Convolutional Neural Networks Based Transfer Learning for Diabetic Retinopathy Fundus Image Classification

Xiaogang Li*†‡§, Tiantian Pang*†‡§, Biao Xiong*†‡§, Weixiang Liu*†‡§, Ping Liang*, Tianfu Wang*†‡§
*Health Science Center, Shenzhen University
†School of Biomedical Engineering, Shenzhen University
‡Guangdong Key Laboratory for Biomedical Measurements and Ultrasound Imaging
§National-Regional Key Technology Engineering Laboratory for Medical Ultrasound
Shenzhen 518060, China

***Abstract*—Convolutional Neural Networks (CNNs) have gained remarkable success in computer vision, which is mostly owe to their ability that enables learning rich image representations from large-scale annotated data. In the field of medical image analysis, large amounts of annotated data may be not always available. The number of acquired ground-truth data is sometimes insufficient to train the CNNs without overfitting and convergence issues from scratch. Hence application of the deep CNNs is a challenge in medical imaging domain. However, transfer learning techniques are shown to provide solutions for this challenge. In this paper, our target task is to implement diabetic retinopathy fundus image classification using CNNs based transfer learning. Experiments are performed on 1014 and 1200 fundus images from two publicly available DR1 and MESSIDOR datasets. In order to complete the target task, we carry out experiments using three different methods: 1) fine-tuning all network layers of each of different pre-trained CNN models; 2) fine-tuning a pre-trained CNN model in a layer-wise manner; 3) using pre-trained CNN models to extract features from fundus images, and then training support vector machines using these features. Experimental results show that convolutional neural networks based transfer learning can achieve better classification results in our task with small datasets (target domain), by taking advantage of knowledge learned from other related tasks with larger datasets (source domain). Transfer learning is a promising technique that promotes the use of deep CNNs in medical field with limited amounts of data.**

***Index Terms*—Convolutional neural networks, transfer learning, fine-tuning, diabetic retinopathy, fundus images, classification.**

I. Introduction

Deep learning has markedly enhanced the state-of-the-art, in a wide variety of applications such as computer vision, speech recognition and natural language processing (NLP). It consists of multiple network layers which are able to learn representations of data with hierarchical levels of abstraction [28]. Particularly, Convolutional Neural Networks (CNNs) have proven to be very powerful machine learning tools for all kinds of computer vision tasks including face recognition and lesion detection from sufficient medical images [21]. It enables learning highly representative and hierarchical image features from a large number of training images. The major power of CNNs exists in its deep architecture [46], [45], [54], which makes CNNs automatically learn mid-level and high-level abstractions from raw data [27]. In image recognition, deep CNNs have achieved the remarkable success, primarily owe to the availability of a mass of annotated data [14] and the use of fast graphics processing units (GPUs) [37]. However, in the medical imaging domain, acquiring data as comprehensively annotated as natural images from ImageNet Large Scale Visual Recognition Challenge (ILSVRC) remains a huge challenge. When such large-scale data are not available, using a very limited number of medical data to train deep CNNs often causes overfitting and convergence problems. Besides, training deep CNNs from scratch requires large computational and memory resources such as expensive GPUs. If these resources are not available, the training will be very time-consuming. Transfer learning is a crucial component in deep CNNs and provides several solutions to these problems for medical imaging applications.

The first solution is to directly use pre-trained CNN models as feature extractors. Some studies demonstrate that generic descriptors extracted from pre-trained CNN models are very effective when recognizing and locating objects from natural images [38], [31]. CNN models that are trained using natural image datasets or other medical datasets are applied to another new medical task. Specifically, the pre-trained CNN models are used as feature generators which extract features from the input images. These extracted features are used to train a new classifier such as random forest, logistic regression. We can also find other relevant research in the medical field. In paper [7], a pre-trained CNN model is used as feature generator for feature extraction. Its chest pathology identification task is performed using a support vector machine classifier. In paper [6], a similar research is also conducted. Besides, results from [19] show that integration of CNN-based features with handcrafted features can improve a nodule detection system performance. Similarly, some papers also show that better deep models can be learned based on CNN transfer learning from ImageNet to other small datasets [38], [55], where ImageNet is an object-level labeled dataset which contains a large number

Corresponding author : Weixiang Liu, Email address : wxliu@szu.edu.cn.

978-1-5386-1937-7/17/$31.00 ©2017 IEEE

of natural images.

The second solution is to use fine-tuning that trains a CNN from a set of weights pre-trained using other data. Specifically, the weights of objective CNN we wish to train can be initialized using the weights from certain pre-trained CNN model with the same architecture, and then several or all network layers of the objective CNN are trained using new data in a supervised manner. In order to classify the interstitial lung diseases automatically, the authors fine-tune all network layers of the pre-trained CNN model [18]. Paper [8] removes the fully connected layers from the pre-trained CNN model and then adds a logistic layer. In training process, the logistic layer is only trained using the annotated data while fixing the parameters from the other layers of the network. This method achieves better results for classification of multi-view mammogram. In paper [11], in order to locate the fetal abdominal standard plane in ultrasound videos (US), the authors train a CNN using enough natural images and then implant the convolutional layers of this CNN into a new CNN called the domain transferred CNN as initialization of settings. Finally, the US samples are used to train the fully connected layers of the domain transferred CNN. Paper [43] explores when transfer learning from pre-trained ImageNet is useful and why it is useful. Tajbakhsh et al. show using a pre-trained CNN model with proper fine-tuning outperforms a CNN trained from scratch, and the layer-wise fine-tuning method provides a practical solution, in order to achieve the best performance for the task with small data. [47]. Experimental results from [43], [47] demonstrate consistently using pre-trained CNN models can achieve better results via fine-tuning, which has been successfully used in some studies [38], [5], [34], [44]. Other applications of deep CNNs in medical image field can be found in the recent survey paper [30].

The rest of this paper is organized as follows. In Section II, we first introduce related work for diabetic retinopathy fundus image classification. Section III presents the details of the method for transfer learning. The dataset description, experimental setup and results are explained in Section IV. At last, Section V presents discussion and conclusion.

II. Related Work

Diabetic retinopathy (DR) is one of the four major blinding diseases. It is a retinal complication caused by diabetes. Because of leak blood and overflow of glucose in the retinal blood vessels, abnormal lesions including microaneurysms, exudations and hemorrhages appear in the retina, which can damage the patient's vision [17]. According to the severity class of DR patients, DR can be divided into five different levels: a) No Diabetic Retinopathy, b) Mild Non-proliferative Retinopathy, c) Moderate Non-proliferative Retinopathy, d) Severe Non-proliferative Retinopathy, e) Proliferative Retinopathy. In the large scale screening, it is very necessary to automatically diagnose DR using computer systems.

In the literature, we can find different techniques and approaches for automatic detection of DR in fundus images. Sanchez et al. use a large median filter to remove the background, and then microaneurysms are detected using two different adaptive thresholds [40]. Haloi et al. [24] put forward mathematical morphology Gaussian scale space and anisotropic diffusion filter methods for image processing. Then, they build Gaussian scale space to extract 22 features that are used to train SVM for detection of exudates. In literature [39], the authors introduce the concept of visual descriptor word bag into the lesion detection process. First, they use a specific algorithm to detect a certain number of points of interest in the training images and use vectors to describe their neighborhoods. Each image is expressed as a collection of points of interest. Then, they build visual dictionary from normal regions and exudate regions, and then generate histogram to represent the original images. At last, they train a SVM classifier to determine whether a test image contains exudates. Besides, paper [25] applies top-down and region growing methods to analyze green channel information in fundus images, and trains a CNN to detect exudates from exudate candidates. In [36], the CNN is first trained using several training images from training set, where inputs of the CNN are raw intensity values of a square window centered in the pixel p that is currently processing. After training, the CNN is applied for each pixel, which aims at classifying each pixel into exudate or non-exudate class. Similarly, the research work [23] uses a deep CNN to classify each pixel of the image either as microaneurysm or non-microaneurysm. For the hemorrhage detection, this paper [50] proposes a dynamic CNN training strategy. At each of training epochs, informative normal samples are dynamically selected from a large number of medical images. And the selected samples are fed into the CNN network to train its parameters using backpropagation algorithm. In DR screening [29], the authors adopt a robust region detector to identify candidate regions, which are likely to contain lesions. Each candidate is transformed into tiles of fixed size. Then, a CNN is trained using these titles. Finally, an image-level diagnosis can be obtained by combining individual lesion results. Authors in paper [35] present a CNN method to implement five class classification of DR from digital fundus images and accurately classify its severity, where the five class classification corresponds to five levels of DR. The CNN model is trained by 80,000 fundus images and achieve a sensitivity of 95% and an accuracy of 75% on 5,000 validation images. In Dong et al. [16] a CNN with nine convolution layers and two fully connected layers is trained using 8,626 fundus images. The CNN achieves accuracy of 75.70% for classifying 1,925 test images into DR and non-DR ones. Alban et al. [2] use two pre-trained CNN models as initialization of objective networks, then training the objective networks using funds images. They provide an analysis of a model for multi-class identification of the severity of DR from fluorescein angiography photographs. The paper [1] proposes a deep-learning enhanced approach for the automatic detection of DR. Its authors make use of multiple CNNs that are trained separately to detect different lesions. This approach achieves significantly better performance to detect referable DR. In this work [22], authors apply deep learning to create an algorithm for detection of DR in retinal fundus images. They use 128,175 retinal fundus images to train a very deep CNN aiming to detecting DR directly from complete fundus images. Experiments are performed on two publicly available datasets and achieve high sensitivity and specificity for detection of referable DR. But the good results are primarily attributed to large-scale annotated training images.

III. Methods

In this study, we use CNNs-based transfer learning to implement fundus image classification of DR. In order to obtain the best classification results as much as possible, we explore different pre-trained CNN models and transfer learning methods. The specific details of the methods for transfer learning will be described in this section.

A. Transfer Learning

Transfer learning is a scheme aiming to improve a learner from one domain by transferring information from a related domain. According to the paper [33], transfer learning is defined by using the following notations. Specifying a definite domain that is represented by D, $D = \{\Omega, P(X)\}$ with $X = \{x_1, x_2, ..., x_n\} \in \Omega$, where Ω and $P(X)$ refer to a feature space and a marginal probability distribution, separately. Given a task T with $T = \{Y, F(*)\}$, Y is a label space and $F(*)$ is an objective predictive function that is learned from the feature vector and label pairs. Specifically, given a source domain D_S with learning task T_S and a target domain D_T with learning task T_T, then transfer learning is the process of improving the learning of the target predictive function $F_T(*)$ in D_T based on the knowledge learned from source domain D_S and learning task T_S, where $D_S \neq D_T$, or $T_S \neq T_T$ [33]. It is worth noting that the single source domain defined here can be extended to multiple source domains. Fig. 1 shows clearly the differences between learning processes of traditional machine learning and transfer learning. Compared to the transfer learning, traditional machine learning often learn knowledge from scratch. Transfer learning transfer the knowledge from source tasks (source domains) to a target task (target domain). In this paper, we define the natural image classification as source domain and the fundus image classification as target domain.

B. Pre-trained CNN Models

ImageNet is a big dataset of over 15 million labeled high-resolution natural images belonging to approximately 22,000 categories [14]. It is used in an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC). In this paper, we refer to the classification tasks that are based on ImageNet as source domain and our fundus image classification tasks as target domain. All pre-trained CNN models we use are trained on ImageNet dataset and are publicly available from this site [1] [51]. For our experiments, in order to explore the effect of depth of the network on experimental results, we select pre-trained CNN

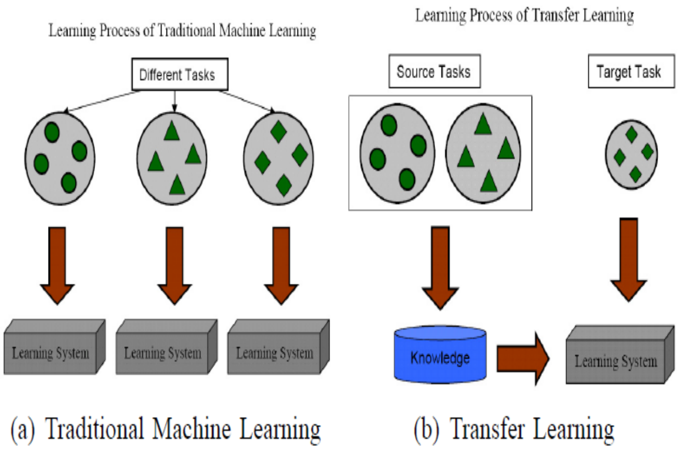


Fig. 1. The differences of the learning between traditional machine learning and transfer learning processes (illustration taken from [33]).

models that have different depths, because it is demonstrated that the representation depth is beneficial for the classification accuracy when the number of layers is substantially increased in the network [45]. The pre-trained CNN models include AlexNet, VggNet, GoogLeNet and their variants. We will briefly describe them. For further details on the models, readers may consult relevant references.

AlexNet is the first significantly successful CNN architecture and achieves 15.3% top-5 classification error on the ILSVRC 2012 [26]. It contains eight network layers. Compared to the existing deep CNNs, VggNet is designed to markedly increase the number of network layers with 16 or 19 layers corresponding to its two architectures, VggNet-vd-16 and VggNet-vd-19 [45]. To reduce the number of weight parameters, 3x3 size convolution filters with a stride of size 1 are utilized in all convolutional layers. In order to understand how different CNNs are compared with each other and with existing state-of-the-art shallow representations, three CNN architectures Vgg-f, Vgg-m and Vgg-s are presented in paper [10]. Each of CNN architectures contains five convolutional layers, three max-pooling layers and three fully connected layers. But they have different filters, strides, pooling sizes and receptive field sizes. Besides, the authors in [10] further train three variants of the Vgg-m network, with lower dimensional Fc2 layers of: 2048, 1024 and 128 dimensions respectively, where three variants are correspondingly named as Vgg-m-2048, Vgg-m-1024 and Vgg-m-128. GoogLeNet is a very deep and complex architecture compared with previous CNN architectures [46]. It designs a new module that is called $Inception$ containing one pooling layer and six convolution layers. Overall, GoogLeNet has 22 layers containing two convolution layers, five pooling layers, two fully connected layers, and nine $Inception$ modules. It is the most successful CNN architecture and achieves 5.5% top-5 classification error on the ILSVRC 2014. Table I shows succinctly the architectures of ten pre-trained CNN models we use in our experiments. Each row or sub-row refers to a layer of the network. For

[1] http://www.vlfeat.org/matconvnet/

TABLE I
PRE-TRAINED CNN ARCHITECTURES USED IN THIS PAPER (SHOWN IN COLUMNS).

ConvNet Configuration									
8 layers	8 layers	8 layers	8 layers	8 layers	8 layers	8 layers	22 layers	16 layers	19 layers
AlexNet	Vgg-f	Vgg-m	Vgg-s	Vgg-m-2048	Vgg-m-1024	Vgg-m-128	GoogLeNet	VggNet-vd-16	VggNet-vd-19
C11-96-s4	C11-64-s4	C7-96-s2	C7-96-s2	C7-96-s2	C7-96-s2	C7-96-s2	C7-64-s2	C3-64-s1 C3-64-s1	C3-64-s1 C3-64-s1
Mp3-s2	Mp3-s2	Mp3-s2	Mp3-s3	Mp3-s2	Mp3-s2	Mp3-s2	Mp3-s2	Mp2-s2	Mp2-s2
C5-256-s1	C5-256-s1	C5-256-s2	C5-256-s1	C5-256-s2	C5-256-s2	C5-256-s2	C3-192-s1	C3-128-s1 C3-128-s1	C3-128-s1 C3-128-s1
Mp3-s2	Mp3-s2	Mp3-s2	Mp2-s2	Mp3-s2	Mp3-s2	Mp3-s2	Mp3-s2	Mp2-s2	Mp2-s2
C3-384-s1 C3-384-s1 C3-256-s1	C3-256-s1 C3-256-s1 C3-256-s1	C3-512-s1 C3-512-s1 C3-512-s1	C3-512-s1 C3-512-s1 C3-512-s1	C3-512-s1 C3-512-s1 C3-512-s1	C3-512-s1 C3-512-s1 C3-512-s1	C3-512-s1 C3-512-s1 C3-512-s1	Ip (3a) Ip (3b)	C3-256-s1 C3-256-s1 C3-256-s1	C3-256-s1 C3-256-s1 C3-256-s1 C3-256-s1
Mp3-s2	Mp3-s2	Mp3-s2	Mp3-s3	Mp3-s2	Mp3-s2	Mp3-s2	Mp3-s2	Mp2-s2	Mp2-s2
							Ip (4a) Ip (4b) Ip (4c) Ip (4d) Ip (4e)	C3-512-s1 C3-512-s1 C3-512-s1	C3-512-s1 C3-512-s1 C3-512-s1 C3-512-s1
							Mp3-s2	Mp2-s2	Mp2-s2
							Ip (5a) Ip (5b)	C3-512-s1 C3-512-s1 C3-512-s1	C3-512-s1 C3-512-s1 C3-512-s1 C3-512-s1
							Ap7-s1	Mp2-s2	Mp2-s2
Fc1-4096 Fc2-4096	Fc1-4096 Fc2-4096	Fc1-4096 Fc2-4096	Fc1-4096 Fc2-4096	Fc1-4096 Fc2-2048	Fc1-4096 Fc2-1024	Fc1-4096 Fc2-128	Fc1-1024	Fc1-4096 Fc2-4096	Fc1-4096 Fc2-4096
FC-1000									
Soft-max									

example, C5-256-s1 specifies a convolution layer with 256 filters of size 5x5, with stride 1. Mp3-s2 denotes a max-pooling layer with the pooling region of size 3x3 and stride 2. For the GoogLeNet, Ap7-s1 indicates an average pooling layer with pooling window size 7x7 and stride 1. Besides, “Ip” refers to the $Inception$ module.

C. Fine-tuning CNN Models for Transfer Learning

Fine-tuning is to train a CNN from a set of weights pre-trained using other data. Specifically, the weights of a pre-trained CNN model are used as initialization of the objective CNN with the same architecture, and then objective CNN can be trained on target data in a supervised manner.

There exist two methods for fine-tuning that will be used in our experiments. The first one is full fine-tuning that refers to fine-tuning all network layers of a CNN model. It is necessary to fine-tune all layers, when the correlation between the target and source domains is not significant. The second is to fine-tune the pre-trained CNN model in a layer-wise manner. In deep CNNs, the lower layers can learn low-level image representations applicable to many vision tasks and the higher layers can learn high-level representations applicable to the specific target task. [54]. Therefore, full fine-tuning may be not necessary for some tasks. Besides, full fine-tuning is extremely time-consuming. But layer-wise fine-tuning method may be able to effectively address this problem. This method starts training the last fully connected layer and then gradually include more network layers during training process until achieving the desired performance. It has been demonstrated that fine-tuning a pre-trained CNN model using the target data can markedly improve the performance [20], where this pre-trained CNN model is trained on source data. When a medium sized dataset is available for the target task, fine-tuning is a recommendable transfer learning scheme for the classification based on CNNs. Fig. 2 shows the pipeline of transferring parameters from the pre-trained CNN model (top row) to the objective CNN, where weight parameters are transferred to our fundus image classification task (target task). After transferring, size of the output layer can be set to the number of target categories, which is able to compensate for different image statistics between the source and target data.

D. Feature Extraction Based on CNN Models for Transfer Learning

CNNs consist of multiple network layers which are able to learn representations of data with hierarchical levels of abstraction. The representation from each layer can be computed from the representation in the previous layer. CNNs can be trained using end-to-end backpropagation algorithm, because it combines feature extraction (learning representations) and classification processes. In general, the convolution layers are considered as feature extractors, and the fully connected layers are seen as a classifier. The lower layers of a CNN learn low-level features, and the higher layers learn high-level features, which are able to describe the whole or part of the object in images. Some studies show that the features extracted from pre-trained CNN models are very effective for many vision tasks [38], [31], [48], [15]. In this paper, pre-trained CNN models are used as feature extractors for our target task. As shown in Fig. 3, each of fundus images is fed into the pre-trained CNN model (top row). The outputs extracted from the internal layers are used as features, which are then used to train a support vector machine classifier (bottom row).

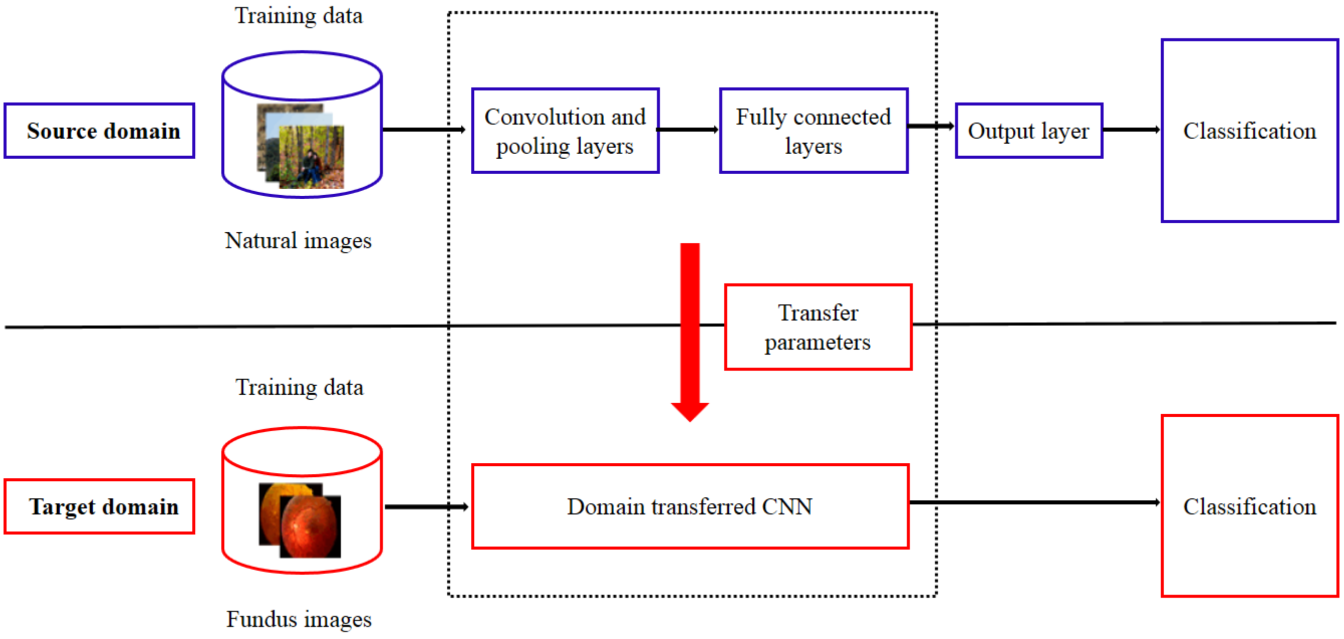


Fig. 2. The pipeline of transferring parameters between CNNs.

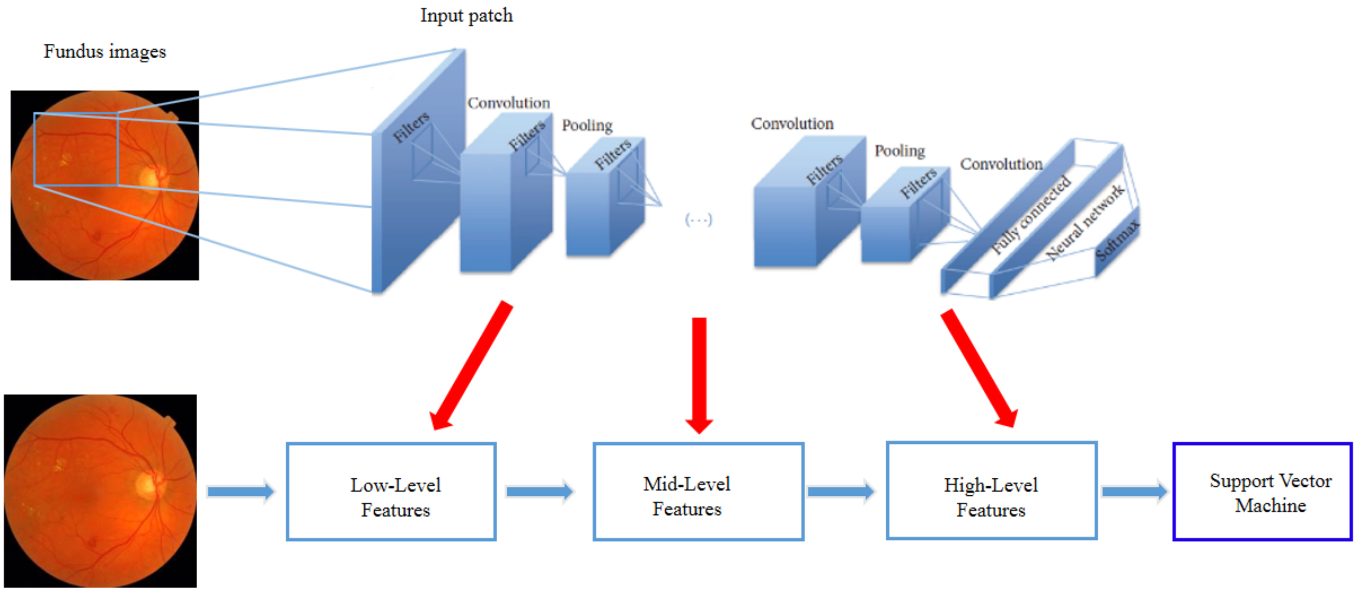


Fig. 3. The pipeline of transfer learning based on the features extracted from pre-trained CNN models.

IV. Experiments and Results

In this section, we carry out experiments on two fundus image datasets using CNNs based transfer learning, which are presented in Section III. We only focus on performing a binary classification task, which is to classify fundus images as normal or abnormal images. In all experiments, the classification performance is evaluated using a 5-fold cross validation method. To ensure a fair evaluation, we use the same data splits for all experiments in this paper. The datasets, the experimental setup and results are described in detail.

A. Dataset Description

1) DR1 Dataset: The publicly available DR1 dataset is provided by the Ophthalmology Department, Federal University of So Paulo, Brazil, containing 1014 color fundus images that are composed of 687 normal images and 327 abnormal images, where abnormal images contain 245 images with bright lesions and 191 images with red lesions. Besides, 109 images have the signs of both bright and red lesions [39]. Images are acquired using a Topcon TRC-50X mydriatic camera. All images have resolution of 640x480 pixels. For each image, three medical experts manually annotate the image as having or not having

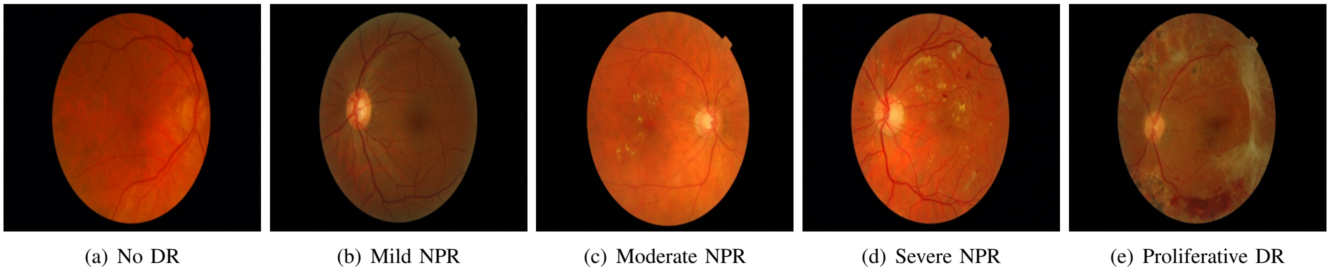

(a) No DR (b) Mild NPR (c) Moderate NPR (d) Severe NPR (e) Proliferative DR

Fig. 4. Levels of diabetic retinopathy (DR) with increasing severity.

bright or red lesions. According to the experts who participate in the evaluation, normal images have no any signs of DR. Abnormal images contain different lesions such as exudate, hemorrhage and microaneurysm.

2) MESSIDOR Dataset: MESSIDOR is also a publicly available dataset [13]. This dataset consists of 1200 eye fundus images acquired at three different sites, 800 with and 400 without pupil dilation. Images are captured at 1440x960, 2240x1488, or 2304x1536 pixels by a color video 3CCD camera. According to the severity class of DR patients, all images are divided into five levels. MESSIDOR is composed of 546 images from DR level 0 (normal), 153 images form DR level 1 (mild), 247 images from DR level 2 (moderate), 254 images from DR level 3 (severe), where the level 3 contains severe non-proliferative retinopathy and proliferative retinopathy. Fig. 4 shows five fundus images from different levels in this dataset, where NPR refers to non-proliferative retinopathy.

B. Data Pre-processing

Experiments we conduct in this paper are based on color fundus images from two public datasets. We find that almost all images have excessive black-space on either side of the eye, which may have an effect on our experiments. Concerning this, we crop a fixed number of pixels from either side of each color fundus image, in order to leave only the field of view the image. Besides, we resize the cropped images to a fixed resolution of 224x224 pixels in our experiments, because all pre-trained CNN models we use take as inputs 224x224 RGB images. Before feeding the resized images into each CNN, one pre-processing for each dataset is to remove an average image from each of the images. Specifically, the average image is provided by current CNN models that are used as feature extractors. However, if we fine-tune pre-trained CNNs on each dataset, the average image will be calculated over training images from this dataset.

C. Fine-tuning CNN Models for Fundus Image Classification

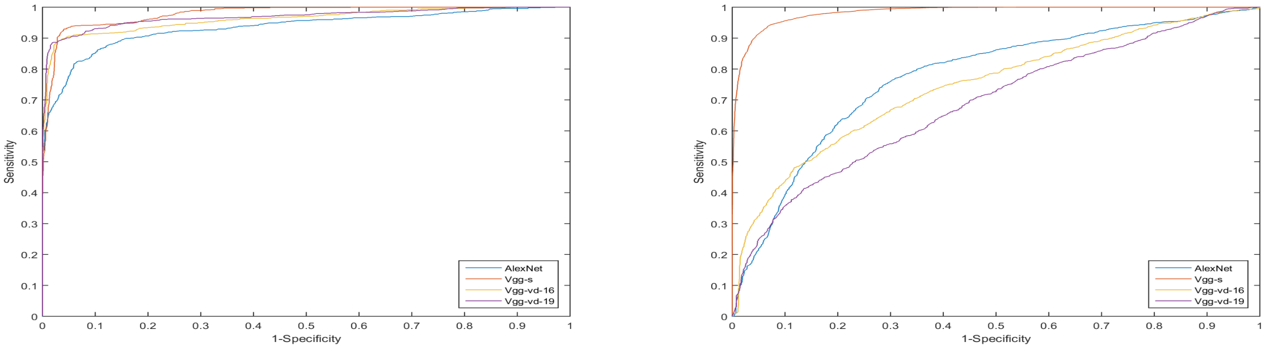
In this part, we fine-tune four pre-trained CNN models so as to implement funds image classification of DR, where the CNN models include AlexNet, Vgg-s, VggNet-vd-16 and VggNet-vd-19. All our fine-tuning experiments are carried out using the MATLAB toolbox called “MatConvNet” [51]. For each pre-trained CNN model, we construct a domain transferred CNN with the same architecture except for the output layer (the last fully connected layer), where the number of neurons in the output layer is set to 2 corresponding to our binary classification task. Then, we implant the parameters of the pre-trained CNN model into the transferred CNN as shown in Fig. 2, and weight parameters of the output layer are initialized using Gaussian distribution. Fine-tuning is done for 30 epochs with stochastic gradient descent using minibatches of size 50. Learning rate starts off at a value of 0.1 and is decreased linearly to 0.0001 over 30 epochs. Weight decay and momentum are set to 0.0005, 0.95, respectively. In order to reduce overfitting, fine-tuning usually need a medium sized dataset. Taking into account our small datasets, we need artificially enlarge the datasets using label-preserving transformations [12]. The methods of data augmentation are vertical flipping, horizontal flipping and rotation with different orientations. This increases the size of our datasets by a factor of 15. For each fundus image dataset, the training set and test set use the same data augmentation methods.

We conduct a full fine-tuning over each of four transferred CNNs, where all layer’s weight parameters are updated during training the network. Each CNN is fine-tuned using the same training protocol and the same implementation. A 5-fold cross validation method is used to evaluate the classification performance. Besides, we calculate four criterions, namely sensitivity (SN, in %), specificity (SP, in %), accuracy (ACC, in %) and area under the ROC curve (AUC). SN and SP are derived based on a threshold 0.5. In each fold, we return all parameters at the last training epoch as our optimal model parameters for cross validation. Experimental results on two datasets are summarized in Table II. On DR1 dataset, the Vgg-s, VggNet-vd-16 and VggNet-vd-19 achieve competitive results which are better than that of AlexNet. But on MESSIDOR dataset, Vgg-s obtains the best results that are very significant. Fig. 5 shows ROC curves of different CNN models performed on DR1 and MESSIDOR using a 5-fold cross validation method. It can be seen that the Vgg-s outperforms other three models.

We attempt to fine-tune Vgg-s in a layer-wise manner rather than full fine-tuning. The reason why we fine-tune Vgg-s is that above experimental results show that Vgg-s outperforms other three models on DR1 and MESSIDOR. Besides, the number of network layers in Vgg-s is less than VggNet-vd-16 and VggNet-vd-19, which helps reduce the risk of overfitting when a large amount of training data is not available. As shown in Table I, Vgg-s has 8 network layers including 5 convolution

TABLE II
CLASSIFICATION RESULTS BASED ON FULL FINE-TUNING.

Model	DR1				MESSIDOR			
	SP	SN	ACC	AUC	SP	SN	ACC	AUC
AlexNet	94.07	81.27	89.75	0.9342	72.44	72.63	72.56	0.7727
Vgg-s	**97.43**	86.47	93.73	**0.9786**	**97.11**	**86.03**	**92.01**	**0.9834**
VggNet-vd-16	94.32	**90.78**	93.17	0.9616	70.86	65.78	68.67	0.7437
VggNet-vd-19	96.49	89.31	**94.12**	0.9684	60.30	64.70	62.19	0.6869



(a)

(b)

Fig. 5. ROC curves for four CNN models based on DR1 (Left) and MESSIDOR (Right).

layers, 3 fully connected layers. Hence we construct a domain transferred CNN that has the same architecture as Vgg-s, except for the Fc8 layer (the output layer) with 2 neurons. Pre-trained parameters of the internal layers (excluding the output layer) of the Vgg-s are then transferred to the domain transferred CNN. Table III shows our layer-wise fine-tuning schemes. In this Table, the letter “U” denotes the weight parameters of current layers are updated during fine-tuning the transferred CNN. The letter “F” indicates weight parameters of current layers are fixed and are not updated. It is worth noting that the weight parameters of the Fc8 layer are always are initialized by Gaussian distribution. For example, the “Fc6 - Fc8” refers to fine-tuning Fc6, Fc7 and Fc8 layers of the domain transferred CNN while other layers are not fine-tuned.

We perform experiments on the augmented datasets that have been described in the previous experiments. The domain transferred CNN is fine-tuned for 30 epochs with stochastic gradient descent using batch size of 50. Similarly, learning rate starts off at a value of 0.1 and is decreased linearly to 0.0001. In order to reduce overfitting, weight decay with a value of 0.0005 is adopted. Momentum is used to speed up learning and is set to 0.95. To evaluate experimental results, the 5-fold cross validation method is also utilized. Besides, the SN, SP, ACC and AUC are also used and are calculated using the same methods described in previous full fine-tuning experiments. The parameters from the last training epoch are still considered as our optimal model parameters. Table IV shows the classification results when layer-wise fine-tuning different layers in the domain transferred CNN. On the DR1 dataset, when layer-wise fine-tuning different layers, the transferred CNN achieves competitive results. But on the MESSIDOR dataset, the differences between classification results are extremely significant. The reason may be that the correlation between the source and target domains is not significant for the MESSIDOR. Hence it is necessary to fine-tune all layers. For example, fine-tuning the “Conv1 - Fc8” or “Conv2 - Fc8” layers can achieve good results on the MESSIDOR dataset. However, the competitive results on the DR1 show that the advantages of deep fine-tuning (fine-tuning many layers) are similar to that of shallow fine-tuning (fine-tuning a few layers). The reason may be that the correlation between the source and target domains is very significant for the DR1. If the distance between the source and target applications is relatively small, shallow fine-tuning such as fine-tuning the last few fully connected layers can be able to obtain the desired classification performance, which is beneficial for the tasks with a very limited number of data and avoids the overfitting and convergence issues to some extent.

D. Feature Extraction Based on CNN Models for Fundus Image Classification

In this part, we use different pre-trained CNN models as feature extractors. As shown in Fig. 3, each fundus image is fed into the pre-trained CNN model (top row). The outputs extracted from the last fully connected layer are used as feature vectors. These feature vectors are then used to train a support vector machine classifier (bottom row), where each of images corresponds to a feature vector. For our experiments, we use the outputs of the last fully connected layer in the pre-trained

TABLE III
LAYER-WISE FINE-TUNING SCHEMES OF THE DOMAIN TRANSFERRED CNN.

Fine-tuning layers	Domain transferred CNN							
	Conv1	Conv2	Conv3	Conv4	Conv5	Fc6	Fc7	Fc8
Conv1 - Fc8	U	U	U	U	U	U	U	U
Conv2 - Fc8	F	U	U	U	U	U	U	U
Conv3 - Fc8	F	F	U	U	U	U	U	U
Conv4 - Fc8	F	F	F	U	U	U	U	U
Conv5 - Fc8	F	F	F	F	U	U	U	U
Fc6 - Fc8	F	F	F	F	F	U	U	U
Fc7 - Fc8	F	F	F	F	F	F	U	U
Fc8	F	F	F	F	F	F	F	U

TABLE IV
CLASSIFICATION RESULTS BASED ON LAYER-WISE FINE-TUNING.

Fine-tuning layers	DR1				MESSIDOR			
	SP	SN	ACC	AUC	SP	SN	ACC	AUC
Conv1 - Fc8	97.43	86.47	93.73	**0.9786**	**97.11**	**86.03**	**92.01**	**0.9834**
Conv2 - Fc8	**97.63**	**88.43**	**94.52**	0.9655	95.23	74.11	87.46	0.9467
Conv3 - Fc8	96.94	87.45	93.79	0.9722	76.15	76.19	76.19	0.8210
Conv4 - Fc8	96.49	88.14	93.66	0.9702	79.70	69.59	75.31	0.8134
Conv5 - Fc8	96.79	87.16	93.60	0.9686	86.02	65.46	77.06	0.8268
Fc6 - Fc8	97.04	85.69	93.27	0.9628	81.38	64.44	74.00	0.8038
Fc7 - Fc8	97.33	85.78	93.50	0.9659	74.91	72.32	73.75	0.8094
Fc8	97.53	82.16	92.35	0.9610	81.23	48.25	66.78	0.7576

CNN as our feature vectors. The reason is that the features learned from higher layers have stronger representation power [38]. Besides, in terms of dimensionality of features, the fully connected layers have smaller dimension compared to convolutional layers. For all pre-trained CNN models, each of features that are extracted from the last fully connected layer has 1000 dimensions. We use the 1000 dimensional features in combination with a support vector machine (SVM) to solve our binary classification task. The SVM with Gaussian kernel is implemented using LibSVM, which is a library for support vector machines [9]. Experiments are performed on the images pre-processed in Section IV, not the augmented images. As with the above experiments. 5-fold cross validation is applied to the evaluation of experimental results. Optimal regularization parameter C and kernel spread gamma γ are tuned using a grid search strategy. The SN, SP, ACC and AUC are also calculated. In Table V, the experimental classification results are achieved based on the features that are extracted from 10 pre-trained CNN models. Overall, when pre-trained CNN models are used as feature extractors, the SVM classifier applied to the extracted features achieves inferior results compared to the domain transferred CNN fine-tuned in a layer-wise manner in Section IV. But on the DR1, the specificity is still kept quite high for all CNN models, which shows the feasibility of detecting DR in fundus images using the features extracted from pre-trained CNNs. In all experiments of this paper, results achieved on the MESSIDOR are not very good, probably because both source and target domains are not significantly related. It is reasonable that when there exists explicit relationship between task domains, transfer learning may be able to help improve the target task [33].

E. Comparison with other automatic DR screening methods

It is difficult to compare our results with other methods, because experiments are not carried out on the same settings. For example, the proportion of images with DR lesions is different, and some important measures are not disclosed, such as accuracy (ACC) and area under the curve (AUC). In spite of this, but sensitivity (SN), specificity (SP) and AUC values can be accepted for mutual comparison. A comparison of our results with existing state-of-the-art results for DR automatic screening is shown in Table VI. We perform a binary classification task that classifies fundus images as normal or abnormal images, where normal and abnormal images are regarded as no DR and DR respectively. For MESSIDOR dataset, our highest results in Table IV are at the bottom of the Table VI. To the best of our knowledge, there is no relevant study that performs a binary classification task based on normal and abnormal images from the DR1 dataset. In Table VI, compared to other latest publications, our approach outperforms some existing methods in terms of the clinically important measures on the MESSIDOR dataset.

TABLE V
CLASSIFICATION RESULTS BASED ON THE FEATURES FROM DIFFERENT CNN MODELS.

Model	DR1				MESSIDOR			
	SP	SN	ACC	AUC	SP	SN	ACC	AUC
AlexNet	91.56	74.58	86.49	0.9193	74.31	51.47	64.17	0.6759
GooLeNet	93.45	**77.66**	88.36	0.9272	**79.37**	31.49	57.25	0.5786
Vgg-f	94.03	75.25	88.07	0.9190	76.30	**56.60**	**67.92**	**0.7339**
Vgg-m	**95.49**	65.73	85.60	0.8935	74.31	54.41	65.60	0.7058
Vgg-m-128	93.16	70.01	85.89	0.8996	73.24	54.95	64.42	0.7064
Vgg-m-1024	93.01	70.64	86.30	0.8978	73.55	56.40	65.50	0.7228
Vgg-m-2048	93.45	76.43	87.28	**0.9307**	71.56	55.86	65.00	0.7153
Vgg-s	94.47	77.05	**88.76**	0.9283	70.64	56.22	67.83	0.7020
VggNet-vd-16	92.43	72.76	85.99	0.9112	73.85	54.58	64.08	0.6905
VggNet-vd-19	93.89	70.34	86.39	0.8958	74.78	55.12	66.58	0.7085

TABLE VI
COMPARISON OF CLASSIFICATION RESULTS WITH OTHER METHODS FOR AUTOMATIC DR SCREENING ON MESSIDOR DATASET.

Model	MESSIDOR				
	SP	SN	ACC	AUC	Detection types
Tang et al. [49]	-	-	-	0.8700	No DR/DR
Snchez et al. [41]	50.00	92.20	-	0.8760	No DR/DR
Seoud et al. [42]	50.00	93.90	-	0.8990	No DR/DR
Antal et al. [4]	91.00	90.00	90.00	0.9890	No DR/DR
Antal et al. [3]	88.00	76.00	82.00	0.9000	No DR/DR
Haloi et al. [23]	96.00	97.00	96.00	0.9880	No DR/DR
Wang et al. [53]	50.00	97.80	90.50	0.9210	No DR/DR
Vo et al. [52]	80.30	91.60	85.80	0.8620	No DR/DR
	85.70	88.20	87.10	0.8700	No DR/DR
Orlando et al. [32]	50.00	76.45	-	0.7325	No DR/DR
	50.00	84.71	-	0.7912	No DR/DR
	50.00	91.09	-	0.8932	No DR/DR
Ours	97.11	86.03	92.01	0.9834	No DR/DR

V. DISCUSSION AND CONCLUSION

In this paper, we use CNNs based transfer learning for a small number of target samples. In order to address DR fundus image classification task, we use different transfer learning methods. Firstly, we implant the parameters of different pre-trained CNN models into the domain transferred CNNs, where the pre-trained CNNs are trained using ImageNet. Then, the fundus images are used to fine-tune the transferred CNNs. Secondly, we consider pre-trained CNN models as feature extractors for fundus images, Outputs of the last fully connected layer are used as features in combination with a support vector machine (SVM) to solve the classification task. Experiments on two public datasets show fine-tuning the CNNs achieves the better classification results. If the difference between source and target domains are small, we suggest layer-wise fine-tuning pre-trained CNNs with a very limited number of data, which can reduce the risk of overfitting. When training the SVM using the extracted features from CNNs also achieves promising results. This is beneficial for the tasks with a very small dataset. In order to improve fundus image classification task, future work is to fuse the features extracted from convolution and fully connected layers with the handcrafted features. It has proven to be very useful in some applications, Overall, Leveraging knowledge learned from a source domain with larger datasets can be able to improve the target task with scarce data.

ACKNOWLEDGMENT

This work is partly supported by Shenzhen Fundamental Research Projects (Nos: JCYJ20160422113119640, JCYJ20140828152830610), and Shenzhen Science and Technology Program (No. CXZZ20140418182638764).

REFERENCES

[1] Michael David Abramoff, Yiyue Lou, Ali Erginay, Warren Clarida, Ryan Amelon, James C Folk, and Meindert Niemeijer. Improved automated detection of diabetic retinopathy on a publicly available

dataset through integration of deep learningdeep learning detection of diabetic retinopathy. *Investigative ophthalmology & visual science*, 57(13):5200–5206, 2016.

[2] Marco Alban and Tanner Gilligan. Automated detection of diabetic retinopathy using fluorescein angiography photographs, 2016.

[3] B Antal and A Hajdu. An ensemble-based system for microaneurysm detection and diabetic retinopathy grading. *IEEE transactions on biomedical engineering*, 59(6):1720–1726, 2012.

[4] Blint Antal and Andrs Hajdu. An ensemble-based system for automatic screening of diabetic retinopathy. *Knowledge-Based Systems*, 60(2):20–27, 2014.

[5] H Azizpour, A. S Razavian, J Sullivan, and A Maki. From generic to specific deep representations for visual recognition. In *Computer Vision and Pattern Recognition Workshops*, pages 36–45, 2015.

[6] Yaniv Bar, Idit Diamant, Lior Wolf, and Hayit Greenspan. Deep learning with non-medical training used for chest pathology identification. In *Proc. SPIE*, volume 9414, page 94140V, 2015.

[7] Yaniv Bar, Idit Diamant, Lior Wolf, Sivan Lieberman, Eli Konen, and Hayit Greenspan. Chest pathology detection using deep learning with non-medical training. In *IEEE International Symposium on Biomedical Imaging*, pages 294–297, 2015.

[8] Gustavo Carneiro, Jacinto Nascimento, and Andrew P Bradley. Unregistered multiview mammogram analysis with pre-trained deep learning models. pages 652–660, 2015.

[9] Chih Chung Chang and Chih Jen Lin. Libsvm: A library for support vector machines. *Acm Transactions on Intelligent Systems & Technology*, 2(3):1–27, 2012.

[10] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *Computer Science*, 2014.

[11] Hao Chen, Dong Ni, Jing Qin, Shengli Li, Xin Yang, Tianfu Wang, and Pheng Ann Heng. Standard plane localization in fetal ultrasound via domain transferred deep neural networks. *IEEE journal of biomedical and health informatics*, 19(5):1627–1636, 2015.

[12] Dan CireşAn, Ueli Meier, Jonathan Masci, and Jürgen Schmidhuber. Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32:333–338, 2012.

[13] Etienne Decencire, Xiwei Zhang, Guy Cazuguel, Bruno Lay, Batrice Cochener, Caroline Trone, Philippe Gain, Richard Ordonez, Pascale Massin, and Ali Erginay. Feedback on a publicly distributed image database: The messidor database. *Image Analysis & Stereology*, 33(3):231–234, 2014.

[14] Jia Deng, Wei Dong, R. Socher, Li Jia Li, Kai Li, and Fei Fei Li. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255, 2009.

[15] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. pages 647–655, 2014.

[16] Jiancheng Dong, Jiancheng Dong, and Jiancheng Dong. Automatic screening of diabetic retinopathy images with convolution neural network based on caffe framework. In *International Conference on Medical and Health Informatics*, pages 90–94, 2017.

[17] O Faust, U R Acharya, E. Y. Ng, K. H. Ng, and J. S. Suri. Algorithms for the automated detection of diabetic retinopathy using digital fundus images: a review. *Journal of Medical Systems*, 36(1):145–157, 2012.

[18] Mingchen Gao, Ulas Bagci, Le Lu, Aaron Wu, Mario Buty, Hoo Chang Shin, Holger Roth, Georgios Z. Papadakis, Adrien Depeursinge, and Ronald M. Summers. Holistic classification of ct attenuation patterns for interstitial lung diseases via deep convolutional neural networks. pages 1–6, 2015.

[19] Bram Van Ginneken, Arnaud A. A. Setio, Colin Jacobs, and Francesco Ciompi. Off-the-shelf convolutional neural network features for pulmonary nodule detection in computed tomography scans. In *IEEE International Symposium on Biomedical Imaging*, pages 286–289, 2015.

[20] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. pages 580–587, 2013.

[21] Hayit Greenspan, Bram Van Ginneken, and Ronald M. Summers. Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. *IEEE Transactions on Medical Imaging*, 35(5):1153–1159, 2016.

[22] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama*, 316(22):2402–2410, 2016.

[23] Mrinal Haloi. Improved microaneurysm detection using deep neural networks. *Computer Science*, 2015.

[24] Mrinal Haloi, Samarendra Dandapat, and Rohit Sinha. A gaussian scale space approach for exudates detection, classification and severity prediction. *Computer Science*, 56(1):3–6, 2015.

[25] Hussain F. Jaafar, Asoke K. Nandi, and Waleed Al-Nuaimy. Automated detection and grading of hard exudates from retinal fundus images. In *Signal Processing Conference, 2011 European*, pages 66–70, 2011.

[26] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *International Conference on Neural Information Processing Systems*, pages 1097–1105, 2012.

[27] Y Lecun, K Kavukcuoglu, and C Farabet. Convolutional networks and applications in vision. In *IEEE International Symposium on Circuits and Systems*, pages 253–256, 2010.

[28] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[29] Gilbert Lim, Mong Lee Li, Hsu Wynne, and Tien Yin Wong. Transformed representations for convolutional neural networks in diabetic retinopathy screening. In *AAAI Workshop on Modern Artificial Intelligence for Health Analytics*, 2014.

[30] G Litjens, T Kooi, B. E. Bejnordi, Setio Aaa, F Ciompi, M Ghafoorian, Van Der Laak Jawm, Ginneken B Van, and C. I. Snchez. A survey on deep learning in medical image analysis. *Cell Reports*, 19(9):1953–1966, 2017.

[31] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1717–1724, 2014.

[32] José Ignacio Orlando, Elena Prokofyeva, Mariana del Fresno, and Matthew B Blaschko. Learning to detect red lesions in fundus photographs: An ensemble approach based on deep learning. *arXiv preprint arXiv:1706.03008*, 2017.

[33] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge & Data Engineering*, 22(10):1345–1359, 2010.

[34] Otavio A. B. Penatti, Keiller Nogueira, and Jefersson A. Dos Santos. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? In *Computer Vision and Pattern Recognition Workshops*, pages 44–51, 2015.

[35] Harry Pratt, Frans Coenen, Deborah M. Broadbent, Simon P. Harding, and Yalin Zheng. Convolutional neural networks for diabetic retinopathy . *Procedia Computer Science*, 90:200–205, 2016.

[36] P Prentasic and S Loncaric. Detection of exudates in fundus photographs using convolutional neural networks. In *International Symposium on Image and Signal Processing and Analysis*, pages 188–192, 2015.

[37] Rajat Raina, Anand Madhavan, and Andrew Y. Ng. Large-scale deep unsupervised learning using graphics processors. In *International Conference on Machine Learning*, pages 873–880, 2009.

[38] Ali S Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf : An astounding baseline for recognition. pages 512–519, 2014.

[39] A Rocha, T Carvalho, H. F. Jelinek, S Goldenstein, and J Wainer. Points of interest and visual dictionaries for automatic retinal lesion detection. *IEEE Transactions on Biomedical Engineering*, 59(8):2244–2253, 2012.

[40] C. I. Sanchez and MMayo A Garcia. Retinal image analysis based on mixture models to detect hard exudates. *Medical Image Analysis*, 13(4):650–658, 2009.

[41] Clara I Sánchez, Meindert Niemeijer, Alina V Dumitrescu, Maria SA Suttorp-Schulten, Michael D Abramoff, and Bram van Ginneken. Evaluation of a computer-aided diagnosis system for diabetic retinopathy screening on public data. *Investigative ophthalmology & visual science*, 52(7):4866–4871, 2011.

[42] L Seoud, T Hurtut, J Chelbi, F Cheriet, and J. M. Langlois. Red lesion detection using dynamic shape features for diabetic retinopathy screening. *IEEE Transactions on Medical Imaging*, 35(4):1116–1126, 2016.

[43] Hoo-Chang Shin, Holger R Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura, and Ronald M Summers. Deep convolutional neural networks for computer-aided detection: Cnn

architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285–1298, 2016.

[44] Hayaru Shouno, Satoshi Suzuki, and Shoji Kido. A transfer learning method with deep convolutional neural network for diffuse lung disease classification. In *International Conference on Neural Information Processing*, pages 199–207, 2015.

[45] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *Computer Science*, 2014.

[46] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. pages 1–9, 2014.

[47] Nima Tajbakhsh, Jae Y Shin, Suryakanth R Gurudu, R Todd Hurst, Christopher B Kendall, Michael B Gotway, and Jianming Liang. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE transactions on medical imaging*, 35(5):1299–1312, 2016.

[48] T Tamaki, J Yoshimuta, M Kawakami, B Raytchev, K Kaneda, S Yoshida, Y Takemura, K Onji, R Miyaki, and S Tanaka. Computer-aided colorectal tumor classification in nbi endoscopy using local features. *Medical Image Analysis*, 17(1):78–100, 2013.

[49] Li Tang, Meindert Niemeijer, Joseph M. Reinhardt, Mona K. Garvin, and Michael D. Abramoff. Splat feature classification with application to retinal hemorrhage detection in fundus images. *IEEE Transactions on Medical Imaging*, 32(2):364–375, 2013.

[50] M. J. van Grinsven, Ginneken B Van, C. B. Hoyng, T Theelen, and C. I. Sanchez. Fast convolutional neural network training using selective data sampling: Application to hemorrhage detection in color fundus images. *IEEE Transactions on Medical Imaging*, 35(5):1273–1284, 2016.

[51] Andrea Vedaldi and Karel Lenc. Matconvnet: Convolutional neural networks for matlab. pages 689–692, 2014.

[52] Holly H. Vo and Abhishek Verma. New deep neural nets for fine-grained diabetic retinopathy recognition on hybrid color space. In *IEEE International Symposium on Multimedia*, pages 209–215, 2017.

[53] Zhe Wang, Yanxin Yin, Jianping Shi, Wei Fang, Hongsheng Li, and Xiaogang Wang. Zoom-in-net: Deep mining lesions for diabetic retinopathy detection. *arXiv preprint arXiv:1706.04372*, 2017.

[54] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. 8689:818–833, 2014.

[55] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. pages 487–495, 2014.