Branching Autoencoder for Saliency Prediction

CV2 Project Paper

Ramtin Nouri

28.04.2021

Abstract

I propose a novel CNN architecture for eye fixation prediction. Using a branching architecture with different branches using different filter sizes in their convolutional layers I could achieve great results.

1 Introduction

In this project we were supposed to build a model for prediction of saliency maps based on eye fixations. We were provided with a dataset based on DUT-OMRON [4]. The inputs are RGB images and outputs are saliency maps created from eye fixations. I build a fully convolutional network with a novel architecture for this task.

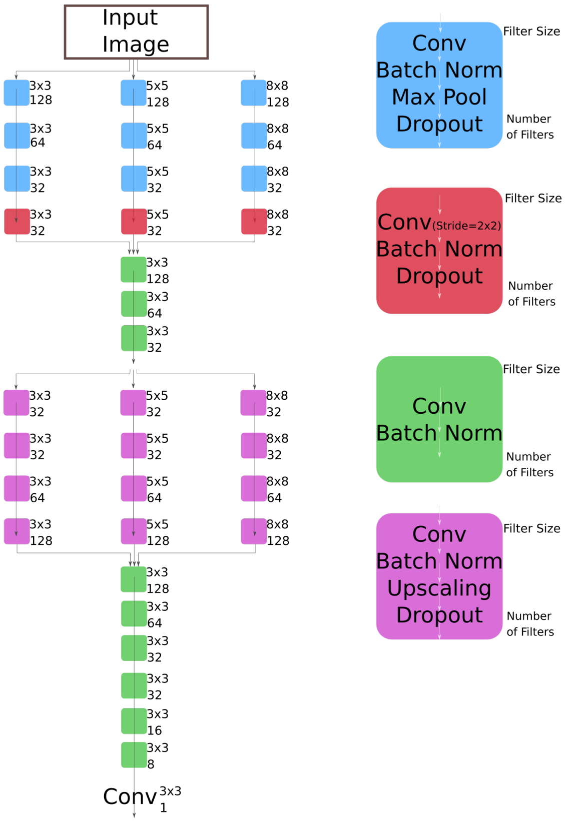

Figure 1: Architecture of neural network. On the left a simplified view of the architecture is shown, where combination of layers have been summarized as displayed on the right. Right to each square the filter size and number of filters used in that convolutional layer is displayed.

2 Model Architecture

The architecture to some extend has an hourglass shape like in an autoencoder. It splits into three branches, where the size of filter used differs (3, 5 and 8). These different filter sizes can be seen as different receptive fields. These three branches have decreasing number of filters which then get concatenated and "summarized" by three further convolutional layers. These could be compared to the dense layer in the middle of an autoencoder as they densify the information. Then again the output is passed to three branches which have an increasing number of filters now and represent the encoder part of the network. Their output is then again passed to a row of convolutional layers decreasing in number of filters leading to one. The output of of last convolutional layer with only one filter is the output of the network.

3 Training

As can be seen in figure 3 each convolutional layer is followed by a batch normalization layer. Every maxpooling or upscaling layer is followed by a dropout layer with a dropout of 0.1.
As loss the mean squared error (MSE) was chosen.
In addition to the provided dataset I also used :

- MIT [3]
- CAT2000 [1]
- Salicon [2]

Datasets without an explicit validation set were split with a 5 : 1 ratio, resulting in over 18616 training samples and 7530 validation samples. All data is also augmented by randomly cropping each side by 25%, flipping the image randomly and normalizing.

4 Results

As figure 2 shows the loss fell below 0.012 MSE with a lowest value of 0.0164 and a mean absolute error of 0.0608.

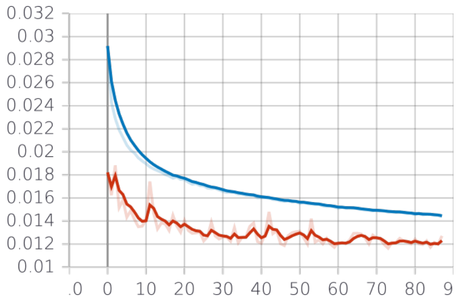

Figure 2: training(blue) and validation(red) loss over epochs. The lighter colors are the actual datapoints and the more saturated curves show the data smoothed out by a value of 0.6.

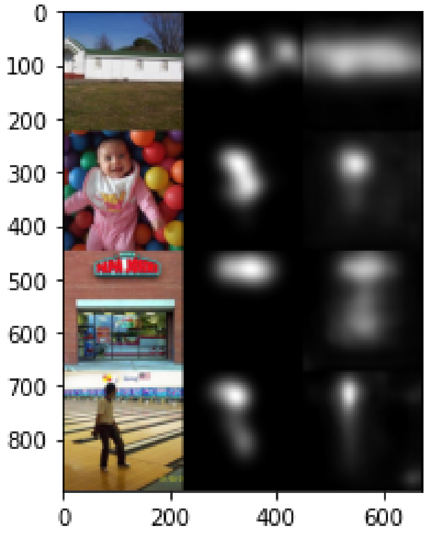

Figure 3: Left column: Input images; Middle column: Ground truth eye fixations; Right column:Predictions

5 Conclusion

I present a novel architecture with great results and room for further improvements.

References

[1] Ali Borji and Laurent Itti. Cat2000: A large scale fixation dataset for boosting saliency research. *CVPR 2015 workshop on "Future of Datasets"*, 2015. arXiv preprint arXiv:1505.03581.

[2] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. Salicon: Saliency in context. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[3] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. In *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[4] Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3166–3173. IEEE, 2013.