Diabetic Retinopathy Classification Using an Efficient Convolutional Neural Network

Jiaxi Gao*, Cyril Leung†, Chunyan Miao‡
* † *Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, Canada*

‡ *Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly*
Nanyang Technological University, Singapore
Email: {jiaxig,cleung}@ece.ubc.ca, ascymiao@ntu.edu.sg

***Abstract*—Diabetic Retinopathy (DR) is a diabetic complication that affects the eyes and may lead to blurred vision or even blindness. The diagnosis of DR through eye fundus images is traditionally performed by ophthalmologists who inspect for the presence and significance of many subtle features, a process which is cumbersome and time-consuming. As there are many undiagnosed and untreated cases of DR, DR screening of all diabetic patients is a huge challenge. Some previous works have applied deep convolutional neural networks(CNNs) to detect DR automatically. However, these methods employed very deep CNNs which require extensive computational resources. In this paper, we proposed a computationally efficient classification system based on efficient CNNs. Our results show that the proposed method achieves or surpasses state-of-the-art methods on two commonly used DR datasets.**

***Index Terms*—Diabetic Retinopathy, Computationally Efficient Convolutional Neural Network**

I. Introduction

Diabetic Retinopathy (DR) is a chronic and progressive diabetic complication which damages the retina. Globally, DR will affect 191 million people by 2030 [1]. It is caused by high blood glucose levels which can lead to blockage or damage in the tiny retinal blood vessels which nourish the retina. In response, the human body attempts to grow new blood vessels in the eye to maintain the nourishment. The new blood vessels are weak and have a high probability of leaking and bleeding [2]. As a result, patients may experience progressive vision disorders from blurred vision to vision loss [3]. Once the vision loss has progressed, it is often permanent. Research has shown that 98% of severe vision loss caused by DR can be prevented with early detection and treatment [4]. Currently, diagnosing DR requires trained ophthalmologists identifying tiny or subtle features such as exudates, microaneurysms or hemorrhages, a process which is challenging and time-consuming. Thus enhancing the accuracy and diagnosis speed using automatic DR classification can potentially have a significant impact on preventing vision disorders caused by DR.

There are several previous works focused on building automatic DR classification systems. Generally, automatic DR screening has evolved from traditional computer vision techniques which combined manually designed feature extraction algorithms and traditional classification algorithms to end-to-end deep learning algorithms. One drawback of the traditional method is that manually designed features are often over-specified, incomplete or require a long time and experience to design and validate. With the rapid increase in data and computation resources, CNN-based methods have significantly improved DR classification performance. Pratt proposed a 13-layer CNN for screening DR [5]. Holly et al. proposed two networks, namely CKML and VNXK, which are based on the networks of GoogLeNet and VGGNet-16 respectively, to classify DR using retinal images on a hybrid color space [6]. Wang et al. proposed the Zoom-In-Net which dealt with DR classification and lesion localization tasks simultaneously [7]. Although the above-mentioned methods achieve good performance on the DR screening task, the training process is very time-consuming and requires vast computation resources. In this paper, we propose a new computationally efficient CNN model MobileNet-Dense and ensemble it with the existing MobilNetV2 [8] to form an effective and efficient DR classification system. Compared to the state-of-the-art methods, our system has two major advantages:

1) **Achieves or surpasses state-of-the-art classification performance on two datasets:** Our system achieves state-of-the-art quadratic weighted kappa (QWK) score [9] on the EyePACS dataset and achieves the best accuracy (Acc) and area under the receiver operating characteristic curve (AUC) scores on the Messidor dataset.
2) **Reduced model complexity :** Compared to the state-of-the-art model on EyePACS dataset, our system requires 32% fewer learnable parameters and 73% fewer Multiply–accumulate operations (MAdds). Compared to the state-of-the-art model on the Messidor dataset, our system requires 83% fewer learnable parameters and 52% fewer MAdds.

The remainder of this paper is organized as follows. Section II introduces the proposed system. Section III describes the implementation details and experiment results. Section IV summarizes our main findings.

II. The proposed DR classification system

Fig 1 shows the general workflow of the proposed DR classification system. It consists of four main steps: preprocessing, data augmentation, model training, and thresh-

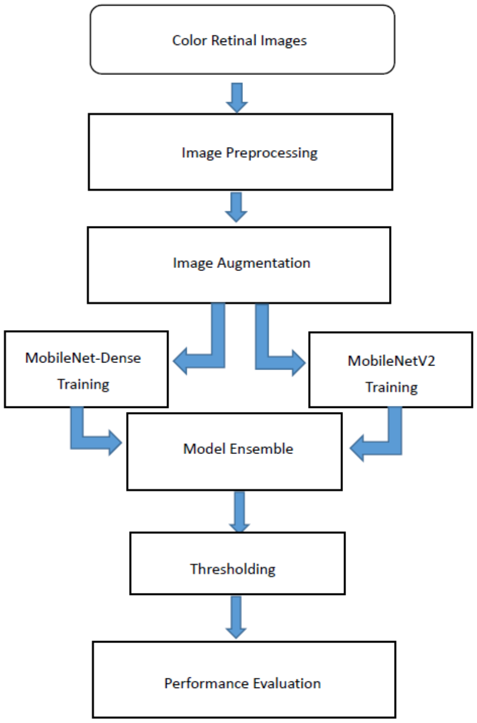


Fig. 1. Proposed automatic DR classification system.

olding optimization. The pre-processing step includes region of interest cropping and image resizing. The retinal image is cropped in order to remove the uninformative black border and resized to a uniform resolution. The image augmentation step expands the training data in order to overcome data imbalance and overfitting problems. The model training step consists of training two computationally efficient CNN networks and training a fully connected neural network to perform the model ensemble. The thresholding optimization step converts the decimal prediction to a categorical prediction using a gradient-free optimization algorithm.

A. Dataset

The EyePACS dataset is sponsored by the California Healthcare Foundation and used in the Kaggle Diabetic Retinopathy Detection Challenge [10]. It is a large DR dataset with high-resolution images taken under different imaging conditions. There are 17653/5453/21335 pairs of color retinal images for the training/validation/test set and the corresponding DR severity levels are provided. Based on the presence of DR lesions, each image is labeled with a severity grade scale from 0 to 4, which represents normal, mild, moderate, severe, and proliferative DR respectively.

The label distribution for the EyePACS training dataset is shown in Fig 2. Random samples from this dataset are shown in Fig 3. Each row contains 3 images of one class.

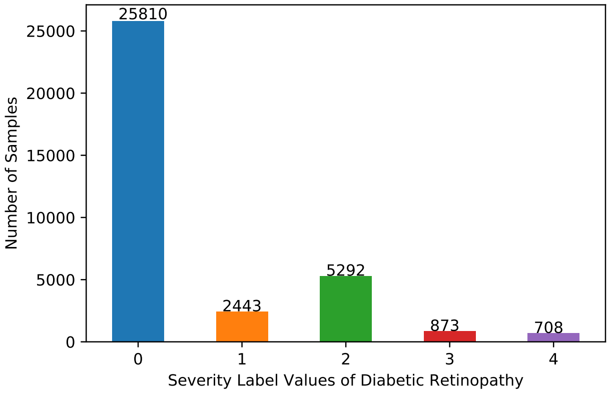


Fig. 2. Distribution of DR Severity Grade in EyePACS training set.

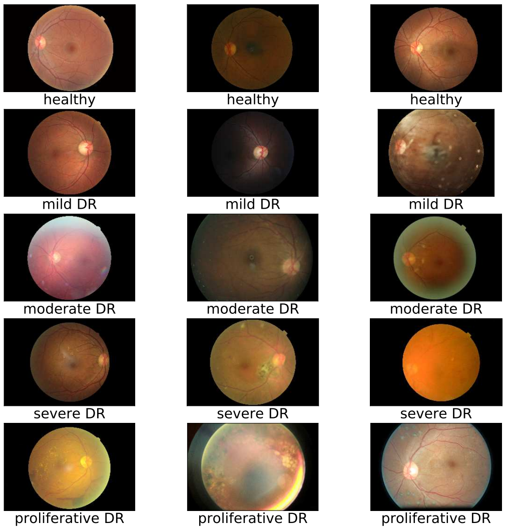


Fig. 3. Sample of retinal images from EyePACS training set [10].

B. Region of Interest Cropping and Resizing

The retinal fundus images contain uninformative black borders as shown in Fig 3. Thus the black border pixels with pixel intensities less than 10 is removed. Then the cropped image is resized to 672 × 672 pixels using bilinear interpolation. The 672 × 672 resolution is used as some of the subtle features of DR may not be captured with a smaller size image.

C. Image Augmentation

As shown in Fig 2, the distribution of the DR severity grades in the EyePACS dataset is very imbalanced, with 73% of the images labeled as healthy, 6% as mild DR, 15% as moderate DR, 2% as severe DR and 2% as proliferative DR. Hence we applied two types of image augmentation techniques to balance the training set. The first type includes geometric augmentation techniques which alter the geometry of the image by mapping the individual pixel values to new destinations [11]. To be more specific, we applied image flipping (horizontally or vertically), image rotation (0-360 degrees) and random cropping (90%). The second type of image augmentation technique is the Fancy PCA color augmentation proposed in [12] which alters the pixel values of the RGB channels. To obtain an augmented image, we first randomly apply one of the geometric augmentation techniques on the input image. The augmented images are then standardized by subtracting the channel means and dividing the channel standard deviations. Then we apply the Fancy PCA color augmentation technique to bring more variety to the training images. We utilized the open-source scikit-image library [13] to implement the discussed data augmentation techniques.

D. CNN Training

We utilized two computationally efficient CNN models to extract DR features. The details of the model structure and training configuration are introduced below.

1) Loss Function: We adopt the quadratic weighted kappa (QWK) [9] score as the evaluation metric since QWK is designed to measure the agreement of two raters on labels with ordinal scales and it has been used for reporting DR classification performance for existing models [7], [14]–[16].

The QWK, k_w, is defined as follows:

$$k_w = 1 - \frac{\sum_{i,j} w_{i,j} o_{i,j}}{\sum_{i,j} w_{i,j} e_{i,j}} \tag{1}$$

where $w_{i,j}$, $o_{i,j}$ and $e_{i,j}$ represent the ij^{th} entry of the weight matrix, observation rating matrix and the expected ratings matrix respectively. We can not optimizing k_w directly using gradient descent algorithm since k_w is non-differentiable. Therefore we consider the DR classification problem as an ordinal regression problem and we use the MSE loss to train the CNN models since both MSE loss and QWK score introduce quadratic penalty to the disagreement between the predicted result and the ground truth label.

2) MobileNetV2: The MobileNetV2 model is constructed using a stack of residual modules [8]. Each residual module is constructed using a stack of inverted residual blocks. The inverted residual block is summarized in Table I. It begins with a Conv1x1 layer which is used for expanding the depth of the input feature map by a factor of t ($t > 1$). Then a depthwise Conv3x3 layer is convolved with the expanded feature map. The last Conv1x1 layer with linear activation is used for compressing (restoring) the depth of the feature map to allow for element-wise addition. If the expansion ratio t is set to less than 1, we obtain the classical residual bottleneck block used in the ResNet model. The inverted residual block is shown to be resistant to information loss and is more memory efficient than the classical residual block [8]. The MobileNetV2 is computationally efficient since the depthwise separable convolution layer is applied throughout the network. It has been shown that a depthwise separable convolution layer uses between 8 to 9 times fewer computations than a standard convolution layer at the cost of a small reduction in performance [17]. The MobileNetV2 model we trained for DR classification is summarized in Table II. We trained MobileNetV2 using a batch size of 4 for 200 epochs with the Adam optimizer [18]. The initial learning rate is set to 10^{-4}, and is reduced to 10^{-5} at 80% of the total number of training epochs. After each epoch, we evaluate the QWK score of the current model checkpoint on the validation set and the model checkpoint which achieves a QWK score higher than 0.810 is saved for further testing and model ensemble. After training 200 epochs, the saved model checkpoints are sorted using the QWK score on the EyePACS validation dataset.

TABLE I
INVERTED RESIDUAL BLOCK [8]

Input	**Operator**	**Output**
$\boldsymbol{F}_{in} : h \times w \times d$	Conv1x1, Relu	$\boldsymbol{F}_1 : h \times w \times (td)$
$\boldsymbol{F}_1 : h \times w \times (td)$	DwConv3x3, stride=1,Relu	$\boldsymbol{F}_2 : h \times w \times (td)$
$\boldsymbol{F}_2 : h \times w \times (td)$	Conv1x1, Linear	$\boldsymbol{F}_3 : h \times w \times d$
$\boldsymbol{F}_{in}$, $\boldsymbol{F}_3$	Element-wise Addition	$\boldsymbol{F}_{out} : h \times w \times d$

3) MobileNet-Dense: Inspired by the empirical observation that DenseNet achieves a similar performance but with fewer MAdds and Parameters than ResNet on ImageNet dataset [19], we modified the existing MobileNetV2 model by replacing the residual connectivity with dense connectivity. We refer this new model as MobileNet-Dense. The MobileNet-Dense model is constructed using a stack of dense modules. Each dense module is constructed using a reduction block followed by a stack of bottleneck blocks. The structure of the bottleneck block and the reduction block are illustrated in Table III and Table IV respectively. The structure of the reduction block and the structure of the bottleneck block follow the structure of inverted residual block [8] in order to prevent information loss. The bottleneck block contains feature concatenation in order to allow for the dense connectivity while the reduction block does not include feature concatenation because it downsamples the width and height of the feature map using a depthwise convolution layer with a stride of 2. Following the design of DenseNet, we let each bottleneck block generate a feature map with k channels. Then the input feature map and the output feature map of the last Conv1x1 layer are concatenated along the channel (depth) axis to allow for the dense connectivity. The hyperparameter growth rate k can be used to trade off performance and model complexity. The overall MobileNet-Dense structure is summarized in Table V. The learning rate setting and model checkpoint saving scheme are the same as those for MobileNetV2.

E. Model Ensemble

Ensemble of machine learning models generally improves the classification system's robustness and accuracy [20]. It mimics the common practice of making decisions based on the decisions of multiple experts. Therefore we applied ensemble learning so as to improve the performance of the classification

TABLE II
THE STRUCTURE OF MOBILENETV2 FOR DR CLASSIFICATION

Input	Operator	M	r	s	t
$672^2 \times 3$	Conv3x3 Layer	40	1	2	-
$336^2 \times 40$	Residual Module	24	1	1	1
$336^2 \times 24$	Residual Module	32	2	2	6
$168^2 \times 32$	Residual Module	40	3	2	6
$84^2 \times 40$	Residual Module	80	4	2	6
$42^2 \times 80$	Residual Module	128	3	1	6
$42^2 \times 128$	Residual Module	208	3	2	6
$21^2 \times 208$	Residual Module	416	1	1	6
$21^2 \times 416$	Conv1x1 Layer	1664	1	1	-
$21^2 \times 1664$	GlobalAvgPool Layer	1664	1	-	-
1664	Fully Connected Layer	256	1	-	-
256	Fully Connected Layer	256	1	-	-
256	Output Layer	1	-	-	-

The input is a $672^2 \times 3$ image. Each line in the table describes a layer or a residual module. Each residual module is constructed using r inverted residual blocks, where the first building block has a stride of s and the following $(r-1)$ building blocks have a stride of 1. M denotes the depth of the output feature map for each layer or sequence. s denotes the stride of the depthwise Conv3x3 layer. Except for the first residual module, a constant expansion rate of $t = 6$ is applied throughout the network.

TABLE III
MOBILENET-DENSE BOTTLENECK BLOCK

Input	Operator	Output
$\boldsymbol{F}_{in} : h \times w \times d$	Conv1x1, Relu	$\boldsymbol{F}_1 : h \times w \times (td)$
$\boldsymbol{F}_1 : h \times w \times (td)$	DwConv3x3, stride=1,Relu	$\boldsymbol{F}_2 : h \times w \times (td)$
$\boldsymbol{F}_2 : h \times w \times (td)$	Conv1x1, Linear	$\boldsymbol{F}_3 : h \times w \times k$
$\boldsymbol{F}_{in}, \boldsymbol{F}_3$	Concatenation	$\boldsymbol{F}_{out} : h \times w \times (d+k)$

TABLE IV
MOBILENET-DENSE REDUCTION BLOCK

Input	Operator	Output
$\boldsymbol{F}_{in} : h \times w \times d$	Conv1x1, Relu	$\boldsymbol{F}_1 : h \times w \times (td)$
$\boldsymbol{F}_1 : h \times w \times (td)$	DwConv3x3, stride=2,Relu	$\boldsymbol{F}_2 : \frac{h}{2} \times \frac{w}{2} \times (td)$
$\boldsymbol{F}_2 : \frac{h}{2} \times \frac{w}{2} \times (td)$	Conv1x1, Linear	$\boldsymbol{F}_3 : \frac{h}{2} \times \frac{w}{2} \times d$

TABLE V
THE STRUCTURE OF MOBILENET-DENSE FOR DR CLASSIFICATION

Input	Operator	M	r	s	k	t
$672^2 \times 3$	Conv3x3 Layer	32	1	2	-	-
$336^2 \times 32$	Dense Block	48	1	1	16	1
$336^2 \times 48$	Dense Module	96	2	2	48	3
$168^2 \times 96$	Conv1x1 Layer	48	1	1	-	-
$168^2 \times 48$	Dense Module	144	3	2	48	3
$84^2 \times 144$	Conv1x1 Layer	72	1	1	-	-
$84^2 \times 72$	Dense Module	216	4	2	48	3
$42^2 \times 216$	Conv1x1 Layer	108	1	1	-	-
$42^2 \times 108$	Dense Module	300	5	2	48	3
$21^2 \times 300$	Conv1x1 Layer	1280	1	1	-	-
$21^2 \times 1280$	GlobalAvgPool Layer	1280	1	-	-	-
1280	Fully Connected Layer	256	1	-	-	-
256	Fully Connected Layer	256	1	-	-	-
256	Output Layer	1	-	-	-	-

The input is a $672^2 \times 3$ image. Each line in the table describes a layer or a dense module. Each dense module is constructed using r bottleneck blocks, where the first bottleneck block is a reduction block and the following $(r-1)$ building blocks are bottleneck blocks. We constructed the standard MobileNet-Dense model using 4 dense modules, where the dense modules contain 2, 3, 4 and 5 bottleneck blocks respectively. Between two adjacent dense modules, a Conv1x1 layer is applied to compress the depth of the feature map. M denotes the depth of the output feature map for each layer or dense module, k is the growth rate which denotes the number of convolution filters applied in the last Conv1x1 layer of each bottleneck block and t denotes the expansion rate of each building block. Except for the first dense block, a constant expansion rate of $t = 3$ and a constant growth rate of $k = 48$ is applied throughout the network.

system. Specifically, the features of each retinal image are extracted from the last fully connected layer (containing 256 nodes) of the Top-M saved checkpoints of MobileNet-Dense and Top-N saved checkpoints of MobileNetV2. Then, we concatenate features from different model checkpoints. In addition, as the EyePACS dataset provides retinal image of both left and right eyes of a patient, we also concatenate the features from both eyes in order to utilize label correlation (Statistics show that more than 87% of the eye pairs have the same DR levels in the EyePACS dataset). In other words, each retinal image has $512(M+N)$ features after concatenation. Then, principal component analysis (PCA) feature reduction is performed on the concatenated features in order to reduce the dimensionality of the features and thereby reducing overfitting and speeding up the training process. We select the number of components such that 99% of the variance is explained. Lastly, we trained a 3-layer fully connected neural network (FCNN) to screen left eyes using the preprocessed features and we trained another 3-layer FCNN to screen right eyes. We used the same structure for these two FCNNs, in which the first hidden layer contains 256 nodes and the second hidden layer contains 128 nodes. The output layer contains one node and MSE loss is used for training. We trained the FCNN model using a batch size of 256 for 120 epochs with the Adam optimizer [18]. A weight decay of 0.025 is applied to reduce overfitting. The initial learning rate is set to 0.0005, and it is successively decreased by a factor of 10 at 40%, 60% and 80% of the total number of training epochs.

F. Threshold Optimization

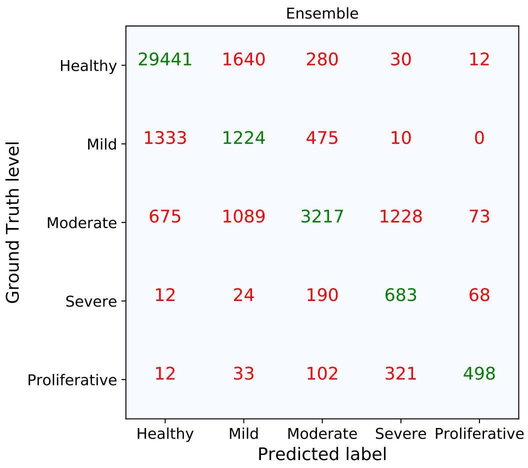
As we are minimizing the MSE loss between the output and ground truth label, the output of our neural network is a decimal value between 0 and 4 (e.g., $y_{pred} = 3.67$). In order to obtain the categorical predictions which are used for calculating the QWK score, we need to threshold the output. The threshold values are given by $[\eta_0 = 0, \eta_1, \eta_2, \eta_3, \eta_4, \eta_5 = 4]$. For any $\eta_i < y_{pred} < \eta_{i+1}$, we interpret y_{pred} to be class i. The optimal threshold values are searched using the gradient-free Powell's method [21] on the EyePACS validation set and then applied on the EyePACS test set to obtain the test QWK score.

III. Results

In this section, we presents the results. Section III-A presents the performance on the EyePACS dataset. Section III-B presents the performance on the Messidor dataset.

A. Performance on the EyePACS Dataset

We use the QWK score to evaluate classification performance and MAdds and number of learnable parameters (Parameters) to measure model complexity. The QWK score, MAdds and Parameters of the proposed system and those of some state-of-the-art models are shown in Table VI. The Zoom-In-Net [7] method achieves the highest QWK score of 0.854 on the EyePACS test set. Unfortunately, we are not able to estimate the number of parameters or MAdds of the Zoom-In-Net based on the information provided in [7]. Therefore we selected Method [16] which achieved 2nd best QWK score on EyePACS test set as the benchmark method to compare performance. It can be seen that the proposed Model Ensemble (2+1) achieves a QWK score of 0.852 compared to a QWK score of 0.849 achieved by benchmark method [16] while using 32% fewer parameters and 73% fewer MAdds. Fig 4 shows the confusion matrix of our best result in order to allow for further comparisons using other evaluation metrics other than QWK score. The results indicate that the proposed system is effective and efficient compared to the benchmark method.



	Healthy	Mild	Moderate	Severe	Proliferative
Healthy	29441	1640	280	30	12
Mild	1333	1224	475	10	0
Moderate	675	1089	3217	1228	73
Severe	12	24	190	683	68
Proliferative	12	33	102	321	498

Fig. 4. Confusion matrix of Model Ensemble on EyePACS test set.

B. Performance on the Messidor Dataset

In order to show the generalization ability of the proposed system, we evaluated the performance of the proposed automatic DR classification system on another widely used independent DR classification dataset, namely the Messidor dataset [22]. The Messidor dataset consists of 1200 retinal images. A retinopathy grade is provided by an ophthalmologist for each retinal image and the grade scales from 0 to 3, representing normal, mild, moderate, and severe DR respectively.

TABLE VI
Performance Comparison on EyePACS test set

Method	**Test QWK**	**Params** [1]	**MAdds** [2]
Method [16]	0.849	11.8M	45.0B
Zoom-In-Net [7]	**0.854**	-	-
MobileNet-Dense	0.825	1.8M	3.7B
MobileNetV2	0.822	4.2M	4.6B
Model Ensemble (1+1) [3]	0.851	6.2M	8.3B
Model Ensemble (2+1) [3]	**0.852**	**8.0M**	**12.0B**
Model Ensemble (1+2) [3]	**0.852**	10.4M	12.9B
Model Ensemble (2+2) [3]	**0.852**	12.3M	16.6B

[1] Parameters is in Millions.
[2] MAdds is in Billions for predicting one image.
[3] Model Ensemble (M+N) refers to the ensemble of Top-M saved checkpoints of MobileNet-Dense and Top-N saved checkpoints of MobileNetV2.

As there are only 1200 images in the Messidor dataset, which is not adequate to train a CNN model. Holly et al. suggested building classifiers using features extracted from the CNN models trained on other DR datasets such as EyePACS dataset [6]. As the DR grade in the Messidor dataset ranges from 0 to 3 while the DR grade in the EyePACS dataset ranges from 0 to 4 , we adopt the same scheme in [6], [7] and perform two binary classification tasks (i.e., Referable versus Non-Referable, Normal versus DR) to demonstrate the generalization ability of the proposed system.

We extract 256-dimensional feature vectors from the last fully connected layer of the MobileNet-Dense and MobileNetV2. Then a logistic regression classifier is trained for binary classification. For Referable/Non-Referable task, images with grade 0 and 1 in the Messidor dataset are considered as Non-Referable, while images with grade 2 and 3 are considered as Referable DR (RDR). 10-fold cross-validation on the entire Messidor dataset is performed as in [6], [7]. For Normal/DR classification task, images with grade 0 in the EyePACS/Messidor datasets are considered as normal, and images with other grades in EyePACS/Messidor dataset are considered as DR. The logistic regression classifier is trained using features extracted from the EyePACS training set and tested on the entire Messidor dataset.

The AUC and Acc scores are used to evaluate the performance. Table VII and Table VIII shows the performance and model complexity of our system and existing methods. It can be seen that both MobileNet-Dense and MobileNetV2 outperform the state-of-the-art SI2DRNet-v1 model in Acc and AUC scores on these two binary classification tasks with fewer MAdds and Parameters. Ensemble of these two models provides a very small performance improvement at the cost of higher MAdds and Parameters. Considering the trade-off between performance and model complexity, we select MobileNet-Dense to compare with the SI2DRNet-v1 model. It can be seen that MobileNet-Dense achieves better performance on these two tasks while using 83% fewer Parameters and 52% fewer MAdds.

IV. Conclusion and Future Work

In this paper, we proposed an effective DR automatic classification system based on computationally efficient CNN models. We evaluated the performance on two widely used DR classification datasets. Experimental results show that our system provides a strong classification performance with significantly smaller model size and significantly less computation complexity than state-of-the-art methods. Our system is more attractive in real clinical settings since it can be trained more efficiently and has less overhead when exporting updated models to clients.

In the future, we plan to deploy the proposed system to resource-constrained platforms such as smartphones or FPGA devices and assess its effectiveness in a clinical setting.

TABLE VII
Performance comparison on the Messidor dataset

Method	DR		RDR	
	AUC	Acc	AUC	Acc
VNXK [6]	0.870	0.871	0.887	0.893
CKML [6]	0.862	0.858	0.891	0.897
Human Expert A [23]	0.922	-	0.94	-
Human Expert B [23]	0.865	-	0.92	-
Zoom-In-Net [7]	0.921	0.905	0.957	0.911
SI2DRNet-v1 [24]	**0.959**	0.905	0.965	0.912
MobileNet-Dense	**0.959**	**0.908**	**0.967**	**0.922**
MobileNetV2	**0.959**	**0.908**	**0.970**	**0.923**
Model Ensemble (1+1)	**0.962**	**0.917**	**0.970**	**0.924**

TABLE VIII
Model Complexity Comparison

Method	Params	MAdds
SI2DRNet-v1 [24]	10.6M	7.7B
MobileNet-Dense	**1.8M**	**3.7B**
MobileNetV2	4.2M	4.6B
Model Ensemble (1+1)	6.2M	8.3B

Acknowledgment

We would like to thank EyePACS and Messidor program partners for providing the retinal image datasets.

This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada under Grant RGPIN 1731-2013, by the UBC Faculty of Applied Science, and by the National Research Foundation, Prime Minister's Office, Singapore under its IDM Futures Funding Initiative and administered by the Interactive and Digital Media Programme Office.

References

[1] Y. Zheng *et al.*, "The worldwide epidemic of diabetic retinopathy," *Indian Journal of Ophthalmology*, vol. 60, no. 5, pp. 428–431, 2012.

[2] "Facts about diabetic eye disease," https://nei.nih.gov/health/diabetic/retinopathy, accessed: 2019-03-20.

[3] Early Treatment Diabetic Retinopathy Study Research Group *et al.*, "Classification of diabetic retinopathy from fluorescein angiograms: ETDRS report number 11," *Ophthalmology*, vol. 98, no. 5, pp. 807–822, 1991.

[4] L. Crossland *et al.*, "Diabetic retinopathy screening and monitoring of early stage disease in australian general practice: tackling preventable blindness within a chronic care model," *Journal of diabetes research*, vol. 2016, 2016.

[5] H. Pratt *et al.*, "Convolutional neural networks for diabetic retinopathy," *Procedia Computer Science*, vol. 90, pp. 200–205, 2016.

[6] H. H. Vo and A. Verma, "New deep neural nets for fine-grained diabetic retinopathy recognition on hybrid color space," in *Proceeding of the IEEE International Symposium on Multimedia (ISM)*. IEEE, 2016, pp. 209–215.

[7] Z. Wang *et al.*, "Zoom-in-net: Deep mining lesions for diabetic retinopathy detection," in *Proceeding of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. Springer, 2017, pp. 267–275.

[8] M. Sandler *et al.*, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 4510–4520.

[9] J. Cohen, "Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit." *Psychological Bulletin*, vol. 70, no. 4, pp. 213–220, 1968.

[10] "Kaggle Diabetic Retinopathy Detection," https://www.kaggle.com/c/diabetic-retinopathy-detection/data, accessed: 2018-04-30.

[11] L. Taylor and G. Nitschke, "Improving deep learning using generic data augmentation," *arXiv preprint arXiv:1708.06020*, 2017.

[12] A. Krizhevsky *et al.*, "Imagenet classification with deep convolutional neural networks," in *Proceedings of the International Conference on Neural Information Processing Systems (NIPS)*. Curran Associates, Inc., 2012, pp. 1097–1105.

[13] S. Van der Walt *et al.*, "scikit-image: Image processing in Python," *PeerJ*, vol. 2, p. e453, 2014.

[14] D. Zhang *et al.*, "Diabetic retinopathy classification using deeply supervised ResNet," in *Proceedings of IEEE SmartWorld, Ubiquitous Intelligence and Computing, Advanced and Trusted Computed, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, 2017.

[15] A. Mathis and B. Stephan, "Competition report of TeamoO," https://www.kaggle.com/c/diabetic-retinopathy-detection/discussion/15617, 2015, online; accessed 30 January 2019.

[16] B. Graham, "Competition report of Minpooling," https://www.kaggle.com/c/diabetic-retinopathy-detection/discussion/15801, 2015, online; accessed 30 January 2019.

[17] A. G. Howard *et al.*, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[19] G. Pleiss *et al.*, "Memory-efficient implementation of Densenets," *arXiv preprint arXiv:1707.06990*, 2017.

[20] C. Zhang and Y. Ma, *Ensemble machine learning: methods and applications*. Springer, 2012.

[21] M. J. Powell, "An efficient method for finding the minimum of a function of several variables without calculating derivatives," *The Computer Journal*, vol. 7, no. 2, pp. 155–162, 1964.

[22] E. Decenciere *et al.*, "Feedback on a publicly distributed image database: the Messidor database," *Image Analysis and Stereology*, vol. 33, no. 3, pp. 231–234, 2014.

[23] C. I. Sánchez *et al.*, "Evaluation of a computer-aided diagnosis system for diabetic retinopathy screening on public data," *Investigative Ophthalmology and Visual Science*, vol. 52, no. 7, pp. 4866–4871, 2011.

[24] Y.-W. Chen *et al.*, "Diabetic retinopathy detection based on deep convolutional neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1030–1034.