OPTIC DISC SEGMENTATION USING A DEEP RESIDUAL FULLY CONNECTED NEURAL NETWORK

Chaitanya Kaul, Suresh Manandhar

Department of Computer Science,
University of York, York, UK,
YO10 5GH

ABSTRACT

Optic disc segmentation is an important first step for detecting various eye diseases such as glaucoma, cataract, etc. The recent advancements in the field of computer vision, thanks to the introduction of convolution based neural network architectures, has been nothing less than revolutionary in solving such problems. In this paper, we present a mixture of residual connections with a unet architecture, as a solution to the problem of optic disc segmentation. The network is tested on data obtained from the Indian Diabetic Retinopathy Image Dataset [1]. The results are compared with the vanilla unet to show the power of residual connections over the previous state of the art network in biomedical image segmentation.

Index Terms— Optic disc segmentation, residual mappings, residual unet.

1. INTRODUCTION

Optic disc (OD) segmentation is a a fundamental task when it comes to processing eye images. Many diseases can cause the optic nerve irrervsible damage and can lead to blindness. Therefore, segmenting the optic disc is an important step in creating a frame of reference for diagnosing optic nerve head pathologies. This, in turn, calls for a reliable OD segmentation technique for automatic screening of optic nerve head abnormalities.

Recent years have witnessed the emergence of deep learning. Methods based on convolution neural networks are quickily becoming the baseline for most computer vision tasks. The work of [2] has suggested that a deeper network, doesnt necessarily mean a better network. They introduced identity mappings to facilitate better learning and to allow the networks to go deeper and also use fewer parameters. [3] Introduced the fully convolutional networks for image segmentation, which used skip connections to combing predictions from the previous layers. The stride was adjused as a hyperparameter to get finer predictions. [4] Introduced an encoder decoder structure called the unet which has found great success in biomedical imaging tasks. We blend the power of residual connections into the structure of the unet to facilitate segmentation of the optic disc from fundus images.

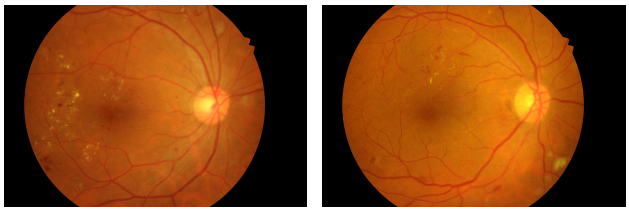


Fig. 1. The following are images obtained from a fundus camera. The bright round patchs in the images are called the optic discs.

The structure of this paper is as follows. In the next section, we introduce the residual unet architecture, which is followed by the the experiments conducted. We end with our results on the dataset, and conclusions.

2. METHODOLOGY

2.1. Architecture

2.1.1. Unet

The unet [4] gets its name from its 'U' shape. To the left is an encoder style structure, and to the right is a decoder style structure. The architecture employs low level information from images to do pixel level classification. The encoder learns a heirarchical representation of the image and the decoder upsamples the image back to its orignal size giving pixel level predictions. The output of a particuar block in the encoder is concatenated with the output of the corresponding decoder layer to facilitate better information flow through the network (see Figure 3). This is termed as a contracting path by the authors and helps capture context and precise segment localization. The task at hand (OD segmentation) has very few training images to work with. As the unet architecture was built with this very problem in mind, it is hence the architecture of choice for this problem.

2.1.2. Residual Mappings

A simple residual connection [2] is the concatenation of the input to the output of a weight layer. It is given by the equation,

$$y_i = h(x_i) + F(x_i, w_i),$$
$$x_{i+1} = f(y_i)$$

Such residual blocks can be stacked one after another between pooling layers to get a Resnet style architecture. However, there are more complex identity mappings which help build deeper and better networks. The advantage of using residual connections is that they help in better gradient flow between layers due to their identity mappings which helps to alleviate the vanishing gradient problem to a huge extent. As the gradient can be propogated better through the layers using these mappings, networks can now be constructed which are deeper and have improved performance. [5] Presents a detailed analysis of different residual mappings. Our residual block can be seen in Figure 2 (a). It contains batch normalization after the input layer. Relu activations are used and there are two convolution operations, with a 3x3 receptive field. Instead of an identity mapping, we concatenate the input to the output following a 1x1 convolution which helps in reducing the number of parameters in the network.

2.1.3. Residual Unet

The residual unet is the unet architecture where the convolutions are replaced by the residual blocks. The combination of residual connections with the unet helps propogate information even better through the network than a vanilla unet (which can be seen through the results). The architecture can be seen in Figure 3.

The input is a 256x256x3 image, passed into a convolution layer. This is followed by a residual block and a max pooling layer. There are 9 residual blocks in the encoder side and 9 in the decoder side. Downsampling is done using the pooling operation in the encoder. The decoder uses the upsampling operation as a means of 'unpooling' to increase the size of the feature maps spatially. Relu activations are used throughout. After the last residual block, a 1x1 convolution is folowed by a sigmoid activation to project the feature map into the desired segmentation. Our network is 'deeper' than the vanilla unet with 33 convolution layers, compared to the 23 of the vanilla unet. The implementation details are decribed in detail in Table 1.

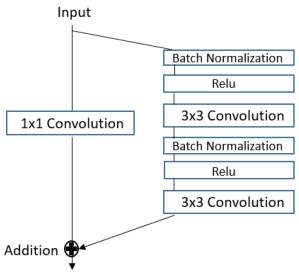
Network Archhitecture		
Layer	Filter	Stride/ Rate
Input	-	-
Convolution	3x3/16	1
Residual	3x3/32	1
MaxPooling	-	(2,2)
Residual	3x3/32	1
Residual	3x3/64	1
MaxPooling	-	(2,2)
Residual	3x3/64	1
Residual	3x3/128	1
MaxPooling	-	(2,2)
Residual	3x3/128	1
Residual	3x3/256	1
Dropout	-	0.2
MaxPooling	-	(2,2)
Residual	3x3/256	1
Residual	3x3/128	1
Dropout	-	0.2
Convolution	2x2/128	1
Concatenation	-	-
Residual	3x3/384	1
Residual	3x3/64	1
Convolution	2x2/64	1
Concatenation	-	-
Residual	3x3/192	1
Residual	3x3/32	1
Convolution	2x2/32	1
Concatenation	-	-
Residual	3x3/96	1
Residual	3x3/16	1
Convolution	2x2/16	1
Concatenation	-	-
Residual	3x3/48	1
Residual	3x3/16	1
Residual	3x3/2	1
Convolution	1x1/1	1

Table 1. Residual unet Architecture. The upsampling stride before the concatenation layers is (2,2). The filter volume in the encoder deontes the output filters of the residual unet. In the decoder section, the filter volume denotes the input to the residual layers and not the output. This is done to denote the output filter size of the filter map concatenation and to make the network easily reproducible.

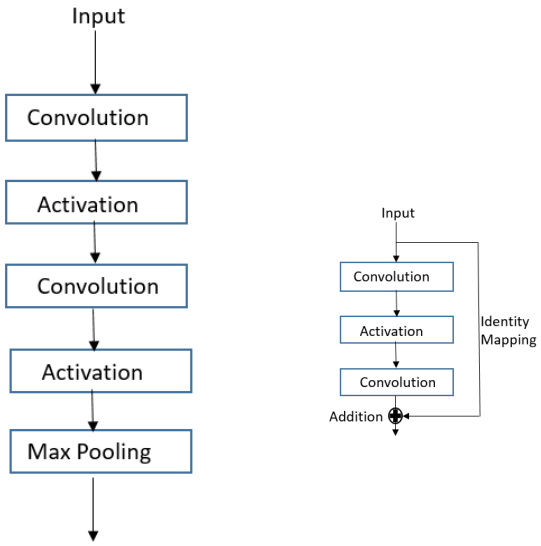
2.2. Loss Function

We train the network on a loss function which is the negative of the global dice index. This is given by,

$$Negative\ Dice\ Loss\ =\ -\frac{2 * |X \cap Y|}{|X| + |Y|}$$



(a) Residual block in the proposed architecture



(b) Unet Conventional data flow (c) Simple identity mapping

Fig. 2. Different Mappings from information propogation in the unet architecture

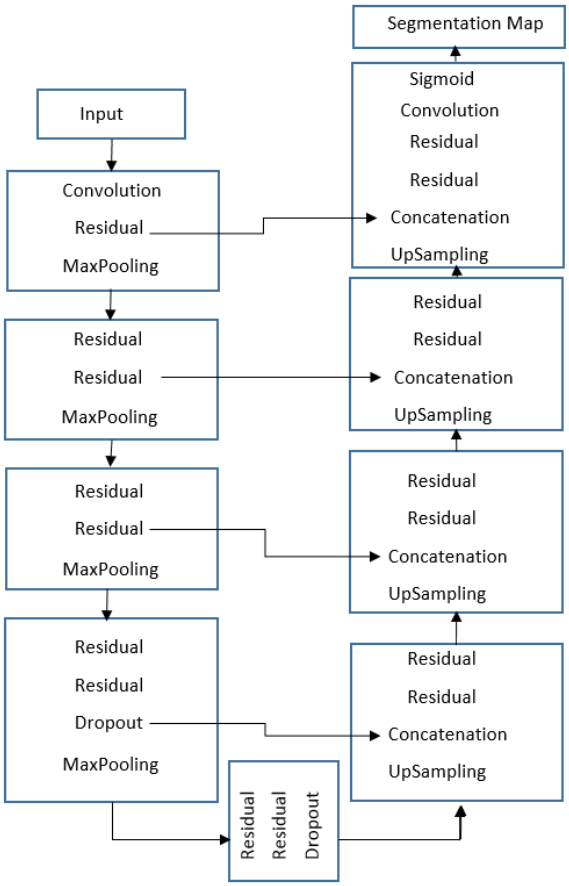


Fig. 3. Proposed Residual Unet Architecture. The convolution layers use relu activation except the last one which uses sigmoid.The output of the residual layers and the first dropout layer is concatenated with the upsampled output in each decoder block.

It is global in the sense that it is trained over the values in a batch rather than each image seperately. It is also easier to optimize as a cost function, as it gets rid of any discontinuties that arise in the cost function due to trying to map a continuous probability to a discrete 0 or 1 value.

2.3. Evaluation Metrics

The evaluation metrics used are the dice index, the jaccard index, sensitivity and specificity. The dice index is discussed in section 2.2. The jaccard index is an intersection over union metric, like the dice index. Sensitivity is defined as the true positives divided by the total positive samples and specificity is true negatives divided by the total negative samples.

3. EXPERIMENTS

3.1. Dataset and Augmentation

The dataset contains 54 very high resolution (4288x2848) fundus images with the optic disc segmented in the ground truths. This amount of data is obviously not enough for a deep learning task, so data augmentation was applied. We appied random zooms, rotations and channel shifts to create a dataset of roughly 1400 images and a validation set of 270 images. The exact same random transformations applied to the images, were also applied to the ground truth masks. As a few optic discs were close to the edge of the images, care was taken to not crop them out during the augmentation process. We initially tried running the architecture with 128x128x1 grayscale images but settled on 256x256x3 preprocessed RGB images as it gave better results.

3.2. Implementation Details

Before the augmentation, data preprocessing was applied. The fundus images are affected by camera illumination problems, and they also have a thick black border which affects the quality of the segmentation. So the image was first scaled to an arbritrary size which reduced the black border, and then mean local colour was subtracted from the image in an attempt to get rid of illumination variations. The images were then re-sized to 256x256x3 which formed the input to the network. Inputting this data into the network saw us run into the exploding gradient problem. This was rectified by standardizing the data. The model was implemented using the keras [6] library with a tensorflow backend. It was run on a Nvidia GTX 1080 with a batch size of 16. The Adam optimizer was used with a learning rate schedule which was determined experimentally. Dropout was used to improve the generalization capacity of the network, with a rate of 0.2, which means that 20% of the nodes were randomly dropped at one iteration. The network converges in about 50 epochs.

3.3. Results and Comparisions with the Vanilla Unet

The results are summarized in Tables 2 and 3. The values of all the evaluation metrics are calculated by comparing the original sized ground truth masks and the predicted masks for the 54 fundus images. It can be seen that the residual unet performs considerably better than the vanilla unet. Some results of images after the optic disc segmented are also shown in Figure 3.

Table 2. Unet Results

Dice Index	Jaccard Index	Sensitivity	Specificity
0.9693	0.9404	0.9987	**0.9999**

Table 3. Residual Unet Results

Dice Index	Jaccard Index	Sensitivity	Specificity
0.9805	**0.9618**	**0.9999**	**0.9999**

4. CONCLUSION

In this paper, we presented out residual unet architecture for the segmentation of optic discs from fundus images. The combination of residual connections with the architecture of the unet provided promising results on the dataset. The residual connections facilitate better gradient flow due to which the accuracy of the residual unet is better compared to its conventional counterpart.

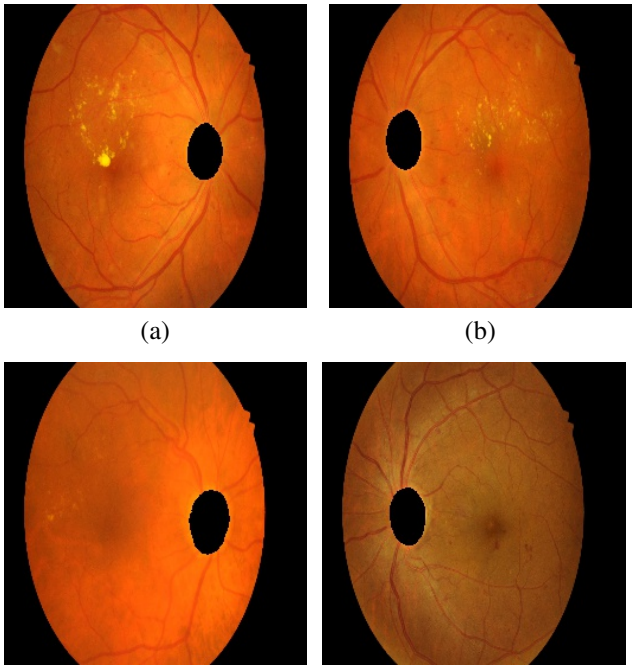


Fig. 4. Fundus images with the optic disc segmented out

5. REFERENCES

[1] "Diabetic retinopathy segmentation and grading challenge," https://idrid.grand-challenge.org/, 2018.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[3] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[4] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Identity mappings in deep residual networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 630–645.

[6] François Chollet et al., "Keras," https://github.com/keras-team/keras, 2015.