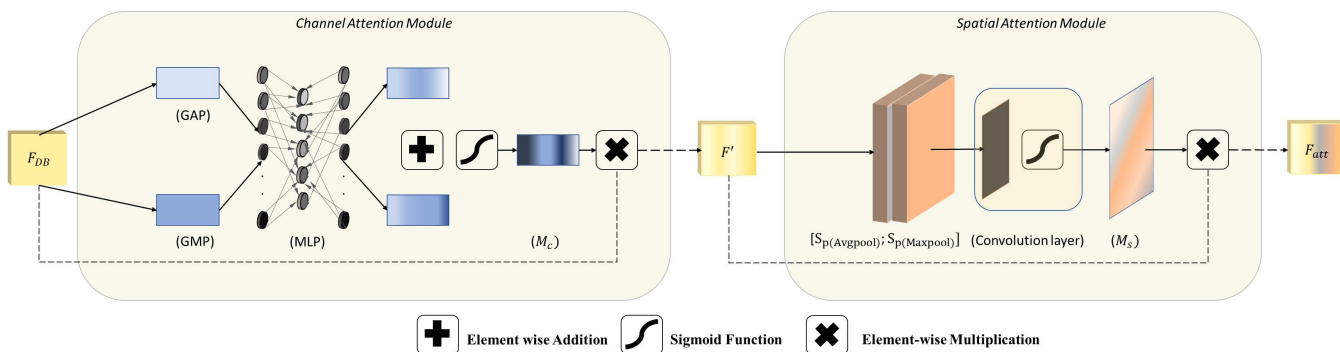


FIGURE 5. Convolutional block attention module illustration.

where

- M number of training samples
- N total number of classes
- w_i weight for class i
- y_m^i target label for training example m for class i
- x_m input image for training example m
- h_θ model with learnable parameters θ

Random horizontal, vertical flipping, and rotation were applied to reduce overfitting and improve the model’s generalizability. Furthermore, it was employed using the on-fly augmentation technique, which means it was utilized as a layer in our network to perform the transformations mentioned during the training phase.

D. ARCHITECTURE

Our algorithm consists of a backbone model (convolutional base) and an attention module. First, the backbone network is used as a feature extractor for the input fundus image, and

then features are refined using Convolutional Block Attention Module (CBAM) for data representation enhancement. Afterward, converting them to a one-dimensional array by averaging each feature map generated by the attention module using Global Average Pooling (GAP) followed by classification head. Fig.4 demonstrates an illustration of our network.

1) DenseNet

DenseNet was used as the main backbone for the proposed approach. Huang et al. [31] demonstrated the robustness of the architecture against the vanishing gradient problem while reducing the number of parameters and reducing over-fitting for smaller datasets. The main idea was to connect CNN layers using a dense connectivity pattern such that each layer has a concatenated input of all preceding feature maps:

$$X_l = H_l([X_0, X_1, \dots, X_{l-1}]) \tag{6}$$

where $[X_0, X_1, \dots, X_{l-1}]$ is the concatenated feature maps to the l^{th} layer, $H_l(\cdot)$ is a hidden layer that exploits consecutive

TABLE 1. Network architecture.

Layer Type	DenseNet + CBAM	
	Output size	Specs.
Augmentation	$256 \times 256 \times 3$	scaling, rotation, flipping
Convolutional	$128 \times 128 \times 64$	7×7 , stride 2
Pooling	$64 \times 64 \times 64$	3×3 max pool, stride 2
Dense Block (1)	$64 \times 64 \times 256$	1×1 conv $\times 6$ 3×3 conv
Transition (1)	$32 \times 32 \times 128$	1×1 conv 2×2 avg pool stride 2
Dense Block (2)	$32 \times 32 \times 512$	1×1 conv $\times 12$ 3×3 conv
Transition (2)	$16 \times 16 \times 256$	1×1 conv 2×2 avg pool stride 2
Dense Block (3)	$16 \times 16 \times 1280$	1×1 conv $\times 32$ 3×3 conv
Transition (3)	$8 \times 8 \times 640$	1×1 conv 2×2 avg pool stride 2
Dense Block (4)	$8 \times 8 \times 1024$	1×1 conv $\times 12$ 3×3 conv
Channel Attention	$8 \times 8 \times 1024$	GAP, MAP $1 \times C/r$ MLP addition layer multiplication layer
Spatial Attention	$8 \times 8 \times 1024$	S_{avg} , S_{max} concatenation layer 7×7 conv multiplication layer
GAP layer	1×1024	-
Dropout	1×1024	-
Softmax layer	1×5	-

operations: batch normalization (BN) [41], followed by a rectified linear unit (RELU) [42], and convolution operation to have a non-linear transformation of the input. Architecture design allows feature reuse based on routing the previous feature maps to the next convolution layer. For pooling, Transition Block (TB) was integrated, consisting of batch normalization, 1×1 convolution, and 2×2 average pooling.

2) CONVOLUTIONAL BLOCK ATTENTION MODULE (CBAM)

CBAM has proved its success in more curated feature generation and performance enhancement [43]. It consists of two sub-modules:

- Channel Attention Module.
- Spatial Attention Module.

The attention module is used to infer two feature maps:

$$F_{att} = (M_s(M_c(F) \otimes F)) \otimes (M_c(F) \otimes F) \quad (7)$$

$F_{att} \in \mathbb{R}^{H \times W \times C}$ is the refined features, $F \in \mathbb{R}^{H \times W \times C}$ is CBAM's input, $M_s \in \mathbb{R}^{H \times W \times 1}$ is a 2-D spatial attention map, \otimes denotes element wise multiplication and $M_c \in \mathbb{R}^{1 \times 1 \times C}$ is 1-D channel attention map:

$$M_c(F) = \sigma(MLP(GAP(F)) + MLP(GMP(F))) \quad (8)$$

where $\sigma(\cdot)$ is Sigmoid function, MLP is shared network with hidden units $\in \mathbb{R}^{C/r \times 1 \times 1}$, C is the number of channels, r is a

Algorithm 1 The Implementation of DenseNet+CBAM Model

Input: Pre-trained DenseNet encoder C with Imagenet weights θ_C , labelled data (X, Y) , α, β, γ , batch size B , class weights W .

Output: θ_A for the attention mechanism A , θ_M for the classification head.

Initialisation : Learning rate l_r

- 1: Apply preprocessing $X' = F_{transform}(X, \alpha, \beta, \gamma)$
- 2: **for** epoch = i from 1 to N **do**
- 3: **for** each mini-batch **do**
- 4: **for** image k in mini-batch b **do**
- 5: Apply on-fly Keras augmentation
- 6: Extract & refine the features
 $z = h_{\theta_A}(h_{\theta_C}(X[k]'))$
- 7: Encode flattened features $z' = h_{\theta_M}(z)$
- 8: Compute $\hat{y}_k = \operatorname{argmax}(\frac{e^{z'_k}}{\sum_{j=1}^N e^{z'_j}})$
- 9: **end for**
- 10: Update MLP via $\theta_M \leftarrow \operatorname{Adam}(\nabla_{\theta_M}(J_{CCE}), \theta_M, W, l_r)$;
- 11: Update A via $\theta_A \leftarrow \operatorname{Adam}(\nabla_{\theta_A}(J_{CCE}), \theta_A, W, l_r)$
- 12: **end for**
- 13: **end for**

compression ratio and GAP (Global Average Pooling), GMP (Global Maximum Pooling) were applied across spatial axes.

$$M_s(F') = \sigma(K^{7 \times 7}([S_{P_{Avg_{pool}}}(F'); S_{P_{Max_{pool}}}(F')])) \quad (9)$$

$F' \in \mathbb{R}^{H \times W \times C}$ is channel's attention module output, $K^{H \times W}$ is a convolution kernel with one filter applied to concatenation of $S_{P_{Avg_{pool}}}$ and $S_{P_{Max_{pool}}}$, where both of them are employed across the channel axis. Fig.5 shows an illustration for CBAM.

3) PROPOSED IMPLEMENTATION

DenseNet169 was selected from the DenseNet family after comparing different reputable pre-trained models. It demonstrated robust performance across all classes due to its nature; as discussed in Section III.D.1, the flow of information from low-level features to the upper layers allowed the model to exploit as many features as possible. A series of experiments were made to choose the best depth to check if we need this high complexity while achieving the best performance, and we decided to reduce the number of convolutional blocks in the fourth dense block to be 12 instead of 32. Exploiting attention mechanisms offer more flexibility to DL algorithms to focus more on the vital information related to the target and discard those not related. CBAM has provided that it is capable of enhancing the model's representational power without increasing the complexity, so we tried different positions for CBAM in our modified DenseNet, and we observed that the best performance is accompanied by positioning CBAM on top of the convolutional encoder plus reducing the training time significantly due to the decrease in spatial dimensions.

TABLE 2. DR severity grading results on APTOS dataset. The best, second best, and third best are marked by italics, boldface, and underline, respectively. M:million.

Method	Accuracy	F1-score*	QWK	No. of parameters
Blended model [28]	81.7%	-	0.711	> 50 M
InceptionResNetV2 [29]	82%	-	-	> 50 M
Modified Xception [25]	83.09%	-	-	> 20 M
AM-InceptionV3 [37]	76.67%	<i>0.7668</i>	-	> 20 M
SFTL model [38]	79.8%	-	-	> 20 M
EfficientNet-B3 [36]	84%	-	0.89	<u>10.8 M</u>
MSA without multi-level feature reuse [30]	<u>82%</u>	-	0.857	15.7 M
Baseline DenseNet169	79%	0.56	0.8287	8.35 M
DenseNet169 + CBAM	81%	0.60	0.8607	8.5 M
DenseNet169 + INS	81%	<i>0.64</i>	<i>0.8916</i>	8.35 M
DenseNet169 + CBAM + INS	<u>82%</u>	0.68	<u>0.888</u>	8.5 M

*Our f1-score was calculated using the macro average version.

TABLE 3. Binary classification results on APTOS dataset. The best, second best, and third best are marked by italics, boldface, and underline, respectively.

Method	Accuracy	Sensitivity	Specificity	QWK
Blended model [28]	97.41%	<u>97%</u>	-	<i>0.9482</i>
AM-InceptionV3 [37]	94.46%	90%	-	-
SFTL model [38]	91.2%	95.1%	85.8%	-
EfficientNet-B3 [36]	-	98%	<u>98%</u>	-
MSA without multi-level feature reuse [30]	<i>98.1%</i>	<i>98.3%</i>	<i>98.2%</i>	-
DenseNet169 + CBAM + INS	<u>97%</u>	<u>97%</u>	98.3%	0.9455

Macro average outcomes were used for the first three metrics.

TABLE 4. Statistics for training and validation datasets.

Class	Train	Test
	3296	366
Normal	1624	181
Mild	333	37
Moderate	899	100
Severe	174	19
PDR	265	30

Four trials were investigated to show the gradual increase in performance:

- Baseline DensNet169.
- DenseNet169 + INS.
- DenseNet169 + CBAM.
- DenseNet169 + CBAM + INS.

Where our baseline has only DenseNet169's modified encoder without attaching CBAM as a supplementary module, moreover as well as not deal with the class imbalance inherited in APTOS data. For the second trial, we demonstrated the effectiveness of using cost-sensitive learning to penalize our model when dealing with minor classes and vice-versa. CBAM was added to DenseNet without using INS to investigate its effectiveness in the third trial. Finally, we investigated the enhancements added by CBAM and INS together. The four experiments had followed the same settings by freezing DenseNet's encoder and using transfer learning to accelerate the training of CBAM and Softmax layers. Fine-tuning was not used in contrast to the conventional framework

when we have a different data domain compared to ImageNet data, and we took our decision based on the interesting results provided by [44], where ImageNet weights demonstrated its robustness as a feature extractor for retinal disease detection. A reduction ratio ($r = 32$) and kernel size ($K^{7 \times 7}$) at channel and spatial modules, respectively for CBAM. Due to its performance, our fourth trial was compared to other state-of-the-art techniques. Detailed information regarding our architecture is demonstrated in Table.1.

E. TRAINING SETTINGS

Our splitting policy was 90% to 10% of our dataset to form a training and validation set. A stratified data splitting technique was exploited to preserve the same distribution to ensure the classes' distribution consistency between the aforementioned subsets and the original set. Table.4 demonstrates the training and validation data statistics. Furthermore, K-fold validation was implemented to have more robust results, and due to the size of the dataset, we used 5-folds to train on 80% and test using 20% of the original dataset at each trial. Furthermore, the maximum number of epochs was limited to 400 while using an early stopping callback to avoid overfitting by saving the best weights corresponding to the minimum validation loss. Finally, we used the exact stratified data splitting mechanism to ensure the same class distribution at each fold.

Our algorithm was implemented using TensorFlow [45] and trained on Tesla V100 GPU provided by Google Co-lab. We trained four networks for 1000 epochs, and with a small

batch size of 32 images, the RGB images are passed to the network after being preprocessed. Furthermore, using Adam optimizer with learning rate 3×10^{-4} , $\beta_1 = 0.9$, $\beta_2 = 0.909$, and weighted CCE that was demonstrated at (5) as a loss function. Specifically, we exploited Sparse (CCE) based on the label encoding found in the dataset. All layers in CBAM were initialized by He normal initializer [46], Dropout layer was set with a rate equal to 0.5 to improve generalizability, and Softmax as a final layer [47]. For severity grading, the highest probability represents the level of the sample, whereas, for binary classification, the output was thresholded at 0.5. We introduce the overall training process of our proposed approach in Algorithm 1.

F. EVALUATION METRICS

Five common metrics were used to evaluate the model’s performance.

1) ACCURACY (ACC)

The percentage of correct predictions that a model can achieve. Accuracy is defined as

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

2) SENSITIVITY (SENS)

is the percentage of positive cases that is classified as actual positive. Identified as follows

$$Sens = \frac{TP}{TP + FN} \tag{11}$$

3) SPECIFICITY (SPEC)

is the percentage of negative cases that are detected as actual negative. Identified as follows

$$Spec = \frac{TN}{TN + FP} \tag{12}$$

4) F1-SCORE (F1)

is the harmonic mean of precision and recall and is identified as

$$F1 = \frac{TP}{TP + \frac{1}{2}(FN + FP)} \tag{13}$$

5) KAPPA-SCORE

to assess the agreement between our model and the original rater. Identified as follows

$$k = 1 - \frac{\sum_{i,j} w_{i,j} O_{i,j}}{\sum_{i,j} w_{i,j} E_{i,j}} \tag{14}$$

where true positives (TP) are the classes classified correctly by the algorithm, true negatives (TN) are samples predicted correctly as negative, false positives (FP) are samples that are miss-classified as a positive class, and false negatives (FN) are samples miss-classified as negative class. $O_{i,j}$ is the observed matrices, and $E_{i,j}$ is the expected one.

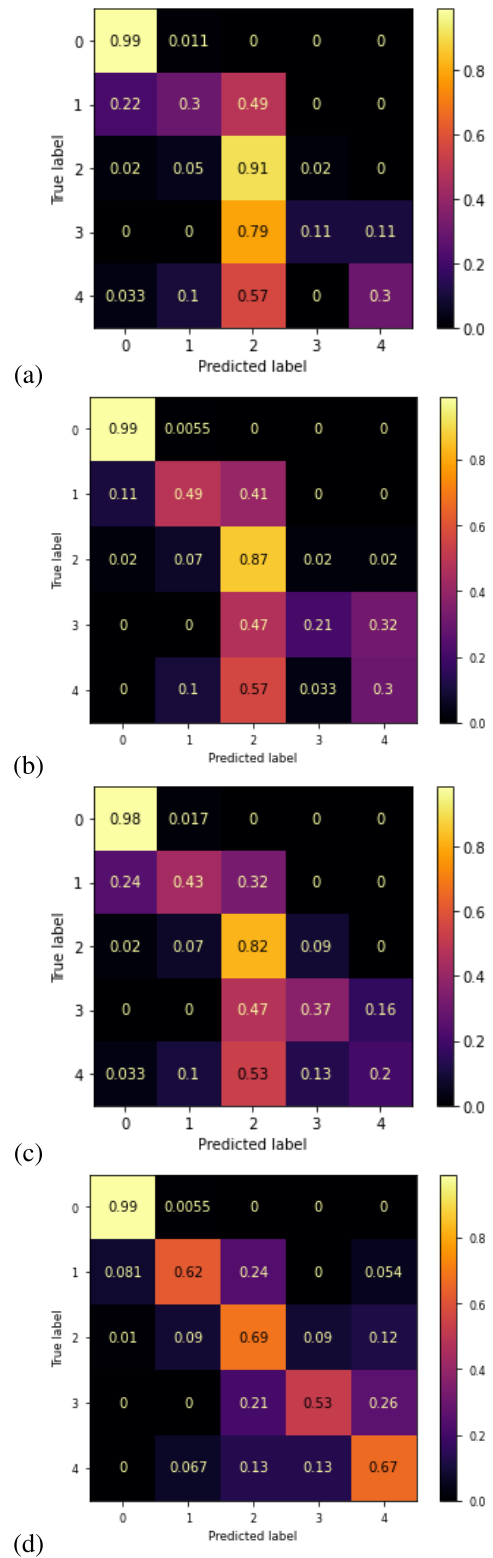


FIGURE 6. Normalized confusion matrices for (a) Baseline DenseNet169 (b) DenseNet + CBAM (c) DenseNet169 + INS (d) DenseNet169 + INS + CBAM.

IV. RESULTS AND DISCUSSIONS

Fig.6 illustrates the performance of our four algorithms. In Fig.6.a, we observe that without the weighted loss function,

it is easier for our model to be distorted and have robust behavior only in detecting major classes (0 and 2) and vice-versa. As can be shown in Fig.6.b, attaching CBAM to our encoder enhanced the detection of classes 1 and 3 by 63.3% and 90.9%, while reducing class 2 only by 4.6%. Class imbalance mitigation allowed better performance, as can be seen in Fig.6.c, class 1,3 detection is enhanced by 43.3% and 236.4% respectively, with respect to the baseline algorithm. Finally, using CBAM with DenseNet169 while adding weighted loss has demonstrated thriving performance across all classes. Regardless of the reduction in class 2 by 14.63%, classes (1,3 and 4) exhibit significant improvements by 44.2%, 43.24%, and 235%. An average QWK and accuracy values of 0.8072 and 72.3% were achieved, respectively, using the 5-fold k-validation technique. As shown in Section III.E, we trained our algorithm only for 400 epochs to reduce the computational cost of training five different models, further training will provide more intact results.

As shown in Table.2, the proposed method outperformed the literature work on the severity grading task and showed comparable results. Our model enhanced accuracy and QWK by 0.4% and 24.9% while decreasing inference time by cutting down the number of parameters by 83% compared to [28]. We achieved almost the same accuracy as [29] while reducing the model size. Our best trial had an increase in accuracy of about 7% compared to the AM-InceptionV3 [37] method. SFTL model achieved high accuracy at the severity grading task. However, they did not tackle the problem of data imbalance. EfficientNet-B3 [36] achieved higher accuracy but only for major classes, while we achieved comparable accuracy in minor classes, and finally, We compared our best trial with the MSA network without multi-level feature reuse [30]. We had almost the same accuracy with an increase in QWK by 3.6%. Furthermore, we achieved a better confusion matrix across all classes than the literature while reducing time and space complexity by a 45% reduction in parameters. Severity grading f1-score was not mentioned in the literature. However, by using CBAM and INS, an enhancement was established by 21.4% with respect to the baseline DenseNet169.

Our algorithm demonstrated robustness against other deep learning architectures for the binary classification task, as shown in Table.3. Above all, the literature did not deal with the class imbalance problem. Most of the algorithms implemented did not consider its effect on quality metrics which provided overestimated outcomes, as most of them were predicting perfectly only for major classes due to ignoring data inherited imbalance. Furthermore, as mentioned in Section III.C, binary grading did not require complex architectures to solve it, our algorithm with lower parameters achieved almost the same metrics compared to other algorithms, plus when we artificially formed two clusters (infected and normal), the classes were balanced which helped literature algorithms to excel in such a task. Moreover, our algorithm exceeds the minimum limits provided by English National Screening Program for sensitivity, and

specificity [48]. Finally, our model achieved low training time (9 seconds/epoch) and relatively high inference speed (1.166 seconds/32 images) compared to the MSA network that achieved 5 seconds exploiting the same batch size.

V. CONCLUSION

In this study, we exploited a new CNN model based on DenseNet169 architecture integrated with CBAM as an additional component to be added for representational power enhancement. The proposed method demonstrated robust performance and comparable quality metrics while reducing the burden of space and time complexity. Furthermore, a 2-D Gaussian filter enhances fundus images' quality. Finally, we used INS to form our weighted loss function to tackle the class imbalance to improve the model's prediction across all classes. For future research direction, we evaluate the performance of different CBAM configurations. Moreover, experimenting with different imbalanced learning techniques and increasing the dataset size will lead to better performance.

REFERENCES

- [1] *Global Report on Diabetes*, World Health Organization (WHO), Geneva, Switzerland, 2016.
- [2] *Diabetes Atlas*, International Diabetes Federation (IDF), Brussels, Belgium, 2019.
- [3] *Diabetes Eye Health a Guide for Health Professionals*, International Diabetes Federation (IDF), Brussels, Belgium, 2017.
- [4] P. Nagaraj, P. Deepalakshmi, R. F. Mansour, and A. Almazroa, "Artificial flora algorithm-based feature selection with gradient boosted tree model for diabetes classification," *Diabetes, Metabolic Syndrome Obesity: Targets Therapy*, vol. 14, pp. 2789–2806, Jun. 2021, doi: 10.2147/DMSO.S312787.
- [5] L. Cheng, X.-H. Wu, and Y. Wang, "Artificial flora (AF) optimization algorithm," *Appl. Sci.*, vol. 8, no. 3, p. 329, Feb. 2018. [Online]. Available: <https://www.mdpi.com/2076-3417/8/3/329>, doi: 10.3390/app8030329.
- [6] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," 2016, *arXiv:1603.02754*.
- [7] N. Gharaibeh, O. M. Al-hazaimeh, A. Abu-Ein, and K. M. O. Nahar, "A hybrid SVM NAÏVE-bayes classifier for bright lesions recognition in eye fundus images," *Int. J. Electr. Eng. Informat.*, vol. 13, no. 3, pp. 530–545, Sep. 2021, doi: 10.15676/ijeeci.2021.13.3.2.
- [8] O. M. Al Hazaimeh, K. M. O. Nahar, B. Al Naami, and N. Gharaibeh, "An effective image processing method for detection of diabetic retinopathy diseases from retinal fundus images," *Int. J. Signal Imag. Syst. Eng.*, vol. 11, no. 4, p. 206, 2018, doi: 10.1504/IJSISE.2018.093825.
- [9] T. Evgeniou and M. Pontil, "Support vector machines: Theory and applications," in *Advanced Course on Artificial Intelligence*, vol. 2049. Berlin, Germany: Springer, 2001, pp. 249–257, doi: 10.1007/3-540-44673-7_12.
- [10] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
- [11] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 10, pp. 1533–1545, Jul. 2014.
- [12] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," 2016, *arXiv:1609.03499*.
- [13] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," 2015, *arXiv:1502.03044*.
- [14] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
- [15] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," 2015, *arXiv:1508.06576*.
- [16] Z. Yan, X. Yang, and K.-T. Cheng, "Joint segment-level and pixel-wise losses for deep learning based retinal vessel segmentation," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 9, pp. 1912–1923, Sep. 2018.

- [17] Y. Wu, Y. Xia, Y. Song, Y. Zhang, and W. Cai, "NFN+: A novel network followed network for retinal vessel segmentation," *Neural Netw.*, vol. 126, pp. 153–162, Jun. 2020.
- [18] H. Zhao, H. Li, S. Maurer-Stroh, and L. Cheng, "Synthesizing retinal and neuronal images with generative adversarial nets," *Med. Image Anal.*, vol. 49, pp. 14–26, Oct. 2018.
- [19] L. Dai, R. Fang, H. Li, X. Hou, B. Sheng, Q. Wu, and W. Jia, "Clinical report guided retinal microaneurysm detection with multi-sieving deep learning," *IEEE Trans. Med. Imag.*, vol. 37, no. 5, pp. 1149–1161, May 2018.
- [20] A. Jain, A. Jalui, J. Jasani, Y. Lahoti, and R. Karani, "Deep learning for detection and severity classification of diabetic retinopathy," in *Proc. 1st Int. Conf. Innov. Inf. Commun. Technol. (ICIICT)*, Apr. 2019, pp. 1–6.
- [21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
- [22] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," 2015, *arXiv:1512.00567*.
- [23] X. Zeng, H. Chen, Y. Luo, and W. Ye, "Automated diabetic retinopathy detection based on binocular siamese-like convolutional neural network," *IEEE Access*, vol. 7, pp. 30744–30753, 2019.
- [24] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learn. Workshop*, vol. 2, Lille, France, 2015.
- [25] S. H. Kassani, P. H. Kassani, R. Khazaeinezhad, M. J. Wesolowski, K. A. Schneider, and R. Deters, "Diabetic retinopathy classification using a modified xception architecture," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol. (ISSPIT)*, Dec. 2019, pp. 1–6.
- [26] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2016, *arXiv:1610.02357*.
- [27] Z. Gao, J. Li, J. Guo, Y. Chen, Z. Yi, and J. Zhong, "Diagnosis of diabetic retinopathy using deep neural networks," *IEEE Access*, vol. 7, pp. 3360–3370, 2019.
- [28] J. D. Bodapati, V. Naralasetti, S. N. Shareef, S. Hakak, M. Bilal, P. K. R. Maddikunta, and O. Jo, "Blended multi-modal deep ConvNet features for diabetic retinopathy severity prediction," *Electronics*, vol. 9, no. 6, p. 914, May 2020.
- [29] A. K. Gangwar and V. Ravi, "Diabetic retinopathy detection using transfer learning and deep learning," in *Evolution in Computational Intelligence*, V. Bhateja, S.-L. Peng, S. C. Satapathy, and Y.-D. Zhang, Eds. Singapore: Springer, 2021, pp. 679–689.
- [30] M. T. Al-Antary and Y. Arafa, "Multi-scale attention network for diabetic retinopathy classification," *IEEE Access*, vol. 9, pp. 54190–54200, 2021.
- [31] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," 2016, *arXiv:1608.06993*.
- [32] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," 2014, *arXiv:1409.0575*.
- [33] Y. Wang, M. Yu, B. Hu, X. Jin, and Y. Li, "Deep learning-based detection and stage grading for optimising diagnosis of diabetic retinopathy," *Diabetes/Metabolism Res. Rev.*, vol. 37, no. 4, p. e3445, 2021, doi: [10.1002/dmrr.3445](https://doi.org/10.1002/dmrr.3445).
- [34] S. Qummar, F. G. Khan, S. Shah, A. Khan, S. Shamshirband, Z. U. Rehman, I. Ahmed Khan, and W. Jadoon, "A deep learning ensemble approach for diabetic retinopathy detection," *IEEE Access*, vol. 7, pp. 150530–150539, 2019, doi: [10.1109/ACCESS.2019.2947484](https://doi.org/10.1109/ACCESS.2019.2947484).
- [35] S. Toledo-Cortés, M. De La Pava, O. Perdómo, and F. A. González, "Hybrid deep learning Gaussian process for diabetic retinopathy diagnosis and uncertainty quantification," 2020, *arXiv:2007.14994*.
- [36] A. Sugeno, Y. Ishikawa, T. Ohshima, and R. Muramatsu, "Simple methods for the lesion detection and severity grading of diabetic retinopathy by image processing and transfer learning," *Comput. Biol. Med.*, vol. 137, Oct. 2021, Art. no. 104795, doi: [10.1016/j.compbiomed.2021.104795](https://doi.org/10.1016/j.compbiomed.2021.104795).
- [37] V. Vives-Boix and D. Ruiz-Fernández, "Diabetic retinopathy detection through convolutional neural networks with synaptic metaplasticity," *Comput. Methods Programs Biomed.*, vol. 206, Jul. 2021, Art. no. 106094, doi: [10.1016/j.cmpb.2021.106094](https://doi.org/10.1016/j.cmpb.2021.106094).
- [38] C. Zhang, T. Lei, and P. Chen, "Diabetic retinopathy grading by a source-free transfer learning approach," *Biomed. Signal Process. Control*, vol. 73, Mar. 2022, Art. no. 103423, doi: [10.1016/j.bspc.2021.103423](https://doi.org/10.1016/j.bspc.2021.103423).
- [39] B. Graham. (2015). *Kaggle Diabetic Retinopathy Detection Competition Report*. [Online]. Available: <https://www.kaggle.com/c/diabetic-retinopathy-detection/discussion/15801>
- [40] G. Hinton and S. Roweis, "Stochastic neighbor embedding," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 833–840.
- [41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.
- [42] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, vol. 15, Fort Lauderdale, FL, USA, 2011.
- [43] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "CBAM: Convolutional block attention module," 2018, *arXiv:1807.06521*.
- [44] M. Raghu, C. Zhang, J. Kleinberg, and S. Bengio, "Transfusion: Understanding transfer learning for medical imaging," 2019, *arXiv:1902.07208*.
- [45] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," 2016, *arXiv:1605.08695*.
- [46] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.
- [47] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 56, pp. 1929–1958, 2014.
- [48] R. Taylor and D. Batey, *The Need to Screen*. Hoboken, NJ, USA: Wiley, 2012, ch. 4, pp. 29–41, doi: [10.1002/9781119968573.ch4](https://doi.org/10.1002/9781119968573.ch4).



MOHAMED M. FARAG received the B.Sc. degree in renewable energy engineering from The University of Ain-Shams, Cairo, Egypt, in 2018. He is currently pursuing the M.Sc. degree in electronics engineering with the German University in Cairo (GUC), Cairo. He is currently working as a Teaching Assistant with the Electronics Department, German University in Cairo. His research and professional interests include data science, medical image processing, machine learning, and deep learning.



MARIAM FOUAD was born in Cairo, Egypt, in 1993. She received the bachelor's degree in electronics engineering from the German University, Cairo, in 2015, and the master's degree in 2017 with the thesis titled "Joint Near-Infrared and Bio-Impedance Spectroscopy Non-Invasive Glucose Monitoring". She is currently pursuing the Ph.D. degree with the Department of Medical Engineering, Ruhr University Bochum in collaboration with the German University in Cairo. Her current research interests include the utilization of deep learning concepts in ultrasound special applications such as harmonic imaging and synthetic data generation.



AMR T. ABDEL-HAMID was born in Cairo, Egypt, in 1974. He received the B.S. degree in electronics and communications engineering from Cairo University, Cairo, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from Concordia University, Canada, in 2001 and 2005, respectively. Currently, he is an Assistant Professor with the Department of Electronics Engineering and the Vice Dean of Student Affairs with the German University in Cairo (GUC), Egypt. His main research interests include the Internet of Things applications and security, system-on-a-chip design and verification, functional verification techniques, tools, and languages, IP watermarking, and security protocols verification.