Automatic Detection of Blood Vessels From Retinal Images Using Convolutional Neural Network

Ville Virkkala, Jarno Leppnen
(Dated: April 28, 2019)

Main causes of blindness, such as diabetes and hypertension, are related to morphological changes of the blood vessels. Thus automatic segmentation of blood vessels from retinal images could greatly improve to make the correct diagnosis in optometric studies. In this work a convolutional neural network is developed to automatically segment blood vessels from retinal images. The developed method produces an accuracy of 85% for the test set. The performance of developed method is also compared against other similar mehthods and against random forest classifier.

I. INTRODUCTION

Several studies have used neural networks in automatic blood vessel detection from retinal images. These methods can be roughly divided into two groups: patch based segmentation of blood vessels[1] and methods based on fully convolutional neural networks[2,3]. In patch based image segmentation the image is traversed through pixel by pixel. For each pixel a patch of fixed size, centered at the pixel, is taken from the image and fed to neural network that classifies the pixel into certain class. In Ref.[1], for each pixel three patches of different sizes from the green channel of image were taken and scaled into same size and used as a three channel input to convolutional neural network. In Ref.[1] the reported accuracy was in most cases clearly better than 90%. The advantages of patch based method are their simplicity and ease of training. However, their main disadvantage is the high computational load when doing inference, because for each pixel separate patch is taken that is fed to neural network. Fully convolutional neural networks[2] are the current state of the art method in image segmentation. The huge advantage of fully convolutional neural network is the huge speed up compared to patch based methods, because the whole image is fed only once as whole to the neural network. In addition there is more contextual information available in fully convolutional neural networks because the whole image is processed at once instead of using smaller patches. In Ref.[3] a modified version of the fully convolutional network in Ref.[2] was used. In Ref.[3] the downsampling of the image was done using the same VGG-16 network as in[2] however the output of each convolution stage was directly upsampled into original size and a binary cross entropy loss was then connected into each upsampled image. The final loss that was optimized is then the sum of the losses connected to each upsampled image. In Ref.[3] the reported accuracy was in all cases better than 95% while the method was much faster, single image segmentation time around 11seconds, than the method of Ref.[1], single image segmentation time around 2000 seconds.

In this work a patch based semantic segmentation method based on convolutional neural networks, resembling that of Ref.[1], is developed to detect blood vessels from retinal images. The performance of the developed method is validated against several test images. In addition performance of the developed method is compared to random forest classifier implemented in scikit-learn python library. The paper is organized as follows. The used data-set and the computational methods are described in detail in sections II and III. In Sec. IV training metrics and performance of the developed method in inference when applied on test images are given. Sec. V is a summary of the results and the differences between the two classifiers are discussed.

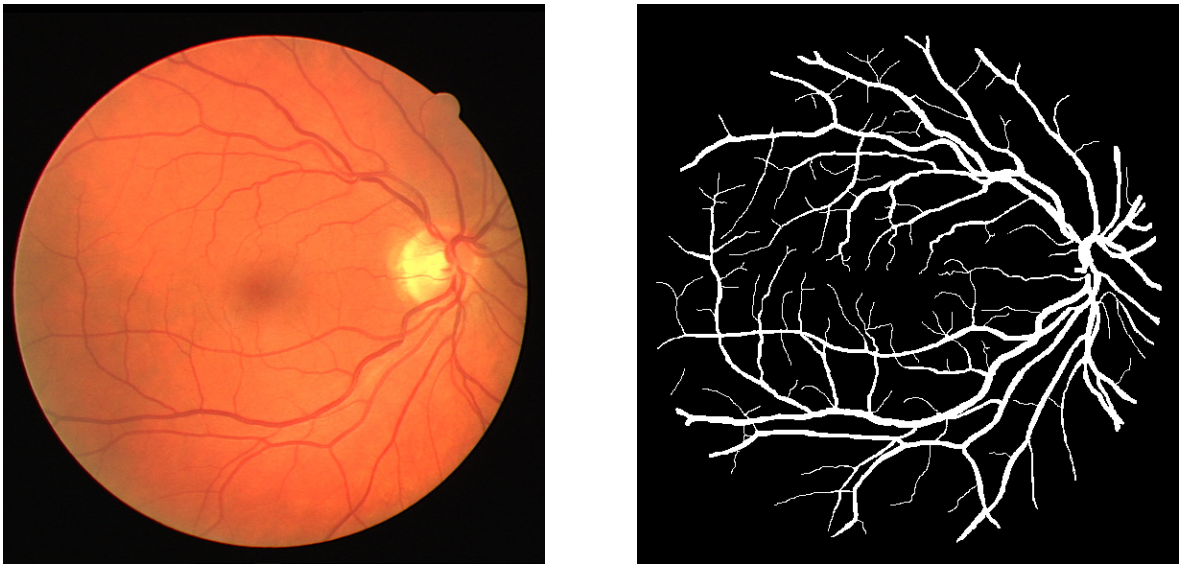
II. USED DATA-SET

The used data set in this work is the publicly available DRIVE data set[?] available in[?] . The data set consist of 40 images of size 565x584 and the corresponding ground truth annotations of blood vessels. The images were obtained from diabetic subjects as a part of diabetic retinopathy screening program in The Netherlands.

The data set was divided into training, validation and test sets containing 28, 6 and 6 images respectively. The images are three channel RGB images, but only the green channel was used in classification, because typically the blood vessels are best visible at the green channel. In classification of pixels patches of size 33x33 centered at the pixel were used. From each image 2000 patches for both pixels corresponding to blood vessels and background pixels were randomly sampled resulting in total 112000 samples for training set, 24000 samples for validation set and 24000 samples for test set respectively.Examples of retinal image that is classified and the corresponding ground truth image are shown in figures 1a and 1b respectively. Examples of patches that corresponds to pixels that are labeled as blood vessels and background are shown in figures 2a-2c and 2d-2f respectively.

III. COMPUTATIONAL METHODS

In this work two different methods were used to classify the songs to different genres. First method is logistic-regression method in which the logistic-loss is minimized iteratively using the gradient descent method. The other



(a) **(b)**

FIG. 1: Example of retinal image used in training (a) and the corresponding ground truth annotation of blood vessels (b).

method used is the Bayes-classifier which classifies the song to certain category that gives the maximum posterior probability with respect to label i. Both methods are described below in detail. In addition we studied the effect of feature extraction and for that purpose we used principal component analysis method to exclude features with little impact.

A. Developed Convolutional Neural Network

B. Training

The developed method above contains three hyper parameters, the number of kernels in first and second convolution layer and the number of layers in first fully connected layer. Rest of the parameters, *i.e.*, the kernel size and patch size were kept fixed. The optimal hyper parameters were found using random search. Ten different configurations were randomly sampled from given intervals that were [10, 35], [10, 35] and [200, 1200] respectively for the hyper parameters. For each configuration the network was trained over 125 epochs using the train data set. In addition an early stopping criterion with tolerance of 0.01 and patience of 2 was used to avoid over fitting of the model. At the end of each epoch the validation loss, evaluated against the validation data set, was evaluated and used as input for early stopping criterion.

Stochastic gradient descent with patch size of 1000 and Adam optimizer[4], initialized with learning rate of 1e-6 was used to optimize the network parameters. The final accuracy of the network, with given hyper parameters, was then obtained evaluating the accuracy against the test set. Final network, used in inference, was then constructed using the hyper parameters corresponding to best accuracy obtained. The final network was then retrained over 250 epochs or until early stop to get full convergence.

C. Inference

IV. RESULTS

A. Optimal hyper parameters and the training accuracy

The results for the hyper parameter search, explained in section III B, are shown in Table I. According to Table I the best accuracy was obtained when the hyper parameters were set to 33x67x355. The final accuracy with these parameters for test set, after training over 250 epochs was 84.8%.

V. CONCLUSIONS

In this work we used logistic-regression and Bayes-classifier to classify songs to different genres based on the music signal's characteristics. For logistic regression the obtained accuracy for test set was 0.67 and logistic-loss 0.27. For the external data set used the obtained accuracy and logistic-loss were 0.65 and 0.178 respectively in the case of logistic-regression. For the Bayes-classifier the obtained accuracy and logistic-loss were 0.53 and 0.33 for the test-data and for external data set 0.32 and 1.17 respectively. According to obtained results both

classifiers performed clearly better than random guess, but remained far from perfect classification. From the two classifiers used the logistic-regression classifier performed clearly better. The logistic classifier also generalized much better to completely new data giving nearly equal performance for test data set and external data set.

Configuration	Accuracy (%)	Configuration	Accuracy (%)
33x67x355	83.8	16x50x981	82.9
29x42x709	83.7	18x63x1048	82.9
29x49x871	83.5	27x58x359	82.4
19x57x1158	83.3	10x64x1176	81.3
20x60x256	83.2	14x49x638	80.1

TABLE I: Results of the random hyper parameter search. The numbers in the configuration column indicates the number of kernels in the first and second convolution layers and the number of neurons in the first fully connected layer respectively. The accuracy is obtained running the trained network with the given hyper parameters against the test data set.

[1] J. H. Tan, U. R. Acharya, S. V. Bhandary, K. C. Chua, and S. Sivaprasa, Journal of Computational Science **309**, 70 (2018).

[2] J. Long, E. Shelhamer, and T. Darrell, in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015) pp. 3431–3440.

[3] K. Hu, Z. Zhang, X. Niu, Y. Zhang, C. Cao, F. Xiao, and X. Gao, Neurocomputings **105**, 179 (2009).

[4] D. P. Kingma and J. Lei Ba, (2015), arXiv:1412.6980.

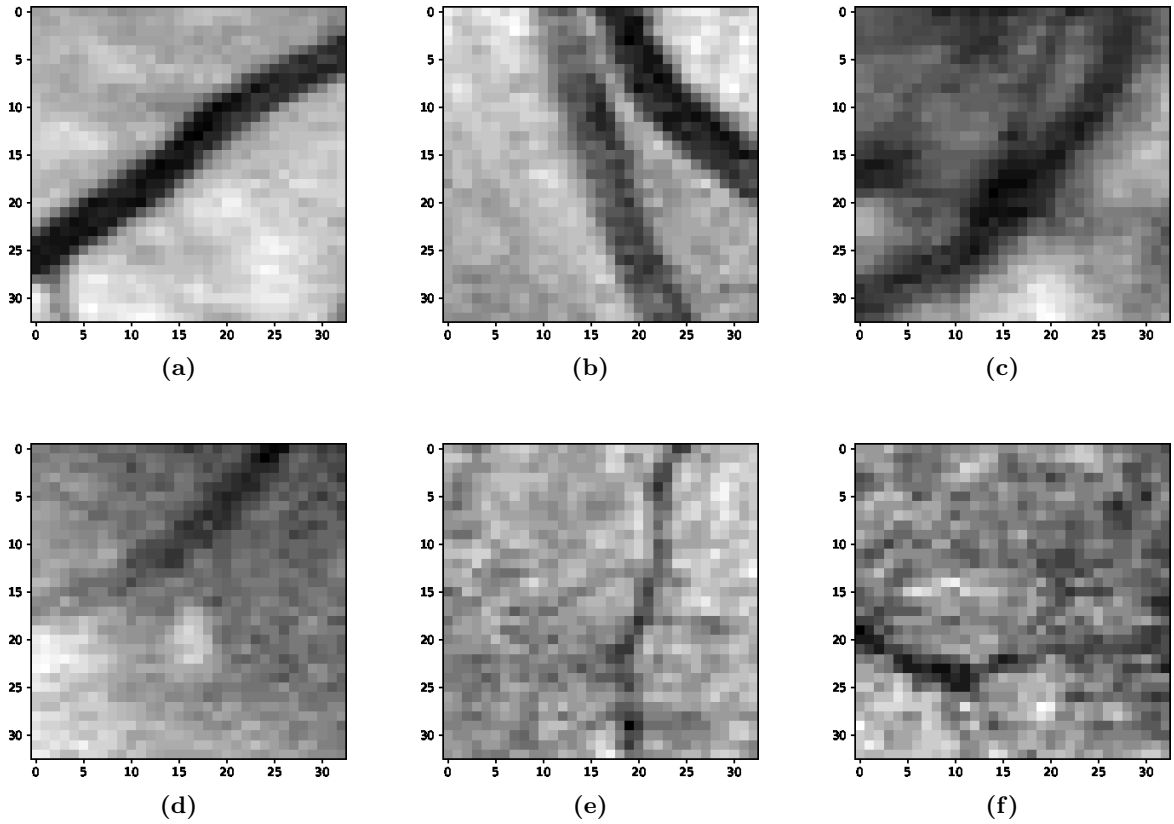


FIG. 2: Examples of patches corresponding to pixels that are labeled as blood vessels (a)-(c) and patches corresponding to pixels that are labeled as background (d)-(f). The pixel that the label is related to is located at the center of patch.