Artificial Intelligence In Medicine 121 (2021) 102176

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

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

Classification of diabetic retinopathy using unlabeled data and knowledge distillation

Sajjad Abbasi [a], Mohsen Hajabdollahi [a], Pejman Khadivi [b], Nader Karimi [a], Roshanak Roshandel [b], Shahram Shirani [c], Shadrokh Samavi [a,c,*]

[a] Department of Electrical and Computer Engineering, Isfahan University of Technology, 84156-8311, Iran
[b] Computer Science Department, Seattle University, Seattle 98122, USA
[c] Department of Electrical and Computer Engineering, McMaster University, L8S 4L8, Canada

ARTICLE INFO

Keywords:
Convolutional neural networks (CNN)
Transfer learning
Knowledge distillation
Teacher-student model
Unlabeled data
Diabetic retinopathy

ABSTRACT

Over the last decade, advances in Machine Learning and Artificial Intelligence have highlighted their potential as a diagnostic tool in the healthcare domain. Despite the widespread availability of medical images, their usefulness is severely hampered by a lack of access to labeled data. For example, while Convolutional Neural Networks (CNNs) have emerged as an essential analytical tool in image processing, their impact is curtailed by training limitations due to insufficient labeled data availability. Transfer Learning enables models developed for one task to be reused for a second task. Knowledge distillation enables transferring knowledge from a pre-trained model to another. However, it suffers from limitations, and the two models' constraints need to be architecturally similar. Knowledge distillation addresses some of the shortcomings of transfer learning by generalizing a complex model to a lighter model. However, some parts of the knowledge may not be distilled by knowledge distillation sufficiently. In this paper, a novel knowledge distillation approach using transfer learning is proposed. The proposed approach transfers the complete knowledge of a model to a new smaller one. Unlabeled data are used in an unsupervised manner to transfer the new smaller model's maximum amount of knowledge. The proposed method can be beneficial in medical image analysis, where labeled data are typically scarce. The proposed approach is evaluated in classifying images for diagnosing Diabetic Retinopathy on two publicly available datasets, including Messidor and EyePACS. Simulation results demonstrate that the approach effectively transfers knowledge from a complex model to a lighter one. Furthermore, experimental results illustrate that different small models' performance is improved significantly using unlabeled data and knowledge distillation.

1. Introduction

Convolutional neural networks (CNNs) are widely used in medical image processing due to their strength in feature extraction and classification [1–5]. CNNs require a large number of labeled training data to be effective. In medical image processing, access to labeled datasets is limited due to privacy and regulatory constraints.

Transfer Learning (TL) approaches rely on knowledge obtained from solving one problem, to solve another problem. Hence, a pre-trained model's model parameters can be transferred to a new model where extensive training data may be lacking [6–8]. In TL, the two models must have a similar structure and architecture, restricting the use of a predefined model.

Knowledge Distillation (KD) was introduced in 2015 to transfer knowledge of a model to another [9]. KD can be used between models with different structures, addressing a significant shortcoming of Transfer Learning. Specifically, knowledge from a complex model (the *teacher*) is transferred to a simpler model (the *student*) by soft labels [10].

Knowledge distillation has exciting applications in expanding the training capabilities of a model. However, more investigation is needed to compare Knowledge Distillation and Transfer Learning. An important question is whether it is possible to use KD as an alternative to the TL. To answer this question, we investigate the application of KD as an alternative for the TL. Our study aims to design a method that has two advantages: (1) an appropriate knowledge transfer technique from a base network to another network (network under transfer), and (2) Designing

* Corresponding author at: Department of Electrical and Computer Engineering, McMaster University, L8S 4L8, Canada.
E-mail address: samavi@mcmaster.ca (S. Samavi).

https://doi.org/10.1016/j.artmed.2021.102176
Received 10 November 2020; Received in revised form 11 September 2021; Accepted 13 September 2021
Available online 17 September 2021

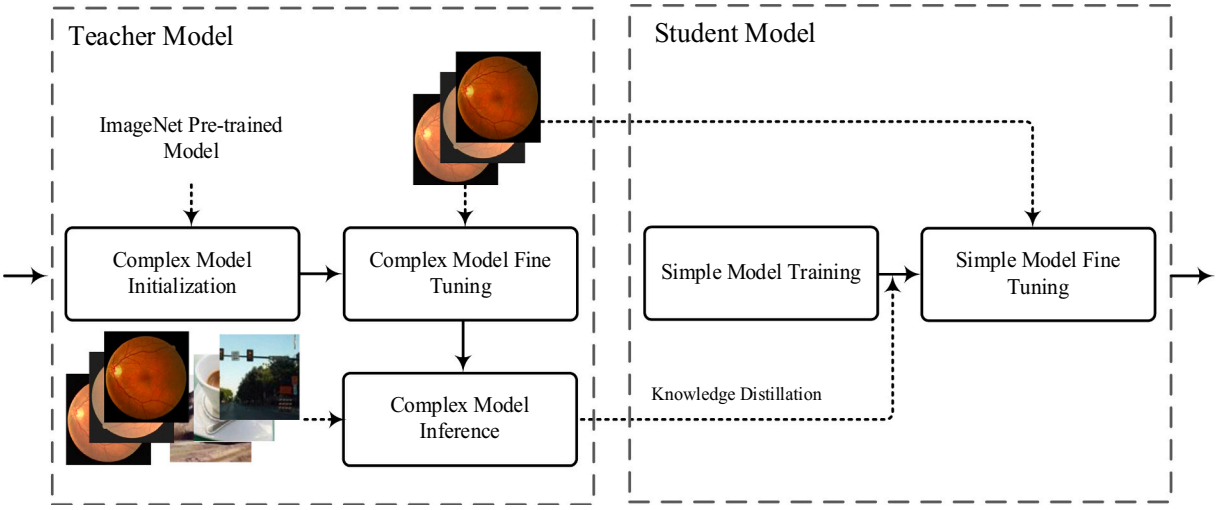

Fig. 1. Block diagram of the proposed method.

the model under transfer with an arbitrary structure. To extract most of the teacher model's knowledge, we also employ unlabeled data during knowledge distillation.

It is possible to create a simple model using the proposed method, which utilizes enough transferred knowledge. To the best of our knowledge, the proposed method is the first study that aims to compensate for the effects of labeled data deficiency in small models used for medical image processing. This method has exciting applications for developing low complexity models that will be implemented in embedded medical imaging devices with low resource budgets. We evaluated our approach using a comprehensive set of experiments to classify diabetic retinopathy (DR) images. DR classification is an application area where sufficient training data does not exist, and our approach could offer significant advances in this domain.

The main contributions of this study can be summarized as follows. First, we proposed a novel approach to use knowledge distillation to transfer knowledge of a complex model, which has many learned parameters, into a simple model. Secondly, we use unlabeled data to transfer enough knowledge to a simple model without extra training. Thirdly, we demonstrate our method's effectiveness by designing a simple and efficient network to analyze Diabetic Retinopathy that can be embedded in medical imaging devices.

The remainder of this paper is structured as follows. First, previous studies in DR classification are briefly described in Section 2. Then, in Section 3, the proposed method for knowledge transfer using an unlabeled dataset is explained. Next, in Section 4, experimental results are presented. Finally, Section 5 is dedicated to the conclusion of this study.

2. Diabetic retinopathy classification

Diabetes is a common disease that could harm the micro-vessels in the human eye retina [11–15]. The advanced stage of this disease can lead to diabetic retinopathy (DR), which is considered a prevalent cause of vision loss. Regular retinal monitoring by an expert can be used to prevent vision loss, which is difficult due to its cost and lack of expert accessibility [16].

Automatic screening and analysis of the retina can be considered as a solution to this problem. The processing of retinal images is conducted based on different methods and techniques. Some examples of methods used classification of retinal images include support vector machine (SVM) [17,18], K-nearest neighbors (KNN) [19,20], and random forest [21]. Among different methods for automatic screening of the retina, the use of CNNs is probably the best approach. CNNs can employ high-level features to map input images to the output. In this regard, in [22], DR detection is realized by semantic segmentation of microaneurysms using a CNN. In [23], red lesions are localized by using CNN working on image patches. After applying image processing methods such as image enhancement, DR is analyzed with a CNN structure. Designing and employing new structures of CNNs dedicated for DR detection could be very useful for better analysis of DR. So, in [24], a densely connected model as a robust structure is designed for better DR classification. In [25], the Inception Res-Net structure is modified, and a pre-trained model is employed for DR analysis. Considering the lesion area has an essential role for better DR detection. Al-Antary in [26] developed a model that extracts features from the different levels with an attention mechanism to consider lesion areas better. In [27], a robust model based on deep residual CNN is developed, which is pre-trained. In [28], the DR classification problem is investigated as a semantic segmentation problem to take the DR lesions into account. So, a model based on Faster Region-based CNN (RCNN) for DR lesion identification is proposed.

From the perspective of model complexity, different networks are proposed in the literature. For example, in [29–31], multiple network structures are utilized parallel or sequentially. Each network could have a part of the image as its input. In [2,32,33], VGG based networks are proposed for DR classification. DR detection requires a structure with a strong feature extracting ability; hence, in [2,33], VGG network parameters are enhanced using transfer learning from a VGG model pre-trained on Image-Net dataset [34,35]. Since pre-trained structures are available in a VGG network, [2,33] were obliged to use a structure such as VGG. By reviewing different CNN structures used for DR analysis, it can be seen that slightly complex networks are employed in many studies. Moreover, we can say that there is no framework for designing a simple structure that can be enriched by the knowledge of complex models.

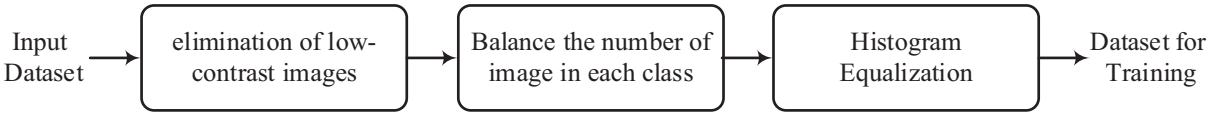

Fig. 2. Image preprocessing.

3. Proposed method

The proposed method is based on three techniques, including transfer learning, knowledge distillation, and employing unlabeled data. In Fig. 1, the proposed method's structure is illustrated, which contains two main parts, including teacher modeling and student modeling. A teacher model is a temporal model used to train the final student model, and at the inference time, only the student model is active. In the following, the proposed method is explained in detail.

3.1. Preprocessing

The process of data preprocessing is provided in Fig. 2. As illustrated in Fig. 2, preprocessing has three main stages: low contrast image elimination, dataset balancing, and histogram equalization. Datasets used for DR analysis have many images captured in different conditions. Hence, there can be some significant problems that have unfavorable influences on model training. One of the most important ones is the existence of several inappropriate and imperfect images among the dataset. There might be a lack of sufficient contrast for a vast number of samples. Furthermore, images of a dataset may not be balanced between all the existed classes. One solution is to eliminate low-contrast images from the training process to address the mentioned problems. Moreover, the same number of images from different classes can be used for model training to yield better balancing.

The elimination process is conducted for the image's standard deviation, where the images are transformed into the grayscale mode. Our experiences have observed that overall standard deviation can be used to demonstrate the contrast of retinal image samples such as the EyePACS dataset [36]. After transforming images into the grayscale mode, the standard deviation is calculated for each image, as illustrated in Eq. (1):

$$StdDev_{IMG} = \sqrt{\frac{\sum_{i=1}^{N}(pixel_i - \mu)^2}{N}} \qquad \mu = \frac{\sum_{i=1}^{N} pixel_i}{N} \tag{1}$$

In which N is the number of pixels included in image IMG.

Before training the network structure, the application of preprocessing and augmentation can be useful for better training. For preprocessing, the same method as performed in [2] is utilized. Histogram equalization of the retinal images increases the contrast of the vessels, especially micro-vessels, and better represents abnormal regions for DR classification. For preprocessing, local histogram equalization is performed separately on each input channel. Then, row-wise and column-wise flipping is used as an augmentation method to increase our training dataset.

3.2. Teacher model

In the teacher modeling stage, a complex model is trained for the DR classification. For training the teacher model, a VGG structure is considered as the teacher. VGG is selected because its pre-trained version on ImageNet is available. At first, a pre-trained VGG model, trained on ImageNet, initializes the teacher model. After that, the target augmented dataset is fed to the teacher model, and the teacher model is trained on the target dataset. In this way, a network with general feature extraction capability specialized on the target dataset is resulted. In this stage, the network structure is ready for knowledge distillation. Thanks to the distillation process, it is possible to train a model in which its structure is different from the teacher model. We need to determine whether it is possible to transfer all the knowledge of a teacher to a student through distillation.

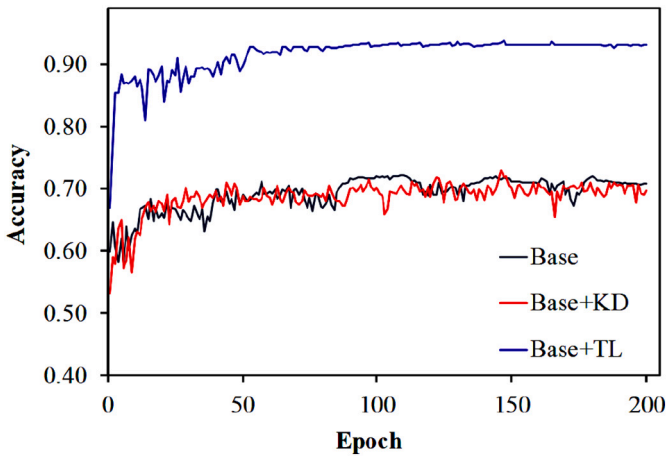

Fig. 3. Comparison of different training VGG models for DR classification. VGG: base, VGG with transfer learning: base + TL, VGG with knowledge distillation: base + KD.

A simple and intuitive experiment is performed to answer the above ambiguity. This experiment considers a VGG network as the teacher and another VGG network as a student model. The VGG teacher model is pre-trained on the ImageNet dataset. The teacher is fine-tuned using a retina dataset aiming to classify them for DR levels. For the student model also a VGG is considered without any pretraining. Then it is trained using the distilled knowledge from the teacher model.

Moreover, for better comparison, a VGG model is trained directly on the same retina dataset for the classification of DR levels. The results of these models are illustrated in Fig. 3. As illustrated in Fig. 3, Base refers to a VGG model which is only trained on the DR dataset. Base+TL refers to the teacher which is a VGG model pre-trained on the ImageNet dataset, and after that, it is trained on the DR dataset. Also, Base+KD refers to a student VGG model which is trained based on knowledge distillation. It can be observed that the VGG model with distillation has slightly better accuracy than the VGG model without any distillation. Indeed, the student model has a lower accuracy far from the teacher's accuracy.

This observation implies that all of the knowledge of a network may not be transferred through distillation. Explicitly, it can be stated that the knowledge transferred to the teacher model is not transferred to the student model through distillation. A simple model's accuracy can be improved using knowledge distillation [37,38]. Also, knowledge distillation works better in conditions with limited training data, as stated in [39,40]. With limited training data, knowledge is not transferred even to a model with the same size as the teacher model. During the knowledge distillation, the student only observes the data in the target dataset and is trained based on the corresponding teacher soft labels. The knowledge related to a pre-trained model and the transfer-learning knowledge can be extracted by observing the images associated with that pre-trained model.

Now let us look at the teacher model, which is ready for the transfer-learning. In the proposed method, better knowledge transfer is performed using the retina images' labels and random natural images labels. The teacher model set labels on random images that are unlabeled at first. In this way, the student model extracts the knowledge of the teacher model from other images. As illustrated in Fig. 1, after fine-tuning the teacher model, knowledge for distillation is provided from both the DR and random images.

3.3. Student model

In the proposed method, we will train a simple model as a student such that maximum information from a teacher model could be utilized. Indeed, training a simple model directly by a massive dataset such as

Algorithm1

Inputs:
D_{RT}: Raw Target Dataset
S_R: A Set of Random Unlabeled Images
S_W: A Set of Weights correspond to Pre-Trained Model on Big Dataset
Definitions:
C_N: Complex model conforming to S_W
S_N: Simple model
D_R: Labeled Dataset ← EMPTY
D_{PT}: Preprocessed Target Dataset ← EMPTY
T_C: Threshold for contrast
Module1: Preprocessing *(D_{RT},T_C)*
D_{PT} ← EMPTY
For *all images, I, in* D_{RT}
If *contrast of I >* T_C:
apply augmentation on I
add I to D_{PT}
End If
End for
Return D_{PT}
Module2: Knowledge Distillation *(Network,* S_R*)*
D_R ← EMPTY
For *all images, I, in* S_R
set Network prediction as the label of I
add I to D_R
End for
Return D_R
1: ***Start***
2: D_{PT} ← *Module1(*D_{RT},T_C*)*
3: Initialize C_N *with* S_W
4: Fine-tune C_N *using* D_{PT}
5: D_R ← *Module2(*C_N, S_R*)*
6: Train S_N *using* D_R
7: Fine-tune S_N *using* D_{PT}
8: ***End***

Fig. 4. Pseudo-code of the proposed method.

ImageNet is a formidable task. Therefore, using unlabeled data could be an alternative to training on a large data set.

In this regard, the student model is selected as a simple model that employs the teacher's knowledge as much as possible. As illustrated in Fig. 1, after training the teacher, the student is trained in two steps. The student follows the same training trend as conducted in the teacher model. In the first step, the student is trained based on the random images' knowledge using knowledge distillation, which simulates the teacher model's transfer learning. After that, the main images are used to fine-tune the student model using the teacher model's soft labels. This stage also simulates the fine-tuning of the teacher model on the main images. Next, we train the student by the main images. Also, the student is trained using the knowledge that is embedded in the teacher model. Finally, the student can be fine-tuned again using hard labels of the main images.

3.4. Proposed methods in the form of pseudo-code

In Fig. 4, the proposed method is represented using pseudo-code. As illustrated in Fig. 4, the procedure is defined clearly in eight consecutive lines. The two modules which are utilized in the algorithm are the Preprocessing and the Knowledge Distillation modules. The pre-processing module takes raw images from the target dataset as input. After selecting well-contrasted images and applying augmentation, contrast enhancement, and brightness improvement, this module produces the final dataset. The final dataset is appropriate for the training of the networks. The complex model is loaded with a pre-trained model and fine-tuned on the final dataset. The knowledge distillation module takes random unlabeled images and the complex model as inputs. After feeding each unlabeled image to the network, the module assigns the network prediction (known as soft label or logit) to the image's label. Eventually, label-assigned images are generated in the form of a dataset. After making a dataset from the unlabeled images, the simple model is trained and fine-tuned on the final dataset.

3.5. Knowledge distillation formulation

Method of knowledge distillation has a vital role in the teacher's proper transfer of knowledge to the student. In [41], a teacher-student model with a conditional method is implemented, where the teacher's predictions are compared with the original labels. If the prediction is correct, then soft labels are used for distillation; otherwise, hard labels are used for that purpose. In the proposed method, we use conditional distillation. Suppose that we have a teacher T with parameters w_T and a student model S with parameters w_S. A set of training sample $D = \{d_1, d_2, \ldots, d_N\}$, and corresponding labels $L = \{l_1, l_2, \ldots, l_N\}$ with ($l_i \in \mathbb{R}^{|C|}$) on DR classification as a target dataset is considered, and C is the set of all possible classes of l_i. Also, a set of random images $R = \{r_1, r_2, \ldots, r_M\}$ without any labels are considered. Two losses can be defined based on Kullback-Leibler (KL) divergence [41]. In KL divergence, in cases where

students attempt to approximate the teacher's predictions, the teacher's parameters are fixed. Accordingly, the first loss is due to the unlabeled data, which is formulated in the following equation:

$$L(w_S)_1 = \frac{-1}{M}\sum_{i=1}^{M}\sum_{j=1}^{C} p(r_i : j|T : w_T) \times log(p(r_i : j|S : w_S)) \quad (2)$$

In Eq. (2), $T : w_T$ and $S : w_S$ represent the teacher network, including parameters of w_T and student network, including parameters of w_S respectively. The symbol $r_i : j$ stands for consideration of label j for image r_i. Accordingly, $p(r_i : j|T : w_T)$ stands for the probability of label j for image r_i predicted by network T, and in the same way for $p(r_i : j|S : w_S)$. The second loss also can be defined due to the labeled data, which is conditional as Eq. (3):

$$L(w_S)_2 = \frac{-1}{N}\sum_{i=1}^{N}\left[\Delta\left(\underset{c \in C}{\text{Argmax}}\,(p(d_i : c|T : w_T)) == l_i\right) \times \left(\sum_{j=1}^{|C|} p(d_i : j|T : w_T)\right) \times \log(p(d_i : j|S : w_S)) + \Delta\left(\underset{c \in C}{\text{Argmax}}\,(p(d_i : c|T : w_T)) \neq l_i\right) \times \log(p(d_i : l_i|S : w_S))\right] \quad (3)$$

The first term of summation indicates the loss due to the samples in which the teacher correctly predicts their labels. The second term indicates the loss of the samples, which the teacher does not correctly predict. In Eq. (3), $\Delta(x)$ is an indicator function, 1 when x is true and 0 when x is false. During training the student model, at first, the student is trained based on the $L(w_S)_1$ to improve the model training capability. At second, the student is fine-tuned using $L(w_S)_2$.

4. Experimental results

Experimental results are conducted in the case of DR classification in retina images. All of the models for DR detection are implemented by Python using the TensorFlow framework. A computer with an Nvidia GPU1080 Ti and 11GB internal memory is used to implement the proposed method to train and test different models. For both the teacher and student models, similar hyper parameters are selected. All the models are designed and trained using the TensorFlow framework. Dropout is used with the value of 25% to prevent overfitting; Relu is used as the activation function and Softmax is applied in order to compute the probabilities and do classification at the final step. Standard batch normalization is set after each layer of the convolutional neural networks. Cross-Entropy is defined as the loss function of the training process, and AdamOptimizer with an initial learning rate of 0.001 and the default decaying rate is employed as the optimization function. The training batch size is set to 8, and all of the models are trained for 100 epochs.

4.1. Datasets

Two datasets, including Messidor and EyePACS, are used for our experiments [36,42]. In the Messidor dataset, there are 1200 RGB images, which we resized to 300 × 300. Since the more training samples, the more model generality, after enhancement, by augmentation, we increased the number of images to 4800.

The EyePACS dataset contains about 35,000 images of different sizes. Some images in EyePACS have a dark area around their borders, which could be harmful to model training. The dark area of these images is cropped, and all of them are resized to 300 × 300. Cropping and resizing can be useful for having fast training with lower resource consumption. Contrast enhancement and removing images with contrasts lower than a threshold from the training process would yield a better performance model. A vast number of images with visually sufficient contrasts are selected to determine the threshold. The average standard deviation of all images is set as the threshold. In the EyePACS dataset, images in which their standard deviation is less than the threshold are eliminated from the dataset. Hence, 35,126 images of the dataset decrease to 25,231, of which 5143 and 20,088 images have labels 1 and 0, respectively. Also, applying a balance between the numbers of different classes is essential to balance the number of image labels seen by the model. For the unlabeled data, a set of natural images are randomly selected from the internet containing 20,000 images resized to 300 × 300. Images of the unlabeled set are fed to the teacher model, as illustrated in Algorithm 1, to set a label to them.

4.2. Evaluation metrics

DR classification accuracy, area under the curve of ROC (receiver operating characteristic curve), and MCC (Matthews Correlation Coefficient) are used to evaluate different structures' performance. MCC and accuracy are used as Eqs. (4)–(5), in which TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative, respectively. A Five-fold cross-validation method is used to have a comprehensive validation. We follow the same definition of DR grading levels for classification, as used in [2,33].

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

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (5)$$

4.3. Models

For the teacher model, a VGG model was employed. However, this model is not able to yield acceptable results on its own. This problem can be due to the lack of data for the training and weak feature extraction capabilities that by using only DR data occurred. In [2,33], transfer learning is used to improve their results. By employing the transfer learning technique, it is possible to provide a better learning capability. To this aim, VGG network parameters that are pre-trained on ImageNet are used to initialize the teacher parameters, and after that DR dataset is used for training. This VGG model has an appropriate detection performance, so it is used as a teacher for training different small student models. This VGG model is used as the teacher model for all of the following experiments.

For the student model, designing small network structures are under consideration. Small structures are different from the VGG network, making it impossible to use a pre-trained VGG model. Moreover, training small models directly on the ImageNet can be a very time-consuming process with a lot of hardware resources. In this

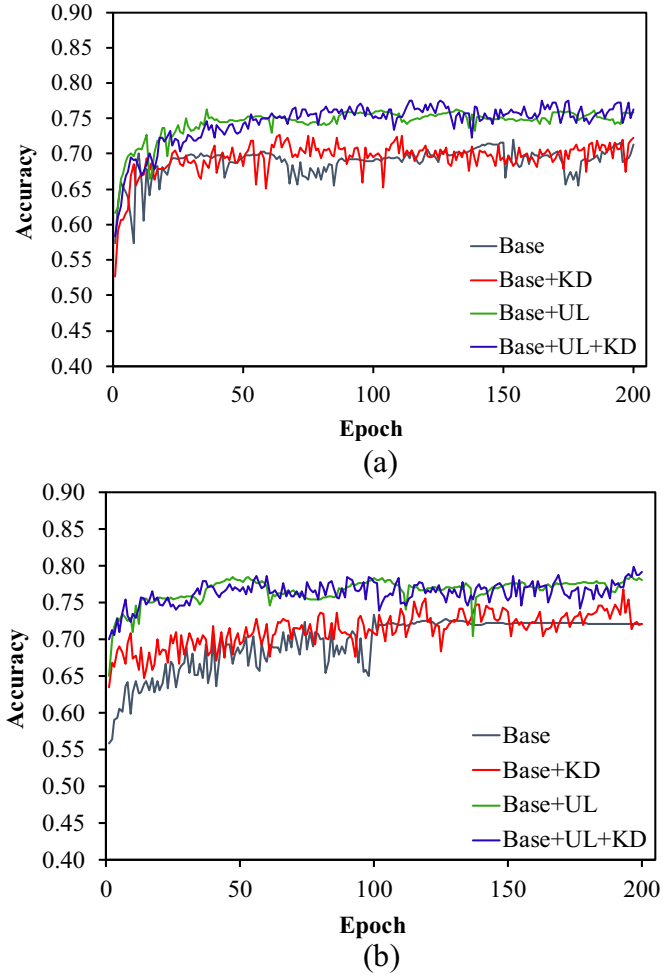

Fig. 5. Accuracies of different training methods for DR classification on Messidor dataset; (a) AlexNet-like, (b) VGG/4.

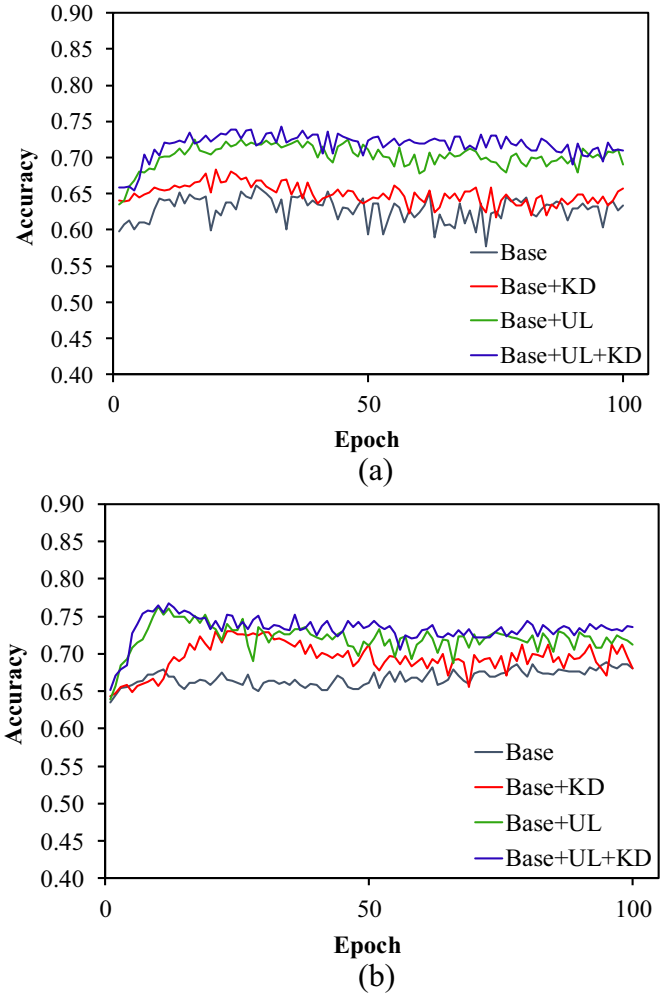

Fig. 6. Accuracies of different training methods for DR classification on EyePACS dataset; (a) AlexNet-like, (b) VGG/4.

experiment, small models are enriched using transfer learning, knowledge distillation, and using unlabeled data.

Two small versions of the VGG network with 16 layers are used, including VGG/4 and VGG/2, in which the number of their filters are divided by 4 and 2, respectively. Also, for better evaluation, a random and small structure with ten convolutional layers is utilized, which is called the "SimpleA" network. The SimpleA network has 20, 20, 30, 30, 40, 40, 160, 160, 250, 250 convolutional filters, in its layers. The conventional training on DR images is named "Base," in which only DR datasets are used for training without any initialization. In the "Base" training method, a model is trained based on DR images as input and the corresponding diabetic labels as the output. In the Base model training, no extra labels, data, and initialization is used. Learning using knowledge distillation is named "KD," in which the student model is trained using knowledge distillation by the VGG teacher model. Teacher and student models work in the inference and training phase, respectively. During training with the KD method, DR images are used as input for both teacher and student models. Output labels for the student training are provided as the soft labels, inference from the teacher model. Employing unlabeled data is called "UL," in which the student model is trained using unlabeled data. Teacher and student models work in the inference and training phase, respectively. During training with the UL method, natural images are used as input for both teacher and student models. Output labels for the student training are provided as the soft

Table 1
AUCs of different methods for DR detection on Messidor.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	78.60	80.18	78.52	77.36	76.96
Base model + KD	83.63	81.59	79.49	78.18	78.57
Base model + UL	85.17	87.09	83.34	82.10	**83.07**
Base model + UL + KD	**86.65**	**88.92**	**85.95**	**83.05**	82.46

Table 2
AUCs of different methods for DR detection on EyePACS.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	69.41	72.66	71.44	65.86	66.81
Base model + KD	75.25	77.95	78.75	67.48	69.25
Base model + UL	79.05	76.33	79.52	72.94	73.27
Base model + UL + KD	**79.51**	**78.96**	**80.71**	**73.78**	**76.04**

Table 3
Accuracies of different methods for DR detection on Messidor.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	73.37	73.58	72.11	71.84	71.95
Base model + KD	76.84	75.37	73.89	71.42	72.74
Base model + UL	78.63	79.58	76.95	76.55	76.21
Base model + UL + KD	**79.89**	**82.32**	**79.16**	**77.05**	**77.58**

Table 4
Accuracies of different methods for DR detection on EyePACS.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	68.85	71.32	70.50	66.68	67.4
Base model + KD	73.04	74.48	74.84	67.56	68.36
Base model + UL	76.40	74.88	75.84	71.44	72.48
Base model + UL + KD	**76.68**	**76.72**	**76.84**	**72.32**	**74.28**

labels, inference from the teacher model. In this way, maximum information is tried to be transferred from the teacher to the student. For better comparison, two structures, including LeNet-like and AlexNet-like, are selected from the [43] and [44], respectively. These models are trained, and corresponding results are reported.

4.4. Detection performance

In Figs. 5 and 6, the results of different training methods for DR classification are illustrated for Messidor and EyePACS datasets, respectively. Figs. 5(a) and 6(a) are related to the AlexNet-like network results. Figs. 5(b) and 6(b) are associated with the VGG/4 network. It can be observed that using unlabeled data have an essential effect on better training of simple networks.

We can assert that using only knowledge distillation in different models and datasets slightly improves network accuracy. Using unlabeled data leads to a suitable improvement of accuracy in all of the models and datasets. Simultaneously using knowledge distillation and unlabeled data, slightly better results are observed compared to using only unlabeled data. Figs. 5 and 6 demonstrate that using unlabeled data could improve the training capability of a model. It can be useful to note that there is a correlation between the teacher and the student in the teacher-student model. So, both the teacher and student models could be underfitted or overfitted. However, these problems are alleviated using the proposed method in two ways. First, a large number of unlabeled data are used, which prevents the student model from overfitting and underfitting. Second, overfitting occurs in the structures with a high depth which in the proposed method designing simple structures are under consideration.

To better compare different methods, three mentioned networks and those from [43,44] are trained for 150 epochs, and their detection performances are reported. In Tables 1 and 2, the results of the AUC for Messidor and EyePACS datasets are reported, respectively, where the best results are bolded. It is observed that for both of Messidor and EyePACS datasets, in all of the models, using unlabeled data causes a significant improvement in the AUC results. In Tables 3 and 4, results of detection accuracy for Messidor and EyePACS datasets are reported which similar results are observed. The improvements are also observed

Table 5
MCC of different methods for DR detection on Messidor.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	45.28	45.71	43.09	42.66	42.62
Base model + KD	52.62	49.62	46.37	42.59	44.12
Base model + UL	56.20	58.14	52.79	51.98	51.12
Base model + UL + KD	**58.79**	**63.79**	**57.26**	**53.17**	**53.98**

Table 6
MCC of different methods for DR detection on EyePACS.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	30.04	34.48	33.44	23.19	25.68
Base model + KD	39.62	43.69	43.46	25.40	28.37
Base model + UL	46.93	42.84	46.15	36.44	37.60
Base model + UL + KD	**47.31**	**47.46**	**48.11**	**37.85**	**41.98**

Table 7
Sensitivity of different methods for DR detection on Messidor.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	64.23	64.47	64.96	66.91	64.72
Base model + KD	71.04	69.58	67.15	72.26	65.45
Base model + UL	71.53	72.01	70.55	70.8	65.93
Base model + UL + KD	**72.5**	**75.42**	**71.28**	**72.74**	**68.61**

Table 8
Sensitivity of different methods for DR detection on EyePACS.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	47.88	45.66	49.33	37.77	42.11
Base model + KD	53.66	**60.44**	54.88	39.66	45.22
Base model + UL	**55.88**	45.66	**59.11**	**53.55**	49.11
Base model + UL + KD	50.77	53.22	58.11	52.44	**52.77**

Table 9
Specificity of different methods for DR detection on Messidor.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	80.33	80.51	75.88	75.69	77.55
Base model + KD	81.26	79.77	80.7	70.68	78.29
Base model + UL	84.04	85.34	81.81	80.33	84.04
Base model + UL + KD	**85.52**	**87.56**	**85.15**	**80.89**	**84.4**

Table 10
Specificity of different methods for DR detection on EyePACS.

Training methods	Networks				
	VGG/4	VGG/2	SimpleA	LeNet-like [43]	AlexNet-like [44]
Base model	80.68	85.68	82.31	83	81.62
Base model + KD	84.00	82.37	86.06	83.25	81.37
Base model + UL	87.93	**91.31**	85.25	81.53	85.62
Base model + UL + KD	**91.25**	89.93	**87.37**	**83.5**	**86.37**

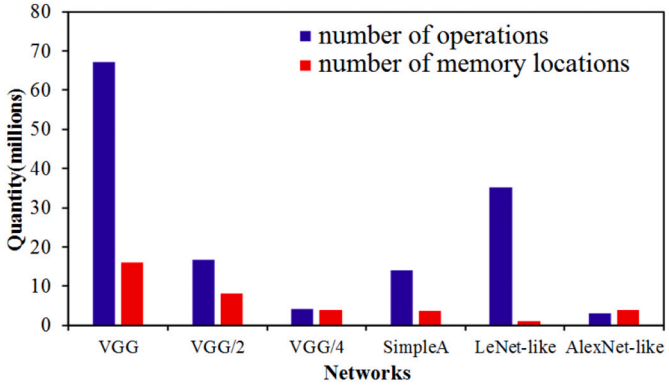

Fig. 7. Complexity of different models.

in MCC results for both employed datasets, as illustrated in Tables 5 and 6. For better comparison in Tables 7–10 sensitivity and specificity of DR detection for Messidor and EyePACS are reported. Generally, Tables 7–10 also show that the results of student models with knowledge distillation and transfer learning are improved.

From Tables 1–10, it can be concluded that using unlabeled data detection accuracy, MCC, and other parameters in all of the simple networks are improved. Furthermore, significant differences are observed between the performance of basic training and training using unlabeled data. Finally, we can say that knowledge of a network can be transferred efficiently from a teacher model to a student model by knowledge distillation and using unlabeled data.

In Table 8, the base model with unlabeled data and knowledge distillation has better sensitivity only in the Alexnet-like structure in the EyePACS dataset. Although in all of the structures, base models with unlabeled data and knowledge distillation have better DR detection metrics, the best sensitivity of DR detection is observed in the base model with unlabeled data. This means the detection of true cases of DR in EyePACS dataset is improved using unlabeled data while it is not improved by including knowledge distillation technique. From the knowledge perspective, it is concluded that all of the knowledge contained in a trained model could not be transferred to another model using knowledge distillation. Table 8 shows that the knowledge about complete detection of DR in the EyePACS dataset was not entirely transferred to some models. This problem is mainly due to the high complexity and variety of images in the EyePACS dataset. In such cases, unlabeled data should be employed to transfer the knowledge from a teacher model to a student. Finally, we can say that the knowledge transfer capability between teacher and student model could be improved using more and more unlabeled data.

4.5. Complexity analysis

Several works have investigated DR detection using deep neural network structures. However, they used models with a vast number of parameters requiring many hardware resources. Complexity analysis is conducted to have an insight into the number of model parameters and amount of memory requirement. Complexity analysis is done on the VGG model used in [32,33], AlexNet-like structure in [44], LeNet-like structure in [43], and our three designed simple networks. The results of this analysis for these structures are presented in Fig. 7. Complexity analysis is conducted based on two crucial perspectives of complexity that existed in the CNN processing. These perspectives include computational complexity as well as memory complexity. Computational complexity indicates the number of necessary computations for CNN processing related to the number of parameters.

From another perspective, the amount of memory required for storing the intermediate feature maps could be considered as the most important and challenging factor for CNN complexity [45,46]. The details of computing complexity for one of our designed models (SimpleA) with 300×300 input images are illustrated in Table 11. As illustrated in Table 11, the SimpleA structure has parameters and feature maps that constitute the computational and memory part of its complexity.

This study's main goal is designing models with low complexity structures with a high capability in knowledge transfer. We used simple models and granted them an appropriate knowledge transfer capability. In Fig. 7, the complexity of the models employed in this study is compared with the VGG model as the widely adopted DR detection model. It can be observed that the model complexities, including computational and memory, are significantly lower in the designed models. Although these models are simple with weak training capabilities, their detection performances are improved using the proposed method.

As previously mentioned, the corresponding detection performance for the models illustrated in Fig. 7 is reported in Table 1–10. It can be observed that there is not a direct relationship between the student's model complexity and detection performance. This is a common event in deep neural network training because some problems may be accrued during the training of different models, such as vanishing gradients which prevent them from reaching a better accuracy. In summary, a more complex model in deep neural networks does not necessarily result in higher accuracy. There is not any difference between the training of teacher and student models except that it is not possible to use all of the data employed to train the teacher to train the student model. This is mainly due to the differences between a heavy and large model such as

Table 12
The time consumed for training different models on Eye-PACS dataset.

Model	Training time (hours)
Base model (VGG)	10.56
Student model (VGG/2) + KD	2.47
Student model (VGG/2) + UL	4.62
Student model (VGG/2) + KD + UL	5.12
Student model (VGG/4) + KD	1.23
Student model (VGG/4) + UL	2.19
Student model (VGG/4) + KD + UL	2.37

Table 11
Structure of SimpleA model and details of computing complexity.

Layer type	Conv	Conv	Pooling	Conv	Conv	Pooling
Parameters	540	3600	–	5400	8100	–
Feature map	20×300^2	20×300^2	20×150^2	30×150^2	30×150^2	30×75^2
Memory	1,800,000	–	450,000	675,000	–	168,750
Layer type	Conv	Conv	Pooling	Conv	Conv	Pooling
Parameters	10,800	14,400	–	57,600	230,400	–
Feature map	40×75^2	40×75^2	40×38^2	160×38^2	160×38^2	160×19^2
Memory	225,000	–	57,760	231,040	–	57,760
Layer type	Conv	Conv	Pooling	Dense	Dense	–
Parameters	360,000	562,500	–	12,800,000	1024	–
Feature map	250×19^2	250×19^2	250×10^2	512	2	–
Memory	90,250	–	25,000	512	2	–

the teacher and a simple model such as the student.

Finally, the time consumed for training different models on the EyePacs dataset is illustrated in Table 12. Student models require less time for training than the teacher model. Also, the training with extra unlabeled data takes longer than other training processes for a student.

5. Conclusion

Considering the outcomes observed in the experimental results section, we see two possibilities that are available by the proposed method. These possibilities include 1) making small trainable models and 2) easing the training process. Small models have less training capability in comparison with the big models. Sometimes the training weakness of the simple models in an application leads us to not using them. Therefore, using small models can be challenging in applications demanding low complexity models with acceptable performance. It was observed from the results that although small models were used, appropriate knowledge from a large and different structure was transferred. The performance of the different small models was improved, and an acceptable DR detection was possible.

We see that this improvement was achieved with a little effort and with unlabeled data without any expensive annotating procedure. Moreover, unlabeled data are used for training small models to make them better generalize on the new tasks.

Furthermore, training on a big dataset was prevented in the proposed method. For training a DR detection model, it was required a pre-trained model on a large dataset such as ImageNet. Although a pre-trained model is used in the transfer learning, applying the transfer learning on a new model and structure is not easily possible and has a high cost. Using the proposed method, transfer learning on a new model does not require extra training, and using any pre-trained model is possible. In this way, different models can be used in transfer learning with high knowledge transfer capabilities.

The proposed method could be applied to different applications such as vessel detection in angiograms [47], image compression [48–51], and image fusion [52]. These are applications that may not have large training datasets, and the proposed method could be helpful.

In summary, we proposed a new method for knowledge transfer from a complex network to an arbitrary simple network. The proposed algorithm employed the soft labels of a random dataset produced by a complex model to extract all of the model information. This information was used to train a simple model that was not able to perform an appropriate classification. Experimental results for DR classification demonstrated that the proposed method for using unlabeled data by simple models improved their accuracy by an average of 6%.

In some medical applications, we need to use portable devices with limited resources. Our proposed approach can be used for knowledge transfer, where the platforms have constraints, the design has to be simple, and the training data is limited.

Declaration of competing interest

The authors declare that there is no conflict of interest.

References

[1] Hajabdollahi M, Esfandiarpoor R, Sabeti E, Karimi N, Soroushmehr SMR, Samavi S. Multiple abnormality detection for automatic medical image diagnosis using bifurcated convolutional neural network. Biomed Signal Process Control 2020;57:101792.

[2] Hajabdollahi M, Esfandiarpoor R, Najarian K, Karimi N, Samavi S, Soroushmehr SM Reza. Hierarchical pruning for simplification of convolutional neural networks in diabetic retinopathy classification. In: IEEE Annual International Conference of the Engineering in Medicine and Biology Society (EMBC); 2019. p. 970–3.

[3] Hajabdollahi M, Esfandiarpoor R, Najarian K, Karimi N, Samavi S, Reza-Soroushmeh SM. Low complexity convolutional neural network for vessel segmentation in portable retinal diagnostic devices. In: IEEE International Conference on Image Processing (ICIP); 2018. p. 2785–9.

[4] Nasr-Esfahani E, Rafiei S, Jafari MH, Karimi NR, Wrobel JS, Samavi S, et al. Dense pooling layers in fully convolutional network for skin lesion segmentation. Comput Med Imaging Graph 2019;vol(78):101658.

[5] Wang Z, Yang J. Diabetic retinopathy detection via deep convolutional networks for discriminative localization and visual explanation. arXiv preprint 2017. arXiv:1703.10757.

[6] Oquab M, Bottou L, Laptev I, Sivic J. Learning and transferring mid-level image representations using convolutional neural networks. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR); 2014. p. 1717–24.

[7] Donahue J, et al. DeCAF: a deep convolutional activation feature for generic visual recognition. In: International Conference on Machine Learning (ICML); 2014. p. 647–55.

[8] Lecun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.

[9] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. arXiv Prepr 2015. arXiv1503.02531.

[10] Abbasi S, Hajabdollahi M, Karimi N, Samavi S. Modeling teacher-student techniques in deep neural networks for knowledge distillation. arXiv Prepr 2019. arXiv1912.13179.

[11] Sahlsten J, et al. Deep learning fundus image analysis for diabetic retinopathy and macular edema grading. Sci Rep 2019;vol.9(1):1–11.

[12] Ting DSW, Cheung GCM, Wong TY. Diabetic retinopathy: global prevalence, major risk factors, screening practices and public health challenges: a review. Clin Experiment Ophthalmol 2016;vol.44(4):260–77.

[13] Hajabdollahi M, Karimi N, Soroushmehr SM Reza, Samavi S, Najarian K. Retinal blood vessel segmentation for macula detachment surgery monitoring instruments. Int J Circuit Theory Appl 2018;46(6):1166–80.

[14] Islam MM, Yang H-C, Poly TN, Jian W-S, Li Y-CJ. Deep learning algorithms for detection of diabetic retinopathy in retinal fundus photographs: a systematic review and meta-analysis. Comput Methods Programs Biomed 2020;191:105320.

[15] Chudzik P, Majumdar S, Calivá F, Al-Diri B, Hunter A. Microaneurysm detection using fully convolutional neural networks. Comput Methods Programs Biomed 2018;158:185–92.

[16] Stolte S, Fang R. A survey on medical image analysis in diabetic retinopathy. Med Image Anal 2020;64:101742.

[17] Adal KM, van Etten PG, Martinez JP, Rouwen KW, Vermeer KA, van Vliet LJ. An automated system for the detection and classification of retinal changes due to red lesions in longitudinal fundus images. IEEE Trans Biomed Eng 2017;65(6):1382–90.

[18] Xu J, et al. Automatic analysis of microaneurysms turnover to diagnose the progression of diabetic retinopathy. IEEE Access 2018;6:9632–42.

[19] Tang L, Niemeijer M, Reinhardt JM, Garvin MK, Abramoff MD. Splat feature classification with application to retinal hemorrhage detection in fundus images. IEEE Trans Med Imaging 2012;32(2):364–75.

[20] Niemeijer M, Abramoff MD, Ginneken B Van. Information fusion for diabetic retinopathy CAD in digital color fundus photographs. IEEE Trans Med Imaging 2009;28(5):775–85.

[21] Acharya UR, et al. Automated diabetic macular edema (DME) grading system using DWT, DCT features and maculopathy index. Comput Biol Med 2017;84:59–68.

[22] Qiao L, Zhu Y, Zhou H. Diabetic retinopathy detection using prognosis of microaneurysm and early diagnosis system for non-proliferative diabetic retinopathy based on deep learning algorithms. IEEE Access 2020;8. pp. 104292–104302.

[23] Zago GT, Andreão RV, Dorizzi B, Salles EOT. Diabetic retinopathy detection using red lesion localization and convolutional neural networks. Comput Biol Med 2020;116:103537.

[24] Riaz H, Park J, Choi H, Kim H, Kim J. Deep and densely connected networks for classification of diabetic retinopathy. Diagnostics 2020;10(1):24.

[25] Gangwar AK, Ravi V. Diabetic retinopathy detection using transfer learning and deep learning. In: Springer evolution in computational intelligence; 2021. p. 679–89.

[26] Al-Antary MT, Arafa Y. Multi-scale attention network for diabetic retinopathy classification. IEEE Access 2021;9:54190–200.

[27] Martinez-Murcia FJ, Ortiz A, Ramírez J, Górriz JM, Cruz R. Deep residual transfer learning for automatic diagnosis and grading of diabetic retinopathy. Neurocomputing 2021;452:424–34.

[28] Nazir T, Irtaza A, Rashid J, Nawaz M, Mehmood T. Diabetic retinopathy lesions detection using faster-RCNN from retinal images. In: IEEE International Conference of Smart Systems and Emerging Technologies-SMARTTECH; 2020, November. p. 38–42. 2020.

[29] Wang Z, Yin Y, Shi J, Fang W, Li H, Wang X. Zoom-in-net: deep mining lesions for diabetic retinopathy detection. In: Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics); 2017. p. 267–75.

[30] Gao Z, Li J, Guo J, Chen Y, Yi Z, Zhong J. Diagnosis of diabetic retinopathy using deep neural networks. IEEE Access 2019;7:3360–70.

[31] Vo HH, Verma A. New deep neural nets for fine-grained diabetic retinopathy recognition on hybrid color space. In: IEEE International Symposium on Multimedia (ISM); 2016. p. 209–15.

[32] Pratt H, Coenen F, Broadbent DM, Harding SP, Zheng Y. Convolutional neural networks for diabetic retinopathy. In: Procedia computer science; 2016. p. 200–5.

[33] Chen Y-W, Wu T-Y, Wong W-H, Lee C-Y. Diabetic retinopathy detection based on deep convolutional neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 2018. p. 1030–4.

[34] Krizhevsky A, Sutskever I, Geoffrey H. Imagenet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 2012:1097–105.

[35] Kandel I, Castelli M. Transfer learning with convolutional neural networks for diabetic retinopathy image classification. A review. Appl Sci 2020;10(6):2021.
[36] "Kaggle: Diabetic retinopathy detection." [Online]. Available:https://www.kaggle.com/c/diabetic-retinopathy-detection, Accessed: 2018-05-14.
[37] Mishra A, Marr D. Apprentice: using knowledge distillation techniques to improve low-precision network accuracy. In: 6th International Conference on Learning Representations, ICLR - Conference Track Proceedings; 2018.
[38] Polino A, Pascanu R, Alistarh D. Model compression via distillation and quantization. In: 6th International Conference on Learning Representations (ICLR) - Conference Track Proceedings; 2018.
[39] Zhu M, Han K, Zhang C, Lin J, Wang Y. Low-resolution visual recognition via deep feature distillation. In: IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings (ICASSP); 2019. p. 3762–6.
[40] Nayak GK, Mopuri KR, Shaj V, Babu RV, Chakraborty A. Zero-shot knowledge distillation in deep networks. arXiv Prepr 2019. arXiv1905.08114.
[41] Meng Z, Li J, Zhao Y, Gong Y. Conditional teacher-student learning. IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings (ICASSP). 2019. p. 6445–9.
[42] Decencière E, et al. Feedback on a publicly distributed image database: the Messidor database. Image Anal Stereol 2014;33(3):231–4.
[43] Chowdhury MS, Taimy FR, Sikder N, Nahid A-A. Diabetic retinopathy classification with a light convolutional neural network. In: International Conference on Comput, Commun, Chem, Mater and Elect Eng (IC4ME2); 2019. p. 1–4.
[44] Hacisoftaoglu RE, Karakaya M, Sallam AB. Deep learning frameworks for diabetic retinopathy detection with smartphone-based retinal imaging systems. Pattern Recognit Lett 2020;135:409–17.
[45] Sze V, Member S, Chen Y, Member S, Yang T. Efficient processing of deep neural networks: a tutorial and survey efficient processing of deep neural networks: a tutorial and survey. Proc IEEE 2017;105(12):2295–23292017.
[46] Hajabdollahi M, Esfandiarpoor R, Khadivi P, Soroushmehr SMR, Karimi N, Samavi S. Simplification of neural networks for skin lesion image segmentation using color channel pruning. Comput Med Imaging Graph 2020;82:101729.
[47] Fazlali HR, Karimi N, Soroushmehr SMR, Sinha S, Samavi S, Nallamothu B, et al. Vessel region detection in coronary X-ray angiograms. In: IEEE International Conference on Image Processing (ICIP); 2015. p. 1493–7.
[48] Karimi N, Samavi S, Soroushmehr SMR, Shirani S, Najarian K. Toward practical guideline for design of image compression algorithms for biomedical applications. Expert Syst. Appl. 2016;56:360–7.
[49] Nasr-Esfahani E, Samavi S, Karimi N, Shirani S. Near lossless image compression by local packing of histogram. In: IEEE International Conference on Acoustics, Speech and Signal Processing; 2008. p. 1197–200.
[50] A. Neekabadi, S. Samavi, S.A. Razavi, N. Karimi, S. Shirani, "Lossless microarray image compression using region-based predictors," IEEE International Conference on Image Processing, vol. 2, pp. II-349, 2007.
[51] Nejati M, Samavi S, Karimi N, Soroushmehr SMR, Najarian K. Boosted dictionary learning for image compression. IEEE Trans Image Process 2016;25(10):4900–15.
[52] Nejati M, Karimi N, Soroushmehr SMR, Karimi N, Samavi S, Najarian K. Fast exposure fusion using exposedness function. In: IEEE International Conference on Image Processing (ICIP); 2017. p. 2234–8.