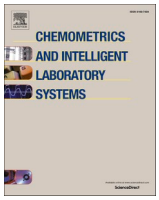
Chemometrics and Intelligent Laboratory Systems 237 (2023) 104820

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

journal homepage: www.elsevier.com/locate/chemometrics



A multi-level fine-tuned deep learning based approach for binary classification of diabetic retinopathy☆

Tawfiq Beghriche, Bilal Attallah *, Youcef Brik, Mohamed Djerioui

LASS Laboratory, Department of Electronics, Faculty of Technology, University of Msila, Algeria

ARTICLE INFO

Keywords:
Diabetic retinopathy (DR)
Classification
Transfer learning
Fine-tuning

ABSTRACT

Diabetes mellitus is a leading cause of diabetic retinopathy (DR), which results in retinal lesions and vision impairment. Untreated DR can lead to blindness, highlighting the need for early diagnosis and treatment. Unfortunately, DR has no cure, and treatments only help to preserve vision. Traditional manual diagnosis of DR retina fundus images by ophthalmologists is time-consuming, costly, and prone to errors. Computer-aided diagnosis methods, such as deep learning, have emerged as popular methods for improving diagnosis and reducing errors. Over the past decade, Convolutional Neural Networks (CNNs) have been shown to perform very well in medical image analysis due to their high ability to extract local features from images. Convolutional neural networks (CNNs) have shown great success in the processing of medical images, including DR color fundus images. In this paper, we proposed a multi-level fine-tuned deep learning based approach for the classification of diabetic retinopathy using three different pre-trained models including: DenseNet121, MobileNetV2, and Xception. The results are provided as classification accuracy, loss metrics, and the performance is compared with state-of-the-art works. The results indicates that the proposed Xception network surpassed its peers' models as well as state-of-the-art methods by achieving the highest accuracy of 97.95% in binary classification of DR images.

1. Introduction

Early disease detection improves the effectiveness of treatment in the healthcare industry. Diabetes is a condition that worsens when the body doesn't produce enough insulin [1]. 425 million adults globally are impacted [2]. The retina, heart, nerves, and kidneys are all impacted by diabetes [1,2]. Diabetes can result in diabetic retinopathy (DR), which enlarges and leaks blood and fluid from the retina's blood vessels [3]. If DR reaches a severe level, vision loss may result. 2.6% of blindness in the world is caused by DR [4]. Diabetes individuals who have had the condition for a long time are more likely to get DR. For diabetic patients to diagnose and treat DR early to reduce the risk of blindness, regular retinal screening is crucial [5]. Different types of lesions that present on a retinal scan can be used to identify DR. Microaneurysms (MA), hemorrhages (HM), and soft and hard exudates (EX) are examples of these lesions [1,6,7].

Actually, machine learning techniques have been widely used in various domains [8–13]. In thiscontext, manyresearchers have usedthem for the diagnosis of diabetes [14–17].Numerous studies have tried to use DL to automatically detect and classify DR lesions. According to the classification approach employed, these methods can be divided into four groups: binary classification, multi-level classification, lesion-based classification, and vessels-based classification.

We may simplify the research done to categorize the DR dataset into only two classes: normal and DR. Using a CNN, K. Xu et al. [18] automatically identified the photos in the Kaggle [19] dataset as either normal or DR images. 1000 photos from the dataset were utilized. Before supplying the images to the CNN, data augmentation and resizing to 224*224*3 were carried out. By applying several transformations, including rescaling, rotation, flipping, shearing, and translation, data augmentation was employed to boost the dataset's image count. Eight CONV layers, four max-pooling levels, and two FC layers made up the CNN architecture. For categorization, CNN's final layer used the SoftMax function. The accuracy of this approach was 94.5%. Three CNNs were trained to classify each image in the study by G. Quellec et al. [20] as referable DR (refer to moderate stage or more) or non-referable DR

☆ _Tawfiq Beghriche, Bilal Attallah, Youcef Brik, Mohamed Djerioui.

* Corresponding author.
E-mail addresses: tawfiq.beghriche@univ-msila.dz (T. Beghriche), bilal.attallah@univ-msila.dz (B. Attallah), youcef.brik@univ-msila.dz (Y. Brik), mohamed.djerioui@univ-msila.dz (M. Djerioui).

https://doi.org/10.1016/j.chemolab.2023.104820
Received 14 February 2023; Received in revised form 29 March 2023; Accepted 4 April 2023
Available online 12 April 2023

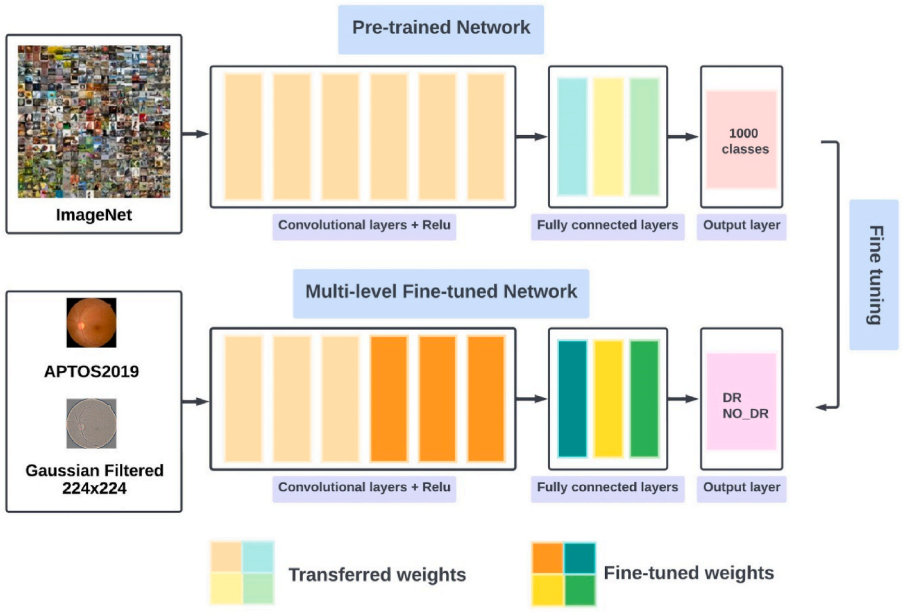


Fig. 1. The proposed fine-tuned network based system for the classification of DR.

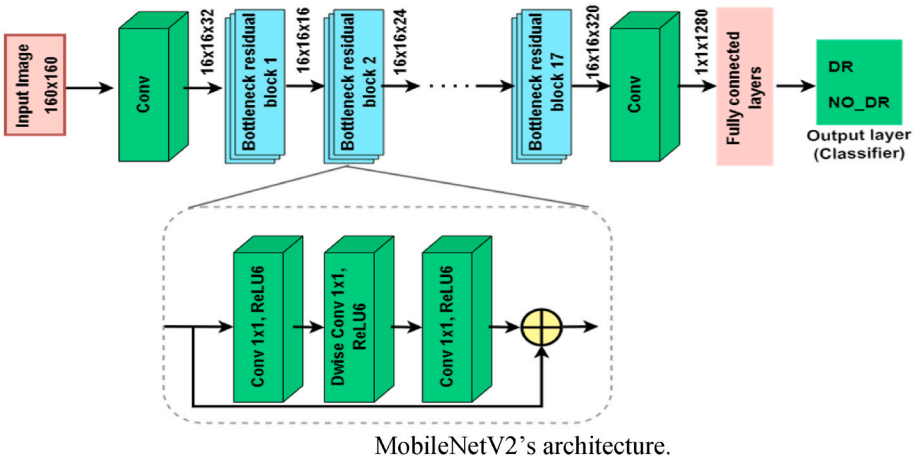


Fig. 2. The MobileNetV2's architecture.

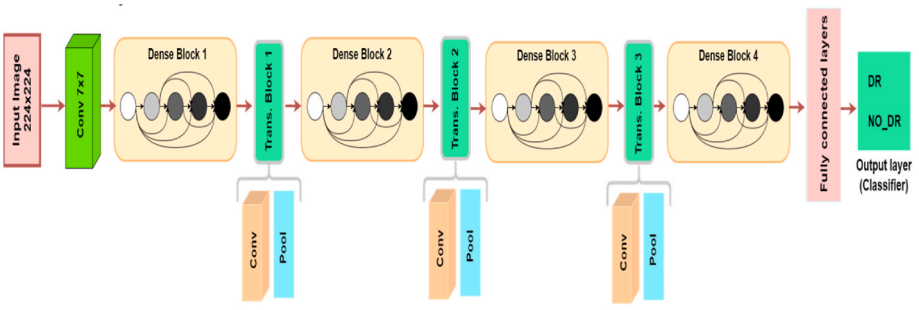


Fig. 3. The DenseNet121's architecture.

(No DR or mild stage) [21]. The photos were gathered from three datasets: private E-ophtha (107,799 images) [22], DiaretDB1 (89 images), and Kaggle (88,702 images). The photos were shrunk, cropped to 448 * 448 pixels, normalized, and had the FOV degraded by 5% during preprocessing. The augmented data were applied after a largeGaussian filter was performed. The two networks of the o O solution [23] and AlexNet [24] were used to pretrain the usedCNNs designs. The CNNs identified MA, HM, soft, and hard EX. Area under the ROC curve for this investigation was 0.954 for Kaggle and 0.949 for E-ophtha. In their study, M. T. Esfahan et al. [25] utilized ResNet34, a well-known CNN, to categorize DR images from the Kaggle dataset [19] as either normal or DR images. On the ImageNet database, ResNet34 is one of the available

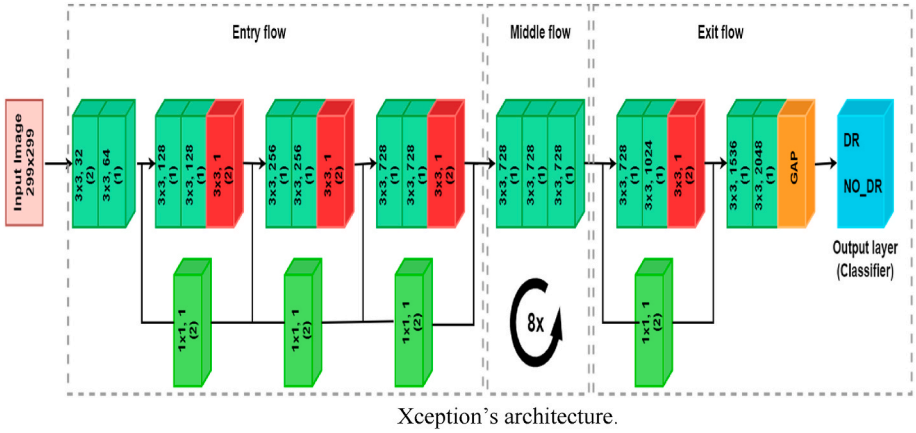


Fig. 4. The Xception's architecture.

Table 1
Fine tuning level Vs number of trainable parameters.

#of trainable parameters\Model	# of Un-freezed layers		
	10 (L1)	20 (L2)	30 (L3)
MobileNetV2	733,761	1,207,361	1,527,681
DenseNet121	169,153	368,193	642,433
Xception	5,498,369	7,328,425	8,942,401

pretrained CNN architectures. To enhance the quality of the photos, they used a number of image preparation techniques. The Gaussian filter, weighted addition, and picture normalization were all used in the image preprocessing. The images were 35,000 in number and 512 × 512 pixels in size. They claimed 85% accuracy and 86% sensitivity. To assess whether an image was referable DR, R. Pires et al. [26] created their own CNN architecture. 16 layers make up the proposed CNN, which is comparable to the pretrained VGG-16 [27] and the o O team [23]. During training, multi-image resolution and two-fold cross-validation were applied. After initializing the weights by the trained CNN on a lower picture resolution, the CNN of the 512*512 input image was trained. In order to lessen overfitting, the CNN was subjected to the drop-out and L2 regularization approaches. The Messidor-2 [28] and DR2 datasets were used to test the CNN after it had been trained on the Kaggledataset [19].Using data augmentation, the training dataset's classes were balanced. When testing the Messidor-2, the work achieved an area under the ROC curve of 98.2%. Three pretrained CNN models were combined in the work by H. Jiang et al. [29], including Inception V3 [30], Inception-Resnet-V2 [31]. To categorize their own dataset as referable DR or non-referable DR using andResnet152 [32]. The Adam optimizer was used to adjust the weights in CNN's training. Adaboost was employed to merge these models. Thedatasetof30,244image-swasresizedto520*520pixels, enhancedandaugmentedbeforebeingfed totheCNNs. The work produced an accuracy of 88.21% and an AUC of 0.946. A weighted pathways CNN (WP-CNN) was developed by Y. Liu et al. [33] to identify referable DR pictures. To balance the classes, they gathered over 60,000 photos that had been classified as referable or non-referable DR and enhanced them numerous times. Before being sent to CNN, these photos were reduced to 299*299 pixels and normalized. The WP-CNN consists of numerous CONV layers with various kernel sizes and various weighted routes that were combined by averaging. With 94.23% accuracy in their dataset and 90.84% accuracy in the STARE dataset, the pretrainedResNet [32], SeNet [34], and DenseNet [35] designs performed worse than the 105-layer WP-CNN. Using two CNN models, G. Zago et al. [36] discovered DR red lesions and DR pictures based on enhanced 65*65 patches. The CNNs employed were a bespoke CNN with five CONV, five max-polling layers, and an FC layer, as well as the trained VGG16 [27]. To categorize patches into red lesions or non-red lesions, these models were trained on the DIARETDB1 [37] dataset and tested on the DDR [38], IDRiD [39], Messidor-2, Messidor [40], Kaggle [19], and DIARETDB0 [41] datasets. After that, a lesion probability map of test images was used to classify if an image had DR or not. For the Messidor dataset, the results of this study produced the best sensitivity of 0.94 and an AUC of 0.912.

In this paper, we proposed a multi-level fine-tuned deep learning based approach for the classification of diabetic retinopathy using three pre-trained models: DenseNet121, MobileNetV2, and Xception. We investigated the impact of various parameters on the performance of each model and identified the ones that yielded the best results. By applying our approach, we were able to achieve better results compared to using each pre-trained model as it is. This study contributes to the field of pre-trained models in two ways. Firstly, it explored the use of three different pre-trained models. Secondly, it demonstrated the effectiveness of multi-level fine-tuning in improving the performance of pre-trained models. Overall, our findings underscore the importance of fine-tuning pre-trained models and suggest that multi-level fine-tuning is a promising approach for achieving better results in image classification tasks.

The remainder of this essay is structured as follows: An overview of the suggested system is given in the second section. Results and analysis are presented in the section that follows. Next, we demonstrate a comparison of state of the art methods. Finally, Section 5 brings the essay to a close.

2. Proposed system

In this paper, we adopted a multi-levelfine-tuning approach for classifying the DR images based on two datasets including APTOS2019, and DR Gaussian Filtered 224*224. These datasets serve as an input to three pre-trained convolutional neural network (CNN) architectures: DenseNet121, MobileNetV2, and Xception.First, we started by initializing the weights of the pre-trained CNNs with the weights of the corresponding ImageNet models. Next, we fine-tuned the CNNs on the DR datasets using lower learning rates, which allows the networks to adjust their weights to the specific characteristics of the DR imagesto identify the robust architecture behind these models which suits the DR classification task. The intuitions behind these techniques are discussed in this section. The overall overview of our proposed method is summarized in Fig. 1, which illustrates the different stages involved in the multi-level fine-tuning approach.

2.1. Datasets description

We used the publicly available APTOS 2019 Blindness Detection [42] to evaluate the proposed method. It contains 3662 retinal fundus

Table 2
Performance assessment of the used pretrained models with and without fine tuning, using both DR APTOS2019, and DR Gaussian Filtered 224 × 224.

Network/Model	Learning rate	# of Un-Freezing layers	Evaluation metrics																			
			MobileNetV2						DenseNet121						Xception							
			DR APTOS2019			DR Gaussian Filtered 224 × 224			DR APTOS2019			DR Gaussian Filtered 224 × 224			DR APTOS2019			DR Gaussian Filtered 224 × 224		
			Training phase		Validation phase	Training phase		Validation phase	Training phase		Validation phase	Training phase		Validation phase	Training phase		Validation phase	Training phase		Validation phase		
			Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
Customized Model	0.01	10 non	96.62	0.1033	96.31	0.1041	95.91	0.1195	94.12	0.2250	95.09	0.1456	95.90	0.1129	94.85	0.1502	93.57	0.1752	**96.32**	**0.1006**	**96.17**	**0.1361**	95.94	0.1191	95.49	0.1123
Fine Tuned	0.001	20 10	94.92	0.9666	93.71	0.9020	94.23	1.0161	95.49	1.0335	98.26	0.0521	96.03	0.1552	98.02	0.0637	94.66	0.2762	97.65	0.1427	96.99	0.1211	97.17	0.0945	97.26	0.2766
		20	93.69	0.9074	97.26	0.3941	92.85	1.1620	96.44	0.6246	99.49	0.0239	97.26	0.2580	99.86	0.0128	95.49	0.7334	98.70	0.1170	96.58	0.8996	98.57	0.1065	97.54	0.9234
		30	49.95	3.3022	49.50	1.0480	49.23	7.6298	50.48	1.1351	99.95	0.0227	97.67	0.2884	99.56	0.0272	95.21	0.8480	99.52	0.0671	95.62	0.5208	97.61	0.1051	94.25	0.4105
Customized Model	0.001	10 non	95.67	0.1201	95.49	0.1045	96.01	0.1288	94.25	0.1609	95.50	0.1293	96.31	0.0990	94.98	0.1524	93.71	0.1652	95.39	0.1259	96.58	0.1175	95.70	0.1305	94.25	0.1414
Fine Tuned	0.0001	20 10	99.39	0.0166	94.94	0.1812	99.90	0.0063	95.49	0.2238	98.23	0.0465	96.58	0.1003	98.23	0.0525	94.94	0.1881	99.66	0.0077	97.67	0.1270	99.73	0.0081	97.13	0.1453
		20	99.28	0.0223	97.67	0.1455	99.15	0.0331	93.57	0.4840	99.49	0.0120	97.13	0.0915	99.69	0.0069	95.49	0.2984	99.56	0.0109	97.54	0.1666	99.93	0.0031	95.08	0.5178
		30	99.08	0.0381	96.58	0.1866	99.04	0.0514	94.66	0.3853	99.59	0.0099	97.13	0.1199	99.83	0.0049	95.90	0.3147	99.80	0.0038	97.95	0.2375	99.90	0.0062	95.49	0.3030
Customized Model	0.0001	10 non	91.81	0.22	93.43	0.1802	91.98	0.2401	91.38	0.2555	89.53	0.268	93.16	0.2079	90.41	0.2538	90.29	0.2432	90.72	0.2762	92.20	0.2520	92.80	0.2582	90.42	0.2726
Fine Tuned	0.00001	20 10	98.36	0.0496	96.17	0.1026	96.76	0.0955	95.35	0.1447	95.02	0.1450	96.31	0.1041	95.02	0.1477	93.97	0.1744	97.71	0.0605	97.54	0.0809	97.54	0.0725	96.31	0.0905
		20	99.28	0.0188	97.26	0.0943	98.50	0.0416	95.49	0.1377	96.18	0.1013	97.26	0.0876	96.42	0.1063	94.66	0.1462	99.56	0.0152	97.81	0.0883	99.25	0.0219	95.90	0.1112
		30	99.66	0.0089	97.54	0.0951	99.45	0.0140	95.35	0.1680	96.96	0.0812	97.40	0.0830	97.00	0.0826	94.80	0.1381	99.73	0.0056	97.54	0.1150	99.93	0.0022	96.58	0.1602

Fig. 5. Performance comparison of the three pre-trained models based on validation Accuracy.

images. All images were clinically rated depending on the severity of DR. APTOS 2019 images were captured in different situations, thus, it has artifacts, blurred images, and varying brightness and resolutions. The severity of DR was divided into five levels, where the first level stands for no DR (a healthy retina) with 1805 images, while the fourth level concerning to proliferative DR (295 images). The remaining levels from second to fourth are in the order of mild, moderate, and severe DR, with the number of images 370, 999, and 193 respectively [43]. In addition to the APTOS 2019 Blindness Detection dataset, the DR Gaussian filtered 224 × 224 dataset [44] was also used in the study. This dataset was created using the APTOS 2019 images and consists of images that were resized to 224 × 224 pixels and filtered using a Gaussian filter. The severity levels of no DR, mild, moderate, severe, and proliferative were maintained in this dataset, as in the APTOS 2019 dataset. The source of the APTOS 2019 Blindness Detection dataset is publicly available, and it was obtained from the Kaggle competition "APTOS 2019 Blindness Detection". It's worth noting that the APTOS 2019 Blindness Detection dataset and the DR Gaussian filtered 224 × 224 dataset were both used to train and evaluate the proposed method. The addition of the DR Gaussian filtered dataset allowed for a more comprehensive evaluation of the method's performance across different image resolutions and preprocessing techniques. Overall, the datasets used in this study provide a diverse and representative collection of retinal fundus images that allow for the evaluation of the proposed method for detecting and classifying diabetic retinopathy.

2.2. Fine-tuned networks

Training a CNN from scratch is difficult, time-consuming and requires a huge amount of data for the model to converge. By this, fine-tuning of pre-trained CNN networks have become an option. Fine tuning is aiming to transferthe knowledge (weights)from CNN model that pre-trained on another dataset likeImageNet [45], that contains over 15 million images belonging to 1000 classes.It includes altering or modifying a part of (or compeletly) the previously trained network to satisfy the considered task, which is classifying the DR images into two categories. Additionally, the classification layer (Softmax) with the 1000 neurons must be removed and replaced with a layer that has neurons equivalent to the number of classes in the required task (2 neurons in our case) [46,47].

Convolutional neural networks (CNNs)architectures arecomposed of set of layers that learn various feature types from the input image. The first few layers are learning the kind of features that almost all images have (i.e. Low-level features)including edges, lines and corners.Therefore, these layers and their weights of the pre-trained modelsare frozen and used directly in the new task. While, the few later layers (fully connected) are picking up the more sophisticated features related to the

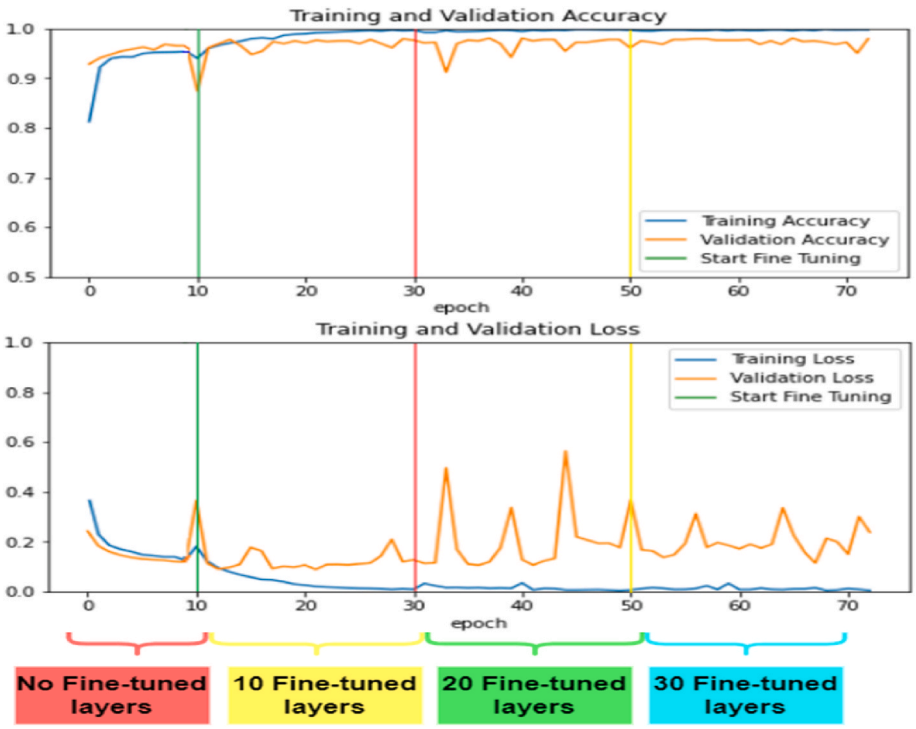


Fig. 6. Epochs vs accuracy and loss metrics in both training and validation phases before and after fine tuning process.

Table 3
Comparison of the proposed system with state of the art in terms of accuracy.

Authors	Method	Used database	Accuracy(%)
[56]	Inception V3	EyePacs	90.9%
[57]	Modified AlexNet	Messidor	96.25
[58]	weighted path CNN (WP-CNN)	Eye fundus images	94.23%
[59]	ConvNet	APTOS2019	97.41%
[60]	Custom CNN	Messidor-2	91.00%
[61]	InceptionV3	APTOS2019	95.56%
[62]	ResNet	APTOS 2019	96.35%
Ours	***Fine-tuned Xception***	***APTOS 2019***	***97.95%***

considered task; that's why we focuson fine tuning these layers first. In the case that CNN is still performing poorly, additional layers (middle) might be unfrozen and retrained until it providessatisfied performances by classifying the imagescorrectly [48,49].A well fine-tuned pre-trained CNN can outperform scratch-trained CNN and it can be more effective spatially whendealing with small datasets [47]. Therefore, threepre-trained networks such as MobileNetV2 [50], DenseNet121 [52], and Xception [53]have been employed in this paper for precise and automatic DR classification.

2.2.1. MobileNetV2

Sandler et al. [50] presented MobileNetV2, which was an improvement above MobileNetV1 [51]. One completely convolution layer, seven inverted residual blocks, one convolution layer, one average pooling layer, and one convolution layer make up the MobileNetV2 architecture's 11 layers. Three layers make up the inverted residual block: a 11 convolution with the ReLU6 activation function, a 1 1 convolution with depthwise separable ReLU6, and a 11 convolution with the linear transformation.As shown in Fig. 2.

2.2.2. DenseNet121

The DenseNet architecture was created in 2017 by Huang et al. [52]. In this architecture, information is acquired in accordance with connections from the current layer (ICIC EXPRESS LETTERS, VOL.15, NO.6, 2021 557) and combined in later levels (referred to as DenseNet). The DenseNet design consists of a convolution layer, pooling layer, three dense blocks, transition layers, one dense block, and a classification layer. Depending on how severe the bottleneck is, the DenseNet layers can range in depth from 121 to 264. The convolution layer after that has its size doubled thanks to the implementation of the convolution layer growth rate. The reduction in the number of parameters is directly influenced by the bottleneck structure's implementation.As shown in Fig. 3.

2.2.3. Xception

Chollet [53] designed the Xceptionarchitecture. The implementation of this architecture, the extreme version of the inception module, addressed deeper networks, computing time, and overfitting concerns. The depthwise separable convolutions are used in the extreme inception module. The three fundamental flows of the Xception architecture are entry, middle, and exit. The entrance flow's first layer is the input image, which has the dimensions 229 by 229 by 3. Following this are three residual connections, followed by 64 and 32 convolution layers using ReLU. The middle flow is connected to eight stacked residual connections. Two depth-wise separable convolutions, global average pooling, and the exit flow are stacked on top of one residual connection as shown in Fig. 4.

3. Experimental results

Nemours experiments have been done to explore the power of thenetworks that werementioned above. A performance based comparison was made between allthe used pre-trained networks before and after using the fine tuning techniques. Fine-tuning levels, epochs number, andthe range of learning rate values were shifted in order to get the best version of a model for precise and automatic DR classification.

3.1. Evaluation metrics

For comparing and evaluating the performance of the models, both the accuracy and the loss function are analysed and calculated.Their

mathematical formats are shown in Eq. (1), and Eq.2 respectively. Where, the True Positive (TP) and the True Negative (TN) stand for the DR images that are correctly labelled. While, False Negative (FN), and False Negative (FN) are referring to wrongly labelled DR images. These parameters (TP, FN, TN and FP)are extracted from the confusion matrix [48,54,55].

Accuracy (Acc): It is the percentage of the correct predictions that a classifier has made compared with the actual values of the target in the validation phase.

$$\mathbf{Acc} = \frac{(\mathbf{TP} + \mathbf{TN})}{(\mathbf{TP} + \mathbf{TN} + \mathbf{FP} + \mathbf{FN})} * \mathbf{100\%} \quad (1)$$

Binary cross-entropy loss (Loss): It is the most widely utilized loss function in classification issues. As the predicted probability approaches the true label, the cross-entropy loss decreases. It evaluates how well a classification model performs when it predicts an outcome that is a probability value between 0 and 1.

$$\boldsymbol{L} = \frac{\mathbf{1}}{\boldsymbol{m}} \sum_{i=1}^{m} (\boldsymbol{y_i} * \boldsymbol{log}(\widehat{\boldsymbol{y_i}}) + (\mathbf{1} - \boldsymbol{y_i}) * \boldsymbol{log}(\mathbf{1} - \widehat{\boldsymbol{y_i}})) \quad (2)$$

3.2. Prediction with fine-tuned networks

In order to evaluate the effectiveness of the proposed approach, three different pre-trained networkssuch as MobileNetV2, DenseNet121, and Xception were analysed and employed for DR prediction. The performance of these models has been verified using both the DR APTOS2019 and DR Gaussian Filtered 224 × 224 datasets.Numerous experiments have been done by varying the values of several hyper-parameters such as the learning rate, the number of epochs, and the number of un-frozen layers (fine tuning level)concerning all the used pre-trained networks. Table 1 shows the number of trainable parameters in each level for each model.

Table 2 shows The models' performance in both training and validation stages, where the MobileNetV2 has achieved the best accuracies of 99.28% and 97.67% respectively by these parameters' values (Lr = 0,0001,E = 20,UF-L = 20).While the DenseNet121 has achieved the best accuracy results of 99.95% (training) and 97.67% (validation) with these hyper-parameters values (Lr = 0,001,E = 20,UF-L = 30). Otherwise, The Xception has achieved promising performance rates with an accuracy of 99.80% in the training phase, and 97.95% in the validation phase with these hyper-parameters values (Lr = 0,0001,E = 20,UF-L = 30).

3.3. Performance comparison

A competitive comparison has been made betweenthe pre-trained models'performances that have been discussed in the previous subsection. Fig. 5 illustrates thiscomparison based on the accuracy performance of used pre-trained models. Obviously, The Xceptionnetwok outperformed its peers' models and it has achieved the highest accuracy of 97.95%. Where the remind networks also performedequally excellent with an accuracy of 97.67%. By this, the Xception architecture will be only taken under consideration for the rest of this paper.

Fig. 6 presents the plot of the accuracy and the loss criteria in the training phase as well as the validation one of the pre-trained Xception network. The pre-trained Xception has obtained the best performance rates (Accuracy = 97.95%, and Loss = 0.2375) in the validation phase with learning rate = 0.0001, when 30 layers have been fine-tuned.

3.4. Comparison with the state-of-the-art

To present how well our methodology in DR prediction performs, we compared it with other works which are reported in the literature. Table 3 has demonstrated that the results obtained by our method is surpassed the other existing works who used the same path work, and

Network/ Model	Learning rate	# of Un-Freezing layers	Evaluation metrics																			
			MobileNetV2						DenseNet121						Xception							
			DR APTOS2019				DR Gaussian Filtered 224 × 224				DR APTOS2019				DR Gaussian Filtered 224 × 224				DR APTOS2019				DR Gaussian Filtered 224 × 224	
			Training phase		Validation phase		Training phase		Validation phase		Training phase		Validation phase		Training phase		Validation phase		Training phase		Validation phase		Training phase	
			Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
Customized Model	0.01	10 non	96.62	0.1033	96.31	0.1041	95.91	0.1195	94.12	0.2250	95.09	0.1456	95.90	0.1129	94.85	0.1502	93.57	0.1752	**96.32**	**0.1006**	**96.17**	**0.1361**	95.94	0.1191	95.49	0.1123
Fine Tuned	0.001	20 10	94.92	0.9666	93.71	0.9020	94.23	1.0161	95.49	1.0335	98.26	0.0521	96.03	0.1552	98.02	0.0637	94.66	0.2762	97.65	0.1427	96.99	0.1211	97.17	0.0945	97.26	0.2766
		20	**93.69**	**0.9074**	**97.26**	**0.3941**	92.85	1.1620	96.44	0.6246	99.49	0.0239	97.26	0.2580	99.86	0.0128	95.49	0.7334	98.70	0.1170	96.58	0.8996	98.57	0.1065	97.54	0.9234
		30	49.95	3.3022	49.50	1.0480	49.23	7.6298	50.48	1.1351	**99.95**	**0.0227**	**97.67**	**0.2884**	99.56	0.0272	95.21	0.8480	99.52	0.0671	95.62	0.5208	97.61	0.1051	94.25	0.4105
Customized Model	0.001	10 non	95.67	0.1201	95.49	0.1045	96.01	0.1288	94.25	0.1609	95.50	0.1293	96.31	0.0990	94.98	0.1524	93.71	0.1652	95.39	0.1259	96.58	0.1175	95.70	0.1305	94.25	0.1414
Fine Tuned	0.0001	20 10	99.39	0.0166	94.94	0.1812	99.90	0.0063	95.49	0.2238	98.23	0.0465	96.58	0.1003	98.23	0.0525	94.94	0.1881	99.66	0.0077	97.67	0.1270	99.73	0.0081	97.13	0.1453
		20	**99.28**	**0.0223**	**97.67**	**0.1455**	99.15	0.0331	93.57	0.4840	**99.49**	**0.0120**	**97.13**	**0.0915**	99.69	0.0069	95.49	0.2984	99.56	0.0109	97.54	0.1666	99.93	0.0031	95.08	0.5178
		30	99.08	0.0381	96.58	0.1866	99.04	0.0514	94.66	0.3853	99.59	0.0099	97.13	0.1199	99.83	0.0049	95.90	0.3147	**99.80**	**0.0038**	**97.95**	**0.2375**	99.90	0.0062	95.49	0.3030
Customized Model	0.0001	10 non	91.81	0.22	93.43	0.1802	91.98	0.2401	91.38	0.2555	89.53	0.268	93.16	0.2079	90.41	0.2538	90.29	0.2432	90.72	0.2762	92.20	0.2520	92.80	0.2582	90.42	0.2726
Fine Tuned	0.00001	20 10	98.36	0.0496	96.17	0.1026	96.76	0.0955	95.35	0.1447	95.02	0.1450	96.31	0.1041	95.02	0.1477	93.97	0.1744	97.71	0.0605	97.54	0.0809	97.54	0.0725	96.31	0.0905
		20	99.28	0.0188	97.26	0.0943	98.50	0.0416	95.49	0.1377	**96.18**	**0.1013**	**97.26**	**0.0876**	96.42	0.1063	94.66	0.1462	**99.56**	**0.0152**	**97.81**	**0.0883**	99.25	0.0219	95.90	0.1112
		30	**99.66**	**0.0089**	**97.54**	**0.0951**	99.45	0.0140	95.35	0.1680	96.96	0.0812	97.40	0.0830	97.00	0.0826	94.80	0.1381	99.73	0.0056	97.54	0.1150	99.93	0.0022	96.58	0.1602

the same performance measures. Among all the previous works we have compared our work to, J.D. Bodapati et al. [4].has used of the ConvNet model that yielded resultswhich was the closest to ours, with an accuracy of 97.41% compared to ours (97.95%) as seen in Table 3.

4. Conclusion

In this paper, we have proposed a multi-level fine-tuned CNN approach that can accurately classify DR images into affected and not affected categories. We evaluated the proposed approach using three different pre-trained networks, namely MobileNetV2, DenseNet121, and Xception, and verified their performance using both the DR APTOS2019 and DR Gaussian Filtered 224 × 224 datasets. Our results show that the fine-tuned Xception network has exceeded its peers' networks achieved by achieving the validation accuracy of 97.95% and outperformed other existing works in the last few years. In the future, the proposed system can be improved in various ways. Firstly, the performance of the proposed method can be further enhanced by exploring other pre-trained networks, such as EfficientNet and ResNet. Secondly, the use of transfer learning and data augmentation techniques can be explored to improve the generalizability of the proposed approach. Lastly, the proposed method can be extended to detect other retinal disorders, such as age-related macular degeneration and glaucoma, which will benefit the medical services.

Funding

The study did not receive any sort of external funding.

Authorship statement

Category 1.

Conception and design of study:Tawfiq Beghriche, Bilal Attallah, Youcef Brika, Mohamed Djerioui; acquisition of data: Tawfiq Beghrichea, Bilal Attallaha, Youcef Brika, Mohamed Djeriouia; analysis and/or interpretation of data: Tawfiq Beghriche, Bilal Attallah, Youcef Brik, Mohamed Djerioui.

Category 2.

Drafting the manuscript: Tawfiq Beghriche, Bilal Attallah, Youcef Brik, Mohamed Djerioui; revising the manuscript critically for important intellectual content:Tawfiq Beghrichea, Bilal Attallaha, Youcef Brika, Mohamed Djeriouia.

Category 3.

Approval of the version of the manuscript to be published (the names of all authors must be listed).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors of this study would like to express their gratitude to University of M'sila, Algeria, LASS Laboratory for their support and assistance in publishing this work.

References

[1] Handbook of Retinal Screening in Diabetes:diagnosis and Management(second ed.), John Wiley & Sons, Ltd Wiley .
[2] International diabetes federation - What Is Diabetes ([Online]).
[3] American academy of ophthalmology-What Is Diabetic Retinopathy?.
[4] R.R. Bourne, et al., Causes of vision loss worldwide, 1990-2010: a systematic analysis, Lancet Global Health 1 (6) (2013) 339–349.
[5] R. Chakrabarti, C.A. Harper, J.E. Keeffe, Diabetic retinopathy management guidelines, Expet Rev. Ophthalmol. 7 (5) (2012) 417–439.
[6] E. T. D. R. S. R. GROUP, Grading diabetic retinopa thy from stereoscopic color fundus photographs- an extension of the modified Airlie House classification, Ophthalmology 98 (5) (1991) 786–806.
[7] P.H. Scanlon, C.P. Wilkinson, S.J. Aldington, D.R. Matthews, A Practical manual of diabetic retinopathy management, in: Wiley-Blackwell, first ed., 2009.
[8] Y. Zhang, X. Xu, Predicting the thermal conductivity enhancement of nanofluids using computational intelligence, Phys. Lett. 384 (20) (2020), 126500.
[9] N. Hameed, A.M. Shabut, M.K. Ghosh, M.A. Hossain, Multi-class multi-level classification algorithm for skin lesions classification using machine learning techniques, Expert Syst. Appl. 141 (2019), 112961.
[10] F. Mohanty, S. Rup, B. Dash, Automated diagnosis of breast cancer using parameter optimized kernel extreme learning machine, Biomed. Signal Process Control 62 (2020) 102–108.
[11] X. Xu, Y. Zhang, Corn cash price forecasting with neural networks, Comput. Electron. Agric. 184 (2021), 106120.
[12] Y. Zhang, X. Xu, Modulus of elasticity predictions through LSBoost for concrete of normal and high strength, Mater. Chem. Phys. 283 (2022), 126007.
[13] S.S. Yadav, S.M. Jadhav, Detection of common risk factors for diagnosis of cardiac arrhythmia using machine learning algorithm, Expert Syst. Appl. 163 (2021). Article ID 113807.
[14] R. Ahuja, S.C. Sharma, M. Ali, A diabetic disease prediction model based on classification algorithms, Annals of Emerging Technologies in Computing 3 (3) (2019) 44–52.
[15] J.J. Khanam, S.Y. Foo, A Comparison of Machine Learning Algorithms for Diabetes Prediction", The Korean Institute of Communications and Information Sciences, Korea, 2021.
[16] A. Choudhury, D. Gupta, A survey on medical diagnosis of diabetes using machine learning techniques, Adv. Intell. Syst. Comput. 740 (2019) 67–78.
[17] M.A. Sarwar, K. Nasir, W. Hamid, M.A. Shah, Prediction of diabetes using machine learning algorithms in healthcare, in: Proceedings of the 24th International Conference on Automation & Computing, IEEE, Newcastle Upon Tyne, UK, 6 September 2018, pp. 1–6.
[18] K. Xu, D. Feng, H. Mi, Deep convolutional neural network-based early automated detection of diabetic retinopathy using fundus image, Molecules 22 (12) (2017) 2054.
[19] Kaggle dataset. https://kaggle.com/c/diabetic-retinopathy-detection.
[20] G. Quellec, et al., Deep image mining fordiabetic retinopathy screening, Med. Image Anal. 39 (2017) 178–193.
[21] T. Kauppiet, The DIARETDB1 diabetic retinopathy database and evaluationprotocol, in: Proceedings of the British Machine Vision Conference, 2007, pp. 1–10, 2007.
[22] E. Decencireet, TeleOphta : machine learning and image processing methods for teleophthalmology, IRBM 34 (2) (2013) 196–203.
[23] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90.
[24] o-O Team, solution [Online]. Available,, https://www.kaggle.com/c/diabetic-retinopathydetection/discussion/15617.
[25] M.T. Esfahani, M. Ghaderi, R. Kafiyeh, Classification of diabetic and normal fundus images using new deep learning method, Leonardo Electron. J. Pract. Technol. 17 (32) (2018) 233–248.
[26] R. Pires, et al., A data-driven approach to referable diabetic retinopathy detection, Artif. Intell. Med. 96 (2019) 93–106.
[27] K.Z.A. Simonyan, Very Deep Convolutional Networks for Large-scale Image Recognition, 2015 arXiv. Preprint., arXiv-1409.
[28] E. Decencière, et al., Feedback on a publicly distributed image database: the Messidor database, Image Anal. Stereol. 33 (3) (2014) 231–234.
[29] H. Jiang, et al., An interpretable ensemble deep learning model for diabetic retinopathy disease classification, 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (2019) 2045–2048. EMBC.
[30] C. Szegedy, et al., Rethinking the inception architecture for computer vision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2818–2826.
[31] C. Szegedy, S. Ioffe, V. Vanhoucke, A. Alemi, Inception-v4, inception-resnet and the impact of residual connections on learning, Proc. AAAI Conf. Artif. Intell. 31 (1) (2017).
[32] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[33] Y.P. Liu, et al., Referable diabetic retinopathy identification from eye fundus images with weighted path for convolutional neural network, Artif. Intell. Med. 99 (2019), 101694.
[34] J. Hu, L. Shen, G. Sun, Squeeze-and-Excitation Networks, IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.
[35] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.
[36] G.T. Zago, R.V. Andreão, B. Dorizzi, E.O.T. Salles, Diabetic retinopathy detection using red lesion localization and convolutional neural networks, Comput. Biol. Med. 116 (2020), 103537.

[37] T. Kauppiet, The diaretdb1 diabetic retinopathy database and evaluation protocol, BMVC 1 (2007).
[38] B.E. Bejnordi, et al., Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer, JAMA 318 (22) (2017) 2199–2210.
[39] P. Porwal, et al., Indian diabetic retinopathy image dataset (IDRiD): a database for diabetic retinopathy screening research, Data 3 (3) (2018) 25.
[40] Messidor. http://messidor.crihan.fr.
[41] T. Kauppi, et al., DIARETDB0: evaluation database and methodology for diabetic retinopathy algorithms, Machine Vision and Pattern Recognition Research Group, Lappeenranta University of Technology 73 (2006) 1–17. Finland.
[42] Blindness Detection. https://www.kaggle.com/c/aptos2019-blindness-detection/, 2019.
[43] A. Sugeno, Y. Ishikawa, T. Ohshima, R. Muramatsu, Simple methods for the lesion detection and severity grading of diabetic retinopathy by image processing and transfer learning, Comput. Biol. Med. 137 (2021), 104795.
[44] https://www.kaggle.com/datasets/sovitrath/diabetic-retinopathy-224x224-gaussian-filtered.
[45] J. Deng, et al., Imagenet: a large-scale hierarchical image database, IEEE conference on computer vision and pattern recognition (2009) 248–255, 2009.
[46] P.A. Vieira, et al., Classification of COVID-19 in X-ray images with genetic fine-tuning, Comput. Electr. Eng. 96 (2021), 107467.
[47] N. Tajbakhshet, Convolutional neural networks for medical image analysis: full training or fine tuning? IEEE Trans. Med. Imag. 35 (5) (2016) 1299–1312.
[48] O.E. Aina, S.A. Adeshina, A.P. Adedigba, A.M. Aibinu, Classification of cervical intraepithelial neoplasia (cin) using fine-tuned convolutional neural networks, Intelligence-Based Medicine 5 (2021), 100031.
[49] J. Howard, S. Ruder, Universal Language Model Fine-Tuning for Text Classification, 2018 *arXiv preprint arXiv:1801.06146*.
[50] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.C. Chen, MobileNetV2: Inverted Residuals and Linear Bottlenecks, 2018 *arXiv.org*, arXiv:1801.04381.
[51] A.G. Howard, MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, 2017 *arXiv.org*, arXiv:1704.04861.
[52] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.
[53] F. Chollet, Xception: Deep Learning with Depthwise Separable Convolutions, 2016 *arXiv.org*, arXiv:1610.02357.
[54] B. Yang, W. Bao, B. Chen, D. Song, Single_cell_GRN: gene regulatory network identification based on supervised learning method and Single-cell RNA-seq data, BioData Min. 15 (1) (2022) 1–18.
[55] B. Yang, W. Bao, B. Chen, PGRNIG: novel parallel gene regulatory network identification algorithm based on GPU, Briefings in Functional Genomics 21 (6) (2022) 441–454.
[56] M.T. Hagos, S. Kant, Transfer Learning Based Detection of Diabetic Retinopathy from Small Dataset, 2019 *arXiv preprint*, arXiv:1905.07203.
[57] T. Shanthi, R. Sabeenian, Modified alexnet architecture for classification of diabetic retinopathy images, Comput. Electr. Eng. 76 (2019) 56–64.
[58] Y.P. Liu, Z. Li, C. Xu, J. Li, R. Liang, Referable diabetic retinopathy identification from eye fundus images with weighted path for convolutional neural network, Artif. Intell. Med. 99 (2019), 101694.
[59] J.D. Bodapati, et al., Blended multi-modal deep convnet features for diabetic retinopathyseverity prediction, Electronics 9 (6) (2020) 914.
[60] J. de La Torre, A. Valls, D.A. Puig, Deep learning interpretable classifier for diabetic retinopathy disease grading, Neurocomputing 396 (2020) 465–476.
[61] V. Vives-Boix, D. Ruiz-Fernández, Diabetic retinopathy detection through convolutional neural networks with synaptic metaplasticity, Comput. MethodsPrograms Biomed. 206 (2021), 106094.
[62] R. Adriman, K. Muchtar, N. Maulina, Performance evaluation of binary classification of diabetic retinopathy through deep learning techniques using texture feature, Proc. Comput. Sci. 179 (2021) 88–94.