A feedforward architecture accounts for rapid categorization

Parnian Taheri[a]

[a] *Basics of Neuroscience,*

Abstract

Animals and humans are capable of recognising rapid visual presentation tasks and rapid categorization. According to the time latency in neuronal activities of our visual pathway, such rapid visual processing is likely to be mostly feedforward. In this experiment, an extended Hubel and Wiesel simple-to-complex cell hierarchy theory is used. The aim of this experiment is to see whether the mentioned theory can reach the accuracy and speed of human in recognizing objects. To meet that goal, the performance achieved by both humans and machine is compared in an animal vs non-animal task.

1. Introduction

Understanding how the brain recognizes objects is a fundamental question in neuroscience, with implications for fields ranging from artificial intelligence to cognitive psychology. The ventral visual pathway, traversing from the primary visual cortex (V1) through extrastriate visual areas II (V2) and IV (V4) to the inferotemporal cortex (IT), has long been implicated in object recognition processes. Recent studies in nonhuman primates have shed light on how this pathway works, revealing important details about how the brain recognizes objects. Across this pathway, there's a noticeable increase in the brain's ability to ignore changes in position and size, as well as a preference for more complex stimuli. We've also observed the brain's capacity to adapt and learn, particularly in areas like IT and the prefrontal cortex (PFC). Although the role of feedback connections in the brain is still debated, recent research suggests that a simple hierarchical system may underlie the brain's ability to recognize objects quickly and accurately.

The model used in this paper is inspired by the Hubel and Wiesel proposal, which is using simple and complex cells or S and C units respectively. S_1 cell is for detecting edges and simple orientations. The simple S units perform a bell-shaped tuning operation over their inputs. On the other hand, C_1 cell is used to make the model insensitive to location and scale by getting max pulling.

2. Dataset

The dataset used for this task contains 1200 images, which are 600 animal and 600 distractor images consists of four categorize; face, close body, medium body and far body. These images are in the form of RGB and they need to be grayscaled. For the learning algorithm, the dataset is split into 4 directories, train and test for both distractors and animals. Also for the psychophysic task, 1200 images are divided into 10 folders for 10 trials.

Figure 1: Animal Pictures - Close Body, Medium Body, Far Body, Head

Figure 2: Non-Animal Pictures - Close Body, Medium Body, Far Body, Head

3. Machine Performance

For analysing the performance of machine in the animal vs non-animal task, a pre-written code was given to us and we had to change it to make it suitable for our dataset. For that reason, few changes is made in that code. Firstly, in order to initialize four variables, train_set.pos, train_set.neg, test_set.pos, test_set.neg, I created four directories each has 300 grayscaled images. For example for train_set.pos I put 300 images of each four categories of animals in the folder. Moreover, due to the error that occured beacuse of the svm part, I commented those lines. If you run the code, after a long time four data will be given which is Xtrain, Xtest, ytrain and ytest, which we are going to run some analyse on. These data are stored in a file named "Data".

3.1. Learning Algorithm

After getting data, it is time for the classification. In order to classify the data, I used a built-in Matlab Application, "Classification Learner". There are so many classifiers in that App, however, because the main research has used SVM, I also choosed Linear SVM.

4. Human Performance

In order to get the humans performance in categorization, a psychophysic task is written. In this task, 1200 images of animal and distractors are shown to the subjects in 10 blocks in a way that a few second a noisy images is shown and after that the image is shown in 0.08 seconds. Images are shown randomly, this means that every time they will be shown in a different order. Subjects are asked to choose whether an image is an animal by pressing right and left arrow of the keyboard. Some data will be stored in two files. One of them is a .txt file that the total accuracy and reaction time, number of true and false answers , ... will be stored another one is a .mat file, which some cells containing true answers, subjects answers, type of the image, name of the image, reaction time, ... will be stored.

4.1. Stored Data

Some data are saved from each task. Firstly, when the subject run the code a directory is created by its name. and a ".txt" file is also created in that folder to save some results. In that file, the name and number of trial is saved in the begining. When the trial is finished, the number of true and false answers in conjunction with the TPR, FPR, TNR and FNR is written in that file. Also, the number of true or false answers for each four categories and the average reaction time is written. Moreover, a ".mat" file is also created. In this file, some cells are stored such as real answer and the subject's answer, file's name, type of the image (body, far, medium, face) and reaction time.

4.2. Confidence Bar

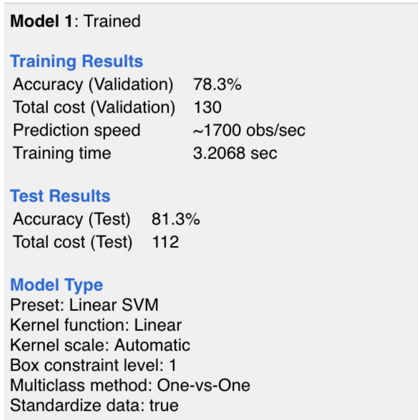
After choosing whether the picture is an animal, there are some bars that shows the confidence of the answer. It means that when you go on each bar the color is changed and when you click on it the next image is shown and your confidence rate is stored. This will be used in the accuracy. It means instead of adding the true answer with one, it will be added with the confidence rate.

5. Results

In the beginning, the results of the written code is studied.

5.1. Learning Code

Here is the accuracy, ROC curve and Confusion Matrix of the Validation and Test when all the data is given as a test.


Model 1: Trained

Training Results

Accuracy (Validation)	78.3%
Total cost (Validation)	130
Prediction speed	~1700 obs/sec
Training time	3.2068 sec

Test Results

Accuracy (Test)	81.3%
Total cost (Test)	112

Model Type
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Figure 3: Accuracy of Validation and Test using Cross Validation

We want to know how does the algorithm work for each categories, therefore after training the model we give each category separately to test the model.

5.2. Psychophysic

After data are gathered from subjects, it's time to analyse it. First, we should load the files and extract the desired data. In order to do that, I load the files and read them line by line and store the numbers in a cell. After that, I get average among all the trials and subjects. Finally, the results are visualized. In the following some of them are shown.

According to the diagrams, the reaction time for both animal and non-animal is higher in Far body. Moreover, the confidence in Medium and Far Body is less than the rest. It is also obvious that the accuracy in Far body is less than the rest and the non-animal pictures give more accuracy than animal. It is also observable that the pattern in both with and without confidence is quite equal.

6. Concluaion

According to the results gotten from both psychophysic 91.5% and learning algorithm 81.3%, we can say that the algorithm is not close to the human performance. Moreover, by the results we understand that human performance is not perfect in Far Bodies.

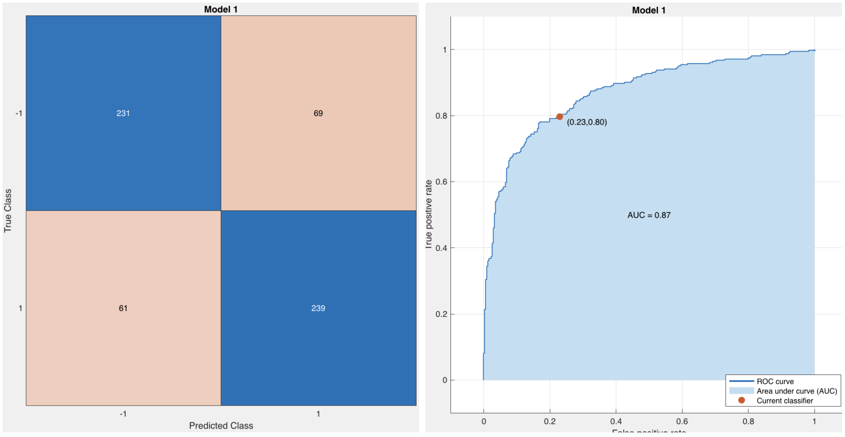


Figure 4: ROC curve and Confusion Matrix of Validatation

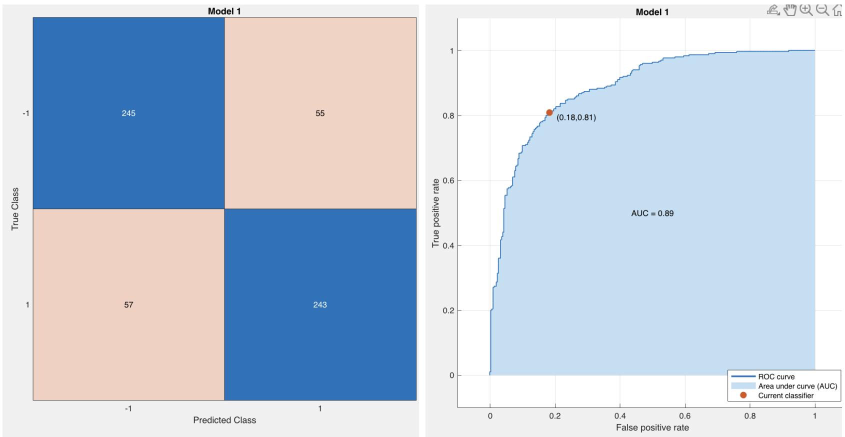


Figure 5: ROC curve and Confusion Matrix of Testset

	Head	Close Body	Medium Body	Far Body	Overall
Animal Reaction time	0.39	0.41	0.48	0.49	0.45
non-Animal Reaction time	0.45	0.45	0.45	0.45	0.45
Confidence accuracy	93.49 %	94.47 %	89.16%	82.56%	89.92%
non-Confidence accuracy	94.56 %	95.45 %	90.56 %	85.44%	91.5%

Table 1: The Reaction Time and Accuracy of animal and non-animal

Model 1: Trained

Training Results
Accuracy (Validation) 80.0%
Total cost (Validation) 120
Prediction speed ~1800 obs/sec
Training time 2.9047 sec

Test Results
Accuracy (Test) 78.0%
Total cost (Test) 33

Model Type
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Figure 6: Head

Model 1: Trained

Training Results
Accuracy (Validation) 80.0%
Total cost (Validation) 120
Prediction speed ~1800 obs/sec
Training time 2.9047 sec

Test Results
Accuracy (Test) 88.0%
Total cost (Test) 18

Model Type
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Figure 7: Close Body

Model 1: Trained

Training Results
Accuracy (Validation) 80.0%
Total cost (Validation) 120
Prediction speed ~1800 obs/sec
Training time 2.9047 sec

Test Results
Accuracy (Test) 86.7%
Total cost (Test) 20

Model Type
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Figure 8: Medium Body

Model 1: Trained

Training Results
Accuracy (Validation) 80.0%
Total cost (Validation) 120
Prediction speed ~1800 obs/sec
Training time 2.9047 sec

Test Results
Accuracy (Test) 71.3%
Total cost (Test) 43

Model Type
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Figure 9: Far Body

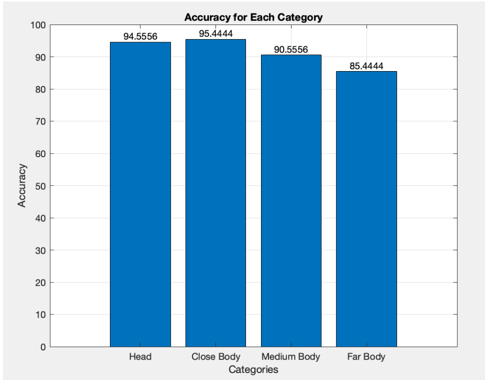


Figure 10: Accuracy of all data without confidence

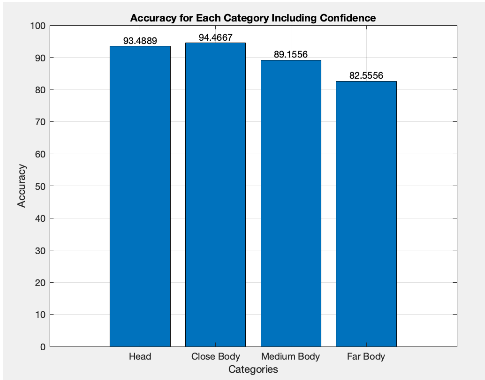


Figure 11: Accuracy of all data with confidence

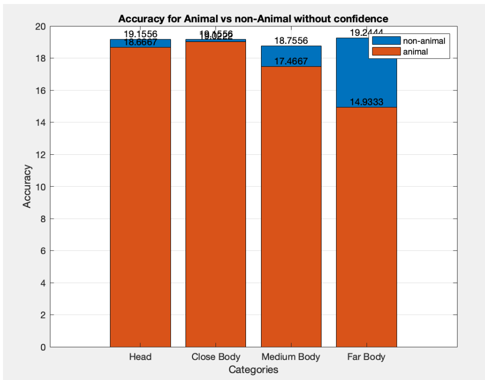


Figure 12: Accuracy of animal vs non-animal without confidence

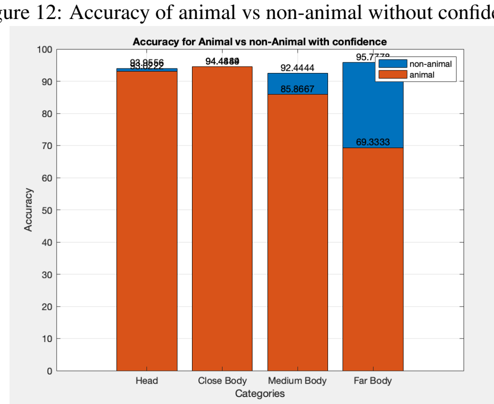


Figure 13: Accuracy of animal vs non-animal with confidence

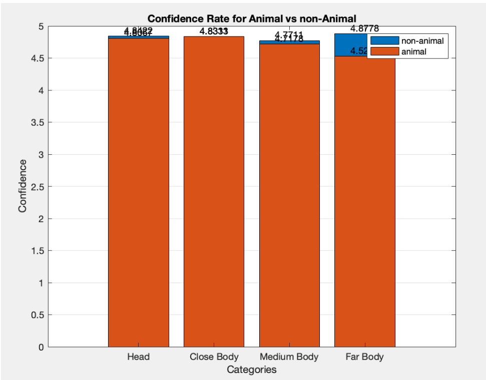


Figure 14: Confidence Rate

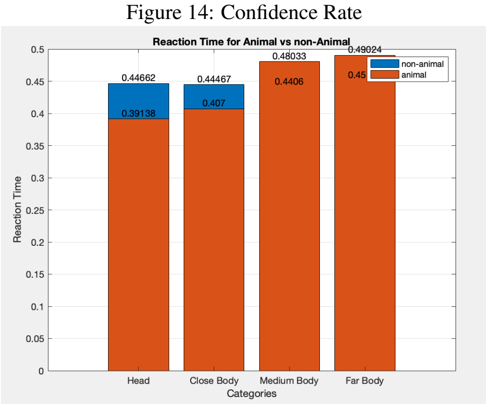


Figure 15: Reaction Time