

Research Article

Diagnosis of Diabetic Retinopathy through Retinal Fundus Images and 3D Convolutional Neural Networks with Limited Number of Samples

Ahsan Bin Tufail ^{1,2}, Inam Ullah ³, Wali Ullah Khan ⁴, Muhammad Asif,⁵ Ijaz Ahmad,⁶ Yong-Kui Ma ¹, Rahim Khan ¹, Kalimullah,⁷ and Md. Sadek Ali ⁸

¹School of Electronics and Information, Harbin Institute of Technology, Harbin, Heilongjiang Province, China

²Department of Electrical and Computer Engineering, COMSATS University Islamabad, Sahiwal Campus, Sahiwal, Pakistan

³College of Internet of Things (IoT) Engineering, Hohai University (HHU), Changzhou Campus, Changzhou 213022, China

⁴Interdisciplinary Centre for Security, Reliability and Trust (SnT)/SigCom, University of Luxembourg, 1855 Luxembourg City, Luxembourg

⁵College of Electronics and Information Engineering, Shenzhen University, Shenzhen, Guangdong 518060, China

⁶School of Pattern Recognition and Intelligent System, Shenzhen Institute of Advance Technology (Chinese Academy of Sciences), Shenzhen, China

⁷Department of Electrical Engineering, University of Science and Technology Bannu, Khyber Pakhtunkhwa, Pakistan

⁸Communication Research Laboratory, Department of Information and Communication Technology, Islamic University, Kushtia-7003, Bangladesh

Correspondence should be addressed to Md. Sadek Ali; sadek@ice.iu.ac.bd

Received 12 September 2021; Revised 24 October 2021; Accepted 2 November 2021; Published 17 November 2021

Academic Editor: Junjuan Xia

Copyright © 2021 Ahsan Bin Tufail et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Diabetic retinopathy (DR) is a worldwide problem associated with the human retina. It leads to minor and major blindness and is more prevalent among adults. Automated screening saves time of medical care specialists. In this work, we have used different deep learning (DL) based 3D convolutional neural network (3D-CNN) architectures for binary and multiclass (5 classes) classification of DR. We have considered mild, moderate, no, proliferate, and severe DR categories. We have deployed two artificial data augmentation/enhancement methods: random weak Gaussian blurring and random shift along with their combination to accomplish these tasks in the spatial domain. In the binary classification case, we have found the performance of 3D-CNN architecture trained by deploying combined augmentation methods to be the best, while in the multiclass case, the performance of model trained without augmentation is the best. It is observed that the DL algorithms working with large volumes of data may achieve better performances as compared to the methods working with small volumes of data.

1. Introduction

Diabetes weakens the blood sugar regulation process inside the human body. In the year 2017, approximately 451 million peoples were suffering from this disease. A higher level of blood sugar severely cripples human body organs leading to risky complications such as coronary episode, vision loss, cataracts, glaucoma, retinopathy, and dementia. A growing population of peoples, irrespective of age, suffering from

diabetes have problems in vision termed as diabetic retinopathy (DR) [1–4].

In clinical settings, four stages are generally involved in the assessment of DR, which are mild, moderate, severe, and proliferative retinopathy, respectively. In the earliest stage, microaneurysms, which are balloon-like structures, are formed in the small veins of the retina which are obstructed in the moderate stage. In the final stages, visual deficiency can occur [2].

Problems associated with DR can be treated in initial stages. The human eye is made up of optic nerves/discs, and the images of the eye can be segmented to get a good classification accuracy of different stages involved in DR [2]. Fundus images display the existence of exudates, hemorrhages, and other eye deficits and are graded manually by a limited number of ophthalmologists whose numbers are shrinking every year [3]. Microaneurysms, hemorrhages, hard exudates, cotton wool spots, neovascularization, and macular edema are some of the characteristics of DR [5, 6]. For the screening of DR, at the level of an image, “normal” category contains no lesion while “abnormal” category contains at least one lesion. A computer-aided diagnostic system can help health care specialists in alleviating variabilities. Deep learning (DL) is a popular method for analysing retinal fundus images [7, 8]. It captures high-level features throughout the learning process effectively adapting to any type of noise [9] thus forming a natural solution for identifying retinal diseases.

Various algorithms have been proposed for the examination of scan reports to diagnose DR [10–12]. Usually, researchers have focused on automatic recognition of lesions associated with DR [13–15]. In [16], the authors deployed convolutional neural networks (CNNs) to obtain a precision of 75% on the validation dataset for classifying DR images in the presence of artificial data augmentation/enhancement techniques. Shanthi and Sabeenian in [17] classified the fundus images employing a modified AlexNet architecture validated using Messidor database. The authors in [18] used transfer learning architectures, such as AlexNet, VGGNet, GoogleNet, and ResNet to reach a recognition rate of 95.68% exploiting publicly available Kaggle platform. A full patch-based CNN architecture is designed in [19] using only 28 retinal images achieving a sensitivity of 0.940. Authors in [2] constructed a 3D capsule network and validated their model on the Messidor dataset to achieve an accuracy of 98.64% on the stage 3 fundus images. In [4], a deep and densely connected network was designed to classify Messidor-2 and EyePACS datasets to attain a precision of 95% on Messidor-2 and 88% on the EyePACS dataset. Researchers in [20] used CNNs to achieve a sensitivity of 90% on the EyePACS dataset and a sensitivity of 87% on Messidor-2 dataset. Sayres et al. [21] deployed DL models achieving a sensitivity of 91% on Messidor-2 and a sensitivity of 94.5% on the EyePACS datasets. The work in [22] employed transfer learning-based VGG-19 architecture to classify 9 retinal diseases and one normal retina class with limited number of samples to obtain an accuracy of 30.5% when considering all the ten categories with the deployment of translation, rotation, and brightness change augmentation methods. The researcher used a deep CNN architecture [23] on the EyePACS dataset achieving a sensitivity of 98% deploying rotation, shearing, flipping, zooming, cropping, Krizhevsky augmentation, and translation as augmentation methods. Shankar et al. [24] proposed a DL model deploying histogram-based segmentation and a synergic network achieving an accuracy of 99.28% on the classification task for the Messidor dataset. Beede et al. [25] conducted a study on the clinical characterization of eye screening workflows

for the detection of diabetic eye disease. They discovered factors such as gradability, Internet speed and connectivity, nursing workflows, and patient experience to be responsible for the model’s performance. Khare et al. [26] proposed a firefly algorithm for dimensionality reduction, principal component analysis for feature extraction, and a deep neural network (DNN) model for the classification of DR to achieving an accuracy of 97%, precision and recall of 96%, specificity of 95%, and a sensitivity of 92% for the binary classification task. Qureshi et al. [27] proposed a DL architecture based on active learning for multiclass (e.g., 5 classes) classification of DR images from EyePACS benchmark for achieving a sensitivity of 92.2%, specificity of 95.1%, *F*-measure of 93%, and an accuracy of 98% on a wide range of fundus images. Das et al. [28] utilized maximal principal curvature, adaptive histogram equalization, morphological opening, and a CNN-based classifier to achieve an accuracy of 98.7% on DIARETDB1 dataset. Li et al. [29] presented a deep ensemble algorithm for the detection of DR using retinal fundus images. They exploited Inception-v4 architecture on Messidor-2 dataset to achieve an area under the curve of 0.994, sensitivity of 0.930, and a specificity of 0.971. Limwattanayingyong et al. [30] compared the screening of DR in a longitudinal setting via DL and human grading. They achieved a prevalence rate of 5.1% using DL while they observe a reduction in prevalence rate on a two-year follow-up. Tsiknakis et al. [31] provided an overview of DR detection based on fundus images discussing several aspects such as datasets, preprocessing techniques, and DL models for the characterization of important lesions. Karakaya and Hacısofuoğlu [32] compared different smart phone-based solutions such as iExaminer, D-Eye, Peek Retina, and iNview finding that field of view is the most important parameter for the detection of DR where iNview provides the largest and iExaminer provides the smallest value for this field.

Few-shot learning has been a topic of considerable interest where very few examples are used to categorize classes especially those classes that are not presented during training [33–35]. Fine-tuning is a common approach for few-shot learning. These systems require complex inference mechanisms due to the processing of complex inductive bias. Meta learning, augmentation/generative methods, transfer learning, and semisupervised methods are some of the typical approaches for this type of learning.

Besides health issues, researchers in academia and industry also investigated other problems in science, technology, and engineering using various DL-based approaches [36–50].

Although the reported studies offer competitive solutions to the binary and multiclass classification of DR, most of them are geared towards utilizing information in the 2D domain. Higher dimensions, such as 3D, offers rich scale and geometry information, which are challenging solutions for computer vision algorithms [51–53]. There is a need for studies to utilize the information offered by higher dimensions for these tasks. To take advantage of these representation learning methods on a limited number of samples [54] in the presence of data augmentation, we have used a 3D-CNN architecture [55] in the spatial domain for one

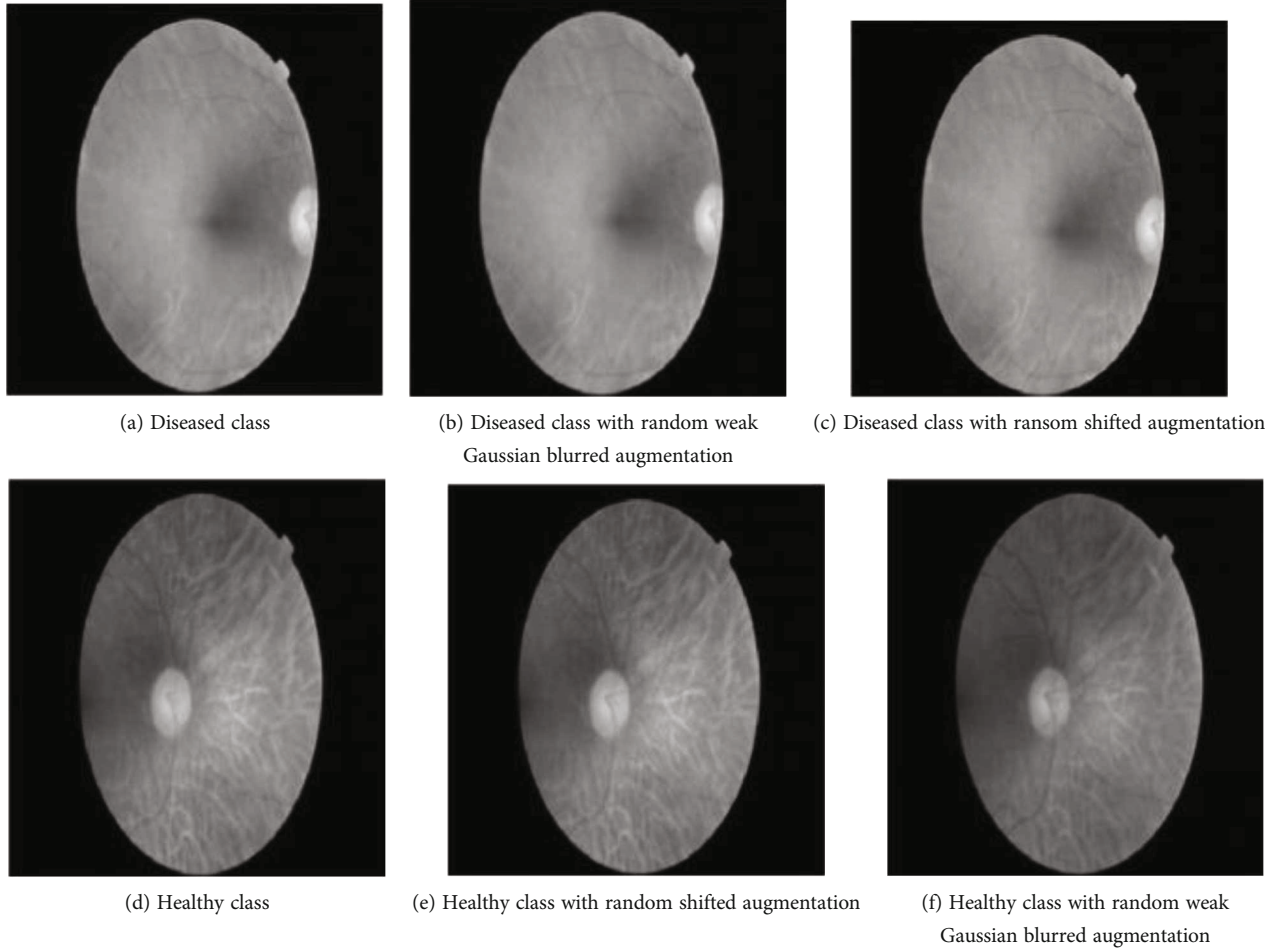


FIGURE 1: Sample images in the TeleOphta database with random shifted and random weak Gaussian blurred augmentation methods for the binary classification task.

binary and one multiclass classification task on the DR datasets. We have employed random weak Gaussian blurring and random shift as data augmentation/enhancement techniques along with their combination to study the impact of these methods on both classification tasks.

This work contributes to the existing literature on the classification of DR in the following ways. To the best of authors' knowledge, very few researches have been carried out in the literature to solve this problem in the 3D domain. This study is designed to achieve that which offers the advantage by considering both spatial and temporal dimensions simultaneously. Impact of different data augmentation schemes on the final classification performance is also worth investigating especially in the 3D domain. Few-shot learning is also a problem worth investigating as limited number of samples is a bottleneck in achieving good classification performances using DL methods.

The remaining sections of this paper are organized as follows. A brief description of the datasets is given in Section 2, while Section 3 provides the details of the methods used in this study. Section 4 presents the details of the conducted experiments. Section 5 provides results and a thorough discussion. Finally, conclusions are drawn in Section 6 from the work presented.

2. Dataset Description

We have used two datasets to carry out the experiments. The first one named TeleOphta [3] is a database of fundus images with exudates and microaneurysm lesions. Using this database, we have constructed 99 3D volumes of healthy subjects and 83 3D volumes of diseased class that show signs of exudates and microaneurysms, and these volumes are split at the subject level [56]. Random shifting and random weak Gaussian blurred augmentation techniques are deployed to enhance the dataset. Some samples of the images are shown in Figure 1. The volume size is $210 \times 210 \times 12$.

The second dataset has Gaussian filtered retina scan images to detect DR with five categories, which are no, mild, moderate, severe, and proliferate. The size of the 3D volumes is $512 \times 512 \times 2$. In this database, we have 262 3D volumes of each of these categories that are split at the subject level [56]. Some samples of the images that are present in the database are shown in Figures 2–4, respectively. We have implemented random shifting and random weak Gaussian blurred augmentation techniques to enhance the dataset. This dataset is taken from the Kaggle website. All the 3D volumes in both these databases are normalized to have intensity values in the range between 0 and 255.

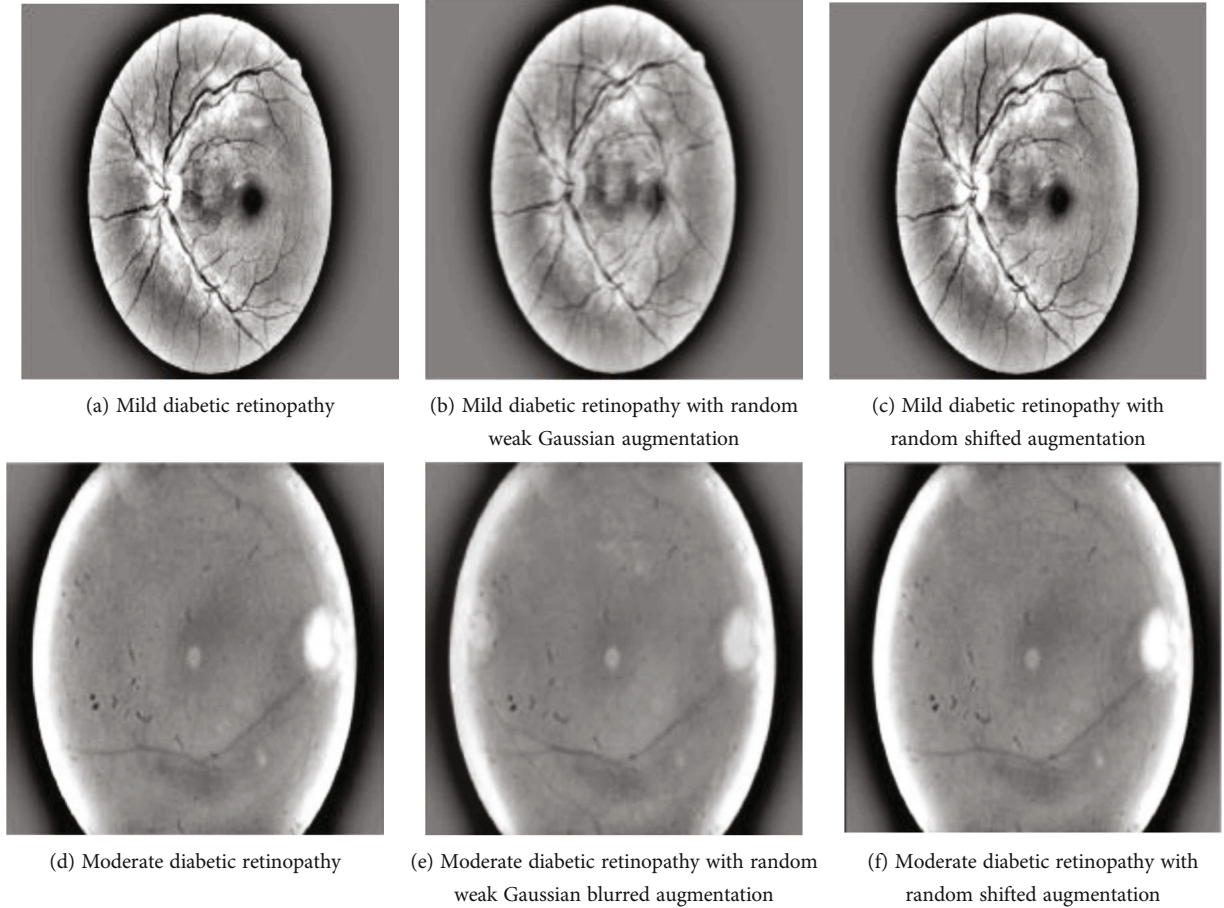


FIGURE 2: Sample images of mild and moderate classes in the Kaggle DR database with random shifted and random weak Gaussian blurred augmentation methods.

3. Methodology

In this work, we have considered both binary and multi-class classification tasks using different DL-based 3D-CNN architectures. These architectures are presented visually in Figures 5 and 6, respectively.

As given in Figure 5, there are small differences between the architectures deployed without augmentation, combined augmentation schemes, and with random weak Gaussian blurring and random shifted augmentation schemes. Feature maps in the convolutional layers are 10 for the combined augmentation scheme, 8 for no augmentation, and 9 for the classification tasks involving random weak Gaussian blurring and random shifted augmentation schemes. Rest of the architectures in Figure 5 are equivalent. An input layer accepts a volume of size $210 \times 210 \times 12$ with rescale-zero-one normalization method that scales the values of the incoming input between 0 and 1 according to the minimum and maximum values per channel. After that, a block that is repeated 5 times named block-A consists of a 3D convolutional layer, batch normalization layer, rectified linear nit (ReLU) activation layer, and a max pooling 3D layer which is used for the extraction, normalization, and downsizing of feature maps. Subsequently, there is a block which is

repeated a single time named block-B consisting of 3 fully connected (FC) layers with number of neurons equal to 300, 150, and 2, one dropout layer with probability 0.1, and, finally, a softmax and a classification layer to culminate the binary classification task. ReLU is equivalent to $\max(x, 0)$. Batch normalization [57] is another technique used for the improvement of training efficiency through a reduction in the statistical difference between the fundus volumes [58]. It contributes to a rapid convergence and a reduction in sensitivity during learning process [59]. Dropout [60] is effective in reducing the overfitting of models by omitting both hidden and visible units during the training process. It is a type of regularization method that prevents complex formation of adaptations on the training data. Weight and bias L2 factors are added to encourage smaller weights and biases by penalizing a network based on the size of weights and biases. Transformation of the input values of the softmax function can be interpreted as probabilities. Table 1 shows detailed 3D-CNN architecture hyperparameters for the binary classification task with 8 feature maps in the convolutional 3D layer.

As given in Figure 6, there are small architectural differences between the architectures deployed using no augmentation, combined augmentation schemes, and with random

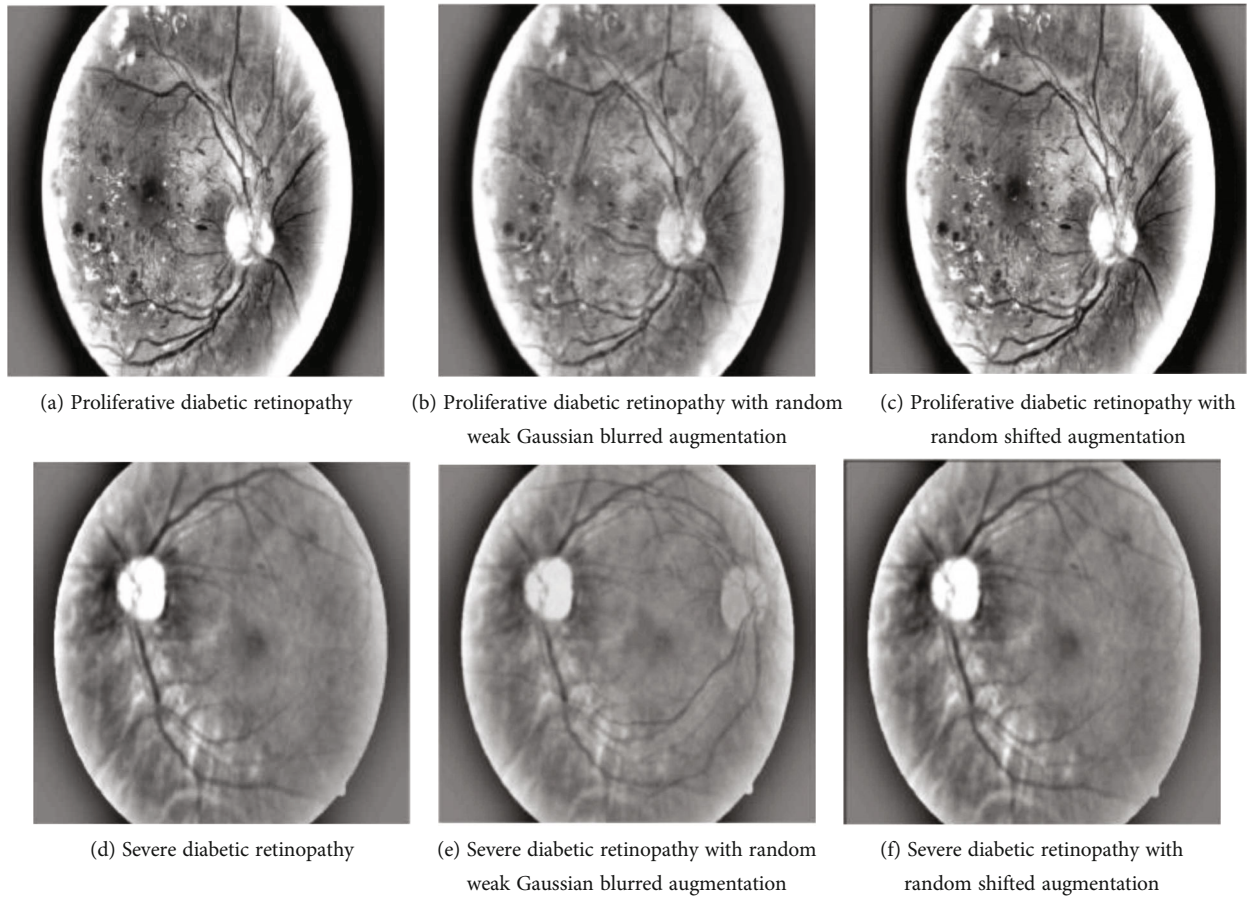


FIGURE 3: Sample images of severe and proliferative classes in the Kaggle DR database with random shifted and random weak Gaussian blurred augmentation methods.

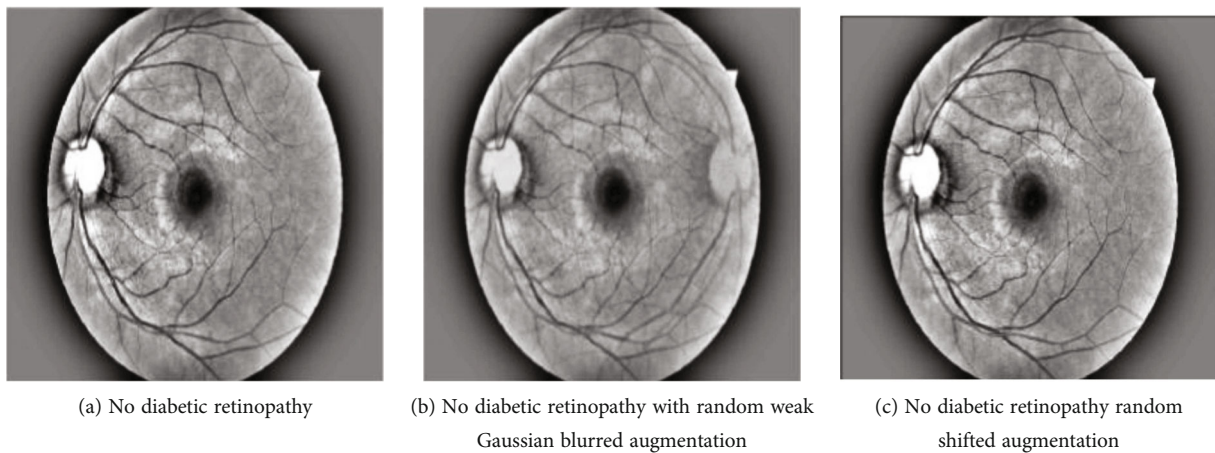


FIGURE 4: Sample images of no DR class in the Kaggle DR database with random shifted and random weak Gaussian blurred augmentation methods.

weak Gaussian blurring and random shifted augmentation schemes. The numbers of features maps in the convolutional layer are 12 for combined augmentations, 10 for no augmentation, and 11 for random weak Gaussian blurring and random shifted augmentation scheme-based classification tasks. The rest of the architectures are equivalent. An input layer

accepts a volume of size $512 \times 512 \times 2$ with zero centre normalization applied to the 3D volume. After that, there comes a block that is repeated 6 times named block-A consisting of a feature extracting convolutional layer, batch normalization layer, exponential linear unit (ELU) activation layer, and a max pooling 3D layer that is used for the extraction,

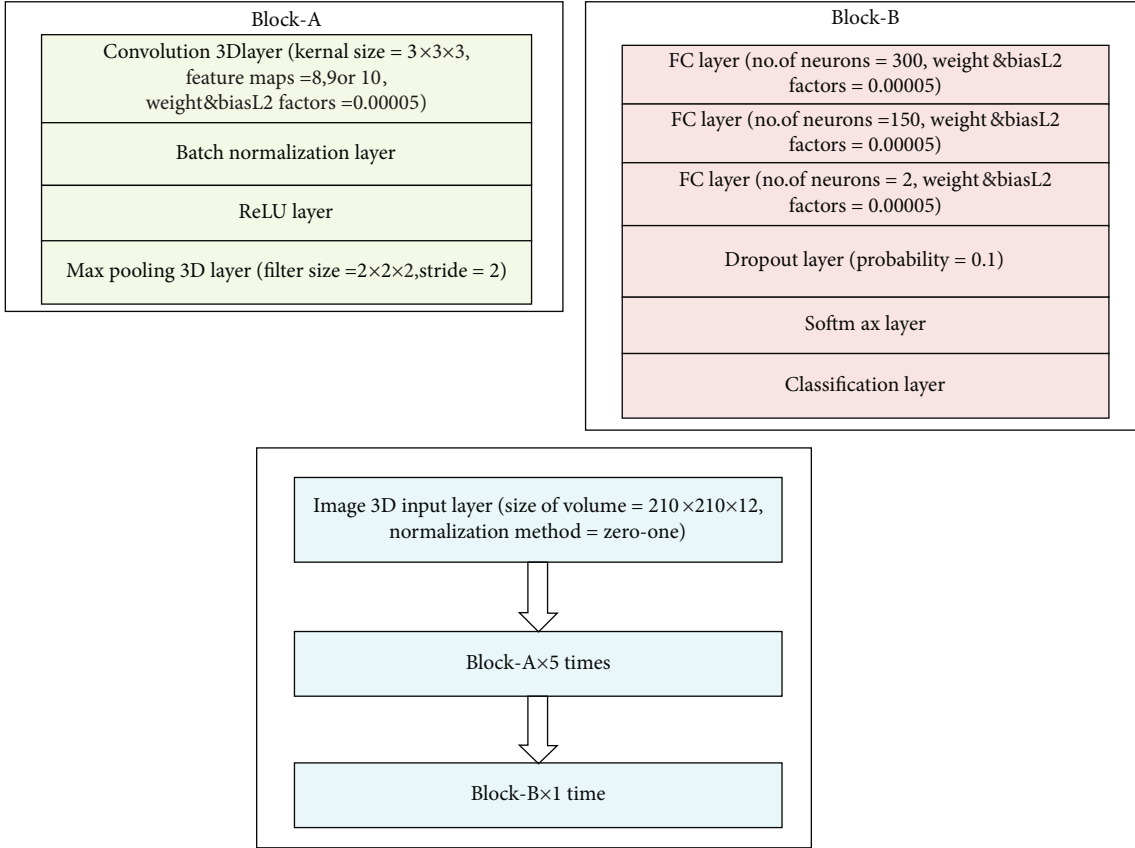


FIGURE 5: Architectures for binary classification (healthy/diseased) tasks without augmentation, with random shifted and random weak Gaussian blurred augmentation schemes, and by combined augmentation schemes. A tenfold cross-validation approach is used for hyperparameter selection.

normalization, and downsizing of feature maps. After that, there is another block that is repeated a single time named block-B consisting of 3 FC layers with 500, 300, and 5 neurons, one dropout layer with probability 0.1, and, finally, a softmax and a classification layer to culminate the multiclass (5 classes) classification task. ELUs [61] solve the vanishing gradient problem by having values in the negative region allowing them to push mean unit activations closer to zero but with lower computational complexity. Mathematically,

$$\text{ELU}(x) = \begin{cases} x, & x \geq 0, \\ \alpha(e^x - 1), & x < 0. \end{cases} \quad (1)$$

Table 2 shows detailed 3D-CNN architecture hyperparameters for the multiclass classification task with 10 feature maps in the convolutional 3D layer.

4. Experiments

We have performed experiments in the spatial domain for both binary and multiclass classification tasks to differentiate between the different categories of DR deploying two data augmentation methods: random weak Gaussian blurring and random shifting. We set the σ value to 1.5 for the ran-

dom weak Gaussian blurring and shift value to 1 or 2 pixels for the random shifting augmentations. We have also combined the training samples of both these augmentation methods. We have carried out the experiments related to the following tasks: (1) binary classification of healthy/diseased classes without augmentation, (2) binary classification of healthy/diseased classes with random weak Gaussian blurring augmentation, (3) binary classification of healthy/diseased classes with random shifting augmentation, (4) binary classification of healthy/diseased classes with combined random shifting and random weak Gaussian blurring augmentation methods, (5) multiclass classification without augmentation, (6) multiclass classification with random weak Gaussian blurring augmentation, (7) multiclass classification with random shifting augmentation, and, finally, (8) multiclass classification with combined random shifting and random weak Gaussian blurring augmentation methods. A number of samples in the training and validation splits are 72 and 8 for each class, respectively, in the case of binary classification task. We have also created a test split and place 19 samples of healthy class and 3 samples of diseased class in this split. To run experiments on the test split, we have deployed complete dataset of 80 samples of each class in the training split and only samples of the test subset in the validation split.

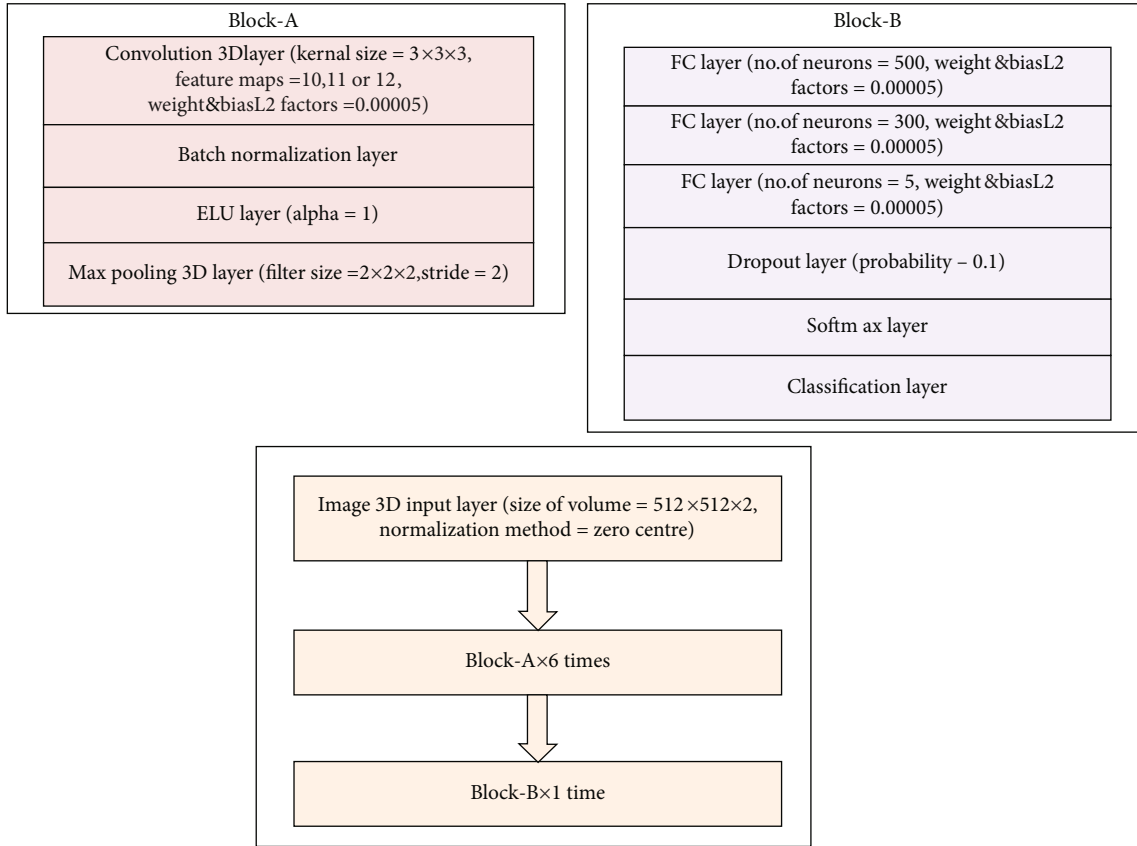


FIGURE 6: Architectures for multiclass classification tasks without augmentation, with random shifted and random weak Gaussian blurred augmentation schemes, and by combined augmentation schemes. A tenfold cross-validation approach is used for hyperparameter selection.

TABLE 1: Architecture hyperparameters for the proposed 3D-CNN model for binary classification task with 8 feature maps in the convolutional layer.

Layer	Filter size	Number of filters	Stride size	Dropout rate	Output size
Conv1+BN+ReLU	$3 \times 3 \times 3$	8	1	—	$8 \times 210 \times 210 \times 12$
MaxPool1	$2 \times 2 \times 2$	—	2	—	$8 \times 105 \times 105 \times 6$
Conv2+BN+ReLU	$3 \times 3 \times 3$	8	1	—	$8 \times 105 \times 105 \times 6$
MaxPool2	$2 \times 2 \times 2$	—	2	—	$8 \times 53 \times 53 \times 3$
Conv3+BN+ReLU	$3 \times 3 \times 3$	8	1	—	$8 \times 53 \times 53 \times 3$
MaxPool3	$2 \times 2 \times 2$	—	2	—	$8 \times 27 \times 27 \times 2$
Conv4+BN+ReLU	$3 \times 3 \times 3$	8	1	—	$8 \times 27 \times 27 \times 2$
MaxPool4	$2 \times 2 \times 2$	—	2	—	$8 \times 14 \times 14 \times 1$
Conv5+BN+ReLU	$3 \times 3 \times 3$	8	1	—	$8 \times 14 \times 14 \times 1$
MaxPool5	$2 \times 2 \times 2$	—	2	—	$8 \times 7 \times 7 \times 1$
FC 1	—	300	—	—	300
FC 2	—	150	—	—	150
FC 3	—	2	—	—	2
Dropout	—	—	—	0.1	2
Softmax	—	—	—	—	2

FC: fully connected; MaxPool: maximum pooling; BN: batch normalization; ReLU: rectified linear unit; Conv: convolutional.

TABLE 2: Architecture hyperparameters for the proposed 3D-CNN model for multiclass classification task with 10 feature maps in the convolutional layer.

Layer	Filter size	Number of filters	Stride size	Dropout rate	Output size
Conv1+BN+ELU	$3 \times 3 \times 3$	10	1	—	$10 \times 512 \times 512 \times 2$
MaxPool1	$2 \times 2 \times 2$	—	2	—	$10 \times 256 \times 256 \times 1$
Conv2+BN+ELU	$3 \times 3 \times 3$	10	1	—	$10 \times 256 \times 256 \times 1$
MaxPool2	$2 \times 2 \times 2$	—	2	—	$10 \times 128 \times 128 \times 1$
Conv3+BN+ELU	$3 \times 3 \times 3$	10	1	—	$10 \times 128 \times 128 \times 1$
MaxPool3	$2 \times 2 \times 2$	—	2	—	$10 \times 64 \times 64 \times 1$
Conv4+BN+ELU	$3 \times 3 \times 3$	10	1	—	$10 \times 64 \times 64 \times 1$
MaxPool4	$2 \times 2 \times 2$	—	2	—	$10 \times 32 \times 32 \times 1$
Conv5+BN+ELU	$3 \times 3 \times 3$	10	1	—	$10 \times 32 \times 32 \times 1$
MaxPool5	$2 \times 2 \times 2$	—	2	—	$10 \times 16 \times 16 \times 1$
Conv6+BN+ELU	$3 \times 3 \times 3$	10	1	—	$10 \times 16 \times 16 \times 1$
MaxPool6	$2 \times 2 \times 2$	—	2	—	$10 \times 8 \times 8 \times 1$
FC 1	—	500	—	—	500
FC 2	—	300	—	—	300
FC 3	—	5	—	—	5
Dropout	—	—	—	0.1	5
Softmax	—	—	—	—	5

FC: fully connected; MaxPool: maximum pooling; BN: batch normalization; ELU: exponential linear unit; Conv: convolutional.

A number of samples in the training and validation splits are 225 and 25 for each class, respectively, in the case of multiclass classification task. We have also created a test split and place 12 samples of each class in this split. To run experiments on the test split, we have employed complete dataset of 250 samples of each class in the training split and only samples of the test subset in the validation split.

For experiments on the binary classification tasks, we have used the following settings: minibatch size is set to 2, initial learning rate is set to 0.001, epochs are set to 50, learning rate schedule is set to piecewise, optimization algorithm is Adam [62], categorical cross-entropy is chosen as a loss function, total number of experiments equals 41, while time taken to perform these experiments is approximately 642 minutes or 10.7 hours.

For experiments on the multiclass classification tasks, we have considered the following settings: minibatch size is set to 2, initial learning rate is set to 0.001, epochs are set to 30, learning rate schedule is set to piecewise, optimization algorithm is Adam, loss function is categorical cross-entropy, total number of experiments equals 41, while time taken to perform these experiments is approximately 8448 minutes or 140 hours.

5. Results and Discussion

For binary classification between healthy and diseased classes, the experimental results are presented in Tables 3 and 4, respectively. As a visual aid, Figure 7 presents the results given in Table 3 while Figure 8 presents the results given in Table 4. We have used accuracy, F_1 -score, Matthews

TABLE 3: Performance metrics for the binary classification tasks without augmentation, with random weak Gaussian blurred augmentation, with random shift augmentation, and with combined augmentations.

Task	Performance metrics
Without augmentation	Accuracy = 55.63%
	F_1 - score = 56.44%
	MCC = 0.1126
	Sensitivity = 57.5%
	Specificity = 53.75%
With random weak Gaussian blurred augmentation	Precision = 55.42%
	Accuracy = 59.38%
	F_1 - score = 60.12%
	MCC = 0.1876
	Sensitivity = 61.25%
With random shifted augmentation	Specificity = 57.5%
	Precision = 59.04%
	Accuracy = 57.5%
	F_1 - score = 58.54%
	MCC = 0.1502
With combined augmentations (along with test split)	Sensitivity = 60%
	Specificity = 55%
	Precision = 57.14%
	Accuracy = 59.89%
	F_1 - score = 63.32%
	MCC = 0.1908
	Sensitivity = 63.64%
	Specificity = 55.42%
	Precision = 63%

TABLE 4: Ranking of methods for the binary classification tasks based on performance metrics without augmentation, with random weak Gaussian blurred augmentation, with random shift augmentation, and with combined augmentations.

Performance metric	Ranking
Accuracy	(1) With combined augmentations
	(2) With random weak Gaussian blurred augmentation
	(3) With random shift augmentation
	(4) Without augmentation
F_1 -score	(1) With combined augmentations
	(2) With random weak Gaussian blurred augmentation
	(3) With random shift augmentation
	(4) Without augmentation
MCC	(1) With combined augmentations
	(2) With random weak Gaussian blurred augmentation
	(3) With random shift augmentation
	(4) Without augmentation
Sensitivity	(1) With combined augmentations
	(2) With random weak Gaussian blurred augmentation
	(3) With random shift augmentation
	(4) Without augmentation
Specificity	(1) With random weak Gaussian blurred augmentation
	(2) With combined augmentations
	(3) With random shift augmentation
	(4) Without augmentation
Precision	(1) With combined augmentations
	(2) With random weak Gaussian blurred augmentation
	(3) With random shift augmentation
	(4) Without augmentation
Overall performance	(1) With combined augmentations
	(2) With random weak Gaussian blurred augmentation
	(3) With random shift augmentation
	(4) Without augmentation

correlation coefficient (MCC), sensitivity, specificity, and precision as performance metrics to assess the performance of different 3D-CNN architectures for this task which are trained from scratch. Mathematically,

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \quad (2)$$

$$F_1\text{-score} = \frac{2TP}{2TP + FP + FN}, \quad (3)$$

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \quad (4)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN}, \quad (5)$$

$$\text{Specificity} = \frac{TN}{TN + FP}, \quad (6)$$

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (7)$$

where TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative samples, respectively. Ranking of the methods for the binary classification tasks based on individual and collective performance metrics is given in Table 4.

As given in Tables 3 and 4, validation on test split is performed with the model trained using combined augmentation methods. The best performing model is the one that is trained using combined augmentation methods, then comes the model that is trained using random weak Gaussian blurred augmentation method, followed by the model trained using random shift augmentation scheme, and, finally, the model that does not use augmentation at all performed the worst. Here, combined augmentations mean combination of both random weak Gaussian blurred augmentation and random shifted augmentation schemes. Except for specificity metric, the rankings for all the methods remained the same which shows strong correlation between these performance metrics.

For the multiclass classification task, we have considered overall accuracy, relative classifier information (RCI), confusion entropy (CEN), index of balanced accuracy (IBA), geometric mean (GM), and MCC as performance metrics. Overall accuracy is the ratio of values that are correctly predicted to the sum of total values.

Tables 5–8 lists the complete statistics of the performance metrics for the multiclass classification tasks. In these tables, class-wise statistics for CEN, IBA, GM and MCC performance metrics as well as overall accuracy and RCI values for each of the four tasks, i.e., without augmentation, with random weak Gaussian blurred augmentation, with random shifted augmentation, and with combined augmentation schemes, are presented. Here, combined augmentations mean combination of both random weak Gaussian blurred augmentation and random shifted augmentation schemes. Finally, the statistics of the task involving test subset validated on the model trained without augmentation are also presented in these tables.

Table 9 lists the summary statistics of the performance metrics for the multiclass classification task, while Figure 9 visually presents the results given in Table 9. In this table, averages of CEN, IBA, GM, and MCC performance metrics are calculated by summing their class-wise values and dividing with 5. The RCI and overall accuracy values remain the same as in Table 5.

Table 10 presents a system of ranking based on the statistics given in Table 9 for the multiclass classification task. As a visual aid, Figure 10 visually presents the results given in Table 10. In this table, ranking based on individual performance metrics as well as an overall ranking obtained by considering the individual performance-based metrics is presented. Overall accuracy, RCI, IBA, GM, and MCC based ranking is obtained by considering the fact that higher values

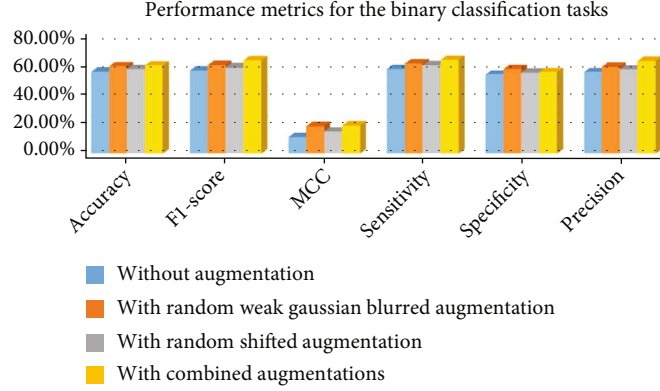


FIGURE 7: Visual presentation of the performance metrics for the binary classification tasks.

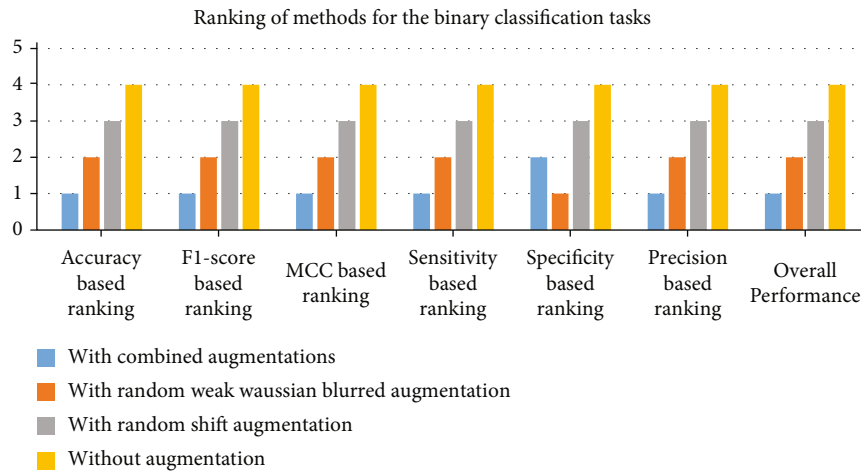


FIGURE 8: Visual presentation for ranking of methods for the binary classification tasks based on performance metrics.

TABLE 5: Overall accuracy, RCI, and class-wise CEN values for the multiclass classification tasks without augmentation, with random weak Gaussian blurred augmentation, with random shift augmentation, and with combined augmentations.

Method	Overall accuracy	RCI	CEN (mild)	CEN (moderate)	CEN (no)	CEN (proliferate)	CEN (severe)
Without augmentation	36.64%	0.0867	0.7752	0.8557	0.7734	0.5673	0.8015
With random weak Gaussian blurred augmentation	31.04%	0.038	0.8086	0.8458	0.8391	0.7008	0.8272
With random shifted augmentation	30.56%	0.0546	0.8259	0.8834	0.8242	0.6418	0.8165
With combined augmentations	33.12%	0.0647	0.793	0.8352	0.8085	0.6155	0.8341
Test set validated on the model trained without augmentation	26.66%	0.1522	0.6992	0.6274	0.9175	0.6904	0.7688

TABLE 6: Class-wise IBA values for the multiclass classification tasks without augmentation, with random weak Gaussian blurred augmentation, with random shift augmentation, and with combined augmentations.

Method	IBA (mild)	IBA (moderate)	IBA (no)	IBA (proliferate)	IBA (severe)
Without augmentation	0.1366	0.0649	0.1232	0.4747	0.1183
With random weak Gaussian blurred augmentation	0.1176	0.09	0.088	0.2458	0.1048
With random shifted augmentation	0.0967	0.0631	0.0833	0.3109	0.1095
With combined augmentations	0.1203	0.1	0.0997	0.3086	0.1031
Test set validated on the model trained without augmentation	0.1475	0.1974	0	0.073	0.1388

TABLE 7: Class-wise GM values for the multiclass classification tasks without augmentation, with random weak Gaussian blurred augmentation, with random shift augmentation, and with combined augmentations.

Method	GM (mild)	GM (moderate)	GM (no)	GM (proliferate)	GM (severe)
Without augmentation	0.5297	0.4162	0.5131	0.7558	0.5056
With random weak Gaussian blurred augmentation	0.4972	0.4523	0.4557	0.6307	0.4815
With random shifted augmentation	0.4635	0.3925	0.4513	0.6817	0.4889
With combined augmentations	0.5001	0.4639	0.4754	0.6891	0.476
Test set validated on the model trained without augmentation	0.4859	0.6038	0	0.4841	0.527

TABLE 8: Class-wise MCC values for the multiclass classification tasks without augmentation, with random weak Gaussian blurred augmentation, with random shift augmentation, and with combined augmentations.

Method	MCC (mild)	MCC (moderate)	MCC (no)	MCC (proliferate)	MCC (severe)
Without augmentation	0.1803	0.043	0.1599	0.4756	0.1484
With random weak Gaussian blurred augmentation	0.1238	0.0639	0.0773	0.3103	0.1093
With random shifted augmentation	0.077	-0.0176	0.0821	0.3933	0.1194
With combined augmentations	0.1264	0.0695	0.1051	0.4232	0.097
Test set validated on the model trained without augmentation	0.03636	0.3015	-0.2629	0.25	0.1666

TABLE 9: Summary statistics of performance metrics for the multiclass classification tasks without augmentation, with random weak Gaussian blurred augmentation, with random shift augmentation, and with combined augmentations.

Task	Performance metrics
Without augmentation	Overall accuracy = 36.64%
	RCI = 0.0867
	Average CEN = 0.75462
	Average IBA = 0.18354
	Average GM = 0.54408
With random weak Gaussian blurred augmentation	Average MCC = 0.20144
	Overall accuracy = 31.04%
	RCI = 0.038
	Average CEN = 0.8043
	Average IBA = 0.12924
With random shift augmentation	Average GM = 0.50348
	Average MCC = 0.13692
	Overall accuracy = 30.56%
	RCI = 0.0546
	Average CEN = 0.79836
With combined augmentations	Average IBA = 0.1327
	Average GM = 0.49558
	Average MCC = 0.13084
	Overall accuracy = 33.12%
	RCI = 0.0647
	Average CEN = 0.77726
	Average IBA = 0.14634
	Average GM = 0.5209
	Average MCC = 0.16424

of these metrics represent better classification while CEN-based ranking is obtained by considering that lower values are desirable as they represent better classification.

In Table 10, it can be observed that the model that performs the best is the one that is trained without augmentation followed by the model that is trained with combined augmentations, followed by the models that are trained with random weak Gaussian blurred and random shifted augmentation methods. Training without augmentation has the best performance considering individual and overall metric-based rankings, while combined augmentations have second best overall performance. Random shifted and random weak Gaussian blurred augmentation methods have equal performances. We can observe strong correlation among these performance metrics as depicted in their rankings where without augmentation and with combined augmentations can be completely specified by only a single performance metric alone. However, there is a clear difference between performances of methods employing random shift augmentation and random weak Gaussian blurred augmentation methods for the multiclass classification tasks when they are observed from individual metric-based performances alone. GM, MCC, and overall accuracy of methods employing random shifted augmentation are the worst while RCI, CEN, and IBA of methods employing random weak Gaussian blurred augmentation are the worst which signifies that these methods have disparities leading to difference in the opinion of these performance metrics. These differences could also be due to the way CNNs generalize to image transformations at a small scale [63].

For the multiclass classification task, we have found that the instances of proliferate DR class have the highest diagnostic performance. The performance of DL architectures is better in the case of binary classification than in the case of multiclass classification tasks, and this result is quite natural. Furthermore, architectures that combine different augmentation methods tend to perform better than those that do not. Furthermore, we have found the performance

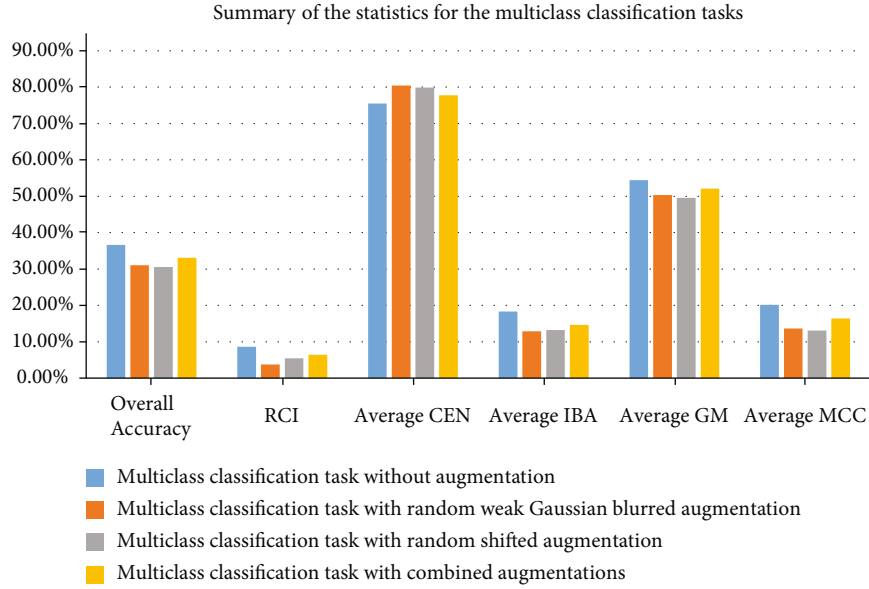


FIGURE 9: Visual presentation for summary statistics of performance metrics for the multiclass classification tasks.

TABLE 10: Ranking of methods based on performance metrics for the multiclass classification tasks without augmentation, with random weak Gaussian blurred augmentation, with random shift augmentation, and with combined augmentations.

Task	Ranking
Without augmentation	Overall accuracy – based ranking = 1
	RCI – based ranking = 1
	CEN – based ranking = 1
	IBA – based ranking = 1
	GM – based ranking = 1
	MCC – based ranking = 1
Overall ranking = 1	
With random weak Gaussian blurred augmentation	Overall accuracy – based ranking = 3
	RCI – based ranking = 4
	CEN – based ranking = 4
	IBA – based ranking = 4
	GM – based ranking = 3
	MCC – based ranking = 3
Overall ranking = 3	
With random shift augmentation	Overall accuracy – based ranking = 4
	RCI – based ranking = 3
	CEN – based ranking = 3
	IBA – based ranking = 3
	GM – based ranking = 4
	MCC – based ranking = 4
Overall ranking = 3	
With combined augmentations	Overall accuracy – based ranking = 2
	RCI – based ranking = 2
	CEN – based ranking = 2
	IBA – based ranking = 2
	GM – based ranking = 2
	MCC – based ranking = 2
Overall ranking = 2	

of architectures trained using random weak Gaussian blurring augmentation to be better than those that are trained using random shifted augmentation as the global sum of the feature maps will not be invariant to translation while performing the operation of convolution.

Architecture engineering has an impact on the performances of classification tasks. It can be observed that, for the binary classification tasks, large number of feature maps in the convolutional layer helps in getting better performances when compared with small number of feature maps in this layer. We can see that combined augmentation methods whose performances are better than other methods used large number of feature maps in the convolutional feature extracting layers. However, interesting observations can be seen for the multiclass classification tasks, where architectures with small number of feature maps help in getting the best performance overall. We can see architectures that did not employ any form of data augmentation performed better than those that employed data augmentation and these architectures employed less number of feature maps in the convolutional feature extracting layers. However, more feature maps in the convolutional layers help in getting better performances on the multiclass classification task as can be seen for the combined augmentations case that outperformed single augmentations for this task by using more complex architecture. In general, we can see the advantages brought by using deeper architectures in comparison with shallower ones when both these tasks (binary and multiclass classifications) are considered.

The suboptimal performance of DL architectures could be explained by the limited number of samples that we have used during training and validation processes [22]. Modern DL architectures require a lot of samples to train without experiencing overfitting issues. Another major limitation of our study is the lack of validation on a multicentre validation set which will prove beneficial in clinical practice. Finally, we

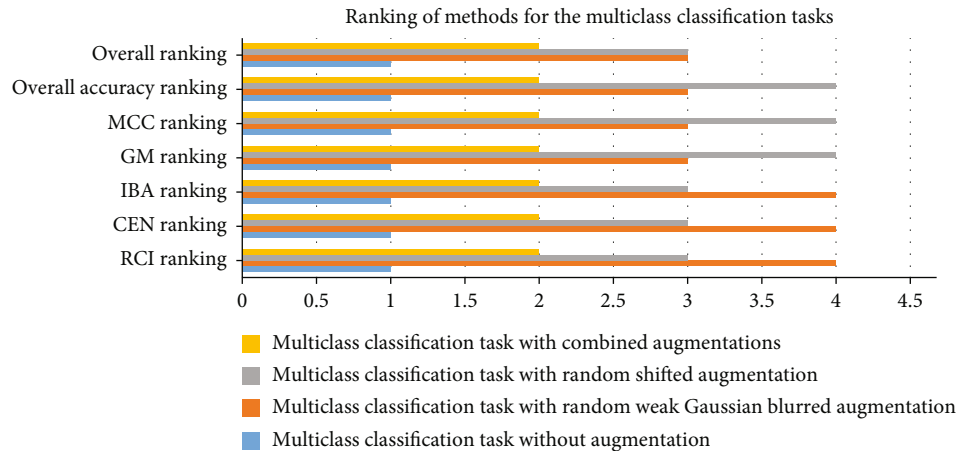


FIGURE 10: Visual presentation for ranking of methods based on performance metrics for the multiclass classification tasks.

hope that this pilot study deploying 3D CNN architectures with data augmentation schemes can be supportive to eye care specialists on the deployment of DL methods in terms of their clinical use.

6. Conclusions

In this research, we have utilized different DL methods to study both binary and multiclass classification problems to differentiate between different stages of DR. We have deployed 10-fold cross-validation approach to select optimal set of hyperparameters for the binary and multiclass classification tasks. For the binary classification task, we have found the performance of architecture trained using combined augmentation methods to be the best while the performance of model trained without any augmentation is found to be the worst. In contrast, in the multiclass case, we have observed the overall performance of model trained without augmentation to be the best while the performance of models trained with a single augmentation method whether random weak Gaussian blurring augmentation or random shifted augmentation to be the worst.

In the future, we will work on other retinal diseases such as retinal detachment using fundus images deploying data augmentation methods such as elastic/plastic deformations as well as other DL-based architectures such as graph convolutional networks. Eye diseases such as age-related macular degeneration, media haze, drusen, myopia, branch retinal vein occlusion, tessellation, epiretinal membrane, laser scars, macular scar, central serous retinopathy, optic disc cupping, central retinal vein occlusion, tortuous vessels, asteroid hyalosis, optic disc pallor, optic disc edema, opticociliary shunt, anterior ischemic optic neuropathy, parafoveal telangiectasia, retinal traction, retinitis, chorioretinitis, exudation, retinal pigment epithelium changes, macular hole, retinitis pigmentosa, and many other eye diseases [64] are affecting a large number of people worldwide, and their accurate and early detection using DL-based methods may allow for palliative care procedures employed by clinicians and medical practitioners.

Data Availability

The data employed to support the findings of this research is publicly available from the Kaggle platform and TeleOphta database.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

- [1] X. Lin, Y. Xu, X. Pan et al., "Global, regional, and national burden and trend of diabetes in 195 countries and territories: an analysis from 1990 to 2025," *Scientific Reports*, vol. 10, no. 1, pp. 1–11, 2020.
- [2] G. Kalyani, B. Janakiramaiah, A. Karuna, and L. V. N. Prasad, "Diabetic retinopathy detection and classification using capsule networks," *Complex & Intelligent Systems*, 2021.
- [3] E. Decenci ere, G. Cazuguel, X. Zhang et al., "TeleOphta: machine learning and image processing methods for teleophthalmology," *IRBM*, vol. 34, no. 2, pp. 196–203, 2013.
- [4] H. Riaz, J. Park, H. Choi, H. Kim, and J. Kim, "Deep and densely connected networks for classification of diabetic retinopathy," *Diagnostics*, vol. 10, no. 1, p. 24, 2020.
- [5] S. Sooraj and M. Bedeuzzaman, "Automatic classification of diabetic retinopathy based on deep learning - a review," in *2020 International Conference on Futuristic Technologies in Control Systems & Renewable Energy (ICFCR)*, pp. 1–5, Malappuram, India, 2020.
- [6] M. R. K. Mookiah, U. R. Acharya, C. K. Chua, C. M. Lim, E. Y. K. Ng, and A. Laude, "Computer-aided diagnosis of diabetic retinopathy: a review," *Computers in Biology and Medicine*, vol. 43, no. 12, pp. 2136–2155, 2013.
- [7] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [8] I. Ahmad, I. Ullah, W. U. Khan et al., "Efficient algorithms for E-healthcare to solve multiobject fuse detection problem," *Journal of Healthcare Engineering*, vol. 2021, Article ID 9500304, 16 pages, 2021.

- [9] C. You, W. Cong, G. Wang et al., "Structurally-sensitive multi-scale deep neural network for low-dose CT denoising," *IEEE Access*, vol. 6, pp. 41839–41855, 2018.
- [10] A. Samanta, A. Saha, S. C. Satapathy, S. L. Fernandes, and Y. D. Zhang, "Automated detection of diabetic retinopathy using convolutional neural networks on a small dataset," *Pattern Recognition Letters*, vol. 135, pp. 293–298, 2020.
- [11] S. R. Shiva, R. R. NilambarSethi, and M. Gadiraju, "Extensive analysis of machine learning algorithms to early detection of diabetic retinopathy," *Materials Today: Proceedings*, 2020.
- [12] G. Saxena, D. K. Verma, A. Paraye, A. Rajan, and A. Rawat, "Improved and robust deep learning agent for preliminary detection of diabetic retinopathy using public datasets," *Intelligence-Based Medicine*, vol. 3-4, article 100022, 2020.
- [13] M. M. Butt, G. Latif, D. N. F. A. Iskandar, J. Alghazo, and A. H. Khan, "Multi-channel convolutions neural network based diabetic retinopathy detection from fundus images," *Procedia Computer Science*, vol. 163, pp. 283–291, 2019.
- [14] M. M. Islam, H.-C. Yang, T. N. Poly, W.-S. Jian, and Y. C. Li, "Deep learning algorithms for detection of diabetic retinopathy in retinal fundus photographs: a systematic review and meta-analysis," *Computer Methods and Programs in Biomedicine*, vol. 191, p. 105320, 2020.
- [15] H. Safi, S. Safi, A. Hafezi-Moghadam, and H. Ahmadi, "Early detection of diabetic retinopathy," *Survey of Ophthalmology*, vol. 63, no. 5, pp. 601–608, 2018.
- [16] A. P. Bhatkar and G. U. Kharat, "Detection of diabetic retinopathy in retinal images using MLP classifier," in *2015 IEEE International Symposium on Nanoelectronic and Information Systems*, pp. 331–335, Indore, India, 2015.
- [17] T. Shanthi and R. S. Sabeenian, "Modified Alexnet architecture for classification of diabetic retinopathy images," *Computers & Electrical Engineering*, vol. 76, pp. 56–64, 2019.
- [18] S. Wan, Y. Liang, and Y. Zhang, "Deep convolutional neural networks for diabetic retinopathy detection by image classification," *Computers & Electrical Engineering*, vol. 72, pp. 274–282, 2018.
- [19] G. T. Zago, R. V. Andreão, B. Dorizzi, and E. O. Teatini Salles, "Diabetic retinopathy detection using red lesion localization and convolutional neural networks," *Computers in Biology and Medicine*, vol. 116, p. 103537, 2020.
- [20] V. Gulshan, L. Peng, M. Coram et al., "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *JAMA*, vol. 316, no. 22, pp. 2402–2410, 2016.
- [21] R. Sayres, A. Taly, E. Rahimy et al., "Using a deep learning algorithm and integrated gradients explanation to assist grading for diabetic retinopathy," *Ophthalmology*, vol. 126, no. 4, pp. 552–564, 2019.
- [22] J. Y. Choi, T. K. Yoo, J. G. Seo, J. Kwak, T. T. Um, and T. H. Rim, "Multi-categorical deep learning neural network to classify retinal images: a pilot study employing small database," *PLoS One*, vol. 12, no. 11, article e0187336, 2017.
- [23] S. M. S. Islam, M. M. Hasan, and S. Abdullah, "Deep learning based early detection and grading of diabetic retinopathy using retinal fundus images," 2018, <https://arxiv.org/abs/1812.10595>.
- [24] K. Shankar, A. R. W. Sait, D. Gupta, S. K. Lakshmanprabu, A. Khanna, and H. M. Pandey, "Automated detection and classification of fundus diabetic retinopathy images using synergic deep learning model," *Pattern Recognition Letters*, vol. 133, pp. 210–216, 2020.
- [25] E. Beede, E. Baylor, F. Hersch et al., "A human-centered evaluation of a deep learning system deployed in clinics for the detection of diabetic retinopathy," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, Honolulu, HI, USA, 2020.
- [26] N. Khare, P. Devan, C. Chowdhary et al., "SMO-DNN: spider monkey optimization and deep neural network hybrid classifier model for intrusion detection," *Electronics*, vol. 9, no. 4, p. 692, 2020.
- [27] I. Qureshi, J. Ma, and Q. Abbas, "Diabetic retinopathy detection and stage classification in eye fundus images using active deep learning," *Multimedia Tools and Applications*, vol. 80, no. 8, pp. 11691–11721, 2021.
- [28] S. Das, K. Kharbanda, S. M, R. Raman, and E. D. D., "Deep learning architecture based on segmented fundus image features for classification of diabetic retinopathy," *Biomedical Signal Processing and Control*, vol. 68, article 102600, 2021.
- [29] F. Li, Y. Wang, T. Xu et al., "Deep learning-based automated detection for diabetic retinopathy and diabetic macular oedema in retinal fundus photographs," *Eye*, 2021.
- [30] J. Limwattanayingyong, V. Nganthavee, K. Seresirikachorn et al., "Longitudinal screening for diabetic retinopathy in a nationwide screening program: comparing deep learning and human graders," *Journal of Diabetes Research*, vol. 2020, Article ID 8839376, 8 pages, 2020.
- [31] N. Tsiknakis, D. Theodoropoulos, G. Manikis et al., "Deep learning for diabetic retinopathy detection and classification based on fundus images: a review," *Computers in Biology and Medicine*, vol. 135, article 104599, 2021.
- [32] M. Karakaya and R. E. Hacisoftoglu, "Comparison of smartphone-based retinal imaging systems for diabetic retinopathy detection using deep learning," *BMC Bioinformatics*, vol. 21, no. S4, p. 259, 2020.
- [33] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Advances in Neural Information Processing Systems*, pp. 3630–3638, Curran Associates, Inc., 2016.
- [34] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems*, pp. 4077–4087, Curran Associates, Inc., 2017.
- [35] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: relation network for few-shot learning," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, Salt Lake City, UT, USA, 2018.
- [36] J. Xia, D. Deng, and D. Fan, "A note on implementation methodologies of deep learning-based signal detection for conventional MIMO transmitters," *IEEE Transactions on Broadcasting*, vol. 66, no. 3, pp. 744–745, 2020.
- [37] K. He, Z. Wang, D. Li, F. Zhu, and L. Fan, "Ultra-reliable MU-MIMO detector based on deep learning for 5G/B5G-enabled IoT," *Physical Communication*, vol. 43, pp. 101181–101187, 2020.
- [38] B. K. Yousafzai, S. Afzal, T. Rahman et al., "Student-performulator: student academic performance using hybrid deep neural network," *Sustainability*, vol. 13, no. 17, p. 9775, 2021.
- [39] W. U. Khan, M. A. Javed, T. N. Nguyen, S. Khan, and B. M. Elhalawany, "Energy-efficient resource allocation for 6G backscatter-enabled NOMA IoV networks," *IEEE Transactions on Intelligent Transportation Systems*, 2021.

- [40] C. Li, J. Xia, F. Liu et al., "Dynamic offloading for multiuser Multi-CAP MEC networks: a deep reinforcement learning approach," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 3, pp. 2922–2927, 2021.
- [41] W. U. Khan, F. Jameel, X. Li, M. Bilal, and T. A. Tsiftsis, "Joint spectrum and energy optimization of NOMA-enabled small-cell networks with QoS guarantee," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 8337–8342, 2021.
- [42] Y. Guo, Z. Zhao, K. He, S. Lai, J. Xia, and L. Fan, "Efficient and flexible management for industrial Internet of Things: a federated learning approach," *Computer Networks*, vol. 192, pp. 108122–108129, 2021.
- [43] W. U. Khan, X. Li, A. Ihsan, M. A. Khan, V. G. Menon, and M. Ahmed, "NOMA-enabled optimization framework for next-generation small-cell IoT networks under imperfect SIC decoding," *IEEE Transactions on Intelligent Transportation Systems*, 2021.
- [44] X. Li, Y. Zheng, W. U. Khan et al., "Physical layer security of cognitive ambient backscatter communications for green Internet-of-Things," *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 3, pp. 1066–1076, 2021.
- [45] W. U. Khan, X. Li, M. Zeng, and O. A. Dobre, "Backscatter-enabled NOMA for future 6G systems: a new optimization framework under imperfect SIC," *IEEE Communications Letters*, vol. 25, no. 5, pp. 1669–1672, 2021.
- [46] F. Jameel, W. U. Khan, N. Kumar, and R. Jantti, "Efficient power-splitting and resource allocation for cellular V2X communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3547–3556, 2021.
- [47] W. U. Khan, F. Jameel, N. Kumar, R. Jantti, and M. Guizani, "Backscatter-enabled efficient V2X communication with non-orthogonal multiple access," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 2, pp. 1724–1735, 2021.
- [48] A. U. Khan, M. Tanveer, W. U. Khan et al., "An enhanced spectrum reservation framework for heterogeneous user in CR-enabled IoT networks," *IEEE Wireless Communications Letters*, *Early Access*, vol. 10, p. 1, 2021.
- [49] A. B. Tufail, Y.-K. Ma, M. K. A. Kaabar et al., "Deep learning in cancer diagnosis and prognosis prediction: a minireview on challenges, recent trends, and future directions," *Computational and Mathematical Methods in Medicine*, vol. 2021, Article ID 9025470, 28 pages, 2021.
- [50] A. B. Tufail, Y.-K. Ma, Q.-N. Zhang et al., "3D convolutional neural networks-based multiclass classification of Alzheimer's and Parkinson's diseases using PET and SPECT neuroimaging modalities," *Brain Informatics*, vol. 8, no. 1, p. 23, 2021.
- [51] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221–231, 2013.
- [52] S. Tang, W. Zhou, L. Chen, L. Lai, J. Xia, and L. Fan, "Battery-constrained federated edge learning in UAV-enabled IoT for B5G/6G networks," *Physical Communication*, vol. 47, pp. 101381–101389, 2021.
- [53] J. Xia, L. Fan, W. Xu et al., "Secure cache-aided multi-relay networks in the presence of multiple eavesdroppers," *IEEE Transactions on Communications*, vol. 67, no. 11, pp. 7672–7685, 2019.
- [54] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *ICML*, pp. 807–814, Omnipress, 2010.
- [55] R. Khan, Q. Yang, I. Ullah et al., "3D convolutional neural networks based automatic modulation classification in the presence of channel noise," *IET Communications*, 2021.
- [56] R. Sayres, N. Hammel, and Y. Liu, "Artificial intelligence, machine learning and deep learning for eye care specialists," *Annals of Eye Science*, vol. 5, p. 18, 2020.
- [57] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning*, PMLR, JMLR, pp. 448–456, France, 2015.
- [58] X. Yin, J. L. Coatrieux, Q. Zhao et al., "Domain progressive 3D residual convolution network to improve low-dose CT imaging," *IEEE Transactions on Medical Imaging*, vol. 38, no. 12, pp. 2903–2913, 2019.
- [59] J. Ming, B. S. Yi, and Y. G. Zhang, "Low-dose CT image denoising using classification densely connected residual network," *KSII Transactions on Internet and Information Systems*, vol. 14, no. 6, pp. 2480–2496, 2020.
- [60] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.
- [61] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," 2015, <https://arxiv.org/abs/1511.07289>.
- [62] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2014, <https://arxiv.org/abs/1412.6980>.
- [63] A. Azulay and Y. Weiss, "Why do deep convolutional networks generalize so poorly to small image transformations?," *Journal of Machine Learning Research*, vol. 20, pp. 1–25, 2019.
- [64] S. Pachade, P. Porwal, D. Thulkar et al., "Retinal fundus multi-disease image dataset (RFMiD): a dataset for multi-disease detection research," *Data*, vol. 6, no. 2, p. 14, 2021.