ADAPTIVE ARTIFICIAL FOVEAL VISUAL ATTENTION

Derek Lee

Professor Curro
ECE-472
Deep Learning

December 17, 2020

Contents

1 Abstract

The concept of modeling Artificial Intelligence based on human intelligence is an idea that has been explored by many in the past. Melício et al. [4] explored applying a biologically-inspired foveal attention model to computer vision. Their model simulated human vision on an image, with high visual acuity near the point of focus, and decreasing visual acuity with increasing distance from that point, with the decrease dependent on a 'fovea size' parameter. I improve on their model by using an adaptive fovea size, along with utilizing an alternative method for computing that adaptive fovea size. I find that using an adaptive fovea size performs better than using a fixed fovea size, with an almost negligible computational cost. However, this does not perform better than using no foveation at all.

2 Introduction

Due to the significant amount of visual information that reaches the eye, the human brain selectively processes the relevant stimuli. When an image is observed, the portion of the image projected onto the fovea is densely sampled, while the periphery is sparsely sampled, limiting the amount of information that the brain has to process. Biological mechanisms, such as saccades, allow the human brain to integrate information from the entire scene.

Similarly, computers also have to process a significant amount of visual information. In images, there is usually information that is irrelevant to the object being classified, which should not be considered. As such, it could be advantageous to 'filter out' this irrelevant information, assisting a neural network by removing potentially distracting information.

My work is inspired by [4], which developed a biologically-inspired foveal attention model. I propose an extension on their model that more effectively focuses on objects in an image. My model utilizes a forward-pass to generate initial top-k labels, followed by a backward-pass to perform object localization, which is used to generate a saliency map. At this point, the model computes an adaptive fovea size, which is used to re-foveate the image. Then, a second forward-pass results in k top-k labels, from which the top-k unique labels are selected.

The main contributions of this paper are the following: First, I evaluate

the performance of my methodology on the same architecture used in [4]. Second, I establish a relationship between performance and the parameter used to compute the adaptive fovea size. Finally, I explore the shortcomings of this model and propose possible solutions for future work.

3 Related Work

The Laplacian Pyramid method, proposed by Burt and Adelson [2], is an image compression method. A Laplacian Pyramid is constructed by applying a Gaussian blur to an image, downsampling that blurred image, and then upsampling the downsampled image and taking the difference between that newly-upsampled image and the original image. All upsampling and downsampling is done by a factor of two. This is repeated to get each subsequent level of the pyramid. With quantization, image compression can be achieved. For the purpose of this paper, a Laplacian Pyramid is used to manipulate the information in an image.

Otsu's method is a technique to perform image thresholding. This algorithm determines an optimal threshold by maximizing the variance between two classes of pixels. This is achieved by computing a histogram of the pixel values in an image, and then performing an exhaustive search to maximize the inter-class variance. This method works best with a bimodal distribution of pixel values.

4 Methodologies

My methodology, inspired by [4], uses their biologically-inspired foveal attention model. The model uses iterative refinement to improve classification. Initially, a foveated image is used in a feed-forward pass through the network. Then, via backward propagation, a saliency map is generated for the top-k classes. Bounding boxes are generated using the saliency maps. The attention is directed to the center of the bounding boxes. The original image is re-foveated at each of those locations and another feed-forward pass is used, resulting in new classifications.

I propose using an adaptive fovea size to focus on relevant sections of the images. During the re-foveation step, the fovea size is calculated dynamically, rather than using a static value.

4.1 Artificial Foveal Visual System (AFVS)

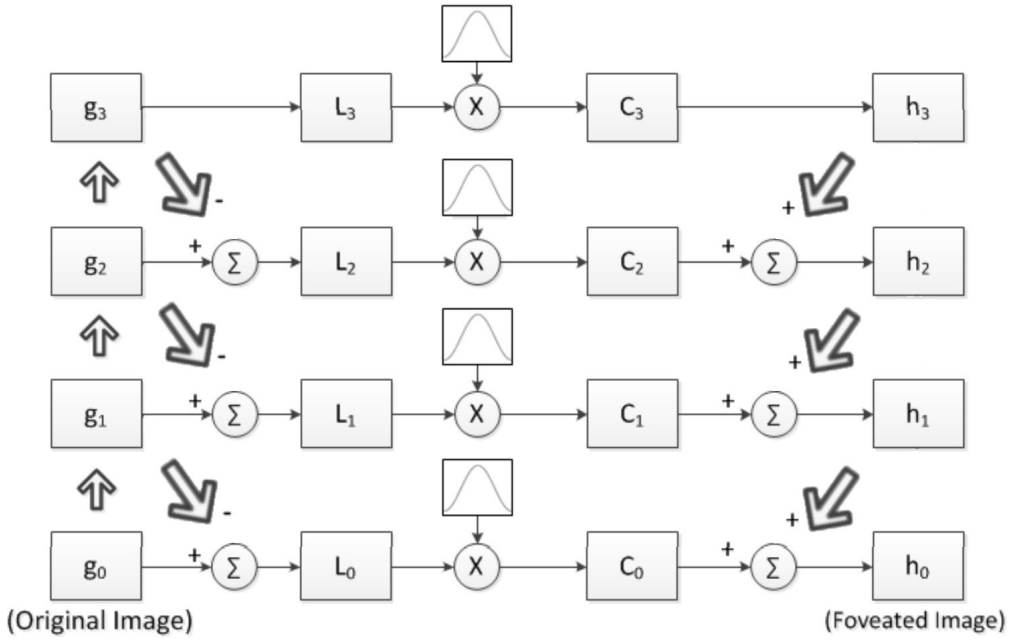


Figure 1: A modified version of a diagram in [1]. A summary of the foveation system using four levels. The thick up arrows represent downsampling followed by a Gaussian blur. The thick down arrows represent upsampling. g_o represents the original image. h_o represents the final, foveated image.

The artificial foveation was based on the Laplacian Pyramid method [2]. To create the artificial foveation, the quantization step involved multiplying each level by an exponential kernel of the form:

$$k(u_o, v_o, f_k) = \mathrm{e}^{-\frac{(u-u_o)^2+(v-v_o)^2}{2f_k^2}}, \quad 0 \leq k \leq K \tag{1}$$

$$f_k = 2^k f_o \tag{2}$$

u_o and v_o are the foveation point, representing the point of focus. f_o is the fovea size, representing the desired amount of high resolution focus.

The saliency map was generated through the backpropagation method proposed by Simonyan, Vedaldi, and Zisserman [5]. The top-5 labels computed during the initial forward-pass were used to generate saliency maps

during the backpropagation step. For each of the top-5 labels, the loss was generated by assigning 1 to the predicted class and 0 to every other class, and taking the difference between that and the logit vector generated through the forward-pass. The model does not use a softmax layer, as recommended by [5]. The gradient was computed with respect to the original image, resulting in a saliency map.

A bounding box was generated using the saliency map. The bounding box is computed by taking the minimum and maximum x and y values of the saliency map. Using the bounding box, the original image is re-foveated. The center of the bounding box is used as the foveation point. The re-foveated image is used in a feed-forward pass. The top-5 labels are taken from each of the re-foveated images, resulting in 5 top-5 labels. The top-5 unique labels are selected are selected from these 25 labels.

4.2 Adaptive Artificial Foveal Visual System

I now introduce an Adaptive Artifical Foveal Visual System. My variation on the foveation system proposed in Melício et al. [4] accounts for the size of the bounding box during the re-foveation step. Instead of only considering the center of the bounding box, this model uses the dimensions of the bounding box to determine the fovea size.

Specifically, the fovea size is:

$$f_o = \alpha * \max(height, width), \quad 0 \leq \alpha \leq 1 \tag{3}$$

$$f_o = \max(30, f_o) \tag{4}$$

where α is a hyperparameter.

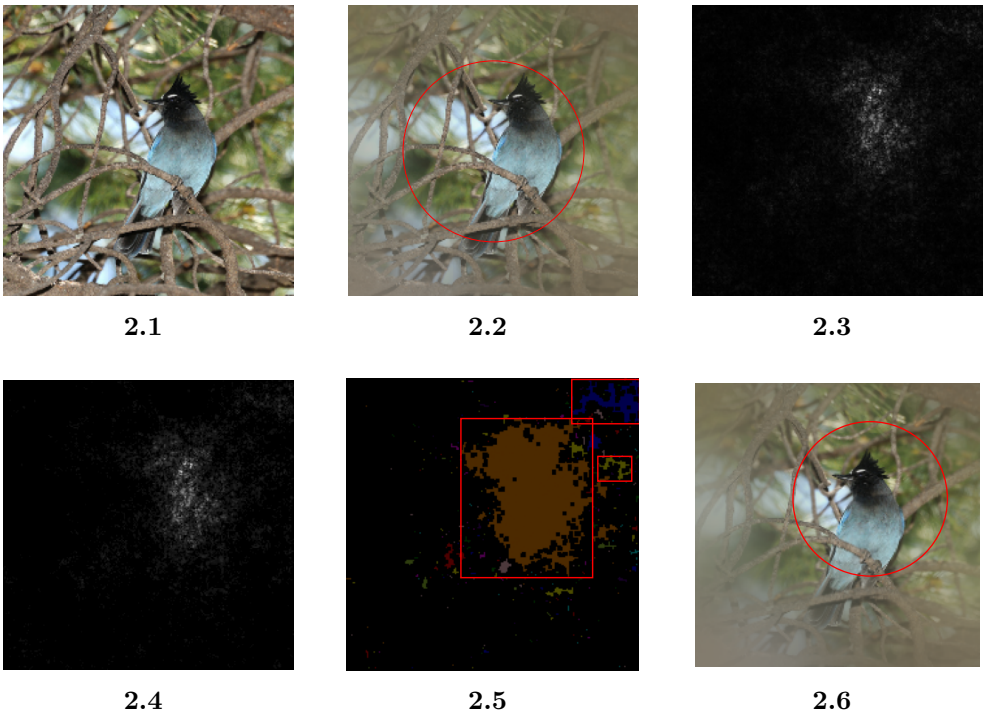
Realistically, α is more constrained. Equation 4 sets an effective limit on the fovea size, because once α is below a certain threshold, $f_o = 30$. In addition, once α is above a certain threshold, there is essentially no foveation, due to the large fovea size.

The purpose of Equation 4 is to provide a minimum bound on the size of the fovea. As noted by [4], a fovea size that is too small is useless, and leads to an accuracy equivalent to random guessing. The goal of the minimum bound to prevent a useless prediction, while also giving the model the flexibility to set a small fovea size to focus on a specific area of the image.

Additionally, instead of using a set threshold to generate a saliency map, I propose finding the largest simply-connected region. First, the saliency

map is thresholded with a threshold value, θ. Any pixel below that threshold is set to 0. Then, Otsu's method was used to threshold the image again. Afterwards, the largest contiguous region was found.

The reasoning for two separate thresholding steps is because Otsu's method assumes a bimodal distribution, which is not necessarily true in a saliency map. Without an initial thresholding step, the largest contiguous region cam span nearly the entire image. With the initial thresholding, the largest contiguous region is more constrained. I speculate that this because a bimodal distribution is forcibly created.



2.1 **2.2** **2.3**

2.4 **2.5** **2.6**

Figure 2: Example of Adaptive AFVS process for "Jay". The red circles represent the focused area simulating the fovea. The red rectangles contain contiguous areas that have an area greater than or equal to 100 pixels.
2.1: Original Image; **2.2**: Foveated Image;
2.3: Saliency Map Before First Thresholding;
2.4: Saliency Map After First Thresholding;
2.5: Simply Connected Regions; **2.6**: Re-foveated Image with $\alpha = 0.5$

Figure 3: Schematic of my model on "Water Snake" using the 5 Top-5 labels. First, the foveated image is passed through the network, generating the top-5 labels. Then, for each label, a bounding box is generated using the saliency map obtained through backpropagation. A second foveation is applied based on each bounding box, with $\alpha = 0.5$. Each of these re-foveated images is passed through the network again, resulting in 5 Top-5 labels. The final Top-5 labels are selected from those 5 Top-5 labels. The red circles represent the focused area simulating the fovea.

Using the 5 Top-5 labels generated during the second-pass, the final predictions are: "Ant", "Slug", "Wolf Spider", "Water Snake", and "Stinkhorn". The correct label ("Water Snake") is predicted during the second-pass.

Using the original Top-5 labels generated during the first-pass, along with the 5 Top-5 labels generated during the second-pass, (6 Top-5) results in different Top-5 labels. The final predictions would be: "Black Widow", "Ant", "Slug", "Wolf Spider", and "Water Snake". Again, the correct label ("Water Snake") is predicted during the second-pass. However, an erroneous label ("Black Widow") is included in the final Top-5, despite none of the second-pass Top-5's including that label.

5 Experiments

I used a pre-trained GoogLeNet, based on the neural network proposed by Szegedy et al. [6], implemented on TensorFlow Hub [3]. Images were resized to 224x224, the default size for the pre-trained GoogLeNet. The value used for θ is the 80th percentile of the saliency values.

The pyramid had 6 levels. This number was selected somewhat arbitrarily. After 5 levels, the subsequent levels on the pyramid have very little variation, due to the small resolution of the levels. It is possible to create a pyramid such that the highest layer is 1x1, but this was not done in this paper.

The following results were obtained on the first 1501-2000 images in the ILSVRC2012 Validation Set.

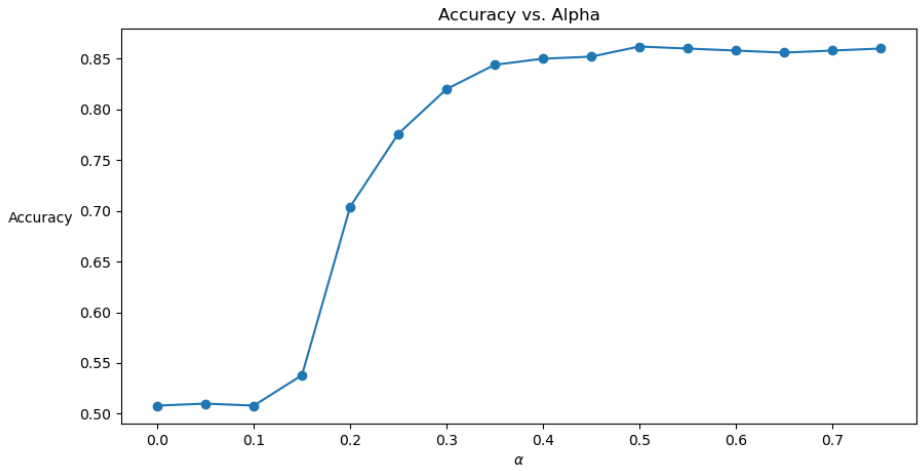


Figure 4: Accuracy as a function of α on the ILSVRC2012 Validation Set

α	**Accuracy**
0.00	0.508
0.05	0.510
0.10	0.508
0.15	0.538
0.20	0.704
0.25	0.776
0.30	0.820
0.35	0.844
0.40	0.850
0.45	0.852
0.50	0.862
0.55	0.860
0.60	0.858
0.65	0.856
0.70	0.858
0.75	0.860

Table 1: Accuracy as a function of α on the ILSVRC2012 Validation Set

As seen in Table 1, the accuracy plateaus around $\alpha = 0.50$. It appears that past this point, the fovea size is so large that there is effectively no foveation in the re-foveated image. Under $\alpha = 0.50$, the accuracy begins to sharply decrease. The accuracy reaches a minimum around $\alpha = 0.10$, likely due to the minimum bound on the fovea size set by Equation 4. It appears that this is due to the significant foveation caused by a small f_o, which obscures important information in the image. $\alpha = 0.5$ appears to be the sweet spot, where there is reasonable foveation; not too little, nor too much.

Method	Initial Foveation	Top-K	Accuracy
Baseline	No	Top-5	**0.868**
AFVS	No	Top-25[1]	0.832
Adaptive AFVS	No	Top-25[1]	**0.856**
AFVS	No	Top-30[2]	0.852
Adaptive AFVS	No	Top-30[2]	**0.862**
Baseline (AFVS)	Yes	Top-5	0.840
AFVS	Yes	Top-25[1]	0.842
Adaptive AFVS	Yes	Top-25[1]	**0.862**
AFVS	Yes	Top-30[2]	0.840
Adaptive AFVS	Yes	Top-30[2]	**0.852**

Table 2: Accuracy on the ILSVRC2012 Validation Set

The AFVS method used a fixed fovea size, $f_o = 70$. This fovea size was used in the backward-pass to re-foveate the image. The Adaptive AFVS used $\alpha = 0.5$.

The Initial Foveation column indicates whether or not the first forward pass used a foveated image. The foveated images pre-processed with $f_o = 70$, $u_o = 112$, and $v_o = 112$.

Top-5 signifies a single forward-pass. Top-25[1] signifies a forward-pass followed by a backward-pass to re-foveate, and another forward-pass using the newly foveated images. Top-30[2] is similar, except that the top-5 labels from the initial forward-pass are used alongside the Top-25[1] labels obtained after the backward-pass.

As can be seen in Table 2, the Adaptive AFVS method performs better than AFVS in every scenario tested. However, similar to [4], the best result is still slightly worse than the Baseline method that did not use any foveation.

[1]Not actually Top-25, but 5 Top-5's
[2]Not actually Top-30, but 6 Top-5's

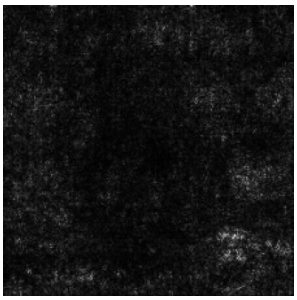
6 Discussion

6.1 Poor Saliency Maps

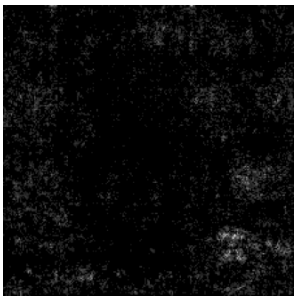
One of the problems with this implementation is that the saliency map generated through backpropagation is not always accurate.

5.1

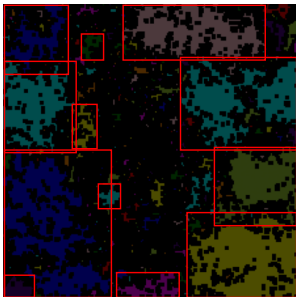
5.2



5.3



5.4



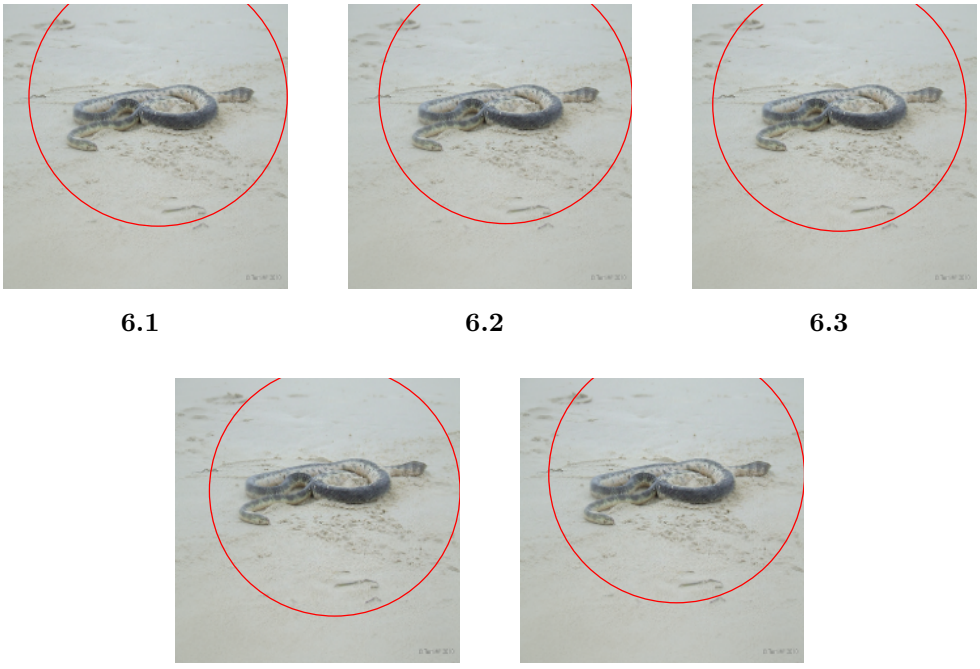
5.5

5.6

Figure 5: Example of a Poor Saliency Map for "Freight Car".
5.1: Original Image; **5.2**: Foveated Image;
5.3: Saliency Map Before First Thresholding;
5.4: Saliency Map After First Thresholding;
5.5: Simply Connected Regions; **5.6**: Re-foveated Image with $\alpha = 0.5$

As can be seen in Figures 5.3 and 5.4, the saliency map is not well-defined. In this example, the object instance comprises the entire image. However, the saliency map is not uniform, with a few clusters in seemingly random locations in the image. This leads to a poor bounding box and thus, a poor re-foveated image. Ideally, the bounding box should be centered in the center of the image, and the bounding box should span the entire image.

6.2 Similar Saliency Maps



6.1 **6.2** **6.3**

6.4 **6.5**

Figure 6: Example of Similar Re-foveations for "Sea Snake".
6.1: "Water Snake"; **6.2**: "Garter Snake"; **6.3**: "Hognose Snake";
6.4: "Rock Python"; **6.5**: "Sea Snake"

As can be seen in Figure 6, the re-foveated images are all very similar. The fovea sizes are almost exactly the same, and without the circles annotating the image, it would be very difficult to tell the difference between the images. This is due to the initial forward-pass resulting in top-5 labels that fall under the same superclass (snakes), or are otherwise very similar. As a result, the backwards-pass produces saliency maps and bounding boxes that are almost identical, resulting in nearly identical re-foveations.

6.3 Errors

Following the procedure outlined in the paper did not result in visually similar images. An additional Gaussian blurring step was used to achieve similar-looking images, with $\sigma_o = 12.5$, and $\sigma_k = 2^k \sigma_o$. This may be due to a miscommunication in the procedure, or a mistake in the implementation.

6.3.1 Mappings

There may be some small error in the calculation of accuracy. The label IDs outputted by the pre-trained GoogLeNet did not correspond to the label IDs provided by ImageNet. A mapping between the label IDs was used. However, the pre-trained GoogLeNet truncated the ImageNet phrases. Thus, some labels were indistinguishable. An example of this is shown in Table 3 and 4.

ILSVRC2012_ID	**WNID**	**Phrases**
782	n03710637	"maillot"
977	n03710721	"maillot, tank suit"

Table 3: ImageNet Label Mappings

Line Number	**Phrases**
639	maillot
640	maillot

Table 4: GoogLeNet Label Mappings

As seen in Table 4, the GoogLeNet Label Mappings have repeated words. However, as seen in Table 3, the two phrases containing "maillot" are "maillot" and "maillot, tank suit". The two phrases in Table 4 cannot be distinguished, so for the purpose of this paper, the labels were mapped sequentially; Line 639 was mapped to ILSVRC2012_ID 782 and Line 640 was mapped to ILSVRC2012_ID 977. It is possible that this may have resulted in an incorrect mapping.

7 Conclusion

In this paper, I proposed a variation on a biologically-inspired model that utilizes human-like foveal vision. My model is an improvement over the previous model proposed by [4]. For an almost negligible computational cost, the accuracy of the model can be increased. In addition, I identity specific shortcomings in the model, and later in this section, suggest future work to address these issues.

Future work can explore utilizing a superclass classifier for the first forward-pass, and a subclass classifier for the subsequent iterations. The labels predicted during the initial forward-pass largely fall under the same superclass, with very similar saliency maps and bounding boxes. Thus, the re-foveation does not lead to a significantly different image for each label.

In addition, because the ILSVRC object instances were mostly centered in the images, the backward-pass did not lead to a significant improvement. It is possible that testing on a dataset with non-centered object instances would show a larger improvement.

8 References

[1] Alexandre Bernardino Ana Filipa Almeida Rui Figueiredo and José Santos-Victor. *Deep Networks for Human Visual Attention: A hybrid model using foveal vision*. Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal, 2017. URL: http://vislab.isr.ist.utl.pt/wp-content/uploads/2017/11/aalmeida-robot2017.pdf.

[2] P. Burt and E. Adelson. "The Laplacian Pyramid as a Compact Image Code". In: *IEEE Transactions on Communications* 31.4 (1983), pp. 532–540. DOI: 10.1109/TCOM.1983.1095851.

[3] Google. *Inception V1*. URL: https://tfhub.dev/google/imagenet/inception_v1/classification/4.

[4] Cristina Melício et al. *Object detection and localization with Artificial Foveal Visual Attention*. Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal, 2018. URL: http://vislab.isr.ist.utl.pt/wp-content/uploads/2018/07/cmelicio_icdl2018.pdf.

[5] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps". In: *arXiv preprint arXiv:1312.6034* (2013).

[6] Christian Szegedy et al. "Going Deeper with Convolutions". In: *Computer Vision and Pattern Recognition (CVPR)*. 2014. URL: `http://arxiv.org/abs/1409.4842`.

9 Appendix

The OpenCV library was used to construct the Laplacian Pyramid. The scikit-image library was used to compute the largest simply-connected region in a saliency map.

Note: All of the following results were obtained using an slightly different foveation method. The method used involved applying a Gaussian blur before downsampling, opposed to the method used to obtain the previous results, which involved apply a Gaussian blur after downsampling.

The following results (in Figure 7 and Tables 5 & 6) were obtained on the first 1501-2000 images in the ILSVRC2012 Validation Set.

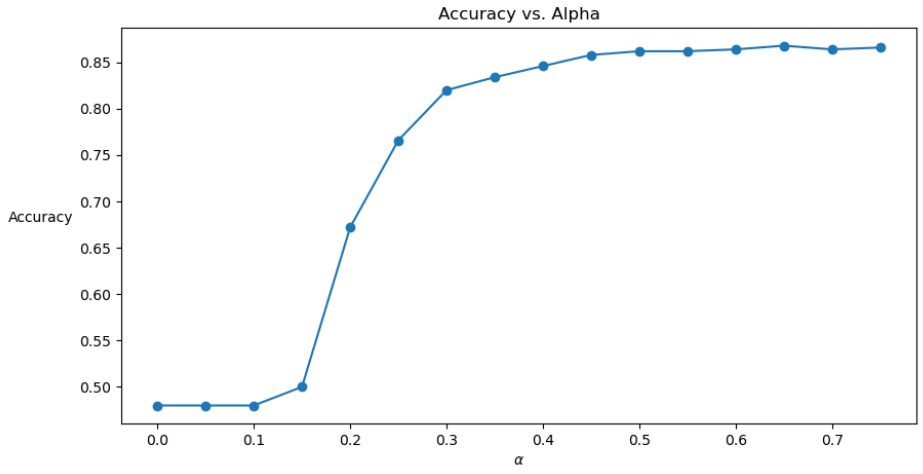


Figure 7: Accuracy as a function of α on the ILSVRC2012 Validation Set

α	Accuracy
0.00	0.480
0.05	0.480
0.10	0.480
0.15	0.500
0.20	0.672
0.25	0.766
0.30	0.820
0.35	0.834
0.40	0.846
0.45	0.858
0.50	0.862
0.55	0.862
0.60	0.864
0.65	0.868
0.70	0.864
0.75	0.866

Table 5: Accuracy as a function of α on the ILSVRC2012 Validation Set

Method	Initial Foveation	Top-K	Accuracy
Baseline	No	Top-5	**0.868**
AFVS	No	Top-25[1]	0.844
Adaptive AFVS	No	Top-25[1]	**0.860**
AFVS	No	Top-30[2]	0.858
Adaptive AFVS	No	Top-30[2]	**0.862**
Baseline (AFVS)	Yes	Top-5	0.840
AFVS	Yes	Top-25[1]	0.842
Adaptive AFVS	Yes	Top-25[1]	**0.862**
AFVS	Yes	Top-30[2]	0.840
Adaptive AFVS	Yes	Top-30[2]	**0.852**

Table 6: Accuracy on the ILSVRC2012 Validation Set

The following results (in Table 7) were obtained by testing on all 50k images in the ILSVRC2012 Validation Set.

Method	Accuracy
Baseline	0.88552
AFVS (Top-5)	0.86214

Table 7: Accuracy on the ILSVRC2012 Validation Set

The AFVS method used a single forward pass. The images were pre-processed with $f_o = 70$, $u_o = 112$, and $v_o = 112$.