Efficient Visual Attention with Deep Learning

Erik Perillo[1], **Esther Luna Colombini**[1]

[1]Institute of Computing (IC) – University of Campinas (Unicamp)
Caixa Postal 6176 – 13.084-971 – Campinas – SP – Brazil

erik.perillo@gmail.com, esther@ic.unicamp.br

Abstract. *The high volume of visual data usually available for autonomous applications contains information that is mostly irrelevant for intelligent agents. Humans perform sensorial filtering through a mechanism called attention. In this work, we present a new fully convolutional neural network architecture designed for detecting visual salience. Experiments carried out with the MIT300 benchmark presented state-of-the-art performance and a parameter reduction of 3/4 compared to similar models.*

Resumo. *O alto volume de dados visuais geralmente disponível para aplicações autônomas contém informações que são, em sua grande parte, irrelevantes para agentes inteligentes. Os seres humanos realizam uma filtragem sensorial através de um mecanismo chamado atenção. Este trabalho descreve uma nova arquitetura de rede neural convolucional projetada para detectar a saliência visual. Os experimentos realizados com o benchmark MIT300 apresentaram desempenho compatível com os melhores modelos e uma redução de parâmetros de 3/4 em comparação com modelos similares.*

1. Introduction

One of the most challenging unsolved problems in Artificial Intelligence is vision. However, it is fundamental for the conception of systems that interact with the real physical world. These systems would be useful for applications such as service robots operating in houses, industry and agriculture, with great potential for the benefit of society.

Vision is remarkably data and computationally intensive. In humans, approximately half of the brain is involved in vision-related tasks [Fixott 1957]. Even our minds can not handle all the sheer amount of sensorial information received every second. In order to deal with this immense amount of data, we have attentional systems, a fundamental mechanism that, among other functions, filters out irrelevant information – either visual or from other senses– and helps us focusing our cognitive processes on what is important at a given moment. These facts are a strong evidence that, in order to help solving the vision problems, attention should be applied.

Visual attention can be defined as the delimitation of a certain spatial region on an image for further cognitive processing [Treisman and Gelade 1980]. The phenomenon emerges from two fundamentally different processes: the *top-down* mechanism that implements our longer-term cognitive strategies by biasing attention according to our interests (e.g. find a red apple in a tree because of hunger, which will make red be more recognizable on the scene), and the *bottom-up* mechanism [Colombini 2014], a process generated through external stimuli that captures one's attention from its conspicuousness level. In this work, we focus on the latter, also named visual saliency.

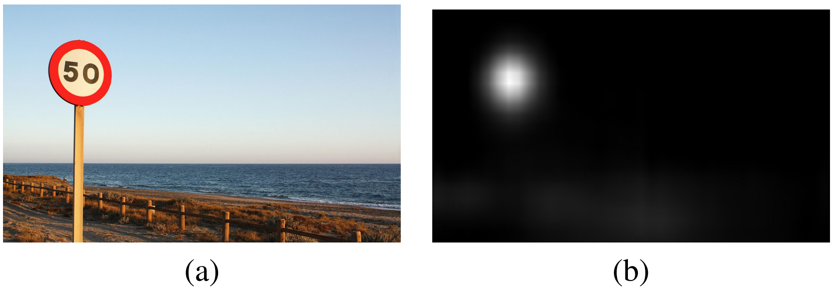

Figure 1. Example of visual saliency. b) is the salience map where brighter pixels represent regions more salient to humans on the original image a).

Visually salient regions on images are usually represented by *saliency maps* (Figure 1). In these maps, images are generated such that areas with high-valued pixels express high saliency on the original image, whereas regions with low-valued pixels represent low saliency. Datasets with such maps are obtained by collecting eye-fixation data from humans while observing the scenes.

1.1. Related work

Early computational models of visual saliency were generally built based on filtering of images for extraction of a pre-selected set of features considered important for *bottom-up* attention. *Vocus* [Frintrop 2005] is a computational model that extracts features shown to be naturally salient to humans such as color/luminance contrast and orientation from different scales of the image.

A rapid change of paradigm occurred around 2015 when *Deep Learning* techniques showed to be very effective in the generation of saliency maps. Models such *Salicon* [Jiang et al. 2015] demonstrated that the use of convolutional neural networks with weights initialized from image classification networks, e.g. *VGG-16* [Simonyan and Zisserman 2014] could lead to salience maps very similar to those generated from humans. *ML-Net* [Cornia et al. 2016] uses the output of different layers of *VGG-16*, combining them in many dimensions and various levels of abstraction. *DeepFix* [Kruthiventi et al. 2015] extends a pre-trained model with new layers that account for global features and center bias, whereas *Salnet* [Pan et al. 2016] explores two models that are simple yet provide good results. These models are usually evaluated and ranked on *MIT saliency benchmark* [Bylinskii et al. 2016a], which uses a variety of metrics to express how close generated saliency maps are to those created from human data. As of today, at least nine out of the ten best models in the ranking use Deep Learning.

1.2. Motivation

Current state of the art models are in general quite expensive computationally, partly because most of them are based on very big pre-trained networks. The convolutional layers of *VGG-16* are composed of around 14.7 million parameters. While pre-trained weights from classification tasks showed to be effective for saliency prediction, it is reasonable to question whether creating a proper network from scratch could yield a smaller amount of parameters that are more efficient for the sole task of salience prediction. Also, there are some ideas from previous work on psychology-based models that were not used in current models but that are considered worthwhile to explore.

1.3. Objectives

This work aims at building a visual saliency model that is a) effective, yielding results similar to other state of the art models, and b) relatively simple and computationally efficient. It is important that both criteria are matched because we aim at extending the model in the future for video and real time applications such as navigating robots.

2. Proposed model

Figure 2 shows the overall architecture of the fully convolutional neural network proposed in this work. It extracts features from increasingly smaller dimensions of the input image.

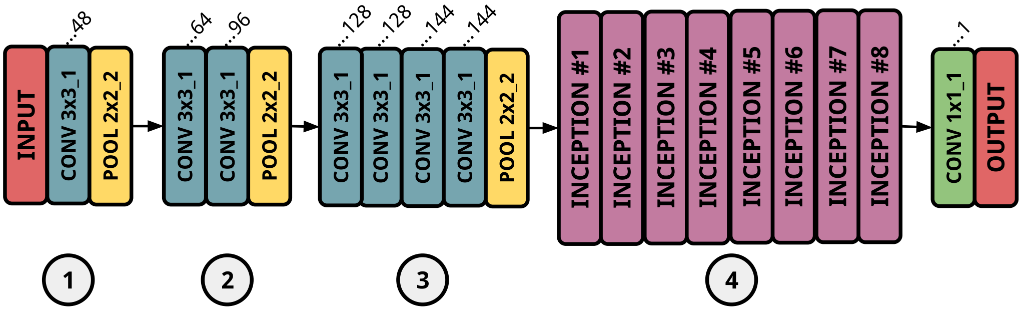

Figure 2. Overview of the network. Filters sizes are in format width×height_stride.

The network is composed of four main blocks:

The first level extracts low level features from the input image, of dimensions $W \times H \times 3$ (width, height, depth), using a single layer with 48 convolution filters with ReLu activation followed by max-pooling that reduces image by a factor of two. It was found that further decreasing the number of filters in this layer considerably hurts performance, which makes sense because it is important to capture high spatial frequency and high contrast information in the context of visual saliency. **The second** level extracts low-medium level features from the input of dimensions $W/2 \times H/2 \times 48$ using two layers with 64 and 96 convolution filters, respectively, followed by ReLU activation and max-pooling. **The third** level extracts medium-high level features from input with dimensions $W/4 \times H/4 \times 96$ using four convolution layers. The first two layers have 128 filters each whereas the last two have 144 filters each. Every convolution layer is followed by ReLu. Max-pooling is carried out at the end. A considerable depth in this level was found to be important for the network's performance. **The fourth** and last level is composed of eight inception blocks that extract high level features from the input with dimensions $W/8 \times H/8 \times 144$. Great level of depth and Inception blocks were found to be very important at this level. A 1×1 convolution makes a linear combination of the output maps at the end of the 8 inception blocks, followed by ReLu, producing the final saliency map of dimensions $W/8 \times H/8 \times 1$.

Figure 3 illustrates the inception architecture [Szegedy et al. 2014] used in each block where filters of size 5×5, 3×3 (both preceded by 1×1 convolutions in order to

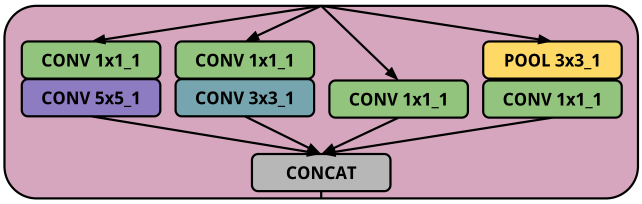

Figure 3. Inception block layout.

reduce the number of input filters), 1×1, and a max-pooling of size 3×3 are applied. Each of these operations is executed in parallel from the same input and the outputs are concatenated at the output. Inception allows the network to use information from different spatial dimensions as well as previous layers (lower level saliency information) in the final map computation, which is considered to be important for visual saliency. The network has a total of 3717841 parameters, a very low number compared to other models. Table 2 details the filter configuration for the inception layers.

Table 1. Number of filters used in each inception block.

Block	pool	conv 1×1	3×3 reduce	conv 3×3	5×5 reduce	conv 5×5
1	96	128	96	192	58	96
2	64	128	80	160	24	48
3	64	128	80	160	24	48
4	64	128	96	192	28	56
5	64	128	96	192	28	56
6	64	128	112	224	32	64
7	64	128	112	224	32	64
8	112	160	128	256	40	80

2.1. Implementation

The network was implemented using *Theano* `0.9.0.dev` along with *Lasagne* `0.2.dev1` on a machine with *Ubuntu 16.04 LTS* and kernel *Linux* `4.8.0-54-generic`. Training was conducted on a GPU *NVIDIA GTX 1080* and the code is available at `https://goo.gl/WzpyYJ`.

2.2. Training

During training, the input images were resized to dimensions $320 \times 240 \times 3$. Moreover, each image is normalized channel-wise by the subtraction of the channel mean and division by the standard deviation. Besides, two aspects are worth mentioning in this pre-training phase: the use of normalizations per image, rather than per dataset, and the conversion of the images colorspace to the LAB colorspace. The choice regarding normalization was made because visual saliency is highly connected to the context of the image, hence saliency depends on the local context. As for the colorspace, *Vocus* [Frintrop 2005] cites that the LAB colorspace is more closely related to human vision once it encompasses red-green, yellow-blue and luminance maps. We conjecture that this colorspace facilitates extraction of important luminance and color contrasts by the learned convolution filters. Prior experiments conducted by our group showed better performance using image-wise normalization and LAB instead of the commonly used RGB.

In order to evaluate the model, the *Correlation Coefficient* between the ground-truth saliency map G and the predicted map P: $CC(P, G) = cov(P, G)/(\sigma(P)\sigma(G))$ was considered. In fact, there is a variety of metrics for evaluating saliency predictions [Bylinskii et al. 2016b], but CC is considered one of the most appropriate because it symmetrically penalizes both false positives and false negatives. However, other typical metrics were also evaluated (as it can be seen in 4).

For training, two datasets were considered: *SALICON* [Jiang et al. 2015], with 15000 images, and *Judd* [Judd 2016], with 1003 images. The network was trained using Stochastic Gradient Descent with Nesterov Momentum of 0.9. SALICON was first used with data augmentation by flipping images horizontally and vertically and the target normalized by mean-std (Last conv layer had ReLu removed in this step). Mean-std normalization of targets was applied because it led to faster convergence. Training iterated for 5 epochs with learning rate of 0.009 and then for 3 epochs with learning rate of 0.001. Then, we switched to using unit normalization on targets. The network was trained for 1 epoch with learning rate of 3×10^{-5} and L2 regularization of 10^{-4}. Finally, Judd dataset was used with data augmentation by flipping images horizontally and the target normalized by unit normalization. Training iterated for 2 epochs with learning rate of 5×10^{-5} and L2 regularization of 3×10^{-5}. Batch sizes were 10 for SALICON and 2 for Judd. The complete training process took around two and a half hours.

3. Results

Prediction took an average time of 8 milliseconds. Figure 4 shows some maps generated by the proposed model. They are generally considerably similar to the ground truth. For evaluation, we submitted the model to the *MIT300 saliency benchmark* that has around 300 images and is commonly used to rank such models. Table 4 shows the resulting values for the most common metrics. The proposed model achieved results comparable to those of the state of the art while having, at least, one fourth of the number of parameters.

4. Conclusion

In this paper, we proposed a novel fully convolutional neural network for the prediction of visual saliency on images. The proposed model architecture and data preprocessing were designed specifically for the task of salience prediction. Our methods showed to be effective, yielding to a network with performance on *MIT300 benchmark* consistently among the ten best results on various metrics while having around $3/4$ less parameters than other state of the art models.

Table 2. State of the art models and metric scores on *MIT300 benchmark*.

Model	Num. parameters	AUC-Judd ↑	CC ↑	NSS ↑	Sim ↑	EMD ↓
Infinite humans	-	0.92	1.0	3.29	1.0	0
DeepFix	≈16.7 million	0.87	0.78	2.26	0.67	2.04
Salicon	≈14.7 million	0.87	0.74	2.12	0.60	2.62
Proposed Model	**3.72 million**	**0.85**	**0.71**	**1.98**	**0.62**	**2.37**
ML-Net	≈15.4 million	0.85	0.69	2.07	0.60	2.53
SalNet	25.8 million	0.83	0.57	1.51	0.52	3.31

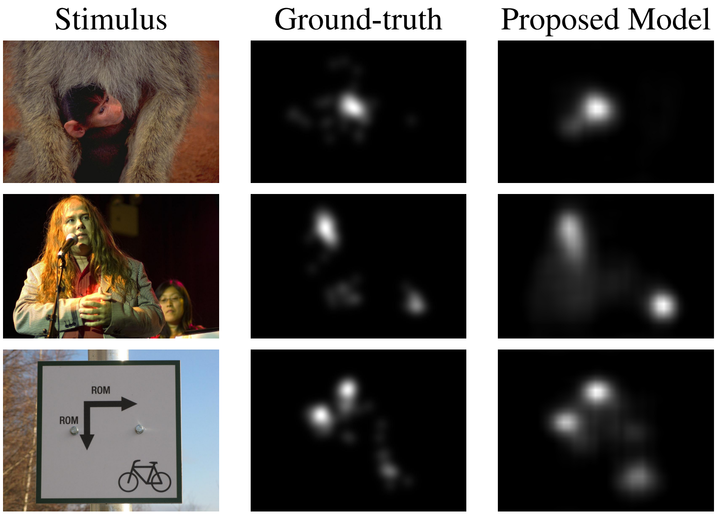

Figure 4. Examples of predictions made by our model.

References

Bylinskii, Z., Judd, T., Borji, A., Itti, L., Durand, F., Oliva, A., and Torralba, A. (2016a). Mit saliency benchmark. `http://saliency.mit.edu/index.html`.

Bylinskii, Z., Judd, T., Oliva, A., Torralba, A., and Durand, F. (2016b). What do different evaluation metrics tell us about saliency models? *arXiv preprint arXiv:1604.03605*.

Colombini, E. L. (2014). *An Attentional Model for Intelligent Robotics Agents*. PhD thesis.

Cornia, M., Baraldi, L., Serra, G., and Cucchiara, R. (2016). A deep multi-level network for saliency prediction. *arXiv preprint arXiv:1609.01064*.

Fixott, R. S. (1957). Evaluation of research on effects of visual training on visual functions. *American Journal of Ophthalmology*, 44:230–236.

Frintrop, S. (2005). Vocus: a visual attention system for object detection and goal-directed search. In *IN LECTURE NOTES IN ARTIFICIAL INTELLIGENCE (LNAI)*. Springer.

Jiang, M., Huang, S., Duan, J., and Zhao, Q. (2015). Salicon: saliency in context. *CVPR*.

Judd, T. e. a. (2016). Learning to predict where people look.

Kruthiventi, S. S. S., Ayush, K., and Babu, R. V. (2015). Deepfix: A fully convolutional neural network for predicting human eye fixations. *arXiv preprint arXiv:1510.02927*.

Pan, J., Sayrol, E., i Nieto, X. G., McGuinness, K., and OConnor, N. (2016). Shallow and deep convolutional networks for saliency prediction. *arXiv preprint arXiv:1603.00845*.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2014). Going deeper with convolutions. *CoRR*, abs/1409.4842.

Treisman, A. M. and Gelade, G. (1980). A feature-integration theory of attention. *Cognit Psychol*.