Deep Clustering For Screening Diabetic Retinopathy

Sangeet Jaiswal

Deep Clustering For Screening Diabetic Retinopathy

DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology
in
Computer Science

by

Sangeet Jaiswal

[Roll No: CS1709]

under the guidance of

Dr. Sushmita Mitra

Professor

Machine Intelligence Unit

Indian Statistical Institute

Kolkata-700108, India

July 2019

To my family and my guide

CERTIFICATE

This is to certify that the dissertation entitled **"Deep Clustering For Screening Diabetic Retinopathy"** submitted by **Sangeet Jaiswal** to Indian Statistical Institute, Kolkata, in partial fulfillment for the award of the degree of **Master of Technology in Computer Science** is a bonafide record of work carried out by him under my supervision and guidance. The dissertation has fulfilled all the requirements as per the regulations of this institute and, in my opinion, has reached the standard needed for submission.

Dr. Sushmita Mitra
Professor,
Machine Intelligence Unit,
Indian Statistical Institute,
Kolkata-700108, INDIA.

Acknowledgments

I would like to show my highest gratitude to my advisor, *Prof. Sushmita Mitra*, Machine Intelligence Unit, Indian Statistical Institute, Kolkata, for her guidance and continuous support and encouragement. She has taught me how to do good research, and motivated me with great insights and innovative ideas.

I would also like to thank *Subhashis Banerjee*, PhD Scholar, Indian Statistical Institute, Kolkata, for his valuable suggestions and discussions.

My deepest thanks to all the teachers of Indian Statistical Institute, for their valuable suggestions and discussions which added an important dimension to my research work.

Finally, I am very much thankful to my parents and family for their everlasting supports.

Last but not the least, I would like to thank all of my friends for their help and support. I thank all those, whom I have missed out from the above list.

Sangeet Jaiswal
Indian Statistical Institute
Kolkata - 700108, India.

Abstract

Deep neural networks have been investigated in learning latent representations of medical images, yet most of the studies limit their approach using supervised convolutional neural network (CNN), which usually rely heavily on a large scale annotated dataset for training. To learn image representations with less supervision involved, we propose a deep clustering algorithm for learning latent representations of medical images. In this work, we present Deep clustering method that jointly learns the parameters of a neural network and the cluster assignments of the resulting features. We iteratively groups the features with a standard clustering algorithm, k-means and uses the subsequent assignments as a supervision to update the weights of the network. We evaluated the learned image representations on a task of classification using a publicly available diabetic retinopathy fundus image dataset. The experimental results show that our proposed method is close to the state-of-the-art supervised ensemble CNN.

List of Figures

List of Tables

Contents

Chapter 1

Diabetic Retinopathy

1.1 Introduction

Diabetic retinopathy, a chronic, progressive eye diseases, has turned out to be one of the most common cause of vision impairment and blindness especially for working ages in the world today [10]. It is estimated to affect over 93 million people. The US Center for Disease Control and Prevention estimated that 29.1 million people in the US have diabetes and the World Health Organization estimates that 347 million people have the disease worldwide*. It usually affects people who have had diabetes for significant number of years [17]. If it is left untreated it could increase the risk of blindness. Diabetic retinopathy can cause blood vessels in the retina to leak fluid or hemorrhage, distorting vision. In its most advanced stage, new abnormal blood vessels proliferate on the surface of the retina, which can lead to scarring and cell loss in the retina.

The progress of DR can be categorized into four stages:

- **Mild nonproliferation retinopathy**: Small areas of balloon-like swelling in the retina's tiny blood vessels, called micro aneurysms, occur at this earliest stage of the disease.

- **Moderate nonproliferation retinopathy**: As the disease progresses, blood vessels that nourish the retina may swell and distort. They may also lose their ability to transport blood.

*https://www.kaggle.com/c/diabetic-retinopathy-detection/overview/description

- **Severe nonproliferation retinopathy**: Many more blood vessels are blocked, depriving blood supply to areas of the retina. These areas secrete growth factors that signal the retina to grow new blood vessels.

- **Proliferative diabetic retinopathy**: At this advanced stage, growth factors secreted by the retina trigger the proliferation of the new blood vessels, which grow along the inside surface of the retina and into the vitreous gel, the fluid that fills the eye. The new blood vessels are fragile, which makes them more likely to leak and bleed. Accompanying scar tissue can contract and cause retinal detachment-the pulling away of the retina from underlying tissue, like wallpaper peeling away from a wall. Retinal detachment can lead to permanent vision loss.

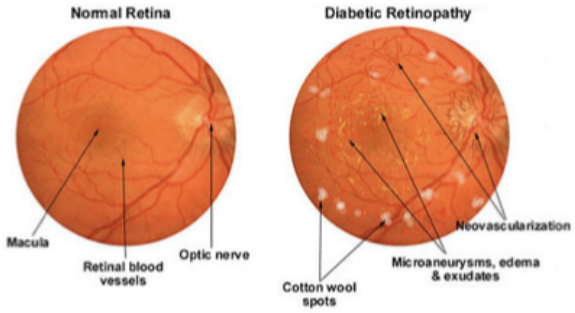

Figure 1.1: Fundus image of a healthy eye (left) and proliferate retinopathy (right).

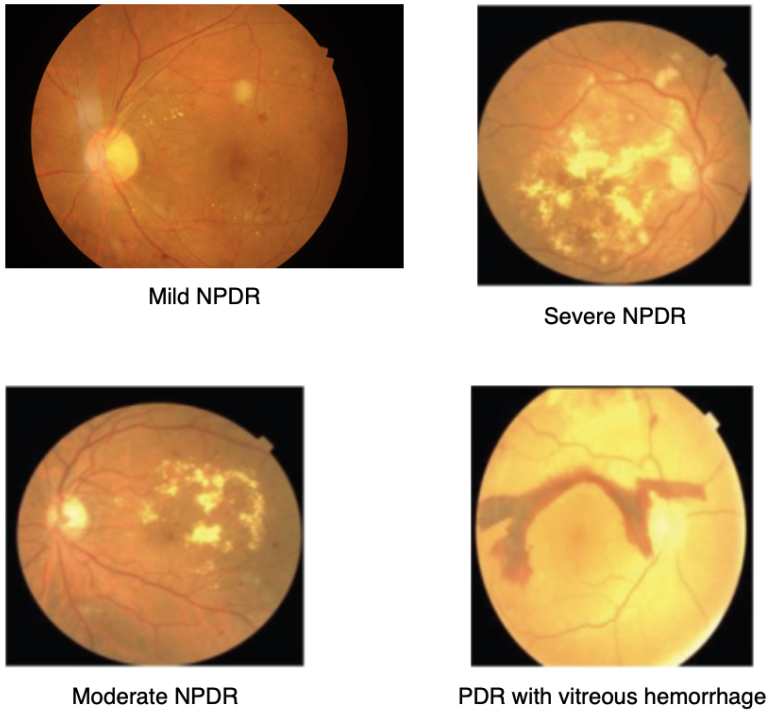

Figure 1.2: Types of Diabetic retinopathy.

1.2 Problem Statement

At present, retinopathy detection system is accomplished by involving a well-trained physician manually detecting vascular abnormalities and structural changes of retina in the retinal fundus images, which are then taken by dilating the retina using vasodilating agent. Due to the manual nature of DR screening methods, however, highly inconsistent results are found from different readers. Therefore there is a need for a comprehensive and automated method of DR screening. Also, accurate detection of DR at the early stage can greatly improve the intervention by clinician, which reduces the risk of vision loss. We propose to reduce the workload on medical specialists by automatically classifying fundus images using deep learning.

1.3 Thesis Outline

The rest of the thesis is organized as follows:

- **Chapter 2**: This chapter discussed the related work done for Diabetic retinopathy grading.
- **Chapter 3**: This chapter provides an outline of dataset statistics, preprocessing and augmentation techniques.
- **Chapter 4**: This chapter compares the performance of various CNN architecture with spatial pyramid pooling layer.
- **Chapter 5**: This chapter presents our Deep clustering based approach for Diabetic retinopathy grading.
- **Chapter 6**: This chapter concludes our analysis and outlines potential shortcomings and future work.

Chapter 2

Related Work

The two-step (i.e., feature extraction and prediction) automated DR detection approaches dominated the field of DR detection for many years. Earlier work using machine learning to diagnose diabetic retinopathy has used classifiers on top of manually designed feature detectors to measure the blood vessels and the optic disc, and to count the presence of abnormalities such as red lesions, microaneurysms, hard exudates, hemorrhages and cotton wool spots. Roychowdhury et al. [18] developed a 3-stage hierarchical architecture using AdaBoost to select the 30 top features out of 78. The 1st stage enhances the image and detects the optic disc, vasculature and red lesions. Stage 2 classifies the lesions as either cotton wool spots, hard exudates, microaneurysms or hemorrhages. Stage 3 counts the features and assigns one of 5 class labels. They achieved 100% sensitivity, 53.16% specificity, and 0.904 AUC.However, these types of approaches have the disadvantage of utilizing limited number of features.

In a similar approach, Acharya et al. 2009 [2] used 331 fundus images for analysis. Five groups were identified: normal retina, mild non-proliferative diabetic retinopathy, moderate non-proliferative diabetic retinopathy, severe non-proliferative diabetic retinopathy, and proliferative diabetic retinopathy. Four salient features blood vessels, microaneurysms, exudates, and haemorrhages were extracted from the raw images using image-processing techniques and fed to the SVM for classification. They achieve an accuracy of 86%, sensitivity of 82% and specificity of 86%. This is in comparison to the group's earlier work [34] that focused on using the area and perimeter of the RGB components of the blood vessels and a neural network to achieve an accuracy of 84%, sensitivity of 92% and specificity of 100%. They also investigated a simpler

approach [1] that did not use retinopathy-specific features. Using higher order spectra (HOS) features they achieved an accuracy of 82%, sensitivity of 83% and specificity of 89%. Nayak et al. [25] performed 3-class classification Image preprocessing, morphological processing techniques and texture analysis methods are applied on the fundus images to detect the features such as area of hard exudates, area of the blood vessels and the contrast to achieve an accuracy of 94%, sensitivity of 90% and specificity of 100%.

This type of approaches are not as effective as the recent deep learning approaches, such as [17, 2]. All these deep learning approaches adopted the standard architecture like GoogLeNet, ResNet, Vgg to build their CNN, based on the experimental results these deep learning approaches significantly outperform the traditional two-step approaches. Pratt et al [28] use a CNN with data augmentation to classify 5 classes of retinopathy on the kaggle dataset of 80000 images. They achieved a sensitivity score of 95% on the dataset with accuracy of 75% on 5000 validation images. Colas et al. [7] described the work of start-up company DreamUp Vision in classifying the same dataset. They achieved an area under the receiver operating characteristics curve(AUROC) of 0.946 with 96.2% sensitivity. In [3] CNN based method was employed to detect microaneurysms a DR stage grading.

Ensemble of CNN was employed to simultaneously detect DR and macular edema by Kori et al.[19]. They employed a variant of ResNet [12] and densely connected networks [14]. To make the model prediction more interpretable, a visual map was generated by Torre et al. [31] using CNN model, which can be used to detect lesion in the tested retinal fundus images. A similar approach was used in [33] along with generation of regression activation map (RAM).

Chapter 3

DR Dataset Description & Pre-processing

3.1 Dataset Statistics

The dataset that we have used is downloaded from the kaggle website.*. The training & test dataset contains 35126 & 53576 high resolution images respectively under a variety of imaging conditions. These retina images were obtained from a group of subjects, and for each subject two images were obtained for left and right eyes, respectively. The labels were provided by clinicians who rated the presence of diabetic retinopathy in each image by a scale of "0, 1, 2, 3, 4", which represent "no DR", "mild", "moderate", "severe", "proliferative DR" respectively. As mentioned in the description of the dataset, the images in the dataset comes from different models and types of camera, which can affect the visual appearance of left vs right.The samples images are shown in Fig 3.1 Also, the dataset doesn't have the equal distribution among the 5 scales. As one can expect, normal data with label "0" is the biggest class in the whole dataset, while "poliferative DR" data is smallest class. Fig 3.2 shows counts of images for different scales in the training dataset.

*https://www.kaggle.com/c/diabetic-retinopathy-detection/

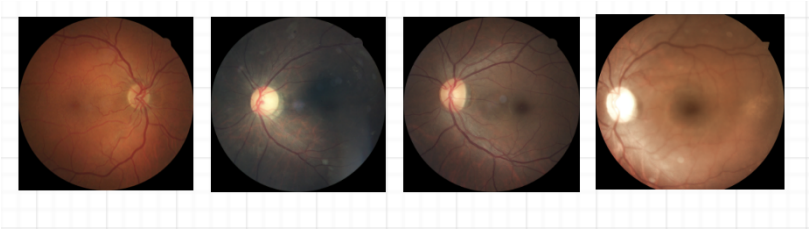

Figure 3.1: Sample images of color retina images dataset.

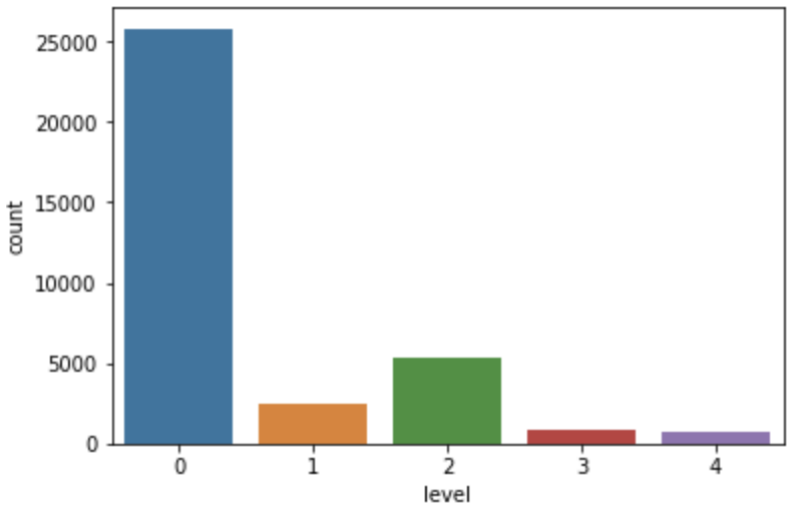

Figure 3.2: Count of images for different scales in the training dataset.

3.2 Data Pre-processing and Augmentation

We normalized our data set by its pixel statistics mean_channel = [0.4568, 0.3276, 0.2462], standard-deviation_channel = [0.2784, 0.2013, 0.1687]. We have also re-scaled the images to 512 x 512 dimensions. The performance of deep neural network is strongly correlated with the size of available training data. Although Kaggle Eye-PACS dataset is largest for retinopathy detection consisting of around 88,702 images, We are to use a very small fraction of it for training containing images for diseases severity grading task with imbalanced classes, requiring us to heavily augment our training data to obtain a model which is stable and not over fitted.

- **Rotation:** A Random rotation between -max_rotate and max_rotate degrees is applied with probability p.
- **Flip:** The image is flipped vertically or horizontal flip is applied.

- **ZoomCrop:** Randomly zoom and crop.
- **RandomResizeCrop:** Randomly resize and crop the image. This transform is an implementation of the main approach used for nearly all wining imagenet entries since 2013, based on Andrew Howard's **Some Improvements on Deep Convolutional Neural Network Based Image Classification**.

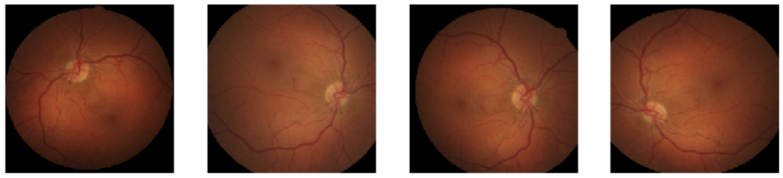

Figure 3.3: Sample images after rotation operation.

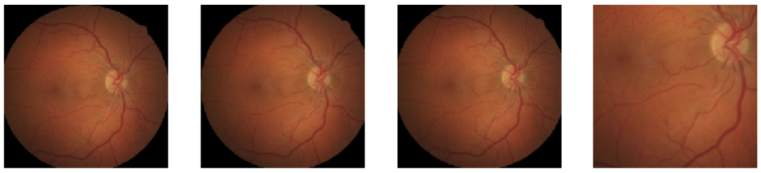

Figure 3.4: Sample images after zoomcrop operation.

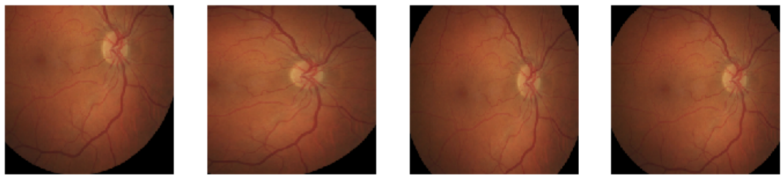

Figure 3.5: Sample images after randomresizecrop operation.

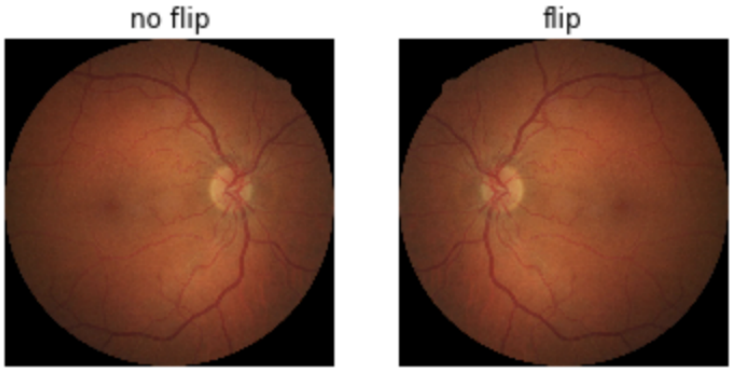

Figure 3.6: Sample images after flip operation.

Chapter 4

ResNet with Class Activation Maps for DR Grading

4.1 Motivation

We generally perceive that "*the deeper the better*" when it comes to convolutional neural network. This makes sense, since the models should be more capable (their flexibility to adapt to any space increases) because they have a bigger parameter space to explore. However, it has been noticed that after some depth, the performance degrades. This was one of the bottlenecks of earlier networks. ResNet gives us the residual learning framework to ease the training of networks that are substantially deeper than those used previously. It has won the 1st place on the ILSVRC 2015 for classification & localization task [13]. This motivates us to use ResNet as a starting point for Diabetic retinopathy grading. The depth of representation is very important for many visual recognition tasks and we have used this deep representation to generate class activation maps to indicate the discriminating image regions used by the CNN to identify that category [36].

4.2 Preliminaries

In this section we will discuss about the ResNet34 architecture, the idea of Class activation maps and spatial pyramid pooling (SPP) layer. We will see how SPP layer helps us training the network with progressive resizing of images, which acts as a

regularizer and reduces the overfit in the model.

4.2.1 Architecture

Let us consider H(x) as the underlying mapping to be fit by a few stacked layers(not necessarily the entire net), with x denoting the inputs to the first of these layers. If one hypothesizes that multiple nonlinear layers can asymptotically approximate complicated functions*, then it is equivalent to hypothesize that they can asymptotically approximate the residual functions, i.e., H(x) - x (assuming input and output are of the same dimensions). So rather than expect stacked layers to approximate H(x), we explicitly let these layers approximate a residual function F(x):= H(x) - x. The original function thus becomes F(x) + x. Although both forms should be able to asymptotically approximate the desired functions(as hypothesized), the ease of learning might be different.

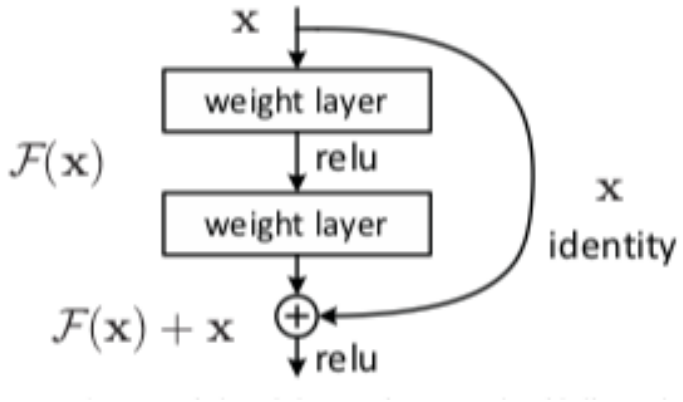

Figure 4.1: Residual learning: a building block [13]

Based on the plain network 4.2 we insert shortcut connections 4.2 which turn the network into its counterpart residual version. The identity shortcuts can be directly used when the input and output are of the same dimensions.

In Fig 4.2 we can see that the ResNet consists on one convolution and pooling step(on orange) followed by 4 layers of similar behavior. Each of the layers follow the same pattern. They perform 3 x 3 convolution with fixed feature map dimensions (F) [64, 128, 256, 512] respectively, by passing the input every 2 convolutions. Furthermore, the width (W) and Height (H) dimensions remain constant during the entire layer.

*still a topic of debate[24]

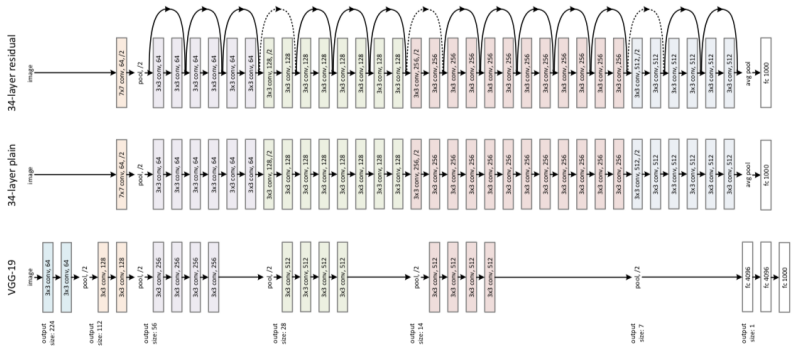

Figure 4.2: Bottom: the VGG-19 model [29] (19.6 billion FLOPs) as a reference. Middle: a plain network with 34 parameter layers (3.6 billion FLOPs). Top: a residual network with 34 parameter layers (3.6 billion FLOPs) [13]

The dotted line is there, precisely because there has been a reduction in the dimension of the input volume. This reduction between layers is achieved by an increase on the stride from 1 to 2, at the first convolution of each layer. Instead of by a pooling operation, which we are used to see as down samplers.

4.2.2 Spatial Pyramid Pooling

Existing deep convolutional neural networks(CNNs) require a fixed-size (e.g., 224 x 224) input image. This requirement is "artificial" and may reduce the recognition accuracy for the images or sub-images of an arbitrary size/scale. Spatial pyramid pooling strategy will eliminate the above requirement. The new network architecture can generate a fixed-length representation regardless of image size/scale. [12]

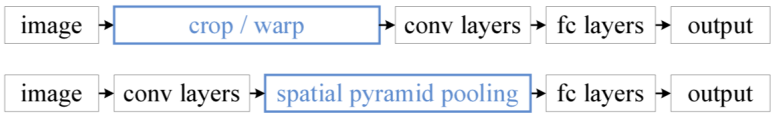

Figure 4.3: Top: a conventional CNN. Bottom: spatial pyramid pooling network structure [12].

A CNN mainly consists of two parts: Convolutional layers, and fully-connected layers that follow. The convolutional layers operate in a sliding-window manner and output feature maps which represent the spatial arrangement of the activations. Convolutional layers do not require a fixed image size and can generate feature maps of any

sizes. On the other hand, the fully-connected layers need to have a fixed-size/length input by their definition. Hence, the fixed-size constraint comes only from the fully-connected layers.

The convolutional layers accept arbitrary input sizes, but they produce outputs of variable sizes. The classifier or fully-connected layers requires fixed-length vectors. To adopt the deep network for images of arbitrary sizes, we replace the last pooling layer with *spatial pyramid pooling layer*.

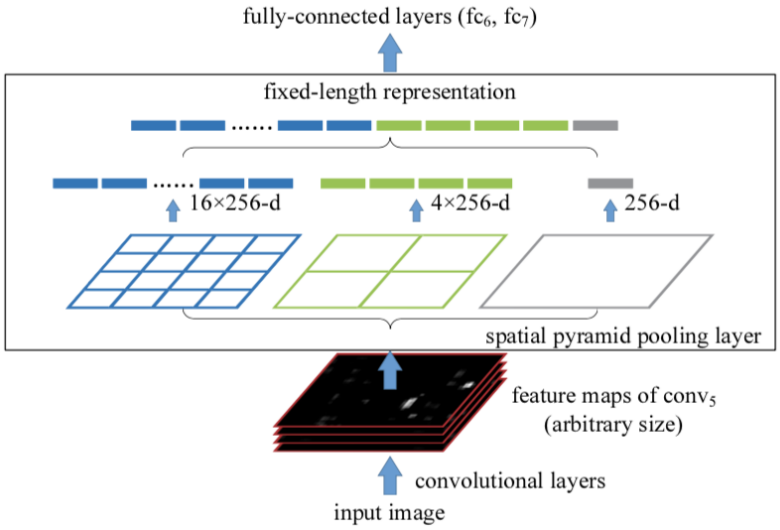

Figure 4.4: **A network structure with a spatial pyramid pooling layer**. Here 256 is a filter number of the conv_5 layer, and conv_5 is the last convolutional layer [12]

Consider the feature maps after conv_5 that have a size of a x a. With a pyramid level of n x n bins, we implement this pooling level as sliding window pooling, where the window size win $=\lceil$ a/n $\rceil$ and stride str $=\lfloor$ a/n $\rfloor$ with $\lceil$. $\rceil$ and $\lfloor$. $\rfloor$ denoting ceiling and floor operations. With an l-level pyramid, we implemented l such layers. The next fully-connected layer (fc_6) will concatenate the l outputs.

In our implementation we have used adaptive average pooing and adaptive max pooling with output size 1.

```
(1): Sequential(
  (0): AdaptiveConcatPool2d(
    (ap): AdaptiveAvgPool2d(output_size=1)
    (mp): AdaptiveMaxPool2d(output_size=1)
  )
```

Figure 4.5: AdaptiveConcatPool2d.

4.2.3 Class Activation Maps

Global average pooling layer proposed in [4], shed light on how it explicitly enables the convolutional neural network (CNN) to have remarkable localization ability despite being trained on image-level labels. While this technique proposed as a means for regularizing training,we find that it is actually builds a generic localizable deep representation that exposes the implicit attention of CNNs on an image. CNNs have the ability to identify exactly which regions of an image are being used for discrimination.

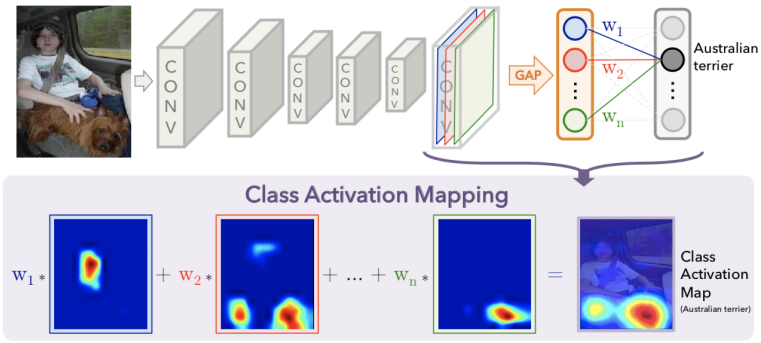

Figure 4.6: Class Activation Mapping: the predicted class score is mapped back to the previous convolutional layer to generate the class activation maps (CAMs). The CAM highlights the class-specific discriminative regions [36]

.

As shown in Fig. 4.6, global average pooling outputs the spatial average of the feature map of each unit at the last convolutional layers. A weighted sum of these values is used to generate the final output. Similary, we compute a weighted sum of the feature maps of the last convolutional layers to obtain our class activation maps. For a given image, let $f_k(x, y)$ represent the activation of unit k in the last convolutional layer at spatial location (x, y). Then, for unit k, the result of performing global average pooling, F^k is $\sum_{x,y} f_k(x, y)$. Thus, for a given class c, the input to the softmax, S_c

is $\sum_k w_k^c F_k$ where w_k^c is the weight corresponding to class c for unit k. Finally the output of the softmax for clss c, P_c is given by $softmax(S_c)$. Here we ignore the bias term: we explicitly set the input bias of the softmax to 0 as it has little to no impact on the classification performance. By plugging $F_k = \sum_{x,y} f_k(x, y)$ into the class score, S_c, we obtain

$$S_c = \sum_k w_k^c \sum_{x,y} f_k(x, y) = \sum_{x,y} \sum_k w_k^c f_k(x, y)$$

.

We define M_c as the class activation map for class c, where each spatial element is given by

$$M_c(x, y) = \sum_k w_k^c f_k(x, y)$$

4.3 Implementation Details

We have added the Adaptive concatenate 2D pooling layer just before the fully connected layer in our architecture which will generate a fixed sized representation of the input image irrespective of its size. We first train the network loaded with pretrained weights of imagenet dataset with 224 x 224 resized images. Now we will do transfer Learning using the pretrained weights of this network and further fine tune the network with 448 x 448 image size. As far as CNN is concerned, we are presenting it with a fresh dataset. Meaning that, even if we were overfitting while training with 224 x 224 size images, we will regularize our model with bigger size images. We will repeat the process with 896 x 896 images, though we didn't get much improved performance with this size. We trained our network for 100 epochs on Nvidia Tesla P100 for 3 days.

4.4 Results

We have compared the results from various CNN architectures with spatial pyramid pooling layer in Table 4.1. We have found that ResNet34 with SPP performed better on test dataset. We have also generated class activation maps (CAM) with ResNet34 to visualize those regions of image which are being used for discrimination. Through class activation maps Fig 4.9 we have understood that network is good at detecting

cotton wool spots, exudates and hemorrhages but unable to detect microaneurysm most of the times. Most of the predictions for class 1, 2 are confused with its neighbourhood class as shown in Fig 4.8. As we can infer from the loss plot Fig 4.7 training network with progressive resizing technique causes spikes in the curve which represents change in the image size. We have found that training the network with image size more than 512 doesn't help much in converging the loss function.

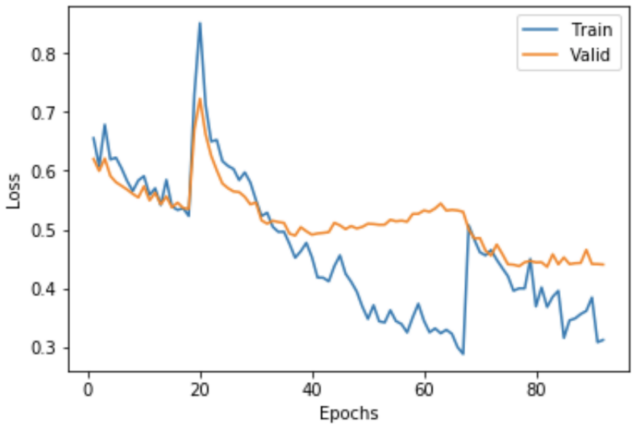

Figure 4.7: Loss Plot:Training and validation loss plot of ResNet34+SPP over 100 epochs.

Model	Quadratic Weighted Kappa
VGG16 + SPP	0.7280
ResNet34	0.7664
Densenet121 + SPP	0.7714
ResNet34 + SPP	0.8022

Table 4.1: Performance of various models with increasing order of kappa score.

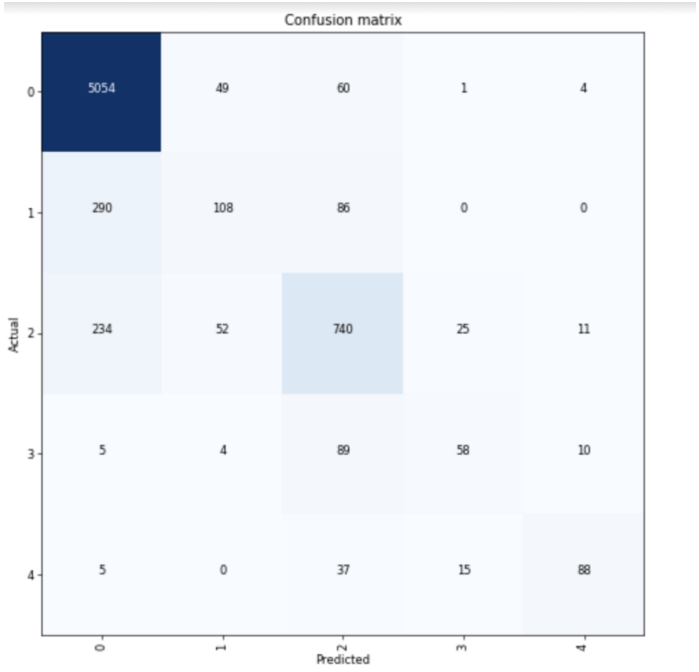

Figure 4.8: Confusion Matrix: Actual vs predicted class labels for validation dataset.

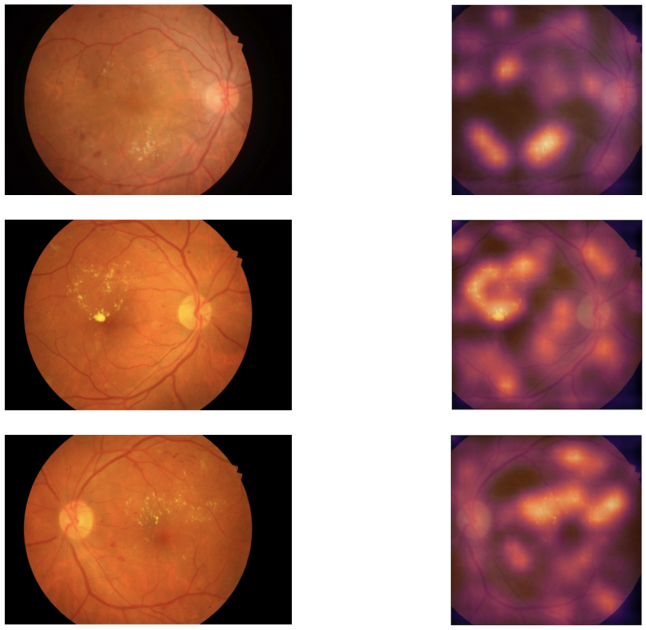

Figure 4.9: Class Activation Maps: Examples of highlighted image regions for most correctly predicted class.

Chapter 5

Deep Clustering For Screening Diabetic Retinopathy

5.1 Introduction to Deep Clustering

Clustering is a class of unsupervised learning methods that has been extensively applied and studied in computer vision. Little work has been done to adapt it to the end-to-end training of visual features on the large scale datasets. In this work, we propose clustering based end-to-end training of Deep neural network without supervision. This method learns simultaneously the parameters of the neural network and the cluster assignment of the resulting features. We showed that we can obtain general purpose discriminating features with clustering framework. Our approach does the weights updating of the convolutional neural network by predicting the cluster assignments. We used k-means, but other clustering approaches can also be used [6]. Unlike supervised methods, clustering has the advantage of requiring little domain knowledge and no supervision.

Unsupervised learning has been widely studied in the Machine Learning [11]. Algorithms for clustering, dimentionality reduction or density estimation are regularly used in the computer vision applications [32]. For example, the "bag of features" model uses clustering on handcrafted local descriptors to produce good image-level features [8]. A key reason for their success is that they can be applied on any specific domain or dataset, like Medical or satellite images, where annotations are not always available in quantity. Our approach, summarized in Fig 5.1, consists in alternating

between clustering of the image descriptor and updating the weights of the convolution network by predicting the cluster assignments. We focus our study on k-means, but other clustering approaches can be used, like Power iteration Clustering (PIC). Unlike standard supervised methods, clustering has the advantage of requiring little domain knowledge and no specific signal from the inputs [35].

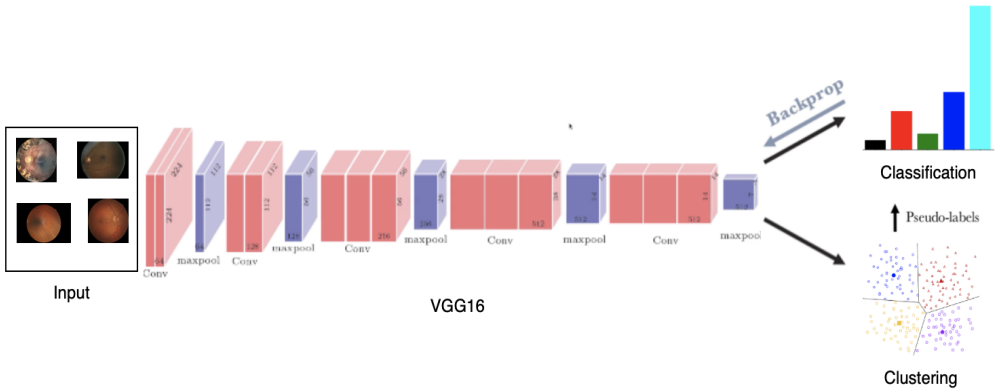

Figure 5.1: Illustration of the proposed method: We iteratively cluster deep features and use the cluster assignment as pseduo-labels to learn the parameters of the convnet [5].

5.2 Method

5.2.1 Unsupervised learning by clustering

Let us denote C_θ the convolution mapping, where θ is the set of convolution parameters. By applying this mapping to an image we will get image features as vector. Given a training set of images, we want to find a parameter θ^* such that the mapping C^*_θ represents good general-purpose features. These parameters are generally learned with supervised data where each image has a corresponding label vector. On top of these features $C_\theta(x_n)$ a parameterized classifier P_γ predicts the true labels. The parameters γ of the classifier and the parameter θ of the convolution mapping are then jointly learned by optimizing the following expression:

$$\min_{\theta,\gamma} \frac{1}{N} \sum_{n=1}^{N} l(P_\gamma(C_\theta(x_n)), y_n)$$

Where l is the loss function. This loss function is minimized using mini-batch stochastic gradient descent [21] and back propagation to compute the gradient [20].

We will cluster the output of the convolution block and use the cluster assignments as "pseudo-labels" to optimize equation 5.2.1. This approach iteratively learns the features and group them. Cluster assignment y_n is jointly learned by d x k centroid matrix C_M for each image n by solving the following optimization problem:

$$\begin{aligned} \min_{C_M \in R^{d \times k}} \quad & \frac{1}{N} \sum_{n=1}^{N} \min_{y_n \in \{0,1\}^k} \quad ||C_\theta(x_n) - C_M y_n||^2 \\ \text{such that} \quad & y_n^T 1_k = 1 \end{aligned}$$

Solving above problem gives a set of optimal assignments y* which we will use as a pseudo-labels.

5.2.2 Algorithm

Algorithm 1 Deep Clustering

Input: Dataset X = $\{x_i\}_{i=1}^N$
Output: Class label y_i^* of $x_i \in X$

Randomly initialize θ, γ
for *epoch* $= 1, \ldots, K$ **do**
  for *batch* $= 1, \ldots,$ *Number of Images* **do**
    $B \leftarrow$ *random samples* $\{x_i\}_{i=1}^b$ *of b images*
    $\{f_i\}_{i=1}^b \leftarrow C_\theta(B)$
  end for
  $\{y_i^*\}_{i=1}^N \leftarrow \min_{C_M \in \mathbb{R}^{d \times K}} \frac{1}{N} \sum_{n=1}^{N} \min_{y_n \in \{0,1\}^K} \|f_n - C_M y_n\|^2$
  $X' \leftarrow \{x_i, y_i^*\}_{i=1}^N$
  for *batch* $= 1, \ldots,$ *Number of images* **do**
    $\min_{\theta,\gamma} \frac{1}{b} \sum_{n=1}^{b} l(P_\gamma(C_\theta(x_n)), y_n^*)$
  end for
end for

5.2.3 Implementation details

We have used standard VGG16 with batch normalization [15] architecture. It consists of five convolutional blocks with 64, 128, 256, 512, 512 filters and three fully connected layers. Unsupervised methods often do not work directly on color and different strategies have been considered as alternative [26]. We apply a fixed linear transformation based on Sobel filters to remove color and increase local contrast [27].

We cluster the resized images features and perform data augmentation 3.2 when training the network. This enforce invariance to data augmentation which is useful for feature learning [9]. The network is trained with dropout [30], a constant step size, an l_2 penalization of the weights θ and a momentum of 0.9. Each mini batch contains 32 images. For the clustering, features are PCA-reduced to 256 dimensions, whitened and l_2 - normalized. We use the k-means implementation of Johnson *et al* [16]. We train the deep cluster model for 150 epochs, which takes 6 days on Nvidia Tesla P100 for VGG16 with batch normalization. For classification task it takes around 2 days on same GPU for 90 epochs.

5.3 Results

The *quadratic weighted kappa*, the state-of-the-art performance matrix for multi class classification and suggested evaluation matrix for DR*. We have compared the performance of various CNN architectures and found that our deep clustering based model performs better Table 5.1. We achieved 0.8105 quadratic weighted kappa on test dataset of kaggle. If we compare the confusion matrix of our previous best model we find that our deep clustering model has performed better in detecting early stages of DR which is a major aspect of this challenge.

*https://www.kaggle.com/c/diabetic-retinopathy-detection/overview/evaluation

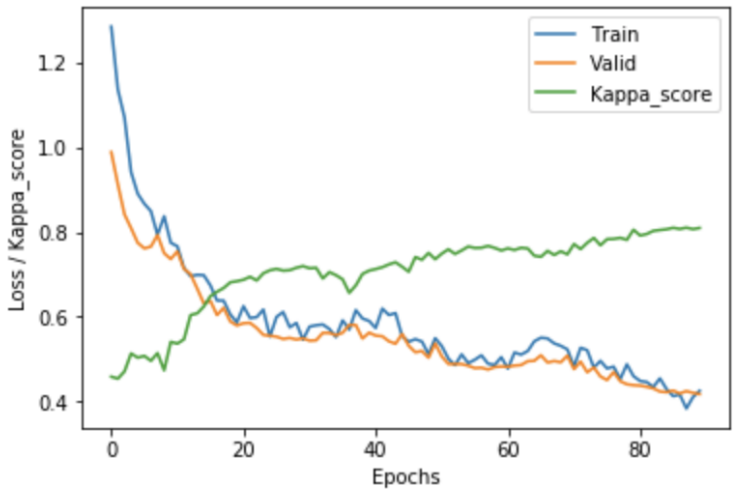

Figure 5.2: Loss Plot: Training and validation loss plot along with kappa score.

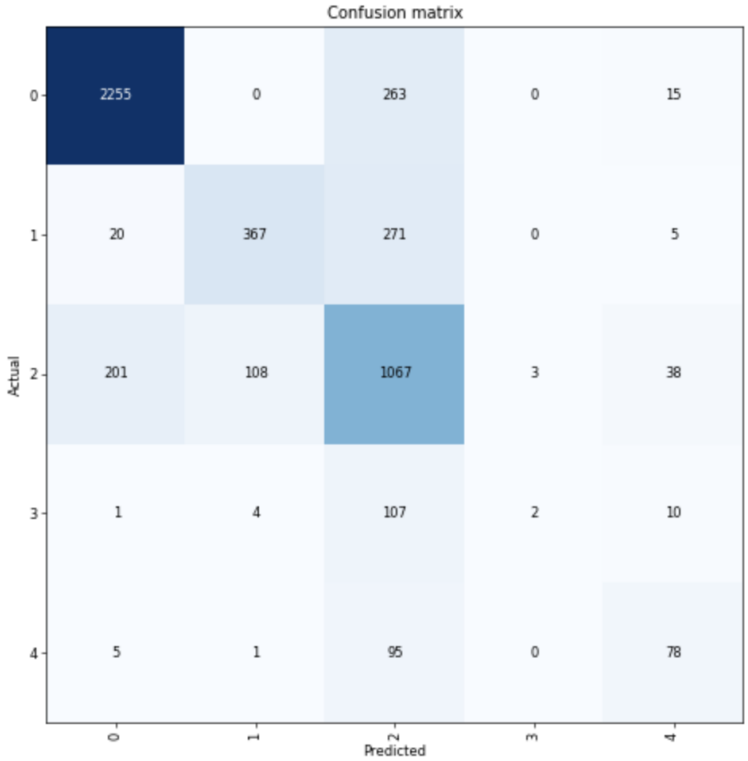

Figure 5.3: Confusion Matrix: Actual vs predicted class labels for validation dataset.

Model	**Quadratic Weighted Kappa**
VGG16 + SPP	0.7280
ResNet34	0.7664
Densenet121 + SPP	0.7714
ResNet34 + SPP	0.8022
VGG16 with deep clustering	0.8105

Table 5.1: Performance of various models with increasing order of kappa score.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

We propose a scalable clustering approach for the unsupervised learning of convnet. It iterates between clustering with k-means the features produced by the convnet and updating its weights by predicting the cluster assignments as pseudo-labels in a discriminative loss. We trained our network on largest publicly available Diabetic retinopathy data provided by EyePACS. It achieves results close to state-of-the-art. Our approach makes little assumption about the inputs, and does not require much domain specific knowledge, making it a good candidate to learn deep representations specific to domains where annotations are scarce.

One of the shortcoming of this model is its training time, it almost took 8 days to perform classification task. Which makes hyper parameter tuning very difficult.

6.2 Future Work

- We can perform morphological image pre-processing like CLAHE(Contrast Limited Adaptive Histogram Equalization) for making uniform intensity variation across image.
- We can consider Power Iteration Clustering (PIC) as an alternative clustering method [22].
- We can consider other networks like ResNet, DenseNet which has deep repre-

sentation of input features to cluster.

- As the misclassification is between neighbouring classes, instead of using PCA before the clustering we can embed the image features in higher dimensions and then applying kernel PCA before clustering. This can lead to better separability of the classes. The weights learn by the network will produce better general purpose features.

References

[1] Acharya, R., Chua, C.K., Ng, E., Yu, W., Chee, C.: Application of higher order spectra for the identification of diabetes retinopathy stages. Journal of Medical Systems 32(6), 481–488 (2008)

[2] Acharya, U.R., Lim, C.M., Ng, E.Y.K., Chee, C., Tamura, T.: Computer-based detection of diabetes retinopathy stages using digital fundus images. Proceedings of the institution of mechanical engineers, part H: journal of engineering in medicine 223(5), 545–553 (2009)

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

[4] Bazzani, L., Bergamo, A., Anguelov, D., Torresani, L.: Self-taught object localization with deep networks. In: 2016 IEEE winter conference on applications of computer vision (WACV). pp. 1–9. IEEE (2016)

[5] Caron, M., Bojanowski, P., Joulin, A., Douze, M.: Deep clustering for unsupervised learning of visual features. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 132–149 (2018)

[6] Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence 40(4), 834–848 (2017)

[7] Colas, E., Besse, A., Orgogozo, A., Schmauch, B., Meric, N., Besse, E.: Deep learning approach for diabetic retinopathy screening. Acta Ophthalmologica 94 (2016)

[8] Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Workshop on statistical learning in computer vision, ECCV. vol. 1, pp. 1–2. Prague (2004)

[9] Dosovitskiy, A., Springenberg, J.T., Riedmiller, M., Brox, T.: Discriminative unsupervised feature learning with convolutional neural networks. In: Advances in neural information processing systems. pp. 766–774 (2014)

[10] Engelgau, M.M., Geiss, L.S., Saaddine, J.B., Boyle, J.P., Benjamin, S.M., Gregg, E.W., Tierney, E.F., Rios-Burrows, N., Mokdad, A.H., Ford, E.S., et al.: The evolving diabetes burden in the united states. Annals of internal medicine 140(11), 945–950 (2004)

[11] Friedman, J., Hastie, T., Tibshirani, R.: The elements of statistical learning, vol. 1. Springer series in statistics New York (2001)

[12] He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE transactions on pattern analysis and machine intelligence 37(9), 1904–1916 (2015)

[13] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)

[14] Huang, G., Liu, S., Van der Maaten, L., Weinberger, K.Q.: Condensenet: An efficient densenet using learned group convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2752–2761 (2018)

[15] Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)

[16] Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with gpus. arXiv preprint arXiv:1702.08734 (2017)

[17] Kertes, P.J., Johnson, T.M.: Evidence-based eye care. Lippincott Williams & Wilkins (2007)

[18] Keshabparhi, S., Koozekanani, D.: Dream: Diabetic retinopathy analysis using machine learning. IEEE J. Biomed. Health Informatics 18(5) (2014)

[19] Kori, A., Chennamsetty, S.S., Alex, V., et al.: Ensemble of convolutional neural networks for automatic grading of diabetic retinopathy and macular edema. arXiv preprint arXiv:1809.04228 (2018)

[20] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11), 2278–2324 (1998)

[21] LeCun, Y.A., Bottou, L., Orr, G.B., Müller, K.R.: Efficient backprop. In: Neural networks: Tricks of the trade, pp. 9–48. Springer (2012)

[22] Lin, F., Cohen, W.W.: Power iteration clustering (2010)

[23] Milligan, D.E., Jacox, M.E.: Matrix-isolation study of the infrared and ultraviolet spectra of the free radical cnn. The Journal of Chemical Physics 44(8), 2850–2856 (1966)

[24] Montufar, G.F., Pascanu, R., Cho, K., Bengio, Y.: On the number of linear regions of deep neural networks. In: Advances in neural information processing systems. pp. 2924–2932 (2014)

[25] Nayak, J., Bhat, P.S., Acharya, R., Lim, C.M., Kagathi, M.: Automated identification of diabetic retinopathy stages using digital fundus images. Journal of medical systems 32(2), 107–115 (2008)

[26] Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: European Conference on Computer Vision. pp. 69–84. Springer (2016)

[27] Paulin, M., Douze, M., Harchaoui, Z., Mairal, J., Perronin, F., Schmid, C.: Local convolutional features with unsupervised training for image retrieval. In: Proceedings of the IEEE international conference on computer vision. pp. 91–99 (2015)

[28] Pratt, H., Coenen, F., Broadbent, D.M., Harding, S.P., Zheng, Y.: Convolutional neural networks for diabetic retinopathy. Procedia Computer Science 90, 200–205 (2016)

[29] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

[30] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research 15(1), 1929–1958 (2014)

[31] de la Torre, J., Valls, A., Puig, D., Romero-Aroca, P.: Identification and visualization of the underlying independent causes of the diagnostic of diabetic retinopathy made by a deep learning classifier. arXiv preprint arXiv:1809.08567 (2018)

[32] Turk, M.A., Pentland, A.P.: Face recognition using eigenfaces. In: Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 586–591. IEEE (1991)

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

[34] Yun, W.L., Acharya, U.R., Venkatesh, Y.V., Chee, C., Min, L.C., Ng, E.Y.K.: Identification of different stages of diabetic retinopathy using retinal optical images. Information sciences 178(1), 106–121 (2008)

[35] Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: European conference on computer vision. pp. 649–666. Springer (2016)

[36] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2921–2929 (2016)