vision

MDPI

Article

Ultrafast Image Categorization in Biology and Neural Models

Jean-Nicolas Jérémie *, Laurent U. Perrinet *

Institut de Neurosciences de la Timone (UMR 7289), Aix Marseille University, CNRS, 13005 Marseille, France
* Correspondence: jean-nicolas.jeremie@univ-amu.fr (J.-N.J.); laurent.perrinet@univ-amu.fr (L.U.P.)

Abstract: Humans are able to categorize images very efficiently, in particular to detect the presence of an animal very quickly. Recently, deep learning algorithms based on convolutional neural networks (CNNs) have achieved higher than human accuracy for a wide range of visual categorization tasks. However, the tasks on which these artificial networks are typically trained and evaluated tend to be highly specialized and do not generalize well, e.g., accuracy drops after image rotation. In this respect, biological visual systems are more flexible and efficient than artificial systems for more general tasks, such as recognizing an animal. To further the comparison between biological and artificial neural networks, we re-trained the standard VGG 16 CNN on two independent tasks that are ecologically relevant to humans: detecting the presence of an animal or an artifact. We show that re-training the network achieves a human-like level of performance, comparable to that reported in psychophysical tasks. In addition, we show that the categorization is better when the outputs of the models are combined. Indeed, animals (e.g., lions) tend to be less present in photographs that contain artifacts (e.g., buildings). Furthermore, these re-trained models were able to reproduce some unexpected behavioral observations from human psychophysics, such as robustness to rotation (e.g., an upside-down or tilted image) or to a grayscale transformation. Finally, we quantified the number of CNN layers required to achieve such performance and showed that good accuracy for ultrafast image categorization can be achieved with only a few layers, challenging the belief that image recognition requires deep sequential analysis of visual objects. We hope to extend this framework to biomimetic deep neural architectures designed for ecological tasks, but also to guide future model-based psychophysical experiments that would deepen our understanding of biological vision.

Keywords: vision; ultrafast animal categorization; deep learning; transfer learning; computational neuroscience; behavior; image categorization; timing

Citation: Jérémie, J.-N.; Perrinet, L.U. Ultrafast Image Categorization in Biology and Neural Models . *Vision* **2023**, *7*, 29. https://doi.org/10.3390/vision7020029

Received: 30 September 2022
Revised: 9 March 2023
Accepted: 15 March 2023
Published: 24 March 2023

1. Introduction

1.1. Biological Vision and Ultrafast Image Categorization

What distinguishes a visual scene that includes an animal from one that does not? This question of "animacy detection" is crucial for the survival of any species, especially in regard to the interactions between prey and predators. This constraint has therefore profoundly shaped the way biological visual systems process retinal input. Of particular importance is the fact that this response must be efficient and fast, while keeping energy requirements to a minimum. In addition, these systems must fit the ecological niche of the system under consideration, with the range of patterns to be recognized being different for, say, a lion, a bird, and a human. It is important to note that this biologically-inspired approach shares some similarities and differences with detection algorithms defined in computer vision. Our goal in this paper is both to propose bio-inspired ultrafast image categorization models and to better understand how biological visual systems can efficiently implement such a task [1].

Therefore, let us first define the task of rapidly detecting an animal in a scene (see Figure 1). This task is routinely used in the study of biological vision in the laboratory (for

a review, see [2]). In its simplest form, it consists of reporting whether an image contains an animal. When applied to generic natural scenes, the task is such that the animal species is arbitrary and can include, for example, birds, insects, or mammals. A further difficulty is that there are large variations in the identity, shape, pose, size, number, and position of animals that may be present in the scene. However, biological visual systems are able to perform such detection efficiently in briefly flashed images, a so-called tachistoscopic presentation, with differential activity in electroencephalogram (EEG) recordings as low as 120 ms for humans [3] or as low as 80 ms for monkeys [4]. Such EEG recordings from an ultrafast image categorization task are openly available [5]. This has also been observed in the differential activity of single neurons in the primate lateral prefrontal cortex [6].

Figure 1. In ultrafast image categorization, the task is to report whether a briefly flashed image contains a class of object, such as an animal [3]. The presentation time can be on the order of 20 ms, and the response is, for example, the pressing or not pressing of a button. Representative images for distractors and targets are shown here for two classes: 'animal' and 'artifact'. Note that these tasks are a priori independent and that an animal target can be either a target or a distractor for the other task. Here, based on Rousselet et al. [7], we did not consider images of humans to be part of the animal class, since they seem to represent a class of their own.

Human categorization of an animal can be performed with high accuracy (generally over 80% correct), very quickly [8], and is robust to geometric transformations [7]. Color seems to have little effect, but some low-level statistics [9], as well as other factors (such as the animal's position and size in the scene) may influence accuracy but not speed [10]. Accuracy is maximal when the animal is in the center of the visual field [11], but performance is still above chance level (at about 60%) at extreme eccentricities of about 70°. Such a task is performed seamlessly in parallel, so that multiple images can be categorized at once [12]. Surprisingly, once the task is learned, novel images are processed as quickly as familiar ones [13]. Given the difficulty of modeling this task, a scientific question is to understand what features in the image configuration are sufficient to produce such an effective behavioral response [14].

1.2. Feed-Forward Models of Ultrafast Image Categorization

Designing the best algorithm to solve ultrafast image categorization, as implemented in biological systems, is one possible way to answer this question. In this case, there are major constraints in the dynamics of vision, especially related to the limits of axonal transduction speed, which can lead to major difficulties in modeling the system [15]. In the case of the ultrafast go/no-go categorization task, two consequences follow from

these physiological constraints: first, the response must be made quickly and therefore must be open-loop, i.e., before the action can take effect; second, since the whole process involves several processing steps before recurrent loops can refine neural activity, the flow of information is predominantly feed-forward [16]. This has been confirmed by EEG recordings of humans performing the task, showing that top-down signals (such as context or expectation) can influence categorization, but that the process is mainly a bottom-up, feed-forward process [17]. We can also expect that there should be a trade-off between accuracy and speed for image categorization algorithms [18].

Given the problem of designing the best algorithm to solve ultrafast image categorization in biologically inspired systems, it was previously shown that such a feed-forward architecture may be sufficient to perform the task [19]. This architecture consists of a sequence of layers that interleave a linear and a nonlinear process. This is similar to the simple and complex sublayers observed in the primary visual cortex. The linear part of the processing is performed by a convolutional operator, hence the name convolutional neural network (CNN) for this class of architecture. The nonlinear operation is often a simple rectifying unit, similar to the integration process that transforms the analog input to a neuron into a (positively defined) firing rate. In these architectures, the layer's resolution generally becomes progressively coarser along the levels of the hierarchy, until a few classification layers provide the final output [20]. The efficiency of this model yielded results comparable to humans performing the task on the same images [19]. Other popular methods use oriented luminance gradient histograms [21], but with a similar architecture, in which a sequence of processing steps in image space is followed by a classification step. Remarkably, these CNN architectures mirror that of the primate visual system, wherein the retinal image is transmitted from the thalamus to the primary visual cortex and then follows a path along the temporal lobe [16,22].

1.3. Related Work

Since their adoption as modeling tools, feed-forward architectures have been instrumental in the breakthrough of deep learning architectures, in particular in providing human-like performance for the PASCAL [23] and IMAGENET [24] challenges, that is, classifying millions of images into over 1000 different categories (labels). An important aspect of these architectures, originally inspired by neuroscience, is that they can be trained in a supervised manner, i.e., by associating each image with a given label in the training phase. This was illustrated for the MNIST challenge of classifying handwritten digits by associating each image of a digit with its recognized value [20]. This training process optimizes a given loss function applied to each pair, which allows the weights of the network to be progressively adjusted using gradient descent. In particular, a CNN such as VGG 16 is a well-optimized architecture for performing this challenge of computer domain categorization IMAGENET [25]. Therefore, we decided to use VGG 16 with IMAGENET as a starting point to better understand the process underlying ultrafast categorization of natural images, while bridging our knowledge between neuroscience and computer science.

The task defined for the IMAGENET [24] challenge could be considered computer-specific, since it requires choosing among 1000 labels, which implies knowing and remembering these 1000 labels to make the choice. Unlike artificial neural networks, which can easily compare these 1000 possibilities simultaneously, one can instead use a subset of behaviorally relevant labels to make the task more relevant to humans. Since we defined a novel task, it is then possible to "re-train" these CNNs to categorize images by defining a novel set of supervised pairs (e.g., an image containing an umbrella associated with the synset "artifact"). For the original IMAGENET [24] challenge, each input–output data pair consists of an input data point (an image from the IMAGENET database) and its corresponding output label (e.g., an image containing an umbrella associated with the label "umbrella"). The idea is to take the knowledge gained from one task and transfer it to a different but related task by using the right training pairs to re-train the CNN; this method is called transfer learning [26]. The advantage of using this method is that one can more

easily explore the space of all possible architectures by adjusting the synaptic weights of the convolutional kernels, but also by testing the meta-parameters of the CNNs, such as the number of layers, the number of channels in each layer, or the coarsening of the visual information along the hierarchy [27]. Note that, at the extreme, even the best CNN network may not be able to learn to categorize an image-independent feature, such as whether the calendar day on which the photo was taken is odd or even. For instance, we will show below that, following that logic, if we define a task consisting of random labels among the 1000 categories of IMAGENET, then none of our tested architectures can learn this task efficiently. Finally, while a drawback of these networks is their lack of interpretability, we will exploit the fact that their raw efficiency gives a lower bound for the possibility of solving a given task.

Indeed, compared to random labels, the situation is different when defining more ecological tasks, such as categorizing animals or artifacts. This method can also be used by changing the definition of the supervision pairs to study changes in task context, rapid categorization, and object interference in the image [28]. A fitting question might be, "Is there an animal in this image?", since it reduces this human–machine bias by reducing the choices while maintaining a sufficiently complex and documented question. Searching for these kinds of categories seems to be a primordial function of the brain [16,29]. For example, using a set of specific stimuli, it has been shown that categories can be found in the brain areas of rhesus monkeys and that these categories can then be learned by artificial neural networks [30]. Our goal here is to obtain a model that is more faithful to the physiological data. In summary, and somewhat counterintuitively, compared to biological systems, it may be more difficult for a neural model to make a choice between only two alternatives, such as detecting an animal in an image, than to choose from 1000 labels [31,32]. This work will allow us to better understand how this is achieved in both biology and computational neuroscience models.

1.4. Main Contributions

What distinguishes an image with an animal from an image without an animal? To answer this scientific question, our work proposes three major contributions. First, to define the psychophysical task, we built a script to build large, arbitrary datasets of images based on IMAGENET [24]. It was defined by selecting labels according to a large semantic graph of English words: WORDNET [33] (see Figure 2). According to our scientific question, we first defined our task as the categorization of an animal in an image. As a control, we also defined an independent task consisting of detecting the presence of any artifact in the image (see Figure 1C,D). Second, we re-trained the existing VGG 16 model on these tasks and compared its performance with experimental data. This allowed us to test the robustness of our networks to different geometric transformations and to compare their accuracy with that observed in the physiological data. In addition, we compared the accuracy for both tasks, individually and jointly. Third, we tested different levels of complexity of such models by performing a gradual removal of layers from the original network. This experiment quantified whether low-level features could be sufficient to categorize animals [34] (although it is known that the global image statistics [35] or the spatial frequency envelope is not sufficient to categorize images [36]) and whether this could be accompanied by a decrease in invariance to geometric deformations. Finally, we discuss how this work can be useful in the design of future physiological experiments and in the design of novel computer vision architectures.

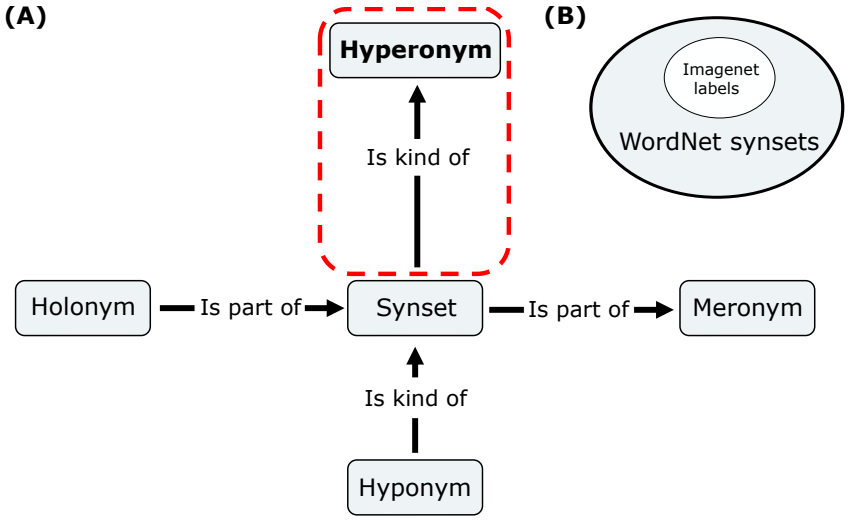

Figure 2. (**A**) Schematic displaying the different semantic links between the synsets of the WORDNET network. We focused on the hyperonym link (frame in red) to build datasets from IMAGENET synsets; for instance, which images are 'kind of' an 'animal'. (**B**) Venn diagram exposing the subset of the IMAGENET labels included in the WORDNET synset's set.

2. Methods

2.1. Building the Dataset Maker Library Using the WORDNET *Hierarchy*

To re-train a deep convolutional network (like VGG 16) for a specific task, one of the most important components is the dataset. We needed a tool that would allow us to generate datasets suitable for answering our question. Therefore, we created a library that, from a keyword, generates a dataset with image folders containing (target) or not containing (distractor) this keyword [37]. For this, we will use the corresponding set of labels from the IMAGENET database [24], which is based on a large lexical database of the English language: WORDNET [33]. The nouns, verbs, adjectives, and adverbs in this database are grouped into a graphical set of cognitive synonyms, synset, each of which expresses a different concept. These synsets are linked to each other using a few conceptual relations (see Figure 2). For example, if we set the dataset maker with the keyword 'animal', we used the hyperonym link to determine that a German Shepherd is a type of dog and that a dog is a type of 'animal', thus defining a hyperonym path. In this example, the synset 'animal' from WORDNET is in the hyperonym path of the label 'German Shepherd' in IMAGENET. Based on this relationship, the dataset creator selected a specific subset of labels in the IMAGENET database to build our datasets. Once the list of labels corresponding to our task was selected, the dataset maker randomly selected from the URLs provided for the IMAGENET challenge [24] to download the images that make up the dataset.

With this tool, we generated datasets according to two given tasks. In particular, we generated the dataset necessary to train the network to answer our question, "Is there an animal in this scene?", by selecting the 'animal' synset. To answer the question, "Is there an artifact in this scene?", we followed the same protocol. As a control, we also created a 'random' dataset, which was generated by randomly selecting 500 labels from the IMAGENET database. The latter was generated to infer the role of the possible links between arbitrary labels by measuring the resulting efficiency of categorization by a deep convolutional network. In summary, we used Dataset Maker to generate three datasets: One based on the 'animal' synset, one based on the 'artifact' synset, and a 'random' one. Each newly generated dataset contains a 'test', 'validation', and 'train' set (with 1200, 800, and 2000 images, respectively). Each set contained a 'target' and a 'distractor' category (both with the same number of images). All networks were trained on the 'training' set and tested during training on the corresponding 'validation' set. We then computed accuracies using the 'test' set. As a control, we also tested the networks on the dataset from

Serre et al. [19], which contains a total of 600 targets (images with an animal) and 600 distractors (images without an animal).

2.2. Transfer Learning

We used the transfer learning method to re-train networks [26]. This method takes the knowledge gained from one task and applies it to a different but related task. We used an existing network that had been pre-trained on a specific task: VGG 16 [25]. This architecture is loaded thanks to the PYTORCH library [38] and trained on the database used to solve the IMAGENET [24] task. We had previously found that this model provided the best trade-off between accuracy and complexity [39]. It also achieves a good model of biological function as measured by the Brain score [40]. Compared to other architectures such as ResNet, VGG 16 stands out as an ideal candidate. Two notable advantages of the transfer learning method are the robustness and the convergence speed for learning the network. This results in lower total execution time and energy consumption. In particular, this method allowed us to save computational time during the learning process and thus experiment with several possible strategies (see Figure 3). We first validated this hypothesis by training a network with random weights: VGG SLS (Supervised Learning from Scratch) as a control.

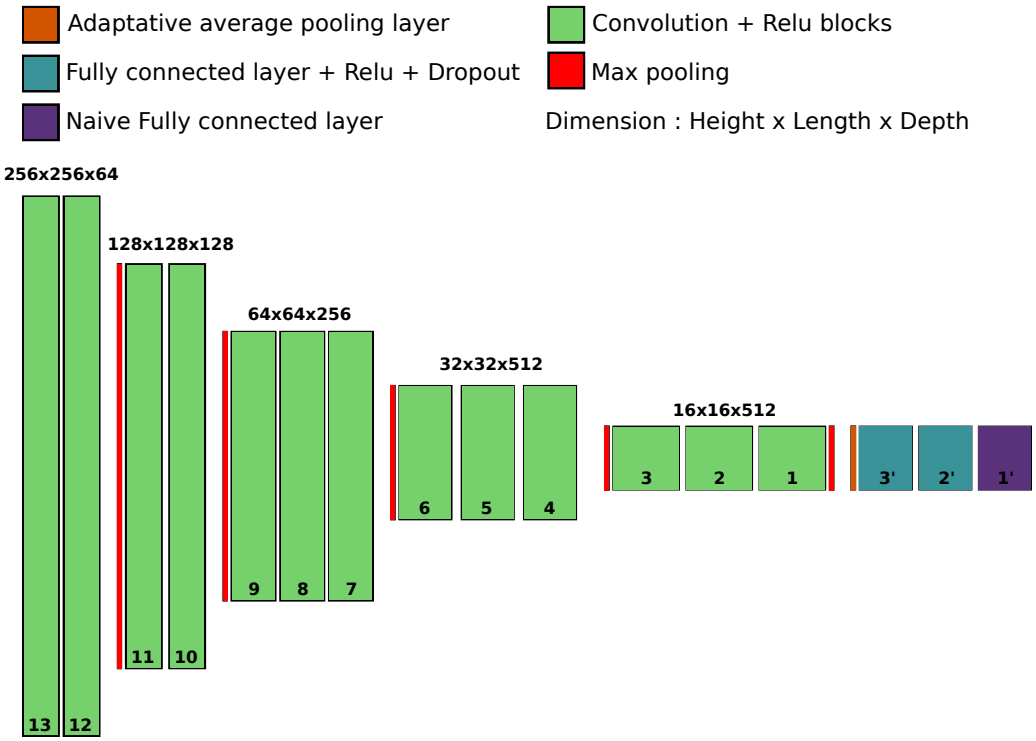

Figure 3. A diagram of the network architecture used in transfer learning. In green, the 5 blocks of 13 convolutional layers noted from 13, at the entrance, to 1, at the junction with the fully connected part, represented jere in blue and purple. The purple layer represents the naive layer substituting for the one in the PYTORCH architecture.

During transfer learning, we kept all layers of the VGG 16 network, since it is already capable of performing feature extraction on natural scenes, and re-trained only the last fully connected layer. In particular, we replaced this last layer trained on IMAGENET (i.e., with a vector of dimension $K = 1000$ that captures the predicted probability of detection for each of the labels in the IMAGENET database) with a layer whose output dimension is simply $K = 1$ and which represents the predicted probability of detection of a new object of interest, i.e., a target rather than a distractor. We then re-trained this fully connected layer, while freezing the weights of the other layers, to match the pre-trained features (the output

of the convolutional layers) with the synset corresponding to the new task implemented by the dataset maker. Following this process, we re-trained a network that we call VGG TLC (Transfer Learning on Classification layers). As a control, we also tested the effectiveness of freezing the remaining layers by completely re-training all layers of the pre-trained network, VGG TLA (Transfer Learning on All layers). Note that these two networks were trained without any form of data augmentation.

Since the network is asked to make a binary decision during training ("Is this synset present in this scene?"), we implemented the loss using the binary cross-entropy loss with logits from the PYTORCH library. We used the stochastic gradient descent (SGD) optimizer from the PYTORCH library and validated parameters such as batch size, learning rate, and momentum by performing a sweep of these parameters for each network. During the sweep, we varied one of these parameters over a given range while leaving the others at their default values for 25 epochs. We chose the parameters' values that gave the best average accuracy on the validation set: batch size = 8, learning rate = 0.00005, momentum = 0.99. Then, to increase the generality of our results, we implemented various preprocessing steps on the inputs to introduce more variation into the training dataset: data augmentation. From the VGG TLC protocol, we tested the effectiveness of this data augmentation using two strategies: first, by re-training a pre-trained network with a set of custom transformations from the PYTORCH library: random horizontal flipping (with $p = 0.5$), random vertical flipping (with $p = 0.5$), a random rotation ($p = 1$), and random grayscale (with $p = 0.5$), such that we trained the VGG TLDA (Transfer Learning with Data Augmentation) model. Then, the input images were distorted using the auto-augment function from the PYTORCH library. This function implements a total of 16 randomly parameterized affine transformations on the inputs to perform data augmentation [41], thus defining the VGG TLAA (Transfer Learning with Auto Augment function) model. Finally, we studied a VGG RANDOM model trained on the 'random' dataset (that is, consisting of two categories defined by randomly chosen labels among the 1000 labels of IMAGENET). Note that, although we implemented all transfer learning strategies on this dataset, as the results were similar for all strategies, we chose to display the networks obtained using the same training protocol as VGG TLAA.

2.3. Pruning

Another network manipulation that we tested is the modification of the CNN architecture. In particular, we tested the effect of pruning the convolutional layers of the pre-trained network VGG 16 to determine the complexity of the features required to categorize a given synset of interest. In fact, the VGG 16 network can be described as a hierarchically organized pipeline: first, a set of convolutional layers, then a set of fully connected layers [25]. The set of convolutional layers is organized into 5 blocks of 13 convolutional layers. Within a block, there is a sequence of convolutional layers followed by a nonlinearity and optionally a normalization (in our case, we did not use or test the batch normalization option). Within each block, the image size and the number of channels are constant. In general, the resolution decreases from block to block using max-pooling operations, while the number of channels increases from 64 at the input to 512 at the fully connected blocks.

Since the final process of the set of convolutional layers is an adaptive pooling function that produces a characteristic image of constant size equal to 7×7, the size and architecture of the fully connected layers were kept constant. Therefore, we defined new networks whose names correspond to the number of layers to be pruned. The network named VGG-1 had only its last convolutional layer block pruned, and then we applied the same learning process as for the network VGG TLAA. We then did the same for the 12 different depth factors. We have chosen the names of the meshes according to the number of layers removed. Thus, the network with one layer removed is called "vgg minus one", i.e., "VGG-1" (from the deepest VGG-1 to the shallowest VGG-12).

2.4. Accuracy

Tha accuracy metric will be used to describe the performance of the model. In effect, the network is expected to output a binary decision ('Is there an animal in the scene?') and is designed to provide the predicted probability of the presence of a synset of interest in the scene. We considered the output a 'target' if the network output was greater than 0.5 (i.e., 50%), otherwise it was considered a 'distractor'. A positive true was defined as the case in which the network categorized a 'target' if it was a 'target', otherwise it defined a positive false. Similarly, a true negative was defined when the network categorized a 'distractor' when it was indeed a 'distractor', otherwise it defined a false negative. Based on these observations, we could determine each time that the networks performed a good categorization and calculate its accuracy as the ratio of the sum of true positives and negatives over the total number of samples tested. To provide a comparison with the state of the art, we tested the VGG 16 and computed its prediction by summing the predictions of the labels belonging to the hyperonymous path of the synsets of interest after the softmax layer, hence VGG LUT (Look Up Table). Accuracy was then computed using the same methodology as for the re-trained networks. We evaluated the accuracy of our different networks on the test set using Equation (1):

$$\text{Accuracy} = \frac{\text{True}_{\text{positive}} + \text{True}_{\text{negative}}}{\text{True}_{\text{positive}} + \text{True}_{\text{negative}} + \text{False}_{\text{positive}} + \text{False}_{\text{negative}}} \quad (1)$$

3. Results

3.1. Performances on Natural Scenes Containing Animals without Transfer Learning

Obviously, testing the initial pre-trained net should be one of the first experiments before re-training the neural networks. If we were to test it on the dataset on which it was trained to categorize an animal, it would indeed perform very well, with a mean accuracy of 0.99 for categorizing an animal (and 0.98 for an artifact; see Figure 4). The goodness of these results is quite stunning compared to human behavioral results and highlights one difference between human and machine intelligence.

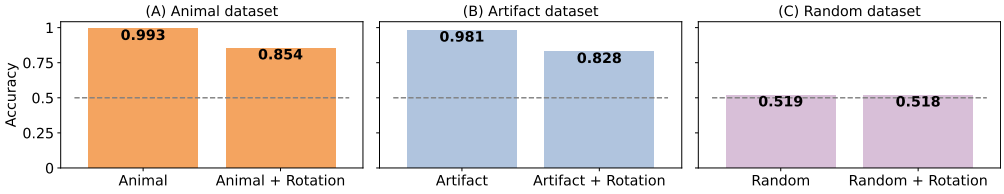

Figure 4. Bar graph representing the accuracy of the VGG LUT network on datasets built with the (**A**) 'animal', (**B**) 'artifact', or (**C**) control 'random' datasets [37]. For each dataset, the network is tested with original images (left) or after applying a random rotation (right). The dotted line represents the chance level for all graphs.

However, as soon as we added a perturbation such as a random rotation (images similar to [42]) into the same dataset, the performance dropped to 0.85 for the presence of an animal and 0.83 for the detection of an artifact, on par with human performance. Note that if the labels chosen in the task definition have no semantic link, as is the case for the test on the 'random' dataset, the network cannot perform a correct categorization, with or without rotation, and it yields an accuracy close to chance level.

3.2. Performances on Natural Scenes Containing Animals with Transfer Learning

We then tested different variations of transfer learning on the task, "Is there an animal in this visual scene?", and show the mean accuracies for different datasets, as summarized in Table 1. First, we have seen that the VGG LUT network seems to be robust, as validated on the dataset used by Serre et al. [19], on which the network achieves a mean accuracy of 0.95. Note that it achieves better performances compared to about 0.84 obtained by the

model designed in Serre et al. [19] and about 0.80 in psychophysics, and this is without any retraining process. Now, let us focus on the network after the transfer learning process, as the VGG TLC, VGG TLA, VGG TLDA, and VGG TLAA reached similar levels of performance on the test set (with 0.97, 0.96, 0.97, and 0.95, respectively) and also maintained robust categorization on the Serre et al. [19] dataset (with 0.94, 0.92, 0.91, and 0.88, respectively). Compared to the VGG SLS, which could only reach 0.64 on the same task, these results show that transfer learning allows us to obtain highly accurate networks for the categorization of a synset of interest. Note that this low performance is only due to the computational limits that we imposed in our study. We then focused on the robustness of the categorization of the different data augmentation strategies (VGG TLC, VGG TLA, VGG TLDA, and VGG TLAA) compared to the state of the art VGG LUT and the expected performance in neurobiological models.

Table 1. Mean accuracies for the ultrafast image categorization of an animal in a scene from a newly built dataset using our library dataset maker with the synset 'animal' (top) or for the Serre et al. [19] dataset (bottom) for different transfer learning strategies.

Dataset Built with Dataset Maker, Synset: Animal							
	LUT	TLC	TLA	TLDA	TLAA	SLS	
Accuracy	0.99	0.97	0.96	0.97	0.95	0.64	
Dataset from Serre et al. [19]							
	LUT	TLC	TLA	TLDA	TLAA	Serre et al.'s model	Human
Accuracy	0.95	0.94	0.92	0.91	0.88	0.84	0.80

3.3. Robustness of the Categorization with Different Geometric Transformations

Since we were looking for the best robustness for this task, we tested VGG TLC, VGG TLA, VGG TLDA, VGG TLAA, and VGG LUT on the newly constructed dataset using our dataset maker library with the synset 'animal'. We applied either a grayscale filter or a vertical or a horizontal reflection to the input (see Table 2). We also tested the robustness to rotation by rotating the image around the center by an angle ranging from $-180°$ to $+180°$ (see Figure 5). All these networks maintained good average accuracy on the returned dataset and on the grayscale dataset (see Table 2). These results were consistent with psychophysical results showing that ultrafast categorization is robust to a grayscale transformation [10]. Only VGG TLDA and VGG TLAA seemed to show robust accuracy at all angles, with peaks in accuracy at the cardinal orientations ($-180°$, $-90°$, $0°$, $90°$, and $180°$), which could be explained by the pre-training weights of the networks, as they correspond to the peaks found in the categorization of VGG 16. We conclude here that data augmentation provides a more robust categorization of the synset of interest by the network, as the VGG TLDA and VGG TLAA achieve better performance in this task. In addition, the protocol used to re-train the network VGG TLAA, with the auto-augment function of the library PYTORCH [41], is also better than our custom data augmentation. The performance of VGG TLAA is very close to that of VGG TLDA, with a tendency for VGG TLAA to be more robust to rotation. Therefore, the VGG TLAA network is the best fit for psychophysical observations due to its stability and robustness of categorization to different image transformations [7,42]. In the following, we therefore focused on exploring the features that this model relies on to perform its categorization.

Table 2. Mean accuracies for ultrafast image categorization of an animal in a scene using various geometric transformation on the input: vertical flip, horizontal flip, grayscale filter. These transformations were implemented using our dataset maker library with the synset 'animal' for four re-trained networks: VGG TLC, VGG TLA, VGG TLDA, and VGG TLAA. It was compared with the state-of-the-art network VGG LUT. All the transformations used here were performed using the PYTORCH library [38].

	LUT	TLC	TLA	TLDA	TLAA
Vertical flip					
Accuracy	0.96	0.94	0.94	0.95	0.93
Horizontal flip					
Accuracy	0.99	0.97	0.96	0.97	0.95
Grayscale filter					
Accuracy	0.96	0.95	0.93	0.95	0.93

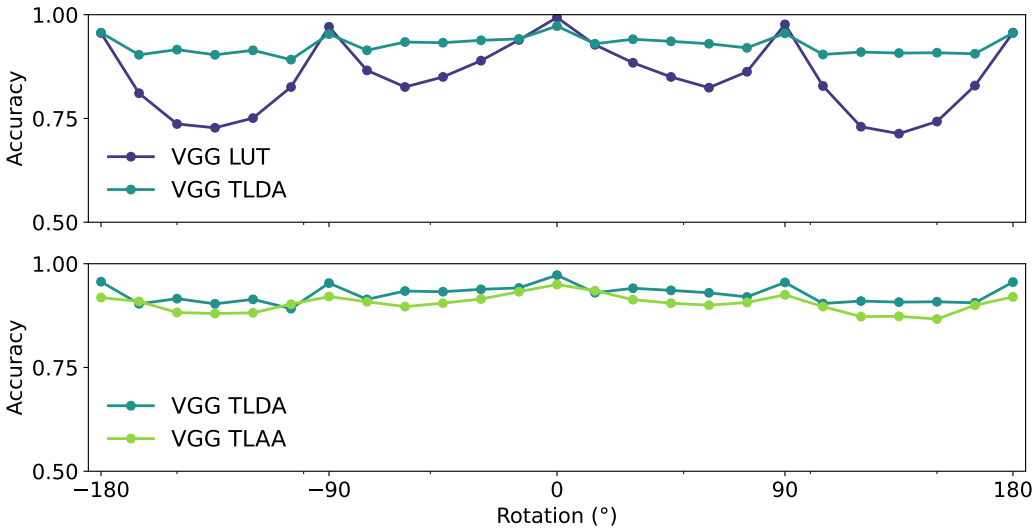

Figure 5. Average accuracy in the test dataset of the different networks for different rotations of the input image. (Top) The VGG LUT displayed with the VGG TLDA networks. (Bottom) The VGG TLAA displayed with the VGG TLDA networks. The rotation is applied around the center with an angle ranging from $-180°$ to $+180°$.

3.4. What Features Are Necessary to Achieve the Task?

We designed an experiment in which we gradually removed layers from a pre-trained network VGG 16 for 12 different "depth" factors. For each level, we tested the re-trained pruned networks to categorize an animal in a scene for our dataset IMAGENET. VGG LUT and VGG TLAA achieved the best accuracy for this task (see Figure 6). The accuracies of the networks remained similar to the performance found by Serre et al. [19] with a slight drop between VGG-9 and VGG-12. This is not a surprise, as their model relied on low-level features [34]. Note that the computational time required to perform the categorization decreased with the depth of the network (in seconds on a Quadro RTX 5000 GPU, we obtained VGG TLAA $= 0.005 \pm 0.0001$ and VGG-8 $= 0.003 \pm 0.0001$) (see Table 3).

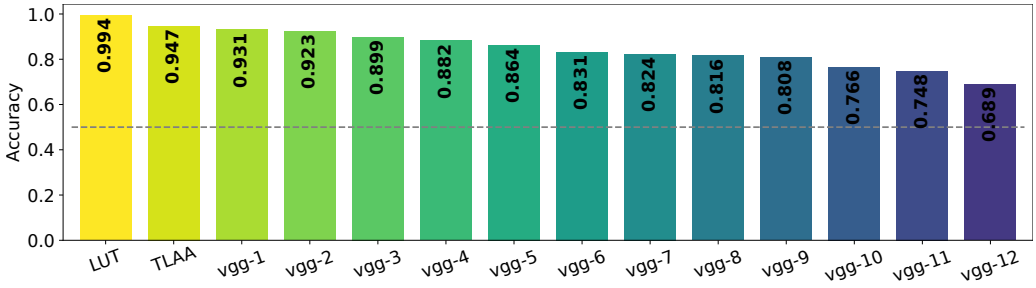

Figure 6. Average accuracy obtained as a function of depth in pruning networks. The networks are re-trained to categorize animals and tested on test datasets based on IMAGENET images constructed using the 'animal' synset. The prefix "vgg-" indicates the number of convolutional layer blocks pruned from the original network (see Section 2.3 for more details). The dotted line represents the chance level for all graphs.

Table 3. Table showing the average time in seconds required for networks to perform a prediction for a 256 × 256 resolution image with a Quadro RTX 5000 GPU.

Mean Time (s)				
VGG TLAA	VGG-1	VGG-2	VGG-3	VGG-4
0.0057 ± 0.0001	0.0055 ± 0.0005	0.0054 ± 0.0008	0.0051 ± 0.0003	0.0047 ± 0.0002
VGG-5	VGG-6	VGG-7	VGG-8	VGG-9
0.0043 ± 0.0001	0.0035 ± 0.0007	0.0031 ± 0.0001	0.0027 ± 0.0003	0.0022 ± 0.0002
VGG-10	VGG-11	VGG-12		
0.0018 ± 0.0007	0.0013 ± 0.0006	0.0008 ± 0.0006		

We also tested all pruned networks on our IMAGENET dataset by rotating the image around the center from $-180°$ to $+180°$; however, the categorization may lose robustness with fewer layers (see Figure 7). Indeed, as the number of layers and the mean accuracy after rotation decreased, the standard deviation of the mean accuracy increased (VGG TLAA $= 0.91 \pm 0.02$, VGG-8 $= 0.73 \pm 0.04$). Although the networks seem to be able to categorize an animal with fewer layers, they seem to trade this advantage for a lower robustness to transformations such as rotations.

To get a better idea of the size of the feature maps needed to categorize an animal in a scene, we tested the networks on a new "shuffled" dataset, where the image had been divided into square patches of different sizes and then blended to generate a new image [43]. Since CNN networks are by definition robust to translation, patch translation should have minimal impact on categorization unless it breaks some necessary patterns in the images. With few layers, the networks should rely on low-level features to perform their categorization, and indeed we obtained an idea of the size of feature maps required for different depths. In fact, between patch sizes 256 × 256 and 64 × 64, the categorization of the networks was robust to this transformation (see Figure 8). However, as soon as we reached the patch size of 32 × 32 pixels, the accuracy of all networks dropped sharply. Furthermore, there seemed to be a transition between deeper and medium networks, as the latter gave better average accuracies for this task. As a consequence, the size of the feature maps needed to perform such a task varies with the depth of the network. For example, VGG TLAA appears to rely on feature map sizes between 32 × 32 and 64 × 64 pixels, as its accuracy drops when we exceed this threshold (see Figure 8); however, further study is needed to quantify this feature map size. In a future application, we could extract feature maps from these low-level layers to better understand the features needed to perform this task. This would allow us to design a stimulus set for a psychological task such as in Thorpe et al. [3]. Such a test could be relevant to whether these features are sufficient to categorize an animal in a flashed scene.

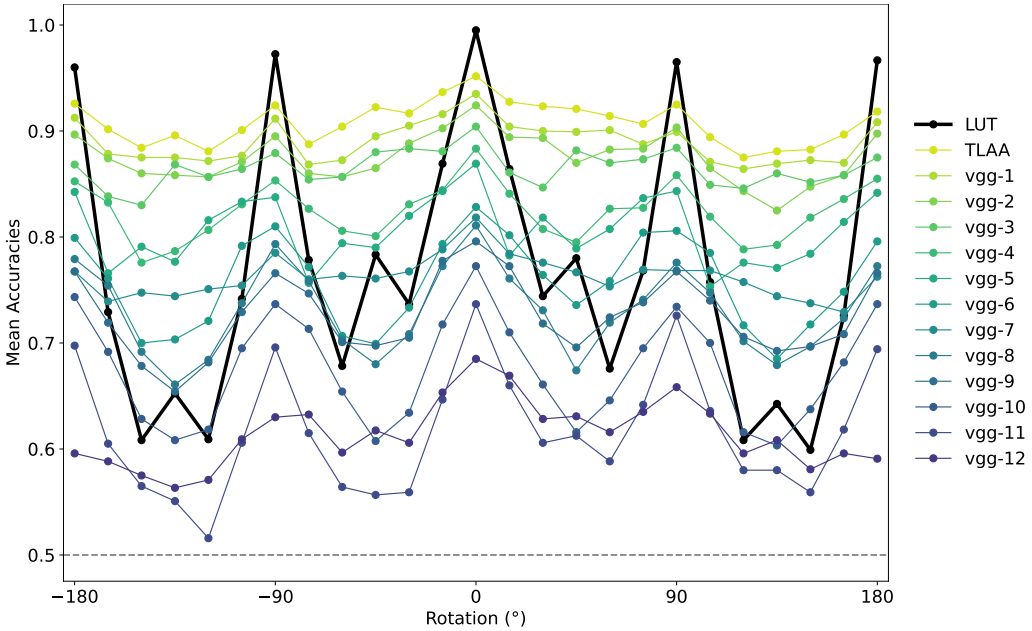

Figure 7. Average accuracy over the test dataset of the re-trained networks for rotations around the center from $-180°$ to $+180°$. These networks were tested on a dataset based on IMAGENET images constructed using the 'animal' synset. The index after "vgg-" indicates the number of convolutional layers pruned in the networks (with vgg-1 being the deepest and vgg-12 being the shallowest).

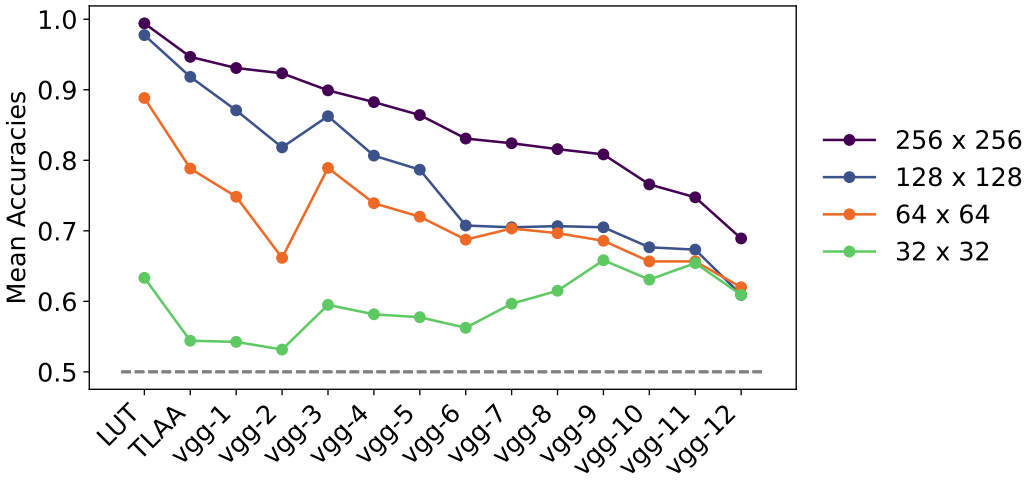

Figure 8. Average accuracy obtained for the differentially pruned networks over the 'shuffled' test dataset, where we applied a shuffled transformation to the input image. We show the results as we decreased the size of the shuffled patches on the images. The networks were retrained to categorize animals and tested on datasets based on IMAGENET images created using the 'animal' synset. The index after "vgg-" indicates the number of convolutional layers pruned in the networks. The dotted line represents the chance level for all plots.

3.5. Dependence of Accuracy Scores between the Two Tasks

We examined dependence of learning performance of VGG TLAA between two tasks by introducing a variation of the synset of interest in the construction of the dataset. We used our dataset maker tool with the keyword 'artifact', thus generating a new network trained to categorize the presence of the 'artifact' synset in a natural scene: VGG ARTIFACT. We displayed the average accuracy of the networks trained to detect the 'animal' synset (here VGG ANIMAL stand for our VGG TLAA) on the dataset constructed with the 'animal' synset (respectively trained to detect the artifact synset tested on the dataset constructed with the artifact synset). Next, we tested the networks trained to detect the

'animal' synset on the dataset constructed with the 'artifact' synset and vice versa. Here, by exposing the predictions for the 'animal' and 'artifact' synsets, we highlight a bias in the composition of the dataset. Although the outputs are independent, the 'animal' images confidently match the 'non-artifact' images (and vice versa), thus facilitating global detection (see Figure 9A,B).

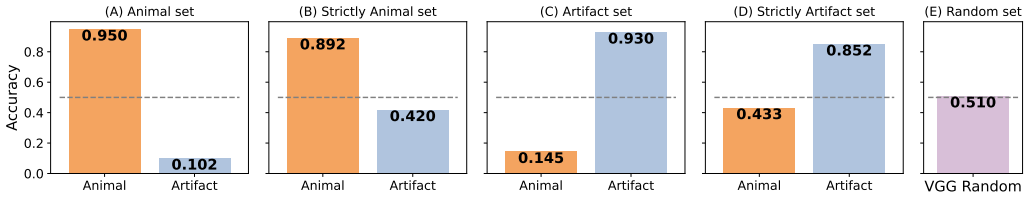

Figure 9. Bar graph representing the accuracy of networks re-trained to categorize an animal: VGG ANIMAL (in orange); re-trained to categorize an artifact: VGG ARTIFACT (in blue); or re-trained to categorize a random distribution of synset: VGG RANDOM (in violet); (**A**) on the dataset built with the 'animal' synset; (**B**) on the dataset built with the 'animal' synset where the distractor "is not an artifact" thus constitutes the "strictly animal" dataset; (**C**) on the dataset built with the 'artifact' synset; (**D**) on the dataset built with the 'artifact' synset where the distractor "is not an animal" thus constitutes the "strictly artifact" dataset; (**E**) on a dataset built with a random distribution of synset (see Figure 3). The dotted line represents the chance level for all graphs.

To infer the influence of this bias on the performance of the network, we generated through the dataset maker a dataset based on the 'animal' synset where, in addition to not being animals, the distractors would also not be 'artifacts'. This defines the 'strictly animal' set (respectively, one defines the 'strictly artifact' set based on the 'artifact' synset where, in addition to not being an artifact, the distractors would also not be an 'animal'). Once this distinction was made, although there is a loss in performance for both networks, they remained fit for their respective tasks by maintaining an accuracy above 0.8. On the other hand, they did not seem to be able to predict the absence of their respective sentences once the ensemble was modified (see Figure 9B,D). These results reinforce the argument that, despite task independence, the composition of the dataset can generate bias in network categorization.

As a control, we tested the VGG RANDOM network on the corresponding dataset (see Section 2.1 for details). As it obtains an average accuracy close to the one obtained with the VGG SLS network, its poor performance can be explained by the fact that the pre-trained weights of the VGG 16 network do not match the new task. Incidentally, this bias is also present in the dataset used by Serre et al. [19]. However, when we compared the performance of the humans on this dataset with the performance achieved by the network on a frame-by-frame basis, we found a high correspondence (about 0.84) in their correct predictions. Indeed, for some images, the networks failed at categorizing but the human succeeded, and vice versa. For some images, both the network and the human succeeded or failed in categorizing an animal, and there were cases where the network was wrong but the humans responded correctly on average (see Figure 10). We have displayed images where one human or both a human and our model failed to categorize an animal in the scene, as this may reflect the specific features that humans or our models rely on to perform their categorization. This close relationship between human and network responses could allow us to select images and design physiological and psychophysical tests to infer the features necessary for such detection.

(**A**) Synset Animal
Target: Human fails

(**B**) Synset Animal
Target: Both fail

(**C**) Synset Animal
Target: TLAA fails

(**D**) Synset Animal
Distractor: Human fails

(**E**) Synset Animal
Distractor: Both fail

(**F**) Synset Animal
Distractor: TLAA fails

Figure 10. Some prototypical example images where either (**A**) humans failed to categorize an animal (or in (**D**), the absence of an animal) in a scene but our model succeeds; (**B**) humans and our model both failed to categorize an animal ((**E**), the absence of an animal) in the scene; (**C**) the model failed to categorize the presence of an animal ((**F**), the absence of an animal) in a scene but the human succeeded. The psychophysical data and the images were taken from the dataset used by Serre et al. [19].

4. Discussion

In this paper, we have shown that we can re-train networks using transfer learning to apply them to an ecological image categorization task and obtain insights on visuo-cognitive processes. Such outcomes could in particular be beneficial when studying impaired systems such as in Autism Spectrum Disorder [44]. These artificial networks achieve accuracies similar to those found in psychophysical responses in humans. In the image processing flow at work in convolution networks, the position of the feature maps has no influence on the activation of receptive fields. Since translation is a shift in the position of the feature maps, these networks are supposed to be robust to translation. However, a transformation by a rotation constitutes then a global perturbation of the features composing the maps. Thus, since the features are different, rotation can lead to the solicitation of different receptive fields. If these new receptive fields are not previously learned, the network will be unable to generalize. This could explain the differences in performance between learning protocols involving or not involving rotations. Furthermore, the robustness of the categorization is comparable to that found in psychophysical data. In particular, we have shown quantitatively that the categorization of the re-trained networks may be robust to transformations such as rotations, reflections, or grayscale filtering, such as is observed in humans [3,7].

We have studied networks that learn to detect if an image contains an animal or an artifact. Two independent networks each re-trained on each of the two categorization tasks used to highlight a link or rather a bias in this categorization. This kind of bias is also found in humans and seems to impact the categorization as well [45] and could be linked to top–down influences [46]. The question of detecting an animal in an image is indeed tightly linked to that of detecting an artifact, allowing for the possibility of the less likely appearances of an animal object (like a teddy bear) or of a non-animal non-object (like a

mountain). The study of this kind of bias could possibly allow for building ecologically-relevant datasets to maximize the learning process of the networks in order to discover more about the features needed for categorization [47].

While the level of 80% correct categorization between humans and machines in this type of task is similar, both could be driven to make different "mistakes", and these particular examples could then be used as subjects for studies in the design of psychophysical tests. In addition, these systematic errors could be a window into some processes in our understanding of primate visual pathways. The last part of our study was based on the search for the features necessary for categorization. We found that, in agreement with the studies of Serre et al. [19], a simple feed-forward network based on low-level features was sufficient to perform categorization efficiently. Moreover, we estimated the size of the features needed to be about 32×32 pixels and 64×64 pixels. Although categorization is still possible at this very low computational cost, we quantitatively show that it gradually loses robustness.

5. Perspectives

One of the main goals of this study was to provide a comparison for an ecological and well-studied task used in visual neuroscience. Although this study focuses on the analysis of categorization, it is a necessary step for a well-known task in the field of vision: visual search. This task consists of the simultaneous localization and detection of a visual target of interest. Applied to the case of natural scenes, visually searching, for example, for an animal (either prey, a predator, or a partner) constitutes a challenging problem due to large variability over the numerous visual dimensions. Previous models managed to solve the visual search task by dividing the image into sub-areas. This is at the cost, however, of computer-intensive parallel processing on relatively low-resolution image samples [48,49]. Taking inspiration from natural vision systems [50], we developed a model that was built over the anatomical visual processing pathways observed in mammals, namely the "what" and the "where" pathways [51]. It operates in two steps; one by selecting a region of interest, before knowing its actual visual content, through an ultrafast/low resolution analysis of the full visual field, and the second providing a detailed categorization of the detailed "foveal" selected region attained with the saccade [52] (see Figure 11). In this perspective, our work would be a deepening of the knowledge and models necessary for the realization of the "what" pathway. Modeling this dual-pathways architecture allows for offering an efficient model of visual search as active vision. In particular, it allows us to fill the gap with the shortcomings of CNNs with respect to physiological performances [53]. In the future, we expect to apply this model to better understand visual pathologies in which there exists a deficiency of one of the two pathways [54] while contributing to the field of computer vision.

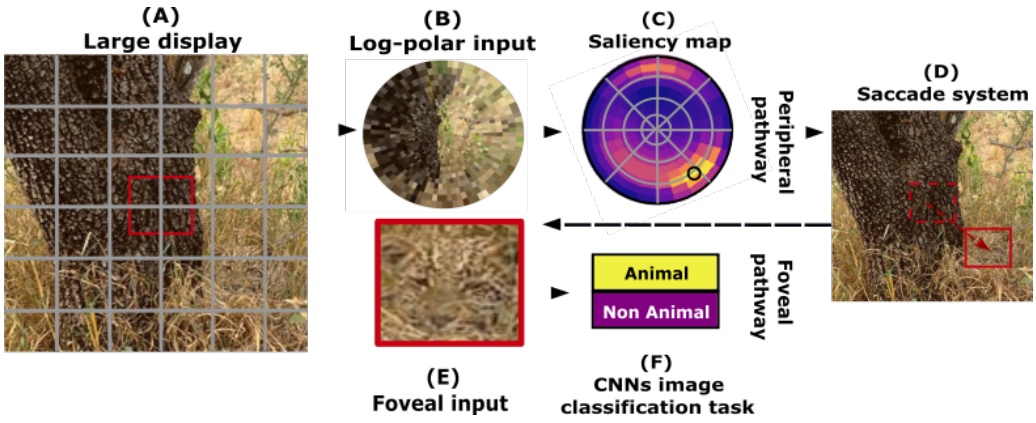

Figure 11. Model built over the anatomical visual processing pathways observed in mammals, namely the "what" and the "where" pathways: the peripheral pathway (top row) is applied to a large

display from a natural scene (**A**): it is first transformed into a retinotopic log-polar input (**B**), and we then learn to return a "saliency map" (**C**). The latter infers, for different positions in the target, the predicted accuracy value that can be reached by the foveal pathway, mimicking the "where" pathway used for global localization. The position with the best accuracy will feed a saccade system (**D**), adjusting the fixation point at the input of the foveal pathway (bottom row). It takes a subsample (**E**) of the large display (**A**), over which a categorization is done (**F**), mimicking the "what" pathway.

Author Contributions: Conceptualization, J.-N.J. and L.U.P.; methodology, J.-N.J.; software, J.-N.J.; validation, J.-N.J. and L.U.P.; formal analysis, J.-N.J. and L.U.P.; investigation, J.-N.J. and L.U.P.; resources, J.-N.J. and L.U.P.; data curation, J.-N.J. and L.U.P.; writing—original draft preparation, J.-N.J. and L.U.P.; writing—review and editing, J.-N.J. and L.U.P.; visualization, J.-N.J.; supervision, L.U.P.; project administration, L.U.P.; funding acquisition, L.U.P. All authors have read and agreed to the published version of the manuscript.

Funding: Authors received funding from the Agence Nationale de la Recherche project number ANR-20-CE23-0021 ("AgileNeuroBot https://laurentperrinet.github.io/grant/anr-anr/", accessed on 15 March 2023) and from the french government under the France 2030 investment plan, as part of the Initiative d'Excellence d'Aix-Marseille Université - A*MIDEX grant number AMX-21-RID-025 "Polychronies https://laurentperrinet.github.io/grant/polychronies/", accessed on 15 March 2023.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This work is made reproducible using the following tools. First, the code reproducing all figures is available at GitHub https://github.com/SpikeAI/2022-09_UltraFastCat/blob/main/Readme.md [55] (accessed on 15 March 2023), and in particular the code at DataSetMaker https://github.com/SpikeAI/DataSetMaker [37] (accessed on 15 March 2023) was used to retrieve images. The paper is available as an arXiv preprint https://arxiv.org/abs/2205.03635 with links to previous versions and to the code (accessed on 15 March 2023). Also find the associated zotero group https://www.zotero.org/groups/4560566/ultrafastcat (accessed on 15 March 2023) used to regroup relevant literature on the subject.

Acknowledgments: For the purpose of open access, the author has applied a CC BY public copyright licence to any author accepted manuscript version arising from this submission.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

(D)CNN	(Deep) Convolutional Neural Network
LUT	Look Up Table
MNIST	Modified or Mixed National Institute of Standards and Technology
SLS	Supervised Learning from Scratch
TLA	Transfer Learning on All layers
TLAA	Transfer Learning with Auto Augment function
TLC	Transfer Learning on Classification layers
TLDA	Transfer Learning with Data Augmentation
VGG	Vision Geometry Group

References

1. Cristóbal, G.; Perrinet, L.U.; Keil, M.S. *Biologically Inspired Computer Vision*; Wiley-VCH Verlag GmbH and Co. KGaA: Weinheim, Germany, 2015. [CrossRef]
2. Fabre-Thorpe, M. The Characteristics and Limits of Rapid Visual Categorization. *Front. Psychol.* **2011**, *2*, 243. [CrossRef] [PubMed]
3. Thorpe, S.; Fize, D.; Marlot, C. Speed of Processing in the Human Visual System. *Nature* **1996**, *381*, 520–522. [CrossRef] [PubMed]
4. Fabre-Thorpe, M.; Richard, G.; Thorpe, S.J. Rapid Categorization of Natural Images by Rhesus Monkeys. *Neuroreport* **1998**, *9*, 303–308. [CrossRef] [PubMed]
5. Delorme, A. Go-Nogo Categorization and Detection Task 2021. OpenNeuro Dataset. Available online: https://openneuro.org/datasets/ds002680/versions/1.2.0 (accessed on 15 March 2023).

6. Freedman, D.J.; Riesenhuber, M.; Poggio, T.; Miller, E.K. Categorical Representation of Visual Stimuli in the Primate Prefrontal Cortex. *Science* **2001**, *291*, 312–316. [CrossRef]
7. Rousselet, G.A.; Macé, M.J.M.; Fabre-Thorpe, M. Is It an Animal? Is It a Human Face? Fast Processing in Upright and Inverted Natural Scenes. *J. Vis.* **2003**, *3*, 440–455. [CrossRef]
8. Kirchner, H.; Thorpe, S.J. Ultra-Rapid Object Detection with Saccadic Eye Movements: Visual Processing Speed Revisited. *Vis. Res.* **2006**, *46*, 1762–1776. [CrossRef]
9. Mirzaei, A.; Khaligh-Razavi, S.M.; Ghodrati, M.; Zabbah, S.; Ebrahimpour, R. Predicting the Human Reaction Time Based on Natural Image Statistics in a Rapid Categorization Task. *Vis. Res.* **2013**, *81*, 36–44. [CrossRef]
10. Zhu, W.; Drewes, J.; Gegenfurtner, K.R. Animal Detection in Natural Images: Effects of Color and Image Database. *PLoS ONE* **2013**, *8*, e75816. [CrossRef]
11. Thorpe, S.J.; Gegenfurtner, K.R.; Fabre-Thorpe, M.; Bülthoff, H.H. Detection of Animals in Natural Images Using Far Peripheral Vision. *Eur. J. Neurosci.* **2001**, *14*, 869–876. [CrossRef]
12. Drewes, J.; Trommershäuser, J.; Gegenfurtner, K.R. Parallel Visual Search and Rapid Animal Detection in Natural Scenes. *J. Vis.* **2011**, *11*, 20. [CrossRef]
13. Fabre-Thorpe, M.; Delorme, A.; Marlot, C.; Thorpe, S. A Limit to the Speed of Processing in Ultra-Rapid Visual Categorization of Novel Natural Scenes. *J. Cogn. Neurosci.* **2001**, *13*, 171–180. [CrossRef] [PubMed]
14. Crouzet, S.M. What Are the Visual Features Underlying Rapid Object Recognition? *Front. Psychol.* **2011**, *2*, 326. [CrossRef]
15. Perrinet, L.U.; Adams, R.A.; Friston, K.J. Active Inference, Eye Movements and Oculomotor Delays. *Biol. Cybern.* **2014**, *108*, 777–801. [CrossRef] [PubMed]
16. Thorpe, S.; Fabre-Thorpe, M. Seeking Categories in the Brain. *Science* **2001**, *291*, 260–263. [CrossRef] [PubMed]
17. Delorme, A.; Rousselet, G.A.; Macé, M.J.M.; Fabre-Thorpe, M. Interaction of Top-down and Bottom-up Processing in the Fast Visual Analysis of Natural Scenes. *Cogn. Brain Res.* **2004**, *19*, 103–113. [CrossRef]
18. Delorme, A.; Richard, G.; Fabre-Thorpe, M. Key Visual Features for Rapid Categorization of Animals in Natural Scenes. *Front. Psychol.* **2010**, *1*, 21. [CrossRef]
19. Serre, T.; Oliva, A.; Poggio, T. A Feedforward Architecture Accounts for Rapid Categorization. *Proc. Natl. Acad. Sci. USA* **2007**, *104*, 6424–6429. [CrossRef]
20. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-Based Learning Applied to Document Recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
21. Rangdal, M.B.; Hanchate, D.B. Animal Detection Using Histogram Orinted Gradient. *Int. J. Recent Innov. Trends Comput. Commun.* **2014**, *2*, 7.
22. Grimaldi, A.; Gruel, A.; Besnainou, C.; Martinet, J.; Perrinet, L.U. Precise Spiking Motifs in Neurobiological and Neuromorphic Data. *Brain Sci.* **2022**, *13*, 68. [CrossRef]
23. Everingham, M.; Eslami, S.M.A.; Van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The PASCAL Visual Object Classes Challenge: A Retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136. [CrossRef]
24. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis. (IJCV)* **2015**, *115*, 211–252. [CrossRef]
25. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
26. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How Transferable Are Features in Deep Neural Networks? *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 3320–3328. [CrossRef]
27. Cichy, R.M.; Khosla, A.; Pantazis, D.; Torralba, A.; Oliva, A. Comparison of Deep Neural Networks to Spatio-Temporal Cortical Dynamics of Human Visual Object Recognition Reveals Hierarchical Correspondence. *Sci. Rep.* **2016**, *6*, 27755. [CrossRef]
28. Joubert, O.R.; Rousselet, G.A.; Fize, D.; Fabre-Thorpe, M. Processing Scene Context: Fast Categorization and Object Interference. *Vis. Res.* **2007**, *47*, 3286–3297. [CrossRef]
29. Kriegeskorte, N.; Mur, M.; Ruff, D.A.; Kiani, R.; Bodurka, J.; Esteky, H.; Tanaka, K.; Bandettini, P.A. Matching Categorical Object Representations in Inferior Temporal Cortex of Man and Monkey. *Neuron* **2008**, *60*, 1126–1141. [CrossRef]
30. Bao, P.; She, L.; McGill, M.; Tsao, D.Y. A Map of Object Space in Primate Inferotemporal Cortex. *Nature* **2020**, *583*, 103–108. [CrossRef]
31. Macé, M.J.M.; Joubert, O.R.; Nespoulous, J.L.; Fabre-Thorpe, M. The Time-Course of Visual Categorizations: You Spot the Animal Faster than the Bird. *PLoS ONE* **2009**, *4*, e5927. [CrossRef]
32. Mack, M.L.; Palmeri, T.J. The Dynamics of Categorization: Unraveling Rapid Categorization. *J. Exp. Psychol. Gen.* **2015**, *144*, 551–569. [CrossRef]
33. Fellbaum, C., Ed. *WordNet: An Electronic Lexical Database*; Language, Speech, and Communication, A Bradford Book; MIT Press: Cambridge, MA, USA, 1998.
34. Perrinet, L.U.; Bednar, J.A. Edge Co-Occurrences Can Account for Rapid Categorization of Natural versus Animal Images. *Sci. Rep.* **2015**, *5*, 11400. [CrossRef] [PubMed]
35. Drewes, J.; Wichmann, F.; Gegenfurtner, K.R. Classification of Natural Scenes Using Global Image Statistics. *J. Vis.* **2005**, *5*, 602. [CrossRef]
36. Wichmann, F.A.; Drewes, J.; Rosas, P.; Gegenfurtner, K.R. Animal Detection in Natural Scenes: Critical Features Revisited. *J. Vis.* **2010**, *10*, 6. [CrossRef] [PubMed]

37. Jérémie, J.N. Online GitHub Repository: Data Set Maker, 2022. Available online: https://github.com/SpikeAI/DataSetMaker (accessed on 15 March 2023).
38. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., dAlché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; pp. 8024–8035.
39. Jérémie, J.N.; Perrinet, L.U. Experimenting with Transfer Learning for Visual Categorization, 2021. Available online: https://laurentperrinet.github.io/sciblog/posts/2021-04-28-experimenting-with-transfer-learning-for-visual-categorization.html (accessed on 15 March 2023).
40. Schrimpf, M.; Kubilius, J.; Hong, H.; Majaj, N.J.; Rajalingham, R.; Issa, E.B.; Kar, K.; Bashivan, P.; Prescott-Roy, J.; Geiger, F.; et al. Brain-Score: Which Artificial Neural Network for Object Recognition Is Most Brain-Like? *bioRxiv Prepr. Serv. Biol.* **2020**. Available online: https://www.biorxiv.org/content/early/2020/01/02/407007.full.pdf (accessed on 15 March 2023). [CrossRef]
41. Cubuk, E.D.; Zoph, B.; Mane, D.; Vasudevan, V.; Le, Q.V. AutoAugment: Learning Augmentation Policies from Data. *arXiv* **2019**, arXiv:1805.09501.
42. Guyonneau, R.; Kirchner, H.; Thorpe, S.J. Animals Roll around the Clock: The Rotation Invariance of Ultrarapid Visual Processing. *J. Vis.* **2006**, *6*, 1. [CrossRef]
43. Biederman, I. Perceiving Real-World Scenes. *Science* **1972**, *177*, 77–80. [CrossRef]
44. Vanmarcke, S.; Van Der Hallen, R.; Evers, K.; Noens, I.; Steyaert, J.; Wagemans, J. Ultra-Rapid Categorization of Meaningful Real-Life Scenes in Adults With and Without ASD. *J. Autism Dev. Disord.* **2016**, *46*, 450–466. [CrossRef]
45. Bogadhi, A.R.; Buonocore, A.; Hafed, Z.M. Task-Irrelevant Visual Forms Facilitate Covert and Overt Spatial Selection. *J. Neurosci. Off. J. Soc. Neurosci.* **2020**, *40*, 9496–9506. [CrossRef]
46. Xu, B.; Kankanhalli, M.S.; Zhao, Q. Ultra-Rapid Object Categorization in Real-World Scenes with Top-down Manipulations. *PLoS ONE* **2019**, *14*, e0214444. [CrossRef]
47. Mehrer, J.; Spoerer, C.J.; Jones, E.C.; Kriegeskorte, N.; Kietzmann, T.C. An Ecologically Motivated Image Dataset for Deep Learning Yields Better Models of Human Vision. *Proc. Natl. Acad. Sci. USA* **2021**, *118*, e2011417118. [CrossRef]
48. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. *arXiv* **2016**, *9905*, 21–37. [CrossRef]
49. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *arXiv* **2016**, arXiv:1506.01497.
50. Mishkin, M.; Ungerleider, L. Object Vision and Spatial Vision: Two Cortical Pathways. *Trends Neurosci.* **1983**, *6*, 414–417. [CrossRef]
51. Daucé, E.; Albigès, P.; Perrinet, L.U. A Dual Foveal-Peripheral Visual Processing Model Implements Efficient Saccade Selection. *J. Vis.* **2020**, *20*, 22. [CrossRef]
52. Yarbus, A. Eye Movements during the Examination of Complicated Objects. *Biofizika* **1961**, *6*, 52–56.
53. New, J.; Cosmides, L.; Tooby, J. Category-Specific Attention for Animals Reflects Ancestral Priorities, Not Expertise. *Proc. Natl. Acad. Sci. USA* **2007**, *104*, 16598–16603. [CrossRef]
54. Wiecek, E.; Pasquale, L.; Fiser, J.; Dakin, S.; Bex, P. Effects of Peripheral Visual Field Loss on Eye Movements During Visual Search. *Front. Psychol.* **2012**, *3*, 472. [CrossRef]
55. Jérémie, J.N.; Perrinet, L.U. Online GitHub repository: SpikeAI/2022-09_UltraFastCat: Ultra-fast Categorization of Image Containing Animals in Biology and Neural Models, 2022. Available online: https://github.com/SpikeAI/2022-09_UltraFastCat (accessed on 15 March 2023).