Automatic Diabetic Retinopathy Classification

Department of Biomedical Engineering, Universidad de los Andes, Bogotá, Colombia

María Alejandra Bravo Sarmiento

Advisor: Dr. Pablo A. Arbeláez E.

Undergraduate Thesis of Biomedical Engineering

November 2016

Abstract

Diabetic retinopathy (DR) is a disease in which the retina is damaged due to augmentation in the blood pressure of small vessels. DR is the major cause of blindness for diabetics. The main symptoms of DR are hemorrhages, exudates, and neovascularization in the retina. It has been shown that early diagnosis can play a major role in prevention of visual loss and blindness. This work proposes a computer based approach for the detection of DR in back-of-the-eye images based on the use of convolutional neural networks (CNNs). Our CNN uses VGG architecture to classify Back-of-the-eye Retinal Photographs (BRP) in 5 stages of DR. Our method combines several preprocessing images of BRP and a vessel segmentation image obtained using the DRIU [10] algorithm. Our best ACA score of obtained was of 50.5% for a 5-way classification.

Key words: *Diabetic retinopathy, convolutional neural networks, retinal vessel segmentation, Back-of-the-eye Retinal Photography.*

1. Introduction

Diabetic retinopathy (DR), or diabetic eye disease, affects most of the patients with diabetes type 1 and some with diabetes type 2. This disease is mainly caused by diabetic microangiopathy, damage of the small blood vessels that oxygenate the retina in the posterior part of the eye of a diabetic patient. The damage in the eye, is due to a metabolic disorder that increases the levels of glucose in a person's blood, causing the person to present high levels of blood pressure. High levels of blood pressure affect the circulatory system of the retina and the light sensitive lining at the back of the eye. This condition, known as DR, can lead to leaking blood and other fluids that cause swelling of retinal tissue and clouding of vision. Patients with DR develop dark spots in the field of vision, blurred vision, difficulty to see at night and some can get completely blind [2].

DR is classified into two principal stages. In the non-proliferative stage of the disease, symptoms are mild or non-existent. During this stage: (1) the blood vessels in the retina are weakened due to the accumulation of glucose and the increase in blood pressure, causing tiny bulges called microanuerysms to protrude from vessels' walls; (2) exudates and edemas become present because of the increase in the permeability of vessels; and (3) some hemorrhages can be detected. Figure 1 shows a back-of-the eye image exhibiting these symptoms. The second proliferative stage, is characterized by a neoformation of weak and fragile vessels in the retina and the crossing between veins and arteries causing wrong blood flow, combined with the symptoms already present during the non-proliferative stages. The need of oxygen causes new vessel formation to irrigate the retina [4]. In some severe cases, DR can lead to diabetic macular oedema when the fluid reaches the macula. Macular oedema can be detected by the presence

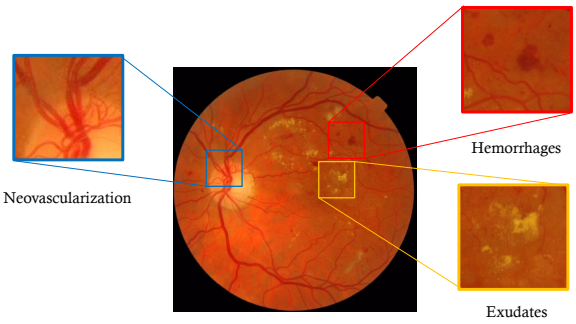

Figure 1. DR visible symptoms

of retinal thickening and swelling within one disc diameter of the center of the fovea [12].

DR is a serious public health problem, being one of the main causes of blindness worldwide and its prevalence is set to continue rising. Today, it is estimated that diabetes affects 2 to 6% of industrial countries' populations and 16% of adults over the age of 65 [3]. Of the global diabetic population, 40-50% could be afflicted by retinopathy [14]. An estimated of 50–65 new cases of blindness per 100,000 people happened every year because of DR [12].

During its history, DR has been categorized as incurable and unpreventable. Nowadays, new technology like Argón's laser and pars plana has opened the possibility of treatment, but early detection is essential for effectiveness [2]. Early detection of sight-threatening retinopathy allows laser therapy to be performed to prevent or delay visual loss, and may be used to encourage improvement in diabetic control. Unfortunately, there are almost no visual symptoms in the early stages of DR, making it difficult to diagnose in time. There exist several diagnostic methods for this disease such as fluorescein angiography, optical coherence tomography, and the most common and less expensive method used, the Back-of-the-eye Retinal Photography (BRF) or Digital Fondus Photography. During fluorescein angiography, a yellow fluorescein dye is introduced into the patient's bloodstream and then several retinal photographs are taken. For optical coherence tomography, light waves are used to capture images of tissues inside the eye that allow to measure the amount of retinal swelling. To take a BRF the pupil is dilatated and then a picture is taken with a high-resolution camera [12].

At present time, the detection of DR is a time-consuming and manual process that requires a trained clinician to examine and evaluate the diagnostic images. Diagnose can take several days depending on the doctor's availability and the number of patients. Experience has proved that there exists a need for an automatic method of DR screening. Recently, image processing and analysis has made grounds in detecting DR, but this is still a new field, with space for improvement in both processing time and accuracy. In previous work [1], we have tried to classify BRF into degrees of DR disease using feature extraction followed by different classifiers such as support vector machines, random forest, nearest neighbors, and convolutional neural networks (CNN) resulting in 43.3%, 39.1%, 39.1%, and 41.6% of average accuracy respectively.

2. Related Work

Initially, diabetic retinopathy detection methods worked extracting handcrafted features of interest in the image. The first approaches in this field aimed at detecting diabetic retinopathy with respect to the textural features of retinal images. For example, Yun *et al.* [17] proposed a method based on detecting and analyzing microaneurysms and hemorrhages. Nayak *et al.* [11] used a combination of hard exudates area and blood vessels by morphological techniques. Reza [13] developed a rule-based algorithm based on hard exudates and cotton wool spots. These pathological regions were segmented by using image processing techniques such as thresholding, morphological reconstruction, and boundary tracing.

Last year, thanks to the California Healthcare Foundation that created a challenge with available dataset, the development of automatic algorithms to classify and detect different degrees of DR grew rapidly. The groups that finished the challenge competition on top all used deep learning and CNNs for building a classification algorithm of DR. The dataset provided BRF images to compare the different algorithms.

The Ben Graham obtained the first place [7] under the name of Min Pooling with a score of 0.8496. This method involved SparseConvNet, a convolutional neural network whose input is the whole image. To train the network, it was necessary to do data augmentation: scaling, skewing and rotating the images. Min Pooling preprocessed the images doing a rescale to have images with eyes of the same radius (300 or 500 pixels), subtracted the local average color, and clipped the image to 90% of its size to remove boundary effects. The final competition model combined three different networks that included: two convolutional networks, using fractional max-pooling [6] involving layers of spacial pooling with fractional parameters, and the other using very deep convolutional neural networks [15]. Each of the three neural networks had between 2.7 to 6.1 millions of weights and implemented a 10% of dropout in the last four layers. Images were used several times to train and test the model. To give the final class Min

Pooling trained a random forest using the predicted scores and other correlated information such as the score of the other person's eye, the variance of the original image and the variance of the preprocessed image.

The team o_O earned second place with 0.8448 of score using a total of three neural network architectures. For this method, images were resized to 768x768, and augmented in number by applying translation, stretching, rotation, flipping and color augmentation. Colors were scaled to have zero mean and unit variance in each image. The algorithm used two types of neural networks with the raw images as their input. For each network, it was calculated three set of weights (best validation score, best kappa, final weight) and entered both right and left eyes. The final score was the average of the six blending outputs. It is important to note that this result wasn't a probability vector for each category, but a single number using regression, usually between 0 and 4, that was rounded to obtain the predicted classification. Since the database was greatly unbalanced, images were first sampled so that each class would be represented equally on average and, as the training progressed, they decreased sampling of rare classes. In order to train the final weights of the CNN, the algorithm was first trained with smaller networks, and used these weights to initialize the more complicated networks.

The third-place group was the Reformed Gamblers, with a score of 0.8394. This method combined the results of nine different convolution networks, including the woks of Benjamin Graham, and Karen Simonyan and Andrew Zisserman. For data augmentation, images were randomly cropped to 85-95% of the original, horizontally flipped, rotated between 0 and 360 degrees, and finally scaled to the desired model input size (between 385, and 767, depending on the network). For non-linearity of the networks, the algorithm used leaky rectifier unit. To facilitate training the algorithm started with pretrained weights and with very small learning rates, which were then increased after several epochs. The output of each network was a number between 0 and 4 and the algorithm used a simple linear model to combine the output of both eyes. Unlike o_O, for the final classification, they didn't simply round the output, but performed a grid search of possible cutoffs to maximize the kappa score [9].

All three groups agreed that neural networks were adequate to classify Diabetic Retinopathy. Likewise, they all used information from both eyes to make the predictions. Another important point was the size of the input image, both o_O and Reformed Gamblers tried inputs of different sizes, and agreed that there was a significant improvement as you increase the size as well as an increase in the processing time. This increase in the score might have being because the microaneurisms were only visible in large scale images, and disappeared in small scale.

3. CNNs for Diabetic Retinopathy Image Classification

3.1. Dataset

The mother dataset was obtained from Kaggle [1], where the California Healthcare Foundation issued a challenge to create an automatic program to detect diabetic retinopathy. This challenge disposes of a dataset of high-resolution retina images taken under a variety of imaging conditions, with 35126 training images and 53487 test images. For annotations, the dataset has the images classified in five groups (0, 1, 2, 3, 4), depending on the disease's severity. Level 0 contains images with no signs of retinopathy, whereas level 4's show advanced symptoms. The scale of each of the degrees of DR correspond to: no DR, mild, moderate, severe and proliferative, form 0 to 4 respectively. The training set is not balanced as shown in Figure 2, with 25810 level 0 images, 2443 for level 1, 5292 of level 2, 873 in level 3, and for 708 in level 4 [5]. In order to allow training and optimization of large capacity models, we split the data into training/annotated and validation. From this database, we reserved a balanced set of 1560 images for validation and the rest were used for training.

It is important to note that the database contained images with very poor quality (poor lighting and unfocused) and also some patients presented diseases different from DR.

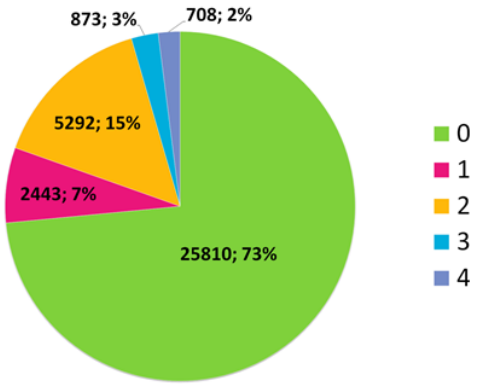

Figure 2. Unbalanced dataset class distribution. Class 0 is almost 73% of all images and class 4 is around 2%.

3.2. Proposed Method

A Neural Network Approach. We approach the DR classification problem by using CNNs. We start from the

[1] https://www.kaggle.com/c/diabetic-retinopathy-detection

VGG16 [15] network, originally designed for large-scale natural image classification into 1000 different classes. This architecture consists of thirteen convolutional layers coupled with Rectified Linear Unit (ReLU) activations, five max pooling layers that separate the network into six scales, and three Fully Connected (FC) layers (as in Fig. 3). For our task, the last fully connected layer at the end of the network was changed to have a 5-response vector that corresponds to each of the degrees of the disease. Our architecture included two dropout layers which made several connections not activated in some iterations of the training process, allowing the model to reduce overfitting. We started our network with the pretrained original VGG16 weights and during the training process all convolutional weights, except for the last ones in Conv5_3, were fixed and no back-propagation was made on them. All the FC layers were also finetuned during training. For the last layer, we used a softmax classifier which calculates the probability for an image of being in each of the five classes.

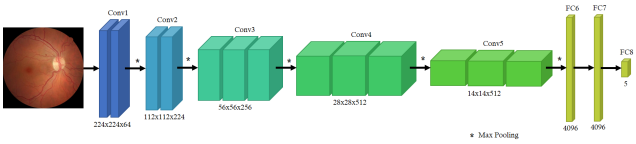

Figure 3. VGG16 fine-tuning architecture used as base network for our method.

To train the network we used a multinomial logistic loss function and implemented it in the same layer as the softmax classifier, Equation (1) shows the definition, where $c \in \{0, ..., 4\}$ the classes. The combination of the softmax layer and the log-loss is useful for numerical stability [16].

$$l(\mathbf{x}, c) = -log\frac{e^{x_c}}{\sum_{k=0}^{4} e^{x_k}} = -x_c + log\sum_{k=0}^{4} e^{x_k}. \quad (1)$$

For training the networks we used publicly available *Caffe* [8] framework, and a NVIDIA TITAN X GPU in which the average execution time for an image was of 0.1ms and the training time for an image to pass through the net was around 1ms, using the VGG-16 architecture.

DRIU and Color Join Architectures Approach. It is known that DR degree classification is mainly performed, between consecutive levels, by the neovascularization present in the retina. For this reason, one of the main characteristics specialists focus on when analyzing BRF is the retinal vascularization. This year Maninis *et al.* [10] developed a Deep Retinal Image Understanding (DRIU) algorithm capable of segmenting retinal vessels with high performance. For this reason, we implemented this algorithm and used it to obtain BRF with segmented

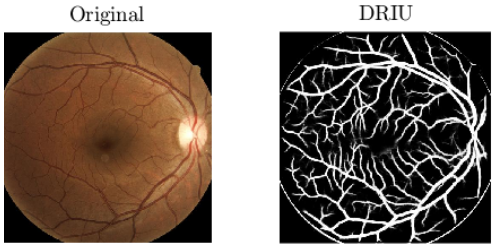

Figure 4. Vessels extraction using DRIU

vascularization. Once we obtained the vessel segmentation image (Fig. 4) we trained several CNN architectures that used both, color (RGB) and vascular (DRIU) information. By introducing the vascular information, as additional layers in the network architecture, we explored the relevance of vessels in automatic classification of DR and expected the method to improve its performance. To find the optimal scale and semantic information to join both images, we tested different architectures described in Sec. 4.4. For all architectures, we used as base CNN VGG.

3.3. Preprocessing

Images from the dataset varied in size between 2500x2000 and 4700x3100 pixels. Due to the number of images, we decided to reduce their size to facilitate handling. First cropped the images to remove the extra black background and resized them to 500x500 pixels. To determine which was the best input for the CNN we tried 5 different preprocessing images shown in Fig. 5:

1. Circle RGB images (rgb): For this set we used the whole image except for the extra black background, and resized it to 224x224.

2. Square images (square): For this set we used the greatest square image inscribed in the eye, and resized it to 224x224.

3. Circle color adjust images (color): For this set we used the same circle images of the first set and subtracted the local average color using a Gaussian filter. The intensities for each channel were centered at 127.

4. Color morphological adjust (morpho): For this set we used the same circle images of the first set and using morphological operations we removed color background to make images look similar. First, we did a closure and an opening of the image with a 60-pixels disk kernel. We subtracted from the original image an average of the morphological transformed images. After subtraction, we did a contrast adjustment using a sigmoid function.

5. Gray scale images (gray): For this set, we used the same circle images in gray scale. This experiment was done to know how important were colors for the classification algorithm.

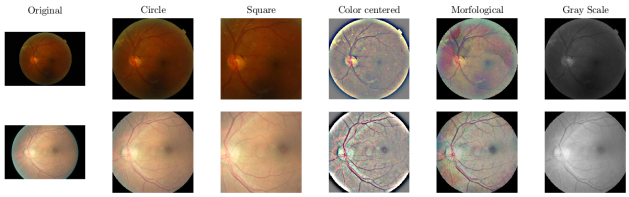

Figure 5. Image preprocessing. Columns correspond to: (1) original image; (2) circle cropped image; (3) square cropped image; (4) color adjust image using a Gaussian filter; (5) color adjust image using morphological operators; and (6) gray scale image.

4. Experimental Validation

4.1. Evaluation Methodology

To evaluate performance of the classification methods, we used a confusion matrix (CM). Each column of the matrix represents the ground truth labels, while the rows correspond to predictions. We count how many images of each class were classified in each category. An ideal model would give as result a confusion matrix with only nonzero entries in the diagonal. It is necessary to normalize the CM by dividing each entry by the total examples in the data base that belong to the predicted class. To have a perfect score, the confusion matrix should be the identity matrix. Finally, the mean of the diagonal gives us the Average Classification Accuracy (ACA), an overall measure of the classification. In the DR classification problem, we have 5 categories; therefore, an ACA above 0.2 shows the algorithm is better than guessing.

4.2. Data augmentation

Since the dataset was not balanced, and for training and testing the network it is better to have balanced set, it was decided to split the database as follows: 1560 images for validation with about 312 images per class; and 33566 images for training (with 25496, 2132, 4980, 561, 397 images per class, corresponding to classes 0 to 4). This split was made making sure both eyes of each subject in the database where in the same set. To balance the training set and to increase the number of images for training, we did data augmentation using rotations from 0 to 360 degrees, zoomming in and out from 0 to 20 pixels, and doing horizontal and vertical flips. This augmentation was done randomly for all images in the training set and, for the classes 1, 2, 3, and 4, images were repeated with different augmentation parameters. The resulting balanced training data set had 127480 images.

4.3. Selection of image preprocessing

To choose the best preprocessing image for the classification task, we trained 5 different CNN with each of the preprocessed methods of Sec. 3.3. Results of training are shown on Fig. 6. For training these CNNs, we used the pretrained weights of VGG-16 from Imagenet challenge and finetuned the fully connected layers with an initialization learning rate of 0.01 that changed every 5 epochs by a power of 10. We trained for 700 iterations (aprox. 14 epochs), with a batch size of 115, using stochastic gradient descent and a momentum of 0.9. We observed similar behaviors for the *rgb, color, gray* and *morpho* sets, around the fifth epoch the nets stopped generalizing and started over-fitting to the training set. Best ACA validation scores are recorded in Table 1, best score was 0.483 obtained using the *color* set. Unlike sets with complete eye images the *square* set started over-fitting around the 2nd epoch and achieved a best ACA score of 0.453 in the 3rd epoch. These behaviors can be observed comparing the loss and ACA graphs of Fig. 5 in which the ACA for training increases significantly more than in validation set.

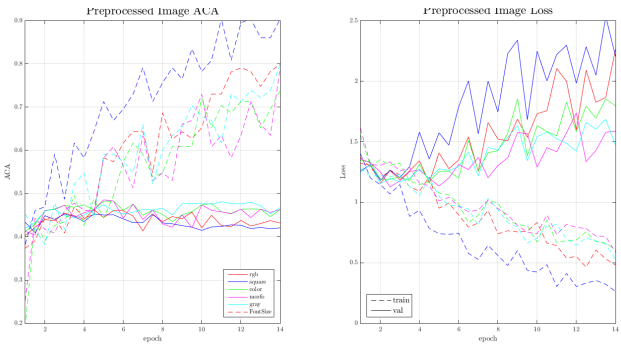

Figure 6. Selection of image preprocessing: Dotted lines correspond to the values for the training set and continuous lines for the validation set.

We also tested the vessel segmentation, DRIU, on images obtained from the two different cuts (circle and square images) to see if the information of the border of the eye was relevant for classification. Results are shown in Fig. 7. Based on these results, we inferred that the information lost with the square cut was relevant because the CNN trained with the square set started over-fitting in the 3rd epoch and achieved low scores, in contrast to the circle images that even in the 10th epoch training did not present a relevant over-fitting but a more stable behavior and learning saturation. The best scores obtained for each of the two networks were 0.396 for the circle one and 0.366 for the square one. For training these CNN we used the pretrained weights of VGG-16 and finetuned the fully connected layers.

We made a further analysis on the classification scores

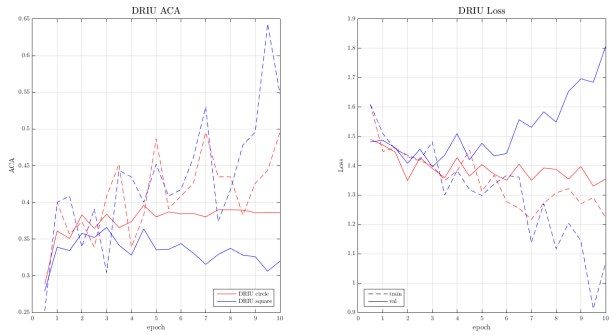

Figure 7. DRIU preprocessing images experiment: Dotted lines correspond to the values for the training set and continuous lines for the validation set.

.

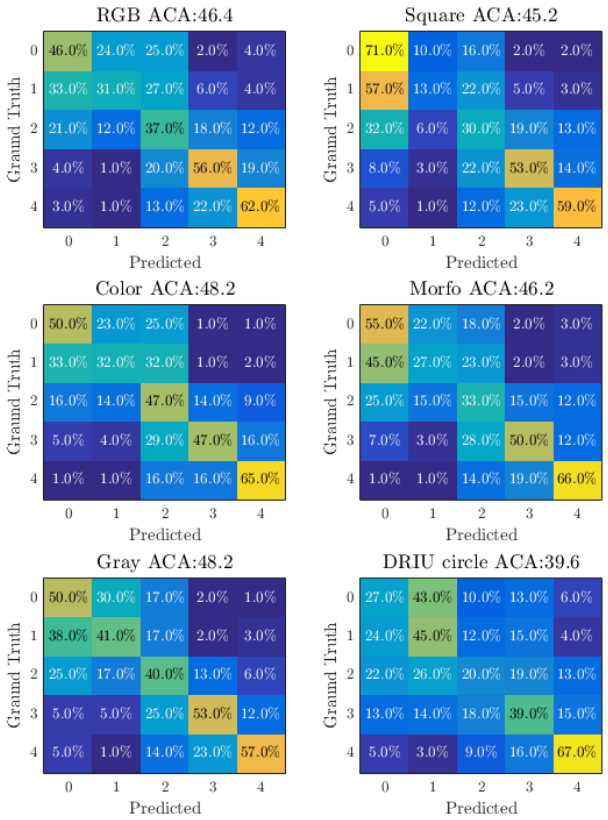

Figure 8. Confusion matrices (CM) of best epoch image preprocessing methods. Each CM corresponds to one preprocessing image experiment.

.

of the different sets based on their best score confusion matrices shown in Fig. 8. Results showed that the *color* and *gray* sets presented a more uniform correct classification score in comparison to the other sets. The *square* image set obtained a very unbalanced classification since most of the predictions were centered in classes 0, 3 and 4. A similar behavior happened for the vessel segmentation images, except for class 2, since it obtained a classification score higher than for the other sets. With the *color* set we obtained a more uniform result than using the raw *rgb* set, but in both cases class 1 was the most confused by the models. The *rgb* and *morpho* sets obtained a similar distribution result. All experiments presented a clear confusion between classes 0, 1 and 2. The confusion between the first three classes was expected since first stages of DR are almost undistinguishable, symptoms are minimum, and differences between stages are not strictly stated.

We found a clear difference between the two crops of the images, in both experiments the results were lower for square images compared with circle images. With these results, we concluded that for some images relevant information, such as the optic disc, hemorrhages, and exudates, were localized on the border of the BRF. Since square images cropped part of the border of the eye, they missed important information for DR detection. For this reason, in the rest of the experiments, we only considered the circular cropped sets.

4.4. Combination of input images

One of our main goals and key algorithmic component of our project was the effective combination of the vessel segmentation, DRIU, and the BRF image to produce a more robust and consistent classification algorithm. We explored three different strategies, as illustrated in Fig. 9. We also tried using different combinations of the preprocessed sets and varied the layers of the CNN that were tuned. These layers were set with an initialization learning rate of 0.01 that changed every 5 epochs by a power of 10. The different architectures used in this work were: early fusion, slow fusion, and late fusion.

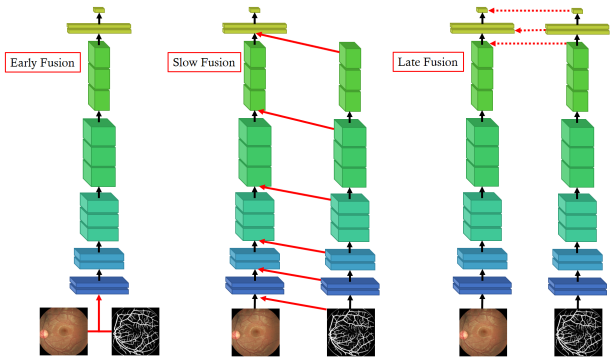

Figure 9. Early, Slow and Late Fusion Architectures. The base network is VGG.

.

Early Fusion. In an early fusion strategy, we introduce the vessel segmentation of DRIU as additional input channel, along with the BRF image, to VGG architecture and re-train the network for the DR classification task. This fusion was tested for all the preprocessing image sets to compare their results. For this CNN, we finetuned the fully connected layers and the first convolutional layer and trained for 12 epochs with a batch size of 64.

Slow Fusion. In the slow fusion strategy, we progressively merge intermediate layers of VGG applied to the vessel segmentation into the VGG architecture applied to BRF image. This was done to take full advantage of the fine and coarse information obtained from the vessel segmentation image. For this architecture, we tried three different fusions: the first-one considering the outputs of the convolutional layers from 1 to 5; the second-one the convolutional layers from 1 to 5 and the additional input channel of the vessel segmentation; and the third-one considering only additional input channel of the vessel segmentation and the 7th fully connected layer. To train this fusion architecture we finetuned the combined layers, depending on the fusion, and all fully connected layers for 14 epochs with a batch size of 32.

Late Fusion. In late fusion, the two networks analyzed the images independently for vessel segmentation image and BRF image, information was then combined in the fully connected layers. We tried different combinations points, after layer 5, 6, 7 and 8. For combination at layer 5, we trained all fully connected layers. For combination at layers 6 and 7, we tried both, training just the fully connected layers after the combined one and training all fully connected layers. Finally, for combination at layer 8 we took the maximum output of the two networks.

5. Results and Discussion

5.1. Combination of input images

In this section, we show the results for each of the combined architectures explained in Sec. 4.4. All best scores for each experiment are registered in Table 1.

Early Fusion. For the early fusion experiments, shown in Fig. 10, we compared the results training with different preprocessed images. For the *rgb* and *square* sets we observed an early overfitting; in contrast, for *color*, *gray* and *morpho*, the network generalized for some epochs. At the end of the 12th epoch, we observed a slight overfitting for all the networks. Best scores are registered in Table 1, *color* and *morpho* obtained the higher scores of 0.478 and 0.472 respectively. *Gray* set obtained 0.449, which showed that vessel segmentation information mixed with color information was more effective than taking the gray images. This is an expected result since exudates and hemorrhages have color characteristics that are no considered in the *gray* set.

Slow Fusion. We tested only *rgb* and *color* sets for slow fusion experiments. In Fig. 10, we observed a similar behavior for networks combined from 0 or 1 to 5th convolutional layers. It was observed that the network hardly learned because even in the training set the ACA did not increase significantly. For the network combined at the beginning and at the end, the behavior was different. This architecture was the only one that showed learning for the training set, in fact for *color* set it started overfitting form the beginning. The best score obtained for slow fusion architectures was 0.462 for *color* set combining at the 1st and 7th layer. For all slow fusion architecture combinations, *color* set obtained a higher classification score than *rgb*.

Late Fusion. For late fusion experiments, we also only combined *color* and *rgb* sets with vessel segmentation images. We combined the two nets in 4 different points. For this networks, weights were initialized using the best scores of the single architectures. Figure 11 shows the results of training, the first graph shows experiments of architectures combined at layers 5 and 6, and the second graph at layers 7 and 8. We found a similar behavior for networks combined at layers 5 and 6 in which the network did not learn. For the networks combined at layers 7 and 8 there was a clear overffiting and no generalization; for networks with more learning parameters (rgb7-lr and rgb7-lr in Fig 11) the overffiting was higher. This underlines the fact that if more parameters are being learned the probability of overfitting is higher. The best score, 0.475, was obtained using the *color* set combining the network at layer 7 and learning all fully connected parameters, with this configuration of parameters we obtained the higher score for *rgb* set of 0.470.

5.2. Fusion of Classifiers

As our final classification method we chose the best neural networks trained, the simple VGG with color and gray sets, the early fusion with color and morpho sets and the late fusion combined at layer 7 with color set, and average their prediction probability scores, (the architectures chosen are in red in Table 1). Then we choose, for each image, the class with higher combined score. The resulting Confusion Matrix is shown in Fig. 12. We observed classes 1 and 2 were extremely difficult to classify correctly, instead class 4 was the easiest. Our final ACA score was 0.505. This combined method obtained the highest score of all because used

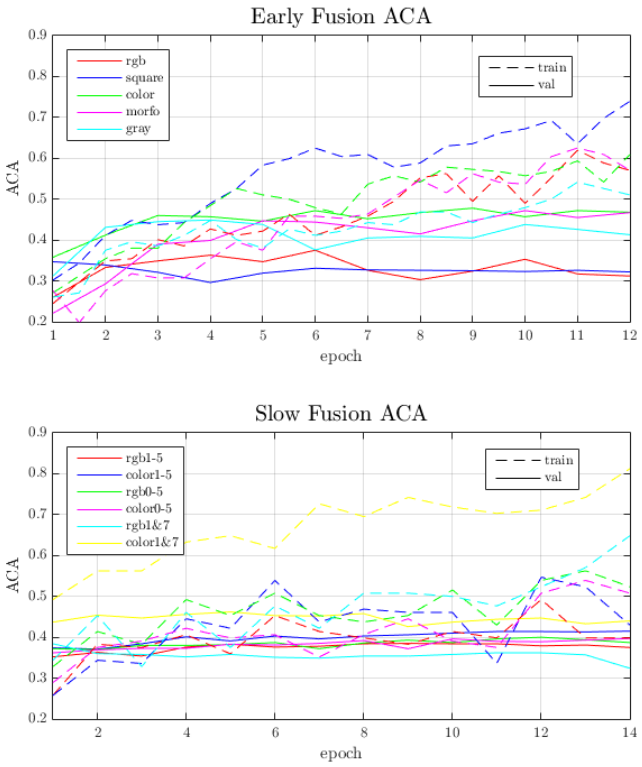

Figure 10. Early and Slow fusion experiments: Top graph correspond early fusion architecture experiments and bottom graph slow fusion architecture experiments.

.

multiple information and different networks. The use of different networks trained with different parameters, made the final method more robust and precise, less sensitive to outliers and capable of generalizing better considering more information.

5.3. Qualitative Results

Figures 13, 14, 15 and 16 show examples of the classification results. Fig. 13 shows two visually similar images, however, the left-hand side image has DR degree 3 and the right-hand side 1. For both cases, our algorithm classified them in degree 1. In our second example, Fig. 14, both images are healthy but our algorithm classified the left-hand side as degree 2, this prediction might be due to the illumination artifacts and the small brown spots it has. For this image, the second higher probability score was class 0 and the scores between the first and second place differ for less than 0.01. The third example of Fig. 15 shows two images in which the left-hand side image looks more seriously ill than the right-hand side image since it exhibits more hemorrhages and exudates. The truth is that the left-hand side image is class 3 and the right-hand side is 4. Both images were predicted in class 4. Our final example, Fig. 16,

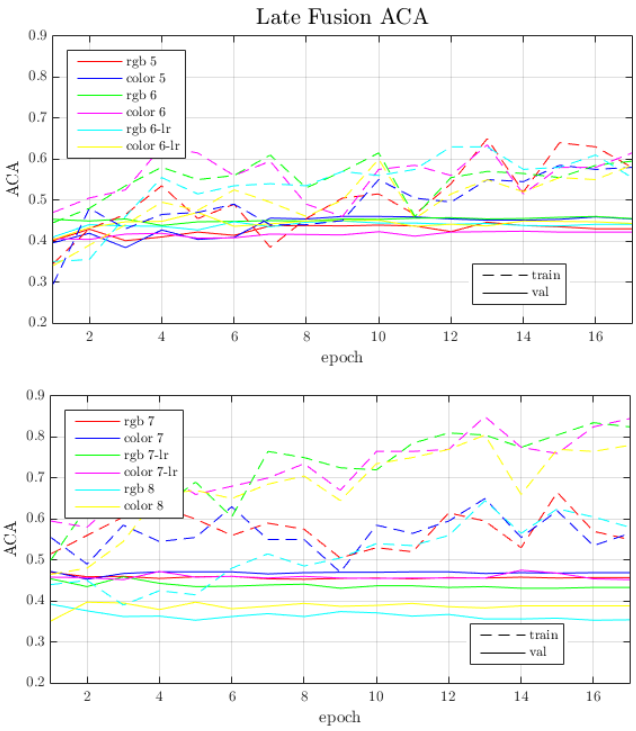

Figure 11. Late fusion experiments: Top graph correspond to combinations at layers 5 and 6 and bottom graph at 7 and 8.

.

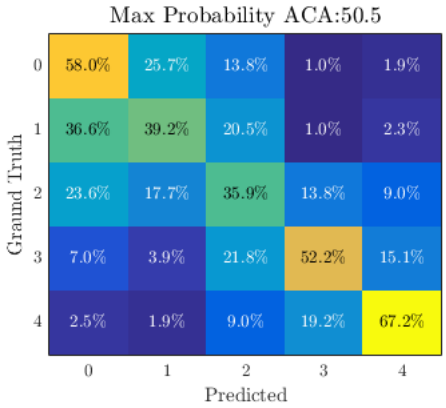

Figure 12. Final Classification Confusion Matrix

.

shows two very similar images which might have similar, or at least consecutive classes. The fact is that the left-hand side image true class is 0 but it was predicted class 1 and the right-hand side image has true and predicted class 3.

This few examples show the complexity of this problem. Even when images seem to have the same degree of DR or a consecutive order, the real classification of the images is not easy to guess just by sight by non-specialists.

Table 1. Experimental ACA Results

Simple	
rgb	0.463
square	0.453
color	0.483
morfo	0.463
gray	0.481
DRIU circle	0.396
DRIU square	0.366

Early	4 channel
rgb+DRIU	0.375
color+DRIU	0.478
square+DRIU	0.347
morfo+DRIU	0.472
gray+DRIU	0.449

Slow	concat 0-5	concat 1-5	concat 0 & 7
rgb+DRIU	0.400	0.385	0.385
color+DRIU	0.408	0.415	0.462

Late	concat 5	concat 6		concat 7		concat 8
		fix weights	learn weights	fix weights	learn weights	max
rgb+DRIU	0.446	0.460	0.449	0.470	0.459	0.392
color+DRIU	0.460	0.470	0.426	0.475	0.472	0.397

.

Figure 13. Classification results: left side image true class 3 predicted 1, right side image true and predicted class 1.

.

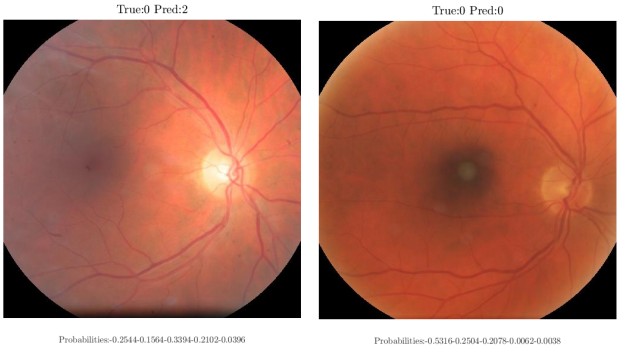

Figure 14. Classification results: left side image true class 0 predicted 2, right side image true and predicted class 0.

.

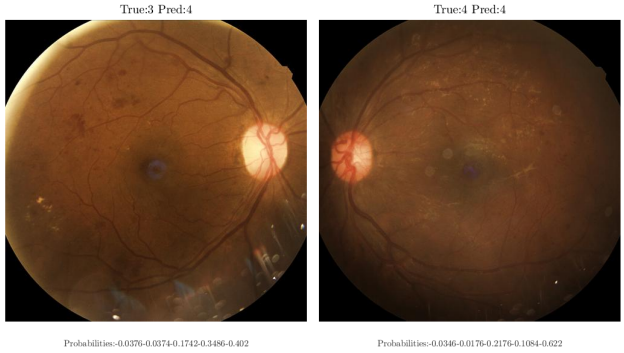

Figure 15. Classification results: left side image true class 3 predicted 4, right side image true and predicted class 4.

.

Figure 16. Classification results: left side image true class 0 predicted 1, right side image true and predicted class 3.

.

6. Conclusion

We obtained an ACA of 0.505, this means that we surpass our previous baseline of 0.433 with SVM and 0.416 with CNN. Although our score is still far from the 0.850 obtained by Ben Graham, we achieved several goals, within this project. Regarding our classification score, we improved significantly. Based on the final confusion matrix we can observe that higher classification scores where obtained for classes 0, 3 and 4. Class 1 had high confusion with classes 0 and 2, and class 2 had high confusion with all classes specially with class 0. The correct classification for class 1 is the most important since it represents the early stages of DR, where early detection is critical. It should be noted that, although class 1 has not the higher score, it is comparable to our baseline and last classifiers [1]. Qualitative results show this problem is not easy to be solve for non-experts in DR and our algorithm predicts reasonable results. Furthermore, neural networks are a classification method with a lot of potential for improvement and even though we did several experiments, there are more parameters tune, either the network architecture, the preprocessing of the images, the data augmentation, using normalization layers, the weight learning parameters among others.

Within this project, other goals were achieved such as learning to use Python software and Caffe package. Using architectures such as DRIU and VGG. Designing and testing different neural network architectures to approach a challenging biomedical problem using computer vision techniques and deep learning.

7. Acknowledgements

I want to thank God for all blessings he has given me: the opportunity of studying in this university and finding my passion; the strength to overcome every challenge; and the amazing people that have been by my side with whom I have grown. I would like to express my sincere gratitude to my advisor Pablo, who has been my mentor during the end of my carrier. Thank to him for his constant teaching, support, motivation, advice, and perspective towards the future. I want to thank my parents who have always supported me and given me courage to work and success in everything I do. To my brother that has been guide, example, motivation, and encouraging me to do my best and never give up. Finally, I want to thank my friends who have lived this journey with me, they have been my support and hope in many moments. This project started in collaboration with Lina María Mejía and it is a continuation of [1].

References

[1] M. A. Bravo. Learning classication methods and applications to medical images. Thesis of Mathematics, Universidad de los Andes, Bogotá, Colombia, 2016.

[2] Á. R. R. Diabética. Pontificia universidad católica de chile. *Boletín de la escuela de medicina*, 31(3), 2006.

[3] M. C. Esteve, M. Fernández, A. Goday, and J. Cano. Revisión de las complicaciones crónicas de la diabetes mellitus en españa. *Jano: Medicina y humanidades*, (1644):27, 2007.

[4] D. S. Fong, L. Aiello, T. W. Gardner, G. L. King, G. Blankenship, J. D. Cavallerano, F. L. Ferris, and R. Klein. Retinopathy in diabetes. *Diabetes care*, 27(suppl 1):s84–s87, 2004.

[5] C. H. Foundation. Diabetic retinopathy detection, September 2015.

[6] B. Graham. Fractional max-pooling. *arXiv preprint arXiv:1412.6071*, 2014.

[7] B. Graham. Kaggle diabetic retinopathy detection competition report, August 6 2015.

[8] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[9] R. K. Jun Xu, John Dunavent. Summary of our solution to the kaggle diabetic retinopathy detection competition, 2015.

[10] K.-K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool. Deep retinal image understanding. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 140–148. Springer, 2016.

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

[12] J. Olson, F. Strachan, J. Hipwell, K. Goatman, K. McHardy, J. Forrester, and P. Sharp. A comparative evaluation of digital imaging, retinal photography and optometrist examination in screening for diabetic retinopathy. *Diabetic medicine*, 20(7):528–534, 2003.

[13] A. W. Reza and C. Eswaran. A decision support system for automatic screening of non-proliferative diabetic retinopathy. *Journal of medical systems*, 35(1):17–24, 2011.

[14] J. Sández, J. Fernández Vigo, A. Díaz, M. Rodríguez, M. Jesús, and J. Barrios. Prevalencia de retinopatía diabética en una población diabética no seleccionada. *Archivos de la Sociedad Española de Oftalmología*, 59(3):277–284, 1990.

[15] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[16] A. Vedaldi and K. Lenc. Matconvnet: Convolutional neural networks for matlab. In *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference*, pages 689–692. ACM, 2015.

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