Abstract—Image processing methods have made great strides in the last decades. Nonetheless, they are still very computationally expensive. For this reason, in order to more efficiently process a scene, biologically inspired approaches are currently being researched. In particular, in this work, we propose a general computational framework inspired by human vision. This framework is capable of performing object recognition tasks by combining Convolutional Neural Networks (CNNs) with foveal vision techniques. It integrates two dissociated visual attentional methodologies that can be performed sequentially: (1) a Foveal Saliency model capable of orienting the focus of attention to the regions of interest, and (2) a Pyramidal Focus model to perform object recognition when the system has already fixated the target. Using a partition of the ILSVRC 2012 validation data set, and a pre-trained GoogLeNet network, we conducted several tests to evaluate the trade-off between the recognition performance and the computational cost, and which model parameters influence it. We demonstrated that by using the Foveal Saliency method there is a significant increase on the classification performance for non-centered objects when using at least two saccade iterations. As for the Pyramid Focus model, although the computation time increases linearly with the number of pyramid levels, it consistently outperforms the classical approach of resizing the image to the size of the network. These promising results highlight the importance of developing biomimetic visual solutions for visual tasks.

Keywords: Visual Attention, Convolutional Neural Networks, Object Recognition, Foveation, Saliency, Gaussian Pyramid

I. INTRODUCTION

The visual acuity attained by the human eye is not uniform. In fact, the resolution of the captured images is higher in the fovea, the small central region of the eye, decaying drastically as we approach the periphery [1]. This anatomical structure acts as a space-variant sensor that samples images with detailed information only at the point of gaze, coding progressively less information further away from it.

This non-uniform visual sampling leads to the need of moving the eyes in order to process an entire scene. However, the amount of visual stimuli in the real world that reaches the eyes (0.1 to 1 Gbits/s for each) is too high for the available brain resources, making it impossible to process an entire scene with the high resolution part of the eye [2]. Hence, humans have mechanisms of visual attention used to reduce the amount of information that is processed. These mechanisms select only the relevant parts of an image and prioritize them in time. These parts are known as the salient parts.

In cognitive psychology [3], visual attention is considered to operate as a two-stage process. In the first stage, attention is distributed uniformly over the visual field and the extraction of saliency information is performed in parallel. In the second stage, the attention is focused to a specific area of the visual scene, and processing is performed in a serial fashion.

 Likewise, in computer vision, robots share similar resource limitations when processing image information in real-time [4]. Conventional computational vision solutions are based on the increase of the number of pixels in an image which results in an increase of the amount of raw data to be processed. Common camera sensors sample the real world uniformly, capturing unnecessary parts of the visual field that can be distracting. The main differences between these systems and the human vision system are the sensor topology and the system feedback.

A new set of techniques known as deep learning has led to exciting new developments in the field of artificial intelligence [5]. Among these techniques Convolutional Neural Networks (CNNs) are of particular interest for computer vision. These networks are inspired biologically on the connectivity pattern between neurons existing on the visual cortex of mammals. Thus, they have made the development of sophisticated systems for specialized vision tasks possible. Despite all this progress, the problem of creating more general visual models that replicate human vision remains challenging. With this goal in mind, much effort has been made towards understanding the efficient human vision system and designing algorithms based on foveal inspired sensors and selective attentional mechanisms of feedback.

The main goal of this work is then to study the need for human inspired vision systems capable of performing object detection and recognition when computational resources are limited. To this end, we developed two methodologies, anatomically and physiologically inspired by the human visual attention mechanism. These two systems integrate a foveal inspired way of sampling the image, an active vision strategy to control the gaze, and a selective visual mechanism to analyse only the relevant regions of the image. Moreover, the brain processing is mimicked by the use of pre-trained CNNs for feature extraction in order to choose where to look and to replicate the high level cognitive object recognition processing.

In our work we developed two dissociated visual attention methodologies that represent a proof-of-concept of an integrated biomimetic solution for detecting object classes present in images: (1) a Foveal Saliency model capable of orienting the focus of attention to the regions of interest, and (2) a Pyramidal Focus model to perform object recognition when the system has already fixated the target. We intend to evaluate the trade-off between the recognition performance and the computational cost, and which model parameters influence it.

II. BACKGROUND

A. Visual Attention

Over the years, several definitions of visual attention have been proposed. However, the one that is most widely accepted today is selective attention. This is due to the attempt of solving the problem of the lack of cognitive resources. This concept consists of processing with higher detail only the relevant sub-regions
of the visual field, called focus of attention, which are focused through selection mechanisms.

A visual stimulus is something present in the visual field that can trigger our attention depending on the degree of importance. The more relevant ones are denominated salient. The orientation of the focus of attention to process regions of interest is called overt attention which is associated with the movement of the eyes, head and body onto those directions. However, it is also possible to attend to peripheral areas without moving the eyes (covert attention). In visual search tasks, covert attention and saccadic eye movements (rapid, simultaneous movements of both eyes in the same direction produced by overt attention) usually have a joint role: a salient feature triggers the attention and through a saccade the eye fixates the region of interest enabling perception with higher resolution. However, the eccentricity effect created by the physical structure of the retina, i.e. a higher resolution in the center relative to the periphery, makes it difficult to detect elements in peripheral areas.

There are two major categories of factors that drive attention to certain objects or locations: bottom-up and top-down factors [6]. Bottom-up factors are driven by stimuli generated by discriminative features of the visual scene. If a feature stands out from its surroundings, it is automatically and involuntarily perceived. This suggests that the saliency of visual field features is formed in the brain before the focused attention itself by a pre-attentive process. Some features are intrinsically more salient in a given context, for example a black ball in the middle of white balls (the salient feature is the color). On the other hand, top-down factors are generated by a goal and are influenced by knowledge, expectations and objectives [7]. Even if objects appear in known settings, the attention driven by these factors is slower because it requires focal attention.

According to Feature Integration Theory (FIT) [8], object perception is made in two stages: (1) a pre-attentive stage where the features are perceived automatically and in parallel across the visual field, followed by (2) a focused attention stage to sequentially process the stimuli locations to identify the objects present in the scene. During the pre-attentive stage, the basic features, such as color, orientation, intensity and contrast are extracted simultaneously and registered in feature maps, through a bottom-up process. These feature maps - topographical maps that highlight image regions corresponding to the respective feature - are merged together in a master map of locations. This map specifies the locations in which features have been detected, but not what objects are there. So in order to identify them, one has to scan sequentially, through focal attention, those salient regions present in the map. At each foveation, features currently at that location are attended and stored in object files. Thus, when the object is familiar, the top-down processing produces associations between the object files and the prior knowledge, resulting in the object recognition.

B. Deep Learning

In recent years there have been great advances in the field of artificial intelligence provided by a new set of learning techniques known as Deep Learning. These advances, according to LeCun, Bengio and Hinton [5], were only achieved due to the development of more powerful, dedicated hardware such as Graphics Processing Unit (GPUs) and the creation of large sets of labeled data [9], [10]. Deep Learning is a useful framework for many different machine learning algorithms to model data with a high level of abstraction through the implementation of deep artificial neural networks inspired by animal brains.

Machine learning is a category of algorithms that allows computers systems to effectively perform a specific task without explicit instructions. Those algorithms are mathematical models that receive input data and based on it use statistical analysis to make predictions or decisions.

Image classification is a classical computer vision task tackled by this kind of algorithms. It consists of assigning a class or probability of belonging to a class to an input image from a set of fixed classes. This task can be very challenging due to the amount of intra-class variability. This problem can be mitigated by using Deep Neural Networks (DNNs), instead of low-level of abstraction features manually selected (such as edges, textures, colors), since they can learn to extract high-level features with increasing abstraction as the number of layers grows.

However, DNNs are characterized for being over-parameterized, requiring a huge amount of labelled data. This data acquisition process often requires manually annotations that can be exhausting and expensive. Thus, in some computer vision problems such as object detection, segmentation, key points detection, performing a weakly supervised learning process from weaker annotations can be useful, for instance localize object bounding boxes only with image class labels (without any bounding box annotations).

C. Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are a class of deep artificial neural networks that are biologically inspired on the visual cortex of mammals. These architectures were first introduced with Neocognitron [11] and have been widely applied in computer vision tasks such as object classification [12] and object detection [13] due to their structure. By this we mean, CNN architectures exploit spatially local correlations by enforcing a sparse local connectivity pattern between neurons of adjacent layers: each neuron is connected to only a small region, the receptive field, of the previous volume. Another advantage of these architectures is parameter sharing that allows a reduction of the number of training parameters and contributes to translational invariance. The intuition behind this is the fact that if a feature detector is helpful in one part of the image, it is also probably useful in another part of the image. Thus parameters are shared along the spatial dimensions (width and height) of the representation.

The architecture of a CNN is typically a stack of different types of layers that apply different transformations to the representation with the objective of extracting task-specific features. Usually the layers are as follows:

1) Convolutional Layer: It is the principal component of a CNN architecture. It is composed of filters with a small receptive field, to extract specific features. Depending on the hyperparameters such as number of filters, kernel size, stride and padding, convolutions are computed from the dot product between the entries of the filter and the input volume thus producing a 2-dimensional activation map of that filter.
2) **Pooling Layer:** The purpose of it is to progressively downsample the representation along spatial dimensions (width and height) to reduce the amount of parameters and computation in the network, and hence controlling overfitting [14]. This layer is periodically inserted in-between successive convolutional layers. The most used types of Pooling are: the Average Pooling, which calculates the average of each region; and the Max Pooling which determines the maximum present in a neighborhood.

3) **Fully Connected Layer:** In this layer each neuron is connected to all activations in the previous layer. This is typically placed at the end of the network to compute the class scores of the image. By assigning a softmax activation function [15] the outputs can be interpreted as probabilities of the initial image belonging to a certain class.

### III. Methodologies

The two methodologies proposed here are influenced by the FIT, already described in section II: a Foveal Saliency (FS) methodology that extracts salient features and then orients the focus to those regions of interest, and then a Pyramid Focus (PF) method that increases the recognition accuracy when the visual system finally fixates a target. To mimic the human visual attention these two methods are based on pre-trained CNN architectures to help recognize the object. Distinct to the traditional way of using these networks, we employ different artificial foveal systems to replicate the human anatomy of the retina. Although these two methods are presented here in a dissociated way, they represent a proof of concept of an integrated biomimetic solution for detecting object classes present in images.

#### A. Foveal Saliency Method

The FS method used in this work corresponds to the one previously developed by us in [16]. The correspondent scheme, presented in Fig. 1, can be decomposed into two phases. First, in a bottom-up fashion, the model performs a first object classification, through a feedforward CNN pass with a foveated image created by an artificial foveal visual system. Then, in a weakly supervised process, through feature saliency extraction, the model computes the next foveation point by performing a feedback propagation in the CNN according to top-down information. These two phases are sequentially repeated to refine the object recognition task.

One of the bottlenecks of classical saliency models is feature selection. The amount of features that exist to be extracted is overwhelming and choosing just a few to have a generalized overhang prediction is quite complicated. For example, Itti’s model [17] extracts pre-defined features through linear filtering such as colors, intensity and orientations that are combined in order to identify the salient location of the object. To overcome this bottleneck, feature selection is employed through a pre-trained CNN model that has learned to extract high-level features in order to produce a more abstract representation of the input.

This methodology is an active vision system capable of improving object detection and recognition that tries to replicate the human saccadic mechanism and the foveal eye anatomy. It was mainly based on the work developed previously by Almeida [18] that was inspired in the object recognition process proposed in *Look and Think Twice* by Cao [19].

Cao’s method attempts to mimic the human visual strategy described in FIT, where after the first glimpse of a scene some relevant parts are inspected with focused attention to identify the object. The process consists in first analysing the full image at a coarse scale through feedforward CNN classification to obtain a set of the most probable object classes. Then in a weakly supervised fashion by feedback propagation of each top predicted classes, the salient regions of the possible object locations are patch cropped and re-classified in a zoomed-in scale. However, when recognizing objects with attention it is important to have some context of a scene, so in Almeida’s work [20] instead of cropping a bounding box around the object, a human like foveated image centered on the object is presented. In this latter work the re-classification scheme proposed in the former is also used.

#### 1) Artificial Foveal Visual System:

The purpose of this artificial foveal visual system is to generate foveated images that are able to replicate the non-uniform distribution of the receptive fields in human eyes. This is represented by the first block in Fig. 1. It was inspired by the Laplacian Pyramid technique proposed by Burt and Adelson [21] for image compression, which is extremely fast and easy to implement and has been applied to real-time image processing and pattern recognition.

First, the artificial foveal visual system begins by generating a Gaussian Pyramid. A Gaussian Pyramid is a collection of images, all arising from a single original image, that are successively lowpass filtered using a Gaussian average (gaussian blur). Each subsequent image level, $g_{k+1}$, is filtered through the convolution of the previous level, $g_k$, with a $5 \times 5$ Gaussian mask $\omega$:

$$g_{k+1}(u,v) = \omega * g_k(u,v) = \sum_{s=-2}^{2} \sum_{t=-2}^{2} \omega(s,t) g_k(u-s,v-t)$$  \hspace{2cm} (1)$$

where $u$ and $v$ are the image level coordinates. The Gaussian mask $\omega$ is computed from the isotropic 2D Gaussian kernel.
represents a significant reduction in the computational cost. Also, decomposed in a series of two 1D Gaussian convolutions, one since it is a very efficient way of obtaining a possible object the resulting images the foveated image is created (Fig. 3).

The Gaussian mask is obtained by discretizing the Gaussian kernel. This theoretically requires an infinitely large convolution kernel, as the Gaussian distribution is non-zero everywhere. Fortunately, this distribution decays abruptly, approaching zero about three as the Gaussian distribution is non-zero everywhere. Therefore, the Gaussian standard deviation and also corresponds to the cut-off frequency of the filtering. The size of the mask influences the amount of blur: a larger size corresponds to a larger convolution mask and thus to a greater degree of filtering. The Gaussian mask is obtained by discretizing the Gaussian kernel. Therefore, the discrete approximation of a Gaussian function with \( \sigma = 1 \), presented in Figure 2, is a \( 5 \times 5 \) mask.

Since the 2D Gaussian kernel is isotropic, it is separable into \( x \) and \( y \) components. This means that the 2D convolution can be decomposed in a series of two 1D Gaussian convolutions, one in the horizontal direction and one in the vertical direction. This represents a significant reduction in the computational cost. Also, the iterative pyramid generation is equivalent to convolving the original image with gaussian kernels with \( \sigma_k = 2^{k-1} \sigma_1 \) where \( k \) is the level of the pyramid and \( \sigma_1 \) is the standard deviation from the first level. In this way, the frequency and therefore the sample interval is reduced by an octave from level to level. This results in a reduction of each image resolution by a factor of 2 along each coordinate direction.

Secondly, the Laplacian pyramid levels are computed as the differences between adjacent Gaussian levels. They represent the error images and result from the up sample of the Gaussian images in order to have compatible resolution to enable the computation of pixel-wise differences.

In the next step, to mimic human vision with a high resolution in a fovea of size \( f_0 \) and a lower resolution in the rest of the retina, each Laplacian level is multiplied by an exponential weighting kernel of the form

\[
k(u, v, f_k) = e^{-\frac{(u-u_0)^2 + (v-v_0)^2}{2f_k^2}}
\]

where \( f_k = 2^k f_0 \) is the exponential kernel standard deviation at the \( k \)-th level and \( (u_0, v_0) \) is the foveation point which defines the focus of attention. Finally by expanding and summing all the resulting images the foveated image is created (Fig. 3).

2) Iterative Weakly Supervised Object Detection: In our model we use a weakly supervised detection process described in [19] since it is a very efficient way of obtaining a possible object location. The process uses class-specific saliency maps to encode the possible location of the object of a given class in an image, and thus can be used for object detection. These saliency maps are extracted through a single backward pass in a CNN trained only on the image labels, so no additional annotations are required (such as object bounding boxes or segmentation masks).

According to Simonyan’s findings [22], it is possible to obtain an image-specific class saliency map via a feedback propagation in the network. This part of the method is present in the second block of the scheme in Fig. 1. Given an image \( I \) and a class \( c \) the CNN classification output score \( S_c(I) \) is highly non linear, therefore it is useful to linearize it through a first-order Taylor approximation in the neighborhood of \( I \) as

\[
S_c(I) \approx G_c^T I + b
\]

where \( b \) is the bias of the model. The term \( G_c = \frac{\partial S_c(I)}{\partial I} \) can be viewed as a measure of how likely it is that pixels of image \( I \) are important for the classification of a class \( c \) and therefore can help detect that class in the image. The pixel derivatives are found by back-propagation until the first input layer, that correspond to the input image. The back-propagated error values are the difference between the output of the CNN softmax layer and the desired output vector, that corresponds to assigning 1 to the element associated with the specific class we want to detect and 0 to all the other inputs. \( G_c \) defines the class specific saliency map of the image \( I \). Since the images used are RGB, a single class saliency value for each pixel \( M_c(i, j) \) is obtained by taking the maximum magnitude of \( G_c \) across all colour channels \( l \),

\[
M_c(i, j) = \max_{l \in \text{rgb}} |G_c(i, j, l)|.
\]
B. Pyramidal Focus Method

The PF method used in this work follows the one previously developed by us in [23]. Assuming the target was already fixated and centered in the visual field, this method corresponds to a recursive process described in FIT that allows for an improvement of the performance on the object recognition task. This architecture takes advantage of the human-inspired foveal vision based on low-pass multiresolution pyramids [24].

1) Artificial Foveal Visual System: The PF model is achieved by Gaussian pyramid coding [21], already explained in section III-A. In order to have more levels for each image, each successive pyramid level contains half of the pixel number of the previous level. This means that the downsampling factor is $\sqrt{2}$ in each coordinate direction, and not the typical downsample factor of 2 used in Gaussian Pyramids. Considering an $n \times m$ input image, the number of Gaussian pyramid levels, $N_L$, is given by the maximum number of times one can reduce the image by a factor of $\sqrt{2}$ in the smaller image dimension, until it reaches a value $N$, where $N \times N$ is the size of the input layer of the CNN pre-trained architecture. Formally, it is represented by

$$N_L = \min \{ \max \{ N_L \in \mathbb{Z} : n2^{-N_L/2} \leq N \}, \max \{ N_L \in \mathbb{Z} : m2^{-N_L/2} \leq N \} \} .$$  \hspace{1cm} (6)

To mimic the human foveal vision each pyramid level is patch cropped in the center with the input size of the network. In this way, the first level of the pyramid represents the fovea with the same resolution as the input image, that decreases as the pyramid level increases (Fig. 4).

2) Joint Object Recognition: In order to identify the object present in the image, each pyramid crop is loaded into a pre-trained network and the output probabilities of the softmax layer from each pyramid crop are averaged with an exponential moving average (EMA) to obtain a final joint prediction. EMA is applied recursively to each increasing level of the pyramid $n$ until it reaches the level $N_L$ according to

$$S(n) = \begin{cases} Y_n, & n = 1 \\ \alpha Y_n + (1 - \alpha)S_{n-1}, & n \geq 1 \end{cases}$$  \hspace{1cm} (7)

where $Y_n$ is the score vector of the prediction with the crop of the pyramid level $n$ and $S(N_L)$ is the score vector of the model prediction of each image. The coefficient $\alpha$ represents the degree of weighting decrease, a constant smoothing factor between 0 and 1. The greater the constant $\alpha$ the greater the penalty of the scores of the prediction with the crop of the lower pyramid level. Therefore, higher levels of the pyramid that have more context information have more emphasis on the final prediction than lower levels. This implementation is better than a simple average filtering mainly for two reasons: (1) the context, since higher levels contain more context information useful for recognition then lower levels; (2) the object size, since for larger objects the crops from lower levels of the pyramid in some images do not catch the entire object.

IV. TESTS AND DISCUSSION

To evaluate the performance of our models we compute the top-1 classification error according to the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) of 2012 metrics [9]. Thus, for each image we consider an error if the ground truth does not match the first class label predicted. Then, the classification error of a model is given by the average error over all images.

We will also estimate the computational cost associated with the object recognition. We define it as the amount of information, expressed simply by the number of bytes encoded in the image, that is going to be processed. This estimation is necessary to evaluate the trade-off between the performance and the computational cost associated with the algorithms. What we mean by this is that we want to first understand the amount of information processed by each model is greater or not than the information processed by the baseline solution. Secondly, in case it is greater, if we can achieve better classification performance.

The dataset used to perform our tests was a partition of the ILSVRC of 2012 validation dataset [9]. The size of the images that can be part of this partition is constrained by the PF properties, since the method needs to generate at least two levels of the Gaussian Pyramid with a downscale factor of $\sqrt{2}$ in each dimension. So, in order to have consistent results across models and to be able to compare them, our dataset partition is composed of the 7100 images selected from the ILSVRC 2012 validation dataset that have a resolution greater than $454 \times 454$.

As explained in section III, our models integrate pre-trained CNN architectures. For matters of consistency, we chose to use always a GoogleNet [25], [26]. Thus, our baseline is a single traditional feedforward pass in that network for an image with uniform resolution. The original image has to be first resized to the resolution of the input layer which is $227 \times 227$.

A. Foveal Saliency Method

We begin by describing the steps of which the model consists:

1) Resize the original image to the CNN architecture input size.
2) Foveate the resized image centered in a foveation point $(u_0, v_0)$ with a certain fovea size $f_0$ via the artificial foveal visual system.
3) Run the CNN model with the foveated image and predict the top $k$ class labels with a feedforward pass.

4) For each of the top $k$ class labels, compute each object’s bounding boxes with top-down back-propagation.

5) For each of the $k$ object location proposals foveate the original image with the foveation point in the center of each bounding box and predict again the top $k$ class labels with a feedforward pass.

6) Given the total $k \times k$ labels and the corresponding confidences, sort them in descending order and choose the top $k$ as final prediction.

7) Repeat from 4.

In the implementation phase we had to choose the values of some hyperparameters. In the artificial foveal visual system we set the Gaussian standard deviation of the first level as $\sigma_1 = 1$ and the number of levels to be five since these values generate a good decay of the blur employed around the foveation point. Then, differing from the previous work [18], where the first foveation point $(u_0, v_0)$ was fixed to be on the image center, we made it a free parameter. In this way we are able to identify objects laying not only on the center but also in the periphery of the image, thus mimicking human-like vision. The fovea size, $f_0$, was also set to be a free parameter. In Fig. 5 we represent different resulting images from our artificial foveal visual system with different fovea sizes $f_0$ and foveation point $(u_0, v_0)$.

For the weakly supervised detection procedure, we had to apply a threshold to the image-specific class saliency map in order to obtain a segmentation mask. Since this parameter does not influence the classification error we set it to $\theta = 0.65$. This value was chosen because it is the one that generates a better compromise between the number of salient pixels in the saliency map and the correct bounding box (discussed in [18]). Also, we chose to do the backpropagation phase with the top-5 class labels, setting $k = 5$. Therefore, we obtain the five possible bounding boxes associated with each class label.

In Fig. 6 we present the weakly supervised detection for an image whose ground truth is red fox. We show the image-specific saliency map, followed by the segmentation mask, and the bounding box of the possible object locations. This was obtained via backpropagation of the top five class labels from the first prediction (kit fox, red fox, grey fox, arctic fox and coyote) without applying the artificial foveal system to the input image. Each saliency map and respective segmentation mask is different. Since the ground truth label is red fox the second image is the one that has more pixels highlighted in the snout, showing that those features are very relevant for intra-class distinction.

The first experiments performed had the goals of analysing the effect of the foveation point $(u_0, v_0)$, the fovea size $f_0$ and the iteration number on the classification performance of the model. In order to understand how the foveation point of the first feedforward pass influences the classification error, we made it vary along a 11 by 11 grid on the resized image domain. Therefore, for each foveation point we foveate with different fovea sizes, compute the classification error, and then average it. Afterwards, we do this for all images and average again, obtaining the classification error for one foveation point. Finally, we repeat this process for all the points considered on the grid.

In Fig. 7 we show, for both the first and the second iteration of the algorithm, the top-1 metrics as a function of the foveation point. The classification error is minimal in the center, and it increases with the distance to it. Also, from the first feedforward classification to the second, we verify that there is an improvement in the performance, and even the first foveation points on the corners manage to achieve better recognition error. These results are a consequence of the ImageNet dataset being mostly composed of images with centered objects, as is demonstrated by the Fig. 8, where the distribution of the center of the bounding boxes in the resized image domain is.

To analyse how the fovea size affects the classification error we varied $f_0$ between 0 and 130 and we ran the model for 6 iterations for $(u_0, v_0) = (113, 113)$ and other 16 foveation points spread in a grid over the resized image. In the previous work [18], the foveation point was fixed to be in the image center, but in natural vision tasks objects can be located in the periphery. As our model iteratively converges to the possible object locations we want to compare the performance for the first foveation point in $(u_0, v_0) = (113, 113)$ against the other chosen positions. For this latter case the classification error is an average over the 16 different foveation positions.

In Fig. 13 we show the top-1 classification error as a function of the fovea size, for 6 iterations of the FS, where the shadow on the right graph represents the uncertainty associated to the different foveation points explained beforehand. Moreover, there is the baseline solution error for the dataset, which we include.
Fig. 7: Top-1 classification error (in %) as a function of the initial foveation point \((u_0, v_0)\). For each point the error was computed over all \(f_0\). On the left is the top-1 error for the first foveation, whereas on the right is the top-1 error for the second foveation.

Fig. 8: ImageNet partition dataset distribution of the center of the bounding boxes in the resized image domain.

In order to understand if our model outperforms it.

On one hand, one of the first aspects we can infer from all the graphs shown is that the error decreases as the fovea size increases, reaching the baseline performance. This was an expected result since smaller fovea sizes represent more blur in the foveated image and therefore less context in the recognition. On the other hand, analyzing the graph of the center foveation (Fig. 13 (a)), it appears that there is practically no reduction of the classification error over iterations, whereas when considering a free initial foveation (Fig. 13 (b)) we verify that our model has a significant gain from the first to the second iteration. If the first foveation is not focused on the location of the object (which in real life is the case, because when we want to find an object we do not know where it is) our model is useful, as it shows the importance of having an active vision system. Also, another way of demonstrating the need for an active vision in this scenario is the fact that with just a single iteration we could not reach the baseline.

From the first iteration, independent of the foveation point, a saliency map is extracted. This map contains important features of images regardless of the applied blur. Although these features are not fine enough to obtain a correct recognition, they are sufficient to obtain a possible location of the object. Therefore, in a second iteration, we already have the foveation point focused on the possible object location, and thus we achieve our best prediction for the fovea size. From this iteration on there is no performance gain. This shows two things: that our model can converge to its best recognition in two iterations, and that the value it converges to is the limit of the performance for each fovea size, i.e., no matter how many iterations we perform the model saturates at around the second.

The value where the error of our model and the baseline match is around \(f_0 = 90\). On one hand, this means that there is also a saturation of the information needed to recognize the object. We can thus achieve the baseline performance without processing all the information contained in the original image. This happens since the artificial foveal visual system reduces the information compressed in the image, since it applies a blur filter and an exponential weighting. We can thus achieve the baseline performance without processing all the information contained in the original image.

In FS method we are not considering any weighting from previous iterations in the current prediction, thus we conclude that it is never going to reach a better performance than the baseline solution. This is because at most, in each iteration the model processes the resized resolution image (that is processed by the baseline solution), the model is only able to approach the baseline solution. However, the objective of this experiment was to show the relevance of having an active vision system capable of directing the focus of attention to the relevant regions of the image, in analogy with human vision.

Since the FS method processes images that are first low-passed by a Gaussian filter and then exponentially weighted in order to create foveated images, one can expect that these images have less information and thus can be more compressed than the respective resized image (with original resolution). The information compression that arises from employing our artificial foveal visual system depends on the fovea size and on the image resolution. The latter we considered fixed, since all images in our experiments had to be first resized to \(227 \times 227\) so that we could load them into the network. Therefore, through the jpeg algorithm [27] we computed the compression gain as a function of the fovea size. This compression gain was obtained from the division between the information encoded on the foveated images and the information encoded on the respective original...
Fig. 9: Top-1 classification error (in %) as a function of the fovea size $f_0$ over 6 model iterations. On the left is the top-1 error for a single first foveation point on the image center $(u_0, v_0) = (113, 113)$, whereas on the right is the top-1 error considering several foveation points spread over a grid. The shadow around the curves represents the uncertainty of the foveation points.

Fig. 10: JPEG Compression (in %) between foveated and cartesian images.

images, for different quality factors. The quality factor is a weight between 1 and 100 that is associated to the image quality generated by the jpeg algorithm. Thus, the higher the quality factor, the better is the image quality.

In Fig. 10 where these results are presented, we can verify that the compression also reaches a saturation fovea, that is consistent with the saturation point of the classification errors. This means that the reason why the model achieves the same performance as the baseline is that for those foveas the amount of information contained in the images is practically the same.

A compression gain of 50% means that the computational cost of processing two foveated images is equivalent to processing the respective original resized image. The best curve to analyse, i.e. with the higher compression, is $Q = 90$, where for a fovea size of $f_0 = 40$ we can achieve that compression gain of 50%. However, for that fovea value the model never reaches the baseline solution performance, as it is shown in Fig. 13. Therefore, we conclude that in terms of efficiency the FS model is worse than a single feedforward classification, meaning that for that same computational cost we only achieve a top-1 classification error of 49.3%, whereas the baseline error is 29.0%.

B. Pyramidal Focus Method

We begin by describing the steps in which the model consists:

1) Compute $N_L$ Gaussian pyramid levels according to the image resolution.

2) Patch crop each pyramid level image in the center with the size of the CNN input layer. This pyramidal representation consists of the foveated image.

3) Run the CNN model for each $N_L$ cropped images.

4) Average the output probabilities obtained from the feedforward classification with each image crop to obtain a joint prediction.

In the joint object recognition part we set the weighting factor of the exponential moving average $\alpha = 0.6$. In Fig. 11 we show three examples of the PF method whose ground truth labels are trimaran, rock beauty and tusker. On the left we present the resized images followed by the patch crops generated by the model. The difference in the number of columns is due to the original image dimensions that provide different number of crops.

One of the experiments performed on the model had the objective of analysing the relation between the classification performance and the number of pyramid levels generated. We also study the behavior of our model relative to the ratio between the object size and the image resolution, where the object size...
Fig. 12: ImageNet partition dataset characteristics. (a) Distribution of the image dimensions. (b) Distribution of the ratio of the object ground truth bounding box/image resolution.

![Graph showing distribution of image dimensions](image1.png)

![Graph showing distribution of area bounding box/image resolution](image2.png)

(a)

(b)

TABLE I: Classification Error (in %) as a function of the image dimensions for top-1 error for our Pyramidal Focus (PF) method and for the baseline solution. The shaded cells point out when our model achieves better results than the baseline.

<table>
<thead>
<tr>
<th>Images Dimensions</th>
<th>Baseline</th>
<th>PF</th>
</tr>
</thead>
<tbody>
<tr>
<td>From 454 × 454</td>
<td>30.5</td>
<td>30.4</td>
</tr>
<tr>
<td>From 643 × 643</td>
<td>27.4</td>
<td>25.9</td>
</tr>
<tr>
<td>From 908 × 908</td>
<td>32.8</td>
<td>29.4</td>
</tr>
<tr>
<td>From 1285 × 1285</td>
<td>27.7</td>
<td>25.5</td>
</tr>
</tbody>
</table>

* The first image dimension corresponds to 2 pyramid levels and so on up to 5 levels.

In Table I we present the classification error as a function of the intervals of image dimensions. We verify that in all cases our model outperforms the baseline error. This is because, although CNNs are robust against scale changes (they are trained for each object class with different scales), the need of resizing the image before loading it in the network can produce distortions in the object. Therefore, as the PF model does not employ any resizing to the level images we have always the original object proportions, improving the recognition of the CNN.

In Fig. 13 we verify that for smaller objects, relative to the image resolution, our PF model works better than the usual traditional single feedforward classification. It is also shown that the bigger the object is relative to the image resolution, the smaller the classification error. This is due to the fact that when the object occupies most of the image, there are usually no distracting objects, contributing to a more accurate recognition. Moreover, objects with a higher ratio can only be fully processed in the latest crops, whereas objects with lower ratios start being processed with a more fine scale in lower levels.

Fig. 13: Top-1 classification Error (in %) as a function of the ratio between the ground truth bounding box area and the image resolution.

![Bar graph showing classification error](image3.png)

V. CONCLUSION

In this work we developed a general computational framework, inspired by human vision, that is capable of performing the task of object recognition by combining Convolutional Neural Networks (CNNs) with foveal vision techniques. This framework integrates two dissociated visual attentional methodologies that can be performed sequentially.

The first one consists of a Foveal Saliency method capable of replicating the nonuniform way of sampling the image and the active strategy of orienting the gaze to regions of interest using attentional mechanisms. The second one is a Pyramidal Focus method that improves the object recognition after the target is centered with the visual field.

For the Foveal Saliency method we were able to conclude that it is necessary to have successive saccades, because in real scenarios, where objects can be anywhere in the image, the results show that the classification performance improves significantly from the first to the second feedforward pass. Additionally, as no performance gain was obtained in consequent iterations, it seems that the method is capable of orienting the gaze to the target in two saccades.

Furthermore, we concluded that the classification performance depends on the fovea size, which is imposed by the artificial foveal visual system. However, the performance falls short of
the baseline solution (a single feedforward pass on a CNN whose input is the uniform resolution image) for smaller foveas, where the information reduction is higher. It was also observed that as the fovea size increases performance reaches a saturation point. This suggests that it is not necessary to store and transmit all the information encoded in the image in order to achieve the models best performance. However, the information reduction obtained was not enough to demonstrate computational gains.

For the Pyramidal Focus method we were able to obtain favorable results in the classification performance where the model outperforms the baseline solution. We conclude that if we are willing to accept an increase in the computational cost, this methodology is a simple and effective way of improving the classification accuracy after centering the objects with the field of view.

Finally, we conclude that our results are promising. Although our computational efficiency was not favorable, since the CNN used was trained to receive images with uniform resolution, we presented a biomimetic visual system that is capable of orienting the gaze to regions of interest and then performing a target classification with foveal focused attention that outperforms the baseline performance. Therefore, in the future, in order to achieve a computational gain, we intend to leverage log-polar like transformations with more compact neural network architectures trained to classify images more efficiently. As opposed to the Cartesian representation of images, Log-polar geometry resembles the structure of the retina, where we are able to have higher sampling rates on the central part of the retina - the fovea - and thus, mimicking more efficiently a foveal sensor.

REFERENCES