Fine-tuning of pre-trained convolutional neural networks for diabetic retinopathy screening: a clinical study

Saboora M. Roshan* and Ali Karsaz

Khorasan Institute of Higher Education,
No. 77, Moalem Blvd., Mashhad, Iran
Email: saboora.m.roshan@gmail.com
Email: Karsaz@khorasan.ac.ir
*Corresponding author

Amir Hossein Vejdani

Navid-e-Didegan Clinic,
Between 7&9 Mollasadra St., Ahmad Abad Blvd.,
Mashhad, Iran
Email: vejdani@drvejdani.com

Yaser M. Roshan

University of British Columbia,
6190 Agronomy Road, Suite 301,
Vancouver, BC V6T 1Z3, Canada
Email: yroshan@mitacs.ca

Abstract: Diabetic retinopathy is a serious complication of diabetes, and if not controlled, may cause blindness. Automated screening of diabetic retinopathy helps physicians to diagnose and control the disease in early stages. In this paper, two case studies are proposed, each on a different dataset. Firstly, automatic screening of diabetic retinopathy utilising pre-trained convolutional neural networks was employed on the Kaggle dataset. The reason for using pre-trained networks is to save time and resources during training compared to fully training a convolutional neural network. The proposed networks were fine-tuned for the pre-processed dataset, and the selectable parameters of the fine-tuning approach were optimised. At the end, the performance of the fine-tuned network was evaluated using a clinical dataset comprising 101 images. The clinical dataset is completely independent from the fine-tuning dataset and is taken by a different device with different image quality and size.

Keywords: deep learning; convolutional neural network; diabetic retinopathy; inception model; clinical study.

Reference to this paper should be made as follows: Roshan, S.M., Karsaz, A., Vejdani, A.H. and Roshan, Y.M. (2020) 'Fine-tuning of pre-trained convolutional neural networks for diabetic retinopathy screening: a clinical study', *Int. J. Computational Science and Engineering*, Vol. 21, No. 4, pp.564–573.

Biographical notes: Saboora M. Roshan received her MSc degree in 2017 from Khorasan Institute of Higher Education, Mashhad, Iran and BSc degree in 2014 from Sadjad University of Technology, Mashhad, Iran. Her main interests are artificial intelligence and machine learning including deep learning, neural networks, and fuzzy systems; computer vision including object recognition and image processing. Currently, she is working as a Research Engineer on electrical industry projects at East Electrical Energy Economics Research Group, Mashhad, Iran.

Ali Karsaz is an Assistant Professor of Electrical Engineering at Khorasan Institute of Higher Education, Mashhad, Iran. He received his PhD and MSc degrees from Ferdowsi University of Mashhad, Iran and BSc degree from University of Poly-Technique, Tehran, Iran. His main interests contain bioinformatics, intelligent control, medical image processing, neural network, genetic algorithms, and nonlinear stochastic dynamic system prediction and estimation.

Amir Hossein Vejdani is an Ophthalmologist at Navid-e-Didegan Clinic, Mashhad, Iran. He performed his Fellowships in the field of cornea-external ocular disease, cataract and refractive surgery at Mashhad Medical University, Iran. He is working closely with R&D engineers to design automated eye-care applications.

Yaser M. Roshan is the Director in Business Development at Mitacs and is the Co-Founder of Ophthalight Digital Solutions Inc. aiming to help people keep their vision by making basic eye care more accessible while providing eye care professionals with advanced mobile health tools that will allow an increase in the frequency, speed, and profitability of eye exams. He received his PhD from the School of Mechatronic Systems Engineering, Simon Fraser University, Canada. He also has worked as a Lecturer at Electrical Engineering Department in Point Park University, USA.

This paper is a revised and expanded version of a paper entitled 'Comparative study of fine-tuning of pre-trained convolutional neural networks for diabetic retinopathy screening' presented at the 24th national and 2nd International Iranian Conference on Biomedical Engineering (ICBME), Amirkabir University of Technology, Tehran, Iran, 30 November to 1 December 2017.

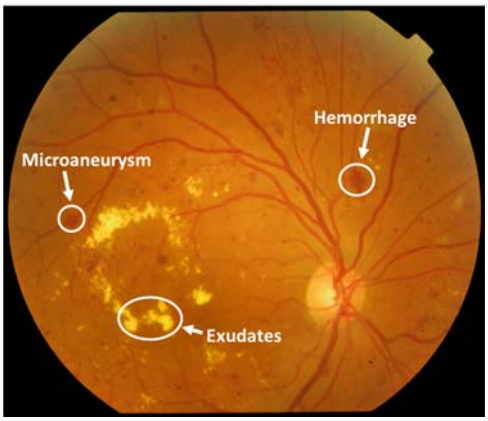
1 Introduction

One of the most important complications of diabetes which damages the small arteries and veins in the retina is diabetic retinopathy (DR). In the early stages, diabetic retinopathy may be asymptomatic, but eventually, it causes blindness if left untreated. In 2015, the number of patients diagnosed with DR and late-stage DR was 145 million and 45 million, respectively (Cho et al., 2017). Evidence shows that by diagnosing DR in early stages it can be treated just by diabetes management and can be prevented from further damages to the retina (Antal and Hajdu, 2012).

Generally, DR is diagnosed by an experienced ophthalmologist using a detailed and highly accurate retinal image (fundus image). Ophthalmologist diagnoses the severity of DR by carefully investigating fundus images and finding the different symptoms of DR, such as haemorrhages, exudates, micro-aneurysms, and neovascularisation. Figure 1 demonstrates an example of a patient's eye with signs of DR. Finding DR signs needs careful and frequent examination, which makes it difficult to diagnose in early stages. Also, in some countries with the shortage of ophthalmologists, trained professionals are not available for examining people's eyes periodically. Therefore, lots of DR cases remain undiagnosed in their early stages (Hani and Nugroho, 2010).

According to recent studies, DR can be diagnosed accurately in early stages by applying automatic diagnosis systems. The main purpose of these systems is to classify DR images from no DR images (Niemeijer et al., 2010). There are multiple approaches in the literature using various algorithms to implement automatic screening of DR. Typically, these methods design their automated diagnostic systems using hand-crafted features. Some of these methods are artificial neural network (ANN) and k-nearest neighbours (Bhatkar and Kharat, 2015; Osareh et al., 2009; Saranya et al., 2012). As an effort in comparing different available techniques, a study is implemented conventional machine learning algorithms using hand-crafted features in which linear support vector machine (SVM) with polynomial kernel of degree three is defined as a reliable classifier that can be used for DR diagnosis (Mohammadian et al., 2017a).

Figure 1 Example of a retinal photo with diabetic retinopathy (see online version for colours)



Generally speaking, feature extraction techniques are complex tasks and require depth knowledge of the images and their differences. Therefore, recent studies are using state-of-the-art neural network named convolutional neural network (CNN) for various fields such as finger vein recognition (Cheng et al., 2017), optimisation of speech recognition system (Weipeng et al., 2019), and especially in medical image analysis (Zhang et al., 2015).

One of the main reasons for implementing CNN in medical applications is that it is able to automatically extract features by using deep multiple layers (Tajbakhsh et al., 2016). Therefore, using CNN in medical diagnosis applications is increasing during recent years. For instance, CNNs were used for grading brain tumours in magnetic resonance imaging (MRI) scans (Pan et al., 2015), predicting the types of polyps during colonoscopy videos (Zhang et al., 2017), classifying interstitial lung disease (ILD) (Shin et al., 2016), and recognising cardiac MRI acquisition plane (Margeta et al., 2017). In another study, ensemble classification with CNN structures was used to extract features and segment retinal blood vessels (Wang et al., 2015) and a study related to severity DR diagnosis using CNN was addressed in Pratt et al. (2016). Also, two different comparative studies of two CNN algorithms to diagnose DR cases are performed in Vo and Verma (2016) and Mohammadian et al. (2017b).

Requiring a large amount of medical training dataset, complications about training of a deep CNN such as convergence and overfitting issues, and requiring large computing power and memory make the full training of a CNN from scratch impractical (Tajbakhsh et al., 2016). In literature, pre-trained CNNs have been fine-tuned as an alternative technique to fully training CNNs and have been used to modify the systems for other applications (Vo and Verma, 2016; Margeta et al., 2017; Tajbakhsh et al., 2016). There are different the state-of-the-art CNN architectures for detection and classification such as CifarNet (Krizhevsky, 2009), Alexnet (Krizhevsky et al., 2012) and GoogleNet (Szegedy et al., 2015a) with different model training parameter values. But GoogleNet model which uses concatenation process is more complex than both CifarNet and AlexNet (Shin et al., 2016). Therefore, in this study, a widely known pre-trained CNN architecture named *Inception-V3* (the latest module version of the GoogLeNet structure) was used for DR screening in Kaggle dataset. Inception-V3 has been previously trained for the application of ImageNet dataset classification and its weights are publicly available (Deep Learning Models, 2015).

CNNs consist of different sets of layers, the earlier layers are related to extract lower-level features from the images including edges and can be useful for various tasks. The other set of layers are trained to extract higher-level features that are more specific for each dataset. Therefore, for implementing DR screening system, the weights of the last layers of the CNNs are fine-tuned to adapt the networks for this very task.

The comparative study in this paper, introduces a better understanding of the effects of re-training various layers of a pre-trained network. Therefore, the images of Kaggle DR dataset, publicly available, were used to retrain the networks and to compare the accuracy of the proposed systems to the aim of choosing the best CNN in performance. (https://www.kaggle.com/c/diabetic-retinopathy-detection/data). Kaggle dataset has a large number of diabetic retinopathy images and is being used by researchers to develop disease diagnosis models (Wang and Jianbo, 2018). Also, in most of the similar literature, the trained networks are being tested on a subset of images in the same dataset. Although, the test dataset is chosen such that it does not overlap with the training dataset but most of the times the trained network is not easily applicable to other datasets. To this end, in this paper the fine-tuned networks were evaluated using a clinical dataset which is completely independent of the original training and testing dataset. The purpose of using the clinical dataset is not only to evaluate the proposed networks, but also to investigate the capability of this system to be employed in local clinics for automatic DR diagnosis. It is noteworthy that the methods used here for fine-tuning and data preparation have been discussed in our previous study and this paper continues our latest research in the field of DR screening and verification of the proposed approach using clinical data (Mohammadian et al., 2017b; https://goo.gl/a2dLbq).

Overall, the contributions of this paper are:

1 employing Inception-V3 as the most practical CNN architecture for DR diagnosis

2 comparing the effects of re-training various layers of Inception-V3

3 evaluating fine-tuned networks using clinical dataset.

The paper continues as follows. Section 2 briefly describes methods that are used for data preparation. Section 3 explains the methodology to compare CNN models. Section 4 introduces the clinical dataset that is utilised to test the proposed networks. The results and comparative analysis of the simulations are discussed in Section 5. Section 6 presents the proposed approach, while Section 7 concludes the paper.

2 Data preparation

2.1 Diabetic retinopathy dataset

In this paper, Kaggle diabetic retinopathy dataset was used to fine-tune the CNNs (Diabetic Retinopathy Detection, 2015). This dataset contains of 35,126 retina images, which have various qualities due to the different models of cameras that have been used for taking the fundus images. Each image is carefully investigated by a clinician for diabetic retinopathy and is rated as follows: no DR – 0; mild DR – 1; moderate DR – 2; severe DR – 3 and proliferative DR – 4. In this study, the aim is to classify the retina images into No DR images (with the label 0) and DR images (with labels 1–4). In addition, the proposed classification networks can be used for diagnosing the severity of the disease as well.

In the first step, Kaggle dataset was being used to retrain the networks and to classify the DR images from no DR ones by comparing performance results of each network. After fine-tuning the network, a clinical dataset which consists of DR and No DR images, was used to test the networks' performances on a completely different and unknown dataset and to investigate which of the proposed systems is the best model to be implemented for automatic DR diagnosis in eye clinics. Therefore, the main aim of utilising models on clinical dataset is to facilitate diagnosis process for practitioners, increase the accuracy of DR diagnosis in physical examinations, and reduce the time of diagnosis for both patients and practitioners.

2.2 Image pre-processing

Various cameras are used to prepare Kaggle images, which cause variations in the images. Therefore, an image pre-processing algorithm using OpenCV package was performed to decrease different camera resolution' effects. In the first step of the pre-processing method which is discussed by Graham (2015), the images were rescaled to have the same radius. The second step was to subtract the local average from the colour of each pixel, which maps the average to 50% grey. The last step was to eliminate the

'boundary effects' by clipping the images' edges (Graham, 2015).

2.3 Data augmentation

In this work, the dataset was enlarged because Inception-V3 is a complex network which requires a large dataset. One of the well-known methods that is used in literature to decrease overfitting is data augmentation (Simonyan and Zisserman, 2014), which is also useful to make networks robust to the noises in the data (Pan et al., 2015). Therefore, data augmentation was used in this work to provide good learning convergence and generalisation which is one of the ways to prevent overfitting. Rotating, shifting, and flipping are common examples that were implemented for data augmentation. Enlarging the dataset by this approach is beneficial, as the augmented images disappear after each training process and there is no need for more memory space. On the other hand, the size of DR images of the Kaggle dataset is not same as the No DR images. Therefore, data augmentation technique was performed to increase the number of DR images.

3 CNN architecture and fine-tuning

As a type of neural networks, CNN consists of convolutional, pooling and fully connected (FC) layers (Wang et al., 2015). The core layer of CNNs is convolutional layer which is the most important part of the network for extracting features from input data, and just like its name, it convolves the input with kernel filters to produce the output named feature map. Neurons in a specific feature map have the same weights and biases. Weights' sharing has two important advantages. First, neurons in a specific feature map can find the same features at different locations of the image. The second advantage is that the number of learning parameters decreases by weights sharing. Subsampling or pooling layer is often placed after convolutional layers and is used to reduce the number of learning parameters that should to be trained, as well as, keeping the most significant information. *Max pooling* and *average pooling* are two different pooling layers that are commonly used in CNNs. The last layer of a CNN is the FC layer, in which, the neurons have full connection to all activations in the former layer. The FC layer is the end of a CNN and it works like a traditional multilayer neural network (Krizhevsky et al., 2012; LeCun et al., 2015).

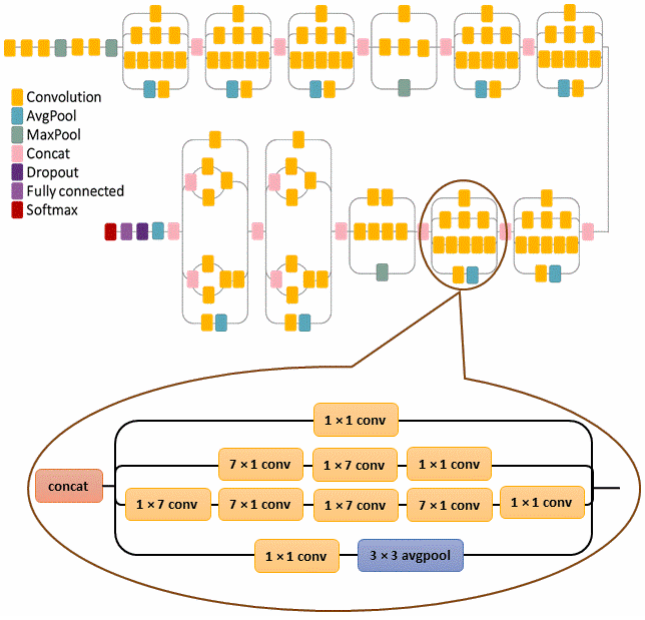
3.1 Fine-tuning

Practically, gathering adequate number of images to train a CNN network from scratch in medical image analysis is almost impossible (Tajbakhsh et al., 2016). Moreover, training a deep network from scratch with a small dataset may cause overfitting problem. Therefore, the most common way to overcome these issues is to re-train the last layers of a fully-trained CNN on a different task and adjusting its weights and parameters for the new classification task. It is noteworthy that the weights and biases of the frozen layers are kept unchanged during the training process, while the parameters of the unfrozen layers are adjusted for the new task. This method is called *fine-tuning* and can be useful to prevent overfitting, and also, due to the small dataset that is used, large computing power and memory are not required. Therefore, in this study, a pre-trained CNN model, *Inception-V3*, was fine-tuned for DR diagnosis application using Keras core library.

3.2 Inception-V3 architecture

The Inception model was designed by Szegedy et al. (2015a). The main difference of inception module from other CNN architectures is its concatenation process. In this process, the output of an inception module is the concatenation of 1×1, 3×3, and 5×5 convolutions that is provided to the input. In a recent publication, Szegedy et al. (2015b) developed Inception-V3, as an updated Inception module. Figure 2 illustrates Inception-V3 schematic diagram. As it is demonstrated in Figure 2, each block consists of three different types of convolutional layer and a pooling layer. The numbers in convolutional layers show the size of the filter's window that is convolved with the input. Overall, Inception-V3 architecture consists of 219 layers which include convolution, pooling and FC layers. More details of the network can be accessed from other references (Szegedy et al., 2015a, 2015b).

Figure 2 Schematic diagram of Inception-V3 (see online version for colours)



3.3 Software and hardware

In this work, Tensorflow, OpenCV, Keras, Numpy, Scikit-learn, and h5py packages were used to implement the image pre-processing and fine-tuning steps to classify DR and no DR images. Keras, a neural network Python library,

includes pre-trained CNNs. To implement Inception-V3, Ubuntu operating system were used for running the Tensorflow and Keras frameworks, and the other above-mentioned packages were utilised for image pre-processing and other related calculations in the Python environment. CNN architecture that was used here was pre-trained on the ImageNet dataset, and its weights are widely used in studies for fine-tuning (Shin et al., 2016). ImageNet has over 1.2 million images with 1,000 separate object categories, which is used for object recognition (Russakovsky et al., 2014).

Designing pre-trained CNNs instead of fully training networks reduces the need for graphics processing unit and external memories. Although, utilising this approach might result in a slightly lower accuracy than training the network from scratch, but it will save time and resources during training process. An Intel i7 core CPU with 8 GB memory was used here for retraining the models which is less expensive and easily available compared to common CNN hardware devices.

4 Clinical data and methodology

In this step, the proposed CNNs were optimised to diagnose DR cases from real fundus images that are captured from the Navid-e-Didegan ophthalmology clinic, Iran (Clinical Dataset, 2016). This dataset contains 101 retinal images, in which 50 of the images are labelled as No DR and 51 images are labelled as DR class by a professional ophthalmologist. Therefore, by applying these labelled images to the fine-tuned networks in the test step, the performance of the networks was examined for the new and unknown clinical dataset, in which all the test images are independent and different from the train data and they have been taken by a different device and different operator.

Table 1 Performance indices

Accuracy	$\frac{TP+TN}{P+N}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
F1-score	$\frac{2TP}{2TP+FP+FN}$

Source: Olson and Delen (2008)

To compare the results of the fine-tuned networks for the clinical data, four performance indices were calculated. These indices are accuracy, precision, recall and F1-score (Olson and Delen, 2008; Nilanjan et al., 2017). Table 1 defines these values as being used in this paper. In this table, *TP*, *FP*, *TN* and *FN* represent true positive, false positive, true negative and false negative results of the classification algorithm, respectively. *P* and *N* represent the labelled DR and no DR samples, respectively, and $P + N$ demonstrates the total number of samples.

Softmax function is usually used in FC layers of CNNs for the final decision of the classifier. Equation (1) demonstrates softmax function for input x_i where N is the number of classes.

$$f(x_i) = \frac{e^{x_i}}{\sum_{i=1}^{N} e^{x_i}} \qquad \forall i \in 1, 2, \ldots, N \tag{1}$$

Generally, the loss function that was used for CNN classification problems is cross-entropy loss function. This function defined as

$$L_i = -\log \frac{e^{f_{y_i}}}{\sum_j e^{f_j}}. \tag{2}$$

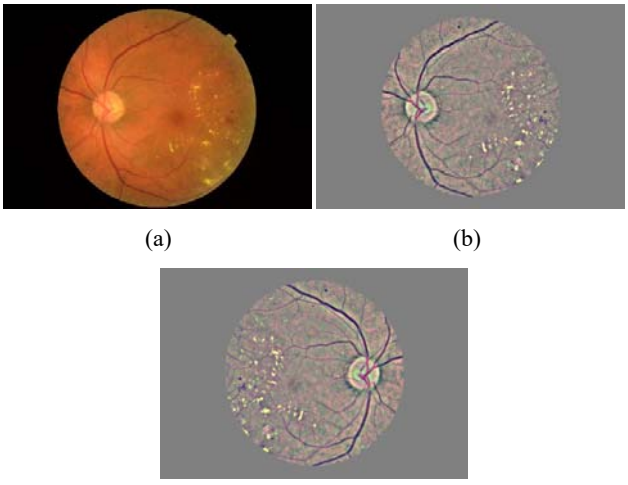
where f_j is the j^{th} element of the vector of class scores f, f_j is the softmax function and can be calculated as equation (1). The full loss of the network is the mean of L_i over all training data.

5 Performance analysis

5.1 Preparing dataset

As the train and test datasets include different fundus images taken with different devices while varying in quality and size, to reduce the variations in images, pre-processing method discussed in Section 2.2 was implemented on both train and test dataset. The result of image pre-processing step is demonstrated in Figures 3(a) and 3(b).

Figure 3 (a) The proliferative DR image (b) The pre-processed image (c) The horizontally flipped image (see online version for colours)



(a) (b)

(c)

As previously discussed, in this study, Kaggle dataset was used to perform the training step of the fine-tuning. There are 35,126 fundus images in this dataset that are divided to 25,810 images of no DR and 9,316 images of DR by an expert. To implement classification algorithms, it is very

common to use the same size of data in each class for training and testing the networks. Therefore, to balance the number of DR images and no DR ones, different image augmentation algorithms were used to increase the size of DR images. By applying the augmentation method that was discussed in Section 2.3 to DR images randomly, we increased the number of these images to 25,619. Figure 3(c) demonstrates an example of a horizontally flipped image.

In order to train the networks, 20 and 80% of the images were chosen randomly for the test and train datasets, respectively. The testing and training sets that were used for each simulation were kept identical to keep the results comparable. The test dataset was being used solely to evaluate the performance of the fine-tuning algorithm in each step. Moreover, the clinical dataset was used to evaluate the performance of each completed fine-tuning process.

5.2 Fine-tuning and retraining networks

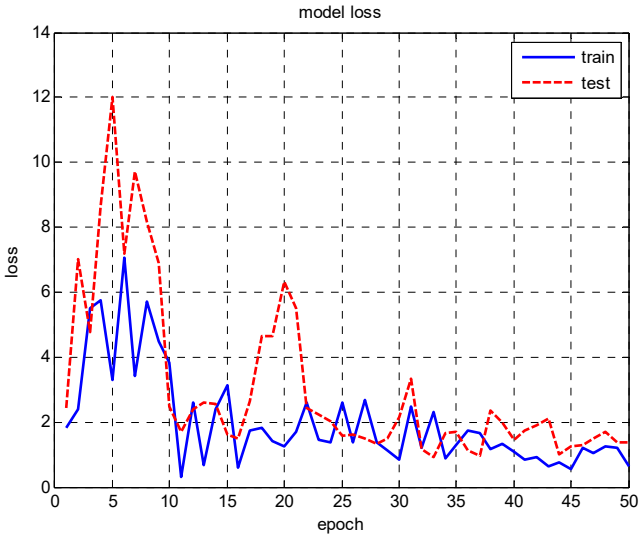
There is a growing interest in implementing fine-tuned CNNs for medical tasks due to lack of sufficient datasets (Esteva et al., 2017; Kieffer et al., 2017; Vesal et al., 2018). A common practice in fine-tuning algorithms is to re-train the last two blocks of the pre-trained networks to fine-tune the CNN. Therefore, the first 172 layers of the Inception-V3 network were kept frozen, while the other layers of the network as well as the FC layer were retrained. In addition, to compare the network' results for various fine-tuned layers of the CNN, two situations where the last three and four blocks were unfrozen and re-trained were also tested. By increasing the number of layers that were re-trained and fine-tuned, the training time was increasing from 3 to 10 hours. The hardware devices that were used for training the models are mentioned in Section 3.3.

To employ CNN for image classification applications, hyperbolic tangent (tanh), sigmoid, exponential linear unit (ELU) and rectified linear unit (RELU) are the four frequently-used activation functions, which we utilised in this study. The accuracy results on the test dataset using these activation functions were compared, while all other parameters were kept constant. Based on the investigation, ELU showed the best performance among these activation functions and were used for the rest of the study. Also, by utilising adaptive moment estimation (ADAM), adaptive gradient algorithm (Adagrad), Nadam (Adam with Nesterov momentum), Adamax (a variant of Adam based on the infinity norm), stochastic gradient descent (SGD) and Root mean square propagation (RMSProp) optimisers; SGD and ADAM demonstrated best performances among all the optimisers. The accuracy results on the test images using these optimisers, while other parameters were kept constant, have been reported in Mohammadian et al. (2017b), and demonstrate the better performance of ADAM and SGD. In addition, ADAM converged faster than SGD with a lower number of epochs required, which makes it the most appropriate optimiser for the proposed networks considering hardware limits. Due to the promising performance of ADAM optimiser and ELU activation function, these two functions were used for performing the simulations. Also, train dataset was augmented by using parameters of height-shift-range, shear-range, zoom-range, and width-shift-range equal to 0.2 and rotation-range equals to 40 degrees with the aim of increasing the model' performances, and also decreasing the overfitting in them.

Multiple block sizes of the pre-trained network were re-trained to investigate the effect of fine-tuning the different layers on the network performance. In the first case, we froze the first 172 layers of the model and fine-tuned the rest of the layers (last two blocks).

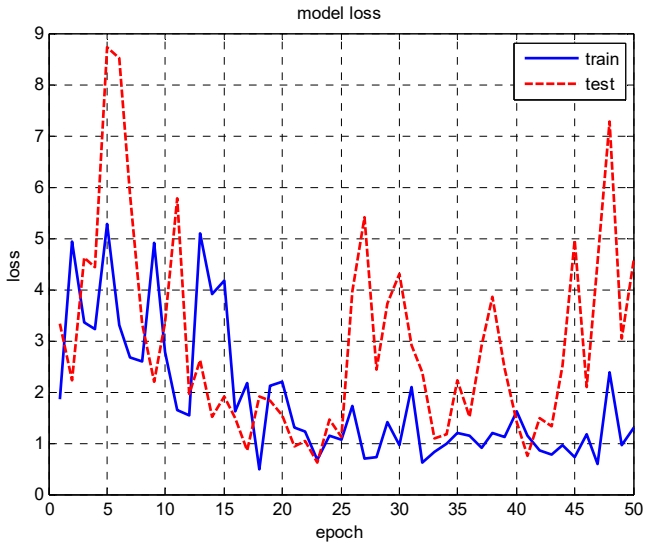
Figure 4 demonstrates the loss results for both train and test datasets over 50 epochs. Note that in this step of comparison, the test dataset is still a subset of Kaggle dataset and the loss results were used to compare the CNNs (Shin et al., 2016). As anticipated, in Figure 4, the loss is decreasing with epochs. Therefore, it can be concluded that re-training the last two blocks may result in an acceptable performance. The fluctuation seen in Figure 4 (and the rest of figures in the paper) is due to the batch size of training data. Increasing the batch size will result in reducing the fluctuation. Experimentally, in this work the batch size was chosen as 16 for all the networks. Increasing the size can decrease the fluctuations but it will violate the hardware limitations.

Figure 4 Network loss for fine-tuning of the last two blocks (the first 172 layers are kept frozen) (see online version for colours)

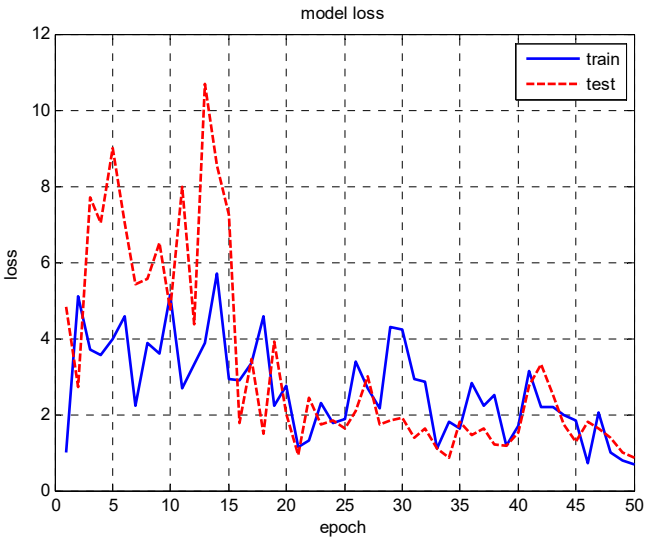


In the second step, the first 158 layers of the model were kept frozen and the last three blocks were fine-tuned. Figure 5(a) demonstrates the loss for this network. As it can be seen in Figure 5(a), although the training loss is decreasing, the test loss is not showing acceptable performance. This behaviour is the typical sign of overfitting the network during training (Shin et al., 2016).

Figure 5 (a) Network loss for fine-tuning of the last three blocks (the first 158 layers are kept frozen) (b) Network loss for fine-tuning of the last three blocks (the first 158 layers are kept frozen), while applying a dropout layer (see online version for colours)



(a)

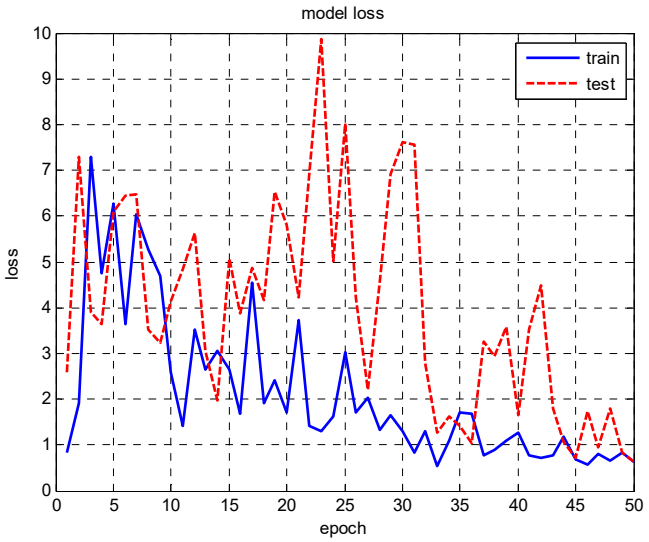


(b)

One of the most useful regularisation techniques to prevent overfitting in CNNs is to apply a dropout layer before the fully-connected layer (see Figure 3). By implementing dropout layer with dropout factor of 0.5, the network's performance is improved for the test dataset as shown in Figure 5(b).

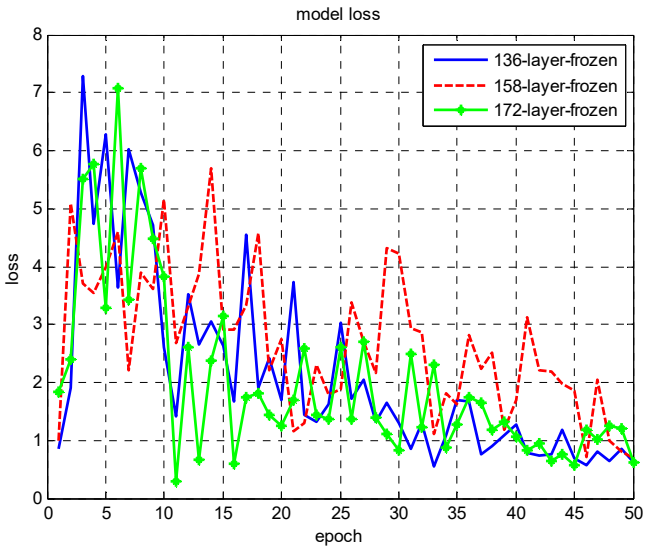
In the last step, the first 136 layers of the model were kept frozen and the last four blocks were fine-tuned. Figure 6 demonstrates the loss graph for this network. For this case, although the number of re-trained layers is more than the previous cases, the loss graphs do not demonstrate acceptable performance. The reason is that the first layers of the original CNN are tuned specifically to extract lower-level features such as edges, shapes, etc. in the base dataset which is relatively larger and more comprehensive. Therefore, re-training these layers with the smaller dataset, and lower epoch number will affect the pre-trained weights that are already well trained and hence it will disturb the accuracy.

Figure 6 Network loss for fine-tuning of the last four blocks (the first 136 layers are kept frozen) (see online version for colours)



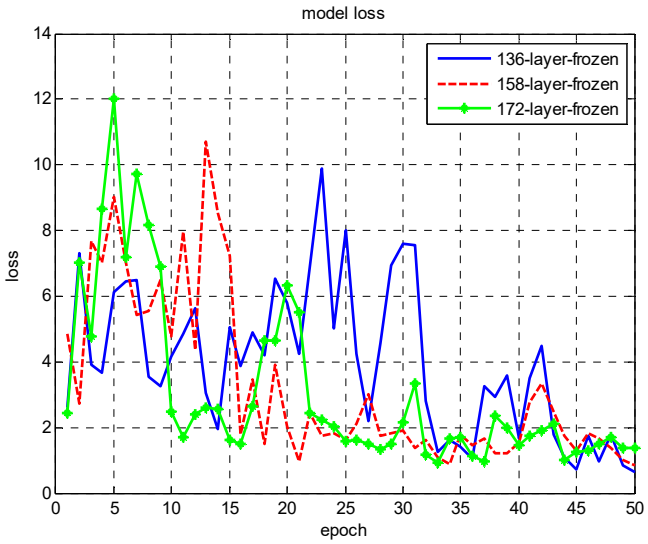
To compare different fine-tuning steps, Figures 7(a) and 7(b) demonstrate loss graphs of the train and the test dataset for all steps, respectively. As shown in Figure 7(a), training loss for all approaches is converging to a fixed number with acceptable performance. Therefore, the difference between networks can be discussed more accurately by investigating the loss graph for test dataset [Figure 7(b)]. Figure 7(b) demonstrates that networks with 158 and 172 frozen layers (fine-tuning the last three and two blocks, respectively) have better results for the test data than the network with 136 frozen layers (fine-tuning the last four blocks).

Figure 7 (a) Network loss for train dataset (b) Network loss for test dataset (see online version for colours)



(a)

Figure 7 (a) Network loss for train dataset (b) Network loss for test dataset (continued) (see online version for colours)



(b)

Overall, fine-tuning the last two or three blocks of Inception-V3 architecture utilising ELU and ADAM as the selected activation function and optimiser, respectively, showed the best classification result for diabetic retinopathy diagnosis in Kaggle dataset.

5.3 Testing the networks with clinical dataset

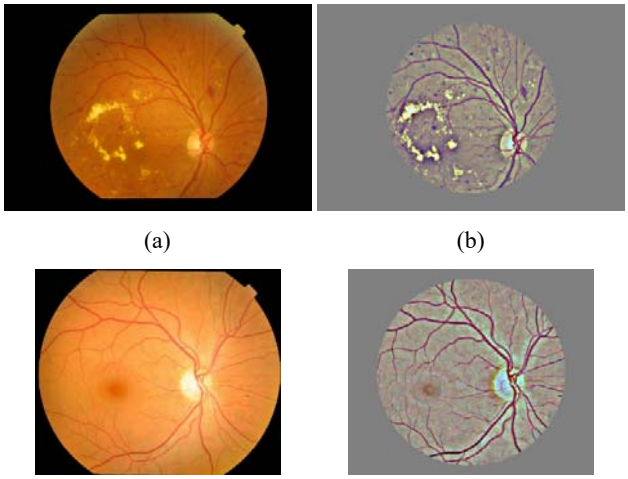
Section 5.2 included the comparative approach to fine-tune the CNN network utilising Kaggle dataset. The next case study is to clinically evaluate the networks and to test the fine-tuned networks using clinical datasets. The first step for using the clinical dataset was to pre-process the images and to diminish the variations in the images, as before. The image pre-processing technique discussed in Section 2.2 was applied to the clinical images and the pre-processed images are demonstrated in Figure 8.

The next step is to apply the pre-processed images to the proposed networks that have already been trained and discussed in Section 5.2. In the comparison step, the performance indices mentioned in Section 4 were used. The result of the comparison is demonstrated in Table 2.

As shown in Table 2, the fine-tuned network after re-training the last three blocks has the best performance results among the other networks, considering all performance indices. However, from the clinical usage perspective of the automatic DR screening approach the recall, which demonstrates the correctness of DR diagnosis, is the most important factor.

As during the screening stage, the goal is to save time for the physician while reducing the number of test images and labelling the ones which are suspicious of DR as well as the ones which are close, having a high recall means that most of the patients will be screened correctly and their images will be labelled for ophthalmologist consideration. As shown in Table 2, re-training the last two blocks of the pre-trained Inception-V3 CNN shows a superior performance in terms of recall, which can facilitate the screening process for ophthalmologists and increase their accuracy and efficiency, while re-training the last three blocks also demonstrates acceptable performance.

Figure 8 (a) Proliferative DR image (b) Pre-processed proliferative DR image (c) No DR image (d) Pre-processed no DR image (see online version for colours)



(a) (b)

(c) (d)

Table 2 Performance indices for the clinical data

Performance indices	*Unfrozen blocks: 2*	*Unfrozen blocks: 3*	*Unfrozen blocks: 4*
Accuracy	0.6733	0.7030	0.5841
Precision	0.6097	0.6438	0.5849
Recall	**0.9803**	**0.9216**	0.6078
F1-score	0.7518	0.7580	0.5961

6 Discussion

Recently, convolutional neural networks are being widely used for medical applications, especially disease diagnosis from images. CNNs are deep networks which require very large datasets to be trained. Also, there is no need to feed them hand-crafted features as the features are extracted through the network, which makes them more suitable for classifying complex datasets. Because large medical data is not available to train deep networks from scratch, it is useful to leverage previously trained networks on a different dataset to classify a new dataset (Tajbakhsh et al., 2016; Margeta et al., 2017). In this study, a pre-trained CNN (Inception-V3), which was trained for object detection, was used for DR diagnosis.

In the first layers of Inception-V3, the lower-level features of the images including edges, lines, and curves are extracted and trained. In the middle layers, another set of the lower level features such as shapes are trained and segmented. These features are general and can be used for many other datasets (such as the DR diagnosis application). In contrast, the higher-level features are extracted and

trained in the last layers of the CNN and retraining these last layers is the key factor of retargeting the system for the new classification problem (Yosinski et al., 2014). So, fine-tuning the last layers of pre-trained networks can make the networks specific to identify the individual features of the new dataset and being able to classify based on them. By fine-tuning the last layers of Inception-V3 on the new dataset, the network is adapted to detect anomalies in fundus images which are the purpose of DR diagnosis system.

Noteworthy is that in Inception-V3, by removing the fully connected layer, the output of the network is the vectors of extracted features. Therefore, other classification algorithms can be trained by providing these features to other common classification methods such as SVM. The feature extraction approach can be used when the dataset is small and is very different from the initial pre-trained dataset. In the case of this work and as the DR-related dataset is large enough, the fine tuning approach is used for DR detection with the description as mentioned above.

To compare the importance of retraining the last layers on the performance of the network, the result of fine-tuning the network for two, three, and four unfrozen blocks were studied. Also, in order to validate the fine-tuned network, a clinical dataset was used without being presented as the training set. In analysing the clinical trial results, the recall index is an important parameter in screening the performance of the automatic disease diagnosis models and shows the proportion of correctly diagnosed patients. Therefore, this index was used in order to compare different network structures and eventually to test the system on the clinical dataset.

7 Conclusions

In this paper, a widely known pre-trained convolutional neural network was fine-tuned for the diabetic retinopathy screening task. The reason for fine-tuning a pre-trained model is to save time and resource during training deep networks, while being able to retarget perfectly on the new dataset and produce acceptable results. This study was developed to compare the effect of re-training different number of layers of the networks on the network's performance in screening diabetic retinopathy cases. Also, to demonstrate the results of the fine-tuned network in clinical applications, a dataset including 101 fundus images, independent of the training and testing dataset, was applied and evaluated. The results of the study can be helpful to determine the best network architecture for diabetic retinopathy screening in real clinical cases.

References

Antal, B. and Hajdu, A. (2012) 'An ensemble-based system for microaneurysm detection and diabetic retinopathy grading', *IEEE Transaction on Biomedical Engineering*, Vol. 59, No. 6, pp. 1720–1726.

Bhatkar, A.P. and Kharat, G.U. (2015) 'Detection of diabetic retinopathy in retinal images using MLP classifier', *International Symposium on Nanoelectronic and Information Systems*, Indore, India, pp.331–335.

Cheng, C., Zhendong, W., Jianwu, Z., Ping, L. and Freeha, A. (2017) 'A finger vein recognition algorithm based on deep learning', *International Journal of Embedded Systems*, Vol. 9, No. 3, pp.220–228.

Cho, N.H. et al. (2017) *Diabetes Atlas*, 8th ed., International Diabetes Federation, Brussels, Belgium.

Clinical Dataset (2016) [online] https://goo.gl/a2dLbq (accessed 25 May 2016).

Deep Learning Models [online] https://github.com/fchollet/deep-learning-models/ (accessed 8 July 2015).

Diabetic Retinopathy Detection [online] https://www.kaggle.com/c/diabetic-retinopathy-detection/data (accessed 8 July 2015).

Esteva, A., Kuprel, B., A. Novoa, R. et al. (2017) 'Dermatologist-level classification of skin cancer with deep neural networks', *Nature*, Vol. 542, pp.115–127.

Graham, B. (2015) Kaggle Competition Report [online] https://kaggle2.blob.core.windows.net/forum-message-attachments/88655/2795/competitionreport.pdf/ (accessed 8 July 2015).

Hani, A.F.M. and Nugroho, H.A. (2010) 'Gaussian Bayes classifier for medical diagnosis and grading: application to diabetic retinopathy', *IEEE EMBS Conference on Biomedical Engineering and Sciences*, Kuala Lumpur, Malaysia, pp.52–56.

Kieffer, B., Babaie, M., Kalra, S. and Tizhoosh, H.R. (2017) 'Convolutional neural networks for histopathology image classification: training vs. using pre-trained networks', *7th International Conference on Image Processing Theory, Tools and Applications*, Montreal, Canada.

Krizhevsky, A. (2009) *Learning Multiple Layers of Features from Tiny Images*, Master thesis, Department of Computer Science, University of Toronto, Canada.

Krizhevsky, A., Sutskever, I. and E. Hinton, G. (2012) 'ImageNet classification with deep convolutional neural networks', *Neural Information Processing Systems Conf. (NIPS)*, Lake Tahoe, Nevada, USA, pp.1097–1105.

LeCun, Y., Bengio, Y. and Hinton, G. (2015) 'Deep learning', *Nature*, Vol. 521, No. 7553, pp.436–444.

Margeta, J., Criminisi, A., Lozoya, R.C., Lee, D. C. and Ayache, N. (2017) 'Fine-tuned convolutional neural nets for cardiac MRI acquisition plane recognition', *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, Vol. 5, No. 5, pp.339–349.

Mohammadian, S., Karsaz, A. and Roshan, Y.M. (2017a) 'A comparative analysis of classification algorithms in diabetic retinopathy', *7th International Conference on Computer and Knowledge Engineering*, Mashhad, Iran, pp.84–89.

Mohammadian, S., Karsaz, A. and Roshan, Y.M. (2017b) 'Comparative study of fine-tuning of pre-trained convolutional neural networks for diabetic retinopathy screening', *2nd International Conference on Biomedical Engineering*, Tehran, Iran, pp.1–6.

Niemeijer, M., Ginneken, B., Creeet, M.J. et al. (2010) 'Retinopathy online challenge: automatic detection of microaneurysms in digital colour fundus photographs', *IEEE Transaction on Medical Imaging*, Vol. 29, No. 1, pp.185–195.

Nilanjan, D., Saddam, A.S., Sayan, C., Prasenjit, M., Achintya, D. et al. (2017) 'Effect of trigonometric functions-based watermarking on blood vessel extraction: an application in ophthalmology imaging', *International Journal of Embedded Systems*, Vol. 9, No. 1, pp.90–100.

Olson, L.D. and Delen, D. (2008) *Advanced Data Mining Techniques*, 1st ed., p.138, Springer, Springer-Verlag Berlin Heidelberg.

Osareh, A., Shadgar, B. and Markham, R. (2009) 'A computational-intelligence-based approach for detection of exudates in diabetic retinopathy images', *IEEE Transaction on Information Technology in Biomedicine*, Vol. 13, No. 4, pp.535–545.

Pan, Y., Huang, W., Lin, Z. et al. (2015) 'Brain tumor grading based on neural networks and convolutional neural networks', *37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Milan, Italy, pp.699–702.

Pratt, H., Coenen, F., Broadbent, D.M., Harding, S.P. and Zheng, Y. (2016) 'Convolutional neural networks for diabetic retinopathy', *International Conference On Medical Imaging Understanding and Analysis*, Loughborough, UK, pp.200–205.

Russakovsky, O., Deng, J., Su, H. et al. (2014) *Imagenet Large Scale Visual Recognition Challenge*, arXiv:1409.0575 (accessed 16 December 2016).

Saranya, K., Ramasubramanian, B. and Mohideen, S.K. (2012) 'A novel approach for the detection of new vessels in the retinal images for screening diabetic retinopathy', *International Conference on Communication and Signal Processing*, Chennai, India, pp.57–61.

Shin, H., Roth, H.R., Gao, M. et al. (2016) 'Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning', *IEEE Transaction on Medical Imaging*, Vol. 35, No. 5, pp.1285–1298.

Simonyan, K. and Zisserman, A. (2014) *Very Deep Convolutional Networks for Large-Scale Image Recognition*, arXiv preprint arXiv:1409.1556 (accessed 8 July 2015).

Szegedy, C., Liu, W., Jia, Y. et al. (2015a) 'Going deeper with convolutions', *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, pp.1–9.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. and Wojna, Z. (2015b) 'Rethinking the inception architecture for computer vision', *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp.2818–2826.

Tajbakhsh, N., Shin, J.Y., Gurudu, S.R. et al. (2016) 'Convolutional neural networks for medical image analysis: fine tuning or full training?', *IEEE Transaction on Medical Imaging*, Vol.35, No. 5, pp.1299–1312.

Vesal, S., Ravikumar, N., Davari, A., Ellmann, S. and Maier, A. (2018) 'Classification of breast cancer histology images using transfer learning'. in Campilho A., Karray F., ter Haar Romeny, B. (Eds.): *Image Analysis and Recognition. ICIAR 2018. Lecture Notes in Computer Science*, Vol. 10882, Springer, Cham.

Vo, H.H. and Verma, A. (2016) 'New deep neural nets for fine-grained diabetic retinopathy recognition on hybrid colour space', *IEEE International Symposium on Multimedia*, San Jose, CA, USA, pp.209–215.

Wang, S., Yin, Y., Cao, G. et al. (2015) 'Hierarchical retinal blood vessel segmentation based on feature and ensemble learning', *Journal of Neurocomputing*, Vol. 149, pp.708–717.

Wang, Z. and Jianbo, Y. (2018) 'Diabetic retinopathy detection via deep convolutional networks for discriminative localization and visual explanation', *AAAI Workshops*.

Weipeng, J., Tao, J., Xingge, Z. and Liangkuan, Z. (2019) 'The optimisation of speech recognition based on convolutional neural network', *International Journal of High Performance Computing and Networking*, Vol. 13, No. 2, pp.222–231.

Yosinski, J., Clune, J., Bengio, Y. and Lipson, H. (2014) 'How transferable are features in deep neural networks?', *Advances in Neural Information Processing Systems 27 (NIPS '14)*, NIPS Foundation.

Zhang, R., Zheng, Y., Mak, T.W.C. et al. (2017) 'Automatic detection and classification of colorectal polyps by transferring low-level CNN features from nonmedical domain', *IEEE Journal of Biomedical and Health Informatics*, Vol. 21, No. 1, pp.41–47.

Zhang, W., Li, R., Deng, H. et al. (2015) 'Deep convolutional neural networks for multi-modality isointense infant brain image segmentation', *NeuroImage*, Vol. 108, pp.214–224.