International Journal of Applied Sciences and Smart Technologies

Volume 3, Issue 1, pages 55–64

p-ISSN 2655-8564, e-ISSN 2685-9432

Stone, Paper, Scissors Mini-Game for AI Pet Robot

Aditya Aspat[1,*], Elton Lemos[1], Abhishek Ghoshal[1]

[1]*Department of Computer Engineering,*
Xavier Institute of Engineering, Mahim, Mumbai, Maharashtra, India
[*]*Corresponding Author: aditya.dir@gmail.com*

(Received 20-07-2020; Revised 19-12-2020; Accepted 19-12-2020)

Abstract

The Artificial Intelligence (AI) Pet Robot is a combination of various fields of computer science. This paper showcases the various functionalities of our AI Pet. Most of the functionalities showcased use the immage processing modules made available through OpenCV. The pet robot has various features such as emotion recognition, follow routine, mini-game etc. This paper discusses the mini-game aspect of the robot. The game has been developed by using VGG16 convolutional network for identification of the action performed by the user. To improve the accuracy we have made use of background subtraction which gives removes all the unwanted objects from the background and gives a simple cutout of the users hand.

Keywords: Pet Robot, VGG16, background subtraction

1 Introduction

Owning a pet comes with a zillion benefits for our physical as well as mental health [1]. Pets give us increased opportunities to go outside, exercise and socialize with people. In today's time such as the pandemic, having a pet by your side gives us

Volume 3, Issue 1, pages 55–64
p-ISSN 2655-8564, e-ISSN 2685-9432

company when we are required to social distance from everyone else. However, with all the benefits of owning a pet, there are some drawbacks as well [2]. Pets can be a source for transmission of various diseases if they are not properly taken care of. Many societies have rules which don't allow pets inside the buildings. They may start barking in the middle of the night which might cause nuisance in some cases to the neighbors. For the reasons given above, we propose the development of an inexpensive pet using modern technologies like artificial intelligence (emotion recognition and face recognition), internet of things (for communicating with the bot and it's movement), image processing (building a mini game) which will seamlessly interact with humans and comes with all the benefits while avoiding the drawbacks of owning a pet.

2 Existing Systems

There are a few pet robots that are already available in the market. Learning about these systems will help better understand how our system compares to them. This section also looks into the different algorithms that have previously been implemented regarding our robot and its functionalities.

There have been multiple attempts at making realistic AI pets with some dating back to 1990s. Tamagotchis were handheld digital pets created in Japan in 1997 [3]. This representation of pets of pets quickly became popular. The colourfully designed creatures would grow differently based on the level of care provided by the user. In more recent times companies like Sony have come up with realistic pets such as the Aibo [4]. It was a slew of features such as facial recognition, emotion detection, automatic battery charging and many more. Pibo is another such AI pet created by Circulus [5]. Pibo's features include weather reports, alarms, notification, taking photos and other interactive functions. There is however one common problem with these products. Their cost is upwards of $1000. One of our goals is to create a fully functioning AI pet by using equipment that is cheap and commercially available.

International Journal of Applied Sciences and Smart Technologies

Volume 3, Issue 1, pages 55–64

p-ISSN 2655-8564, e-ISSN 2685-9432

3 Implementation Methodology

The architecture of our pet closely resembles a living pet. Our pet has 4 major parts:

- The Brain: The computer acts as the brain and helps with all the computation.
- The Spine: All the messages flow through it, the Pi4B acts as a kernel.
- The Eyes: The NOIR camera are the eyes of our bot. They help in capturing images.
- The Limbs: The Arduino UNO and its motors acts as the limbs and perform various movements as per the commands given to it.

Hardware Design

- Vision Unit:

  This unit is responsible for what the bot sees. It helps in capturing the images and sending them to the computer for Image Processing. It comprises of the Raspberry 4 and its camera module. We have also used a NOIR camera which helps the bot in capturing images in low light. The camera's video stream is hosted using the RPi-Cam-Web-Interface. The computer captures this stream and uses it for various processing features

- Motor Unit:

  This unit is responsible for the movement of our bot. It comprises of various actuators like DC Motor and Servos which are controlled by the Arduino UNO. The power for the motor unit is provided by a 12V battery. The Arduino oversees the motor operations but the final decision as to what movement should be performed is taken by the Computer.

- Communication Unit:

  This Unit is the tunnel for information transfer from the Computer to the Pi4B and to the Arduino UNO and vice versa. This Unit consists of a Serial Interface between the Pi4B and the Arduino UNO which is used by the Pi4B to instruct the Arduino or forward instructions given by the Computer. There is also a Socket connection between the Pi4B and the Computer for wireless communication using TCP.

Volume 3, Issue 1, pages 55–64
p-ISSN 2655-8564, e-ISSN 2685-9432

- Interfacing with Robot:

  The robot has a joystick for an arm. If the user shakes hands with the robot, the Robot goes into listening mode and hears any commands given by the user. If the robot cannot understand, the robot will inform the user that it couldn't understand and then continue what it was doing previously. The list of commands that the robot can obey are:

- Stop:

  On receiving a STOP command, the robot will stop whatever it is doing and will enter the face detection mode.

- Tell me the time:

  The UTC time is grabbed from the servers and is displayed on the screen as well as spoken out loud by the bot.

- Read "book name":

  The robot makes use of the Google Text-to-Speech module to read out books to the user.

- Follow me:

  On receiving the "Follow Me" command, the bot starts tracking the user's movement and follows him or her while maintaining a safe distance.

- Play a game:

  Saying this will start the game of stone, paper and scissors. The action performed by the bot is displayed on the screen attached to it.

4 Implementation

This section elaborates on the algorithms we have used for implementation of the stone, paper and scissors game. How they work, and how integrating all of them makes the game easy to play for anyone.

4.1 VGG16

We have used the VGG16 convolution network model proposed by K. Simonyan and

Volume 3, Issue 1, pages 55–64

p-ISSN 2655-8564, e-ISSN 2685-9432

A. Zisserman from the University of Oxford in the paper "Very Deep Convolutional Networks for Large-Scale Image Recognition" [6]. The model achieves 92.7% top-5 test accuracy in ImageNet, which is a dataset of over 14 million images belonging to 1000 classes. The architecture [7] of VGG16 is as in Figure 1.

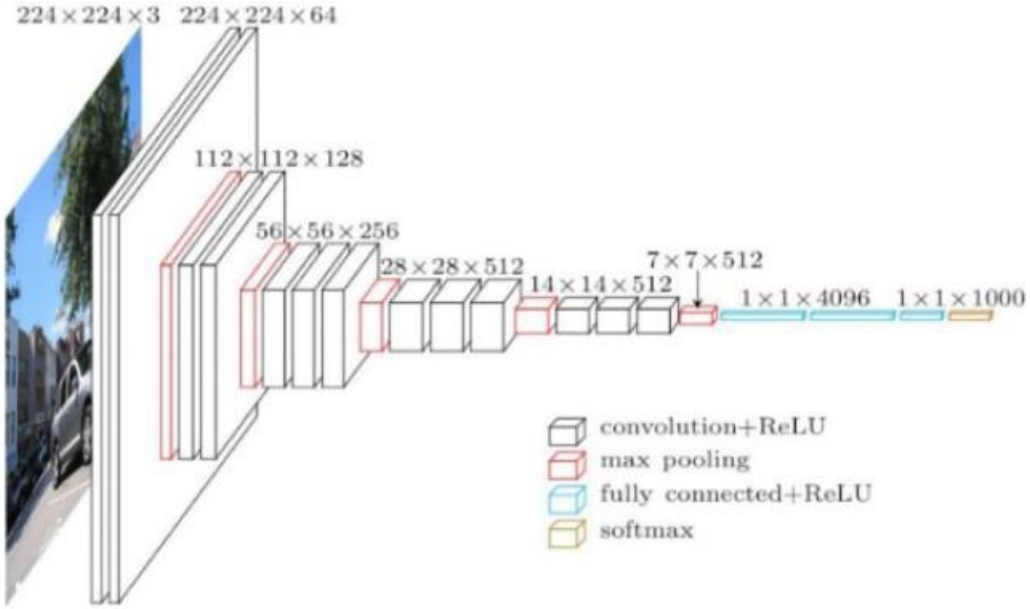

Figure 1. VGG16 architecture

4.2 Training

Deep convolutional neural network models may take days or even weeks to train on very large datasets. A way to short-cut this process is to re-use the model weights from pre-trained models that were developed for standard computer vision benchmark datasets, such as the ImageNet image recognition tasks. Top performing models can be downloaded and used directly, or integrated into a new model for your own computer vision problems. The way to do this is Transfer Learning where we apply pre-trained models on our own dataset to get a state of the art model. We used 550 images with variation for all the three cases and achieved an accuracy of 98% in the validation set. This provides a very good platform to test the model in a real time scenario. As we have already seen in previous implementations, the real time accuracy of models is affected due to the deviations from ideal test conditions. Training images are shown in Figure 2.

International Journal of Applied Sciences and Smart Technologies

Volume 3, Issue 1, pages 55–64

p-ISSN 2655-8564, e-ISSN 2685-9432

Figure 2. Training images

```
acc: 0.5817 - val_loss: 0.9739 - val_acc: 0.6077

acc: 0.6691 - val_loss: 0.7196 - val_acc: 0.6731

acc: 0.8196 - val_loss: 0.3430 - val_acc: 0.8692

acc: 0.9337 - val_loss: 0.1310 - val_acc: 0.9588

acc: 0.9733 - val_loss: 0.0670 - val_acc: 0.9661

acc: 0.9871 - val_loss: 0.0881 - val_acc: 0.9782

acc: 0.9814 - val_loss: 0.1783 - val_acc: 0.9564
```

Figure 3. Model accuracy

In Figure 3, metrics for some of the epochs are shown. acc is the accuracy of the model while training the model, that's is the total number of accurately classified images divided by the number of images in the training set. Val loss is the value of the cost function for the validation data. Val acc is the accuracy of the validation set, that is the total number of accurately classified images divided by the total number of images in the validation set. Some of the barriers we faced while performing real time testing were related to both hardware and software. These have been mentioned in the real time testing explanation below.

5 Results and Discussion

In this section, we provide our research results about Real Time Results for Mini-Game. As shown in the fowchart (see Figure 4), right after we turn on the camera, we take a snapshot of an empty background. We make use of background subtraction [8] to take a cutout of the hand. We need to make sure no object in the background does not move, especially within the boundaries of the blue square. In the image below, the left side is the normal image and the right side shows a continuous background subtracted

Volume 3, Issue 1, pages 55–64
p-ISSN 2655-8564, e-ISSN 2685-9432

image. Since the person is the only thing in the image that is moving, an outline comes around the person.

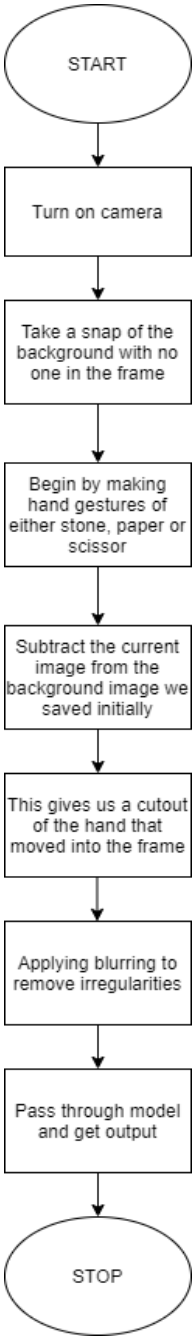

Figure 4. Flowchart for realtime testing

What we do next, is that we use the empty frame we captured to detect any new objects moving into the blue square, i.e. our hand. We cut out this part of the image and

Volume 3, Issue 1, pages 55–64

p-ISSN 2655-8564, e-ISSN 2685-9432

apply blurring so that we reduce the noise that we encounter due to the shortcomings of the camera. As it gets darker, the graininess of the image will increase giving rise to even more noise.

1. The program takes the image of the still background. This will be used later to subtract from the real time image (see Figure 5).

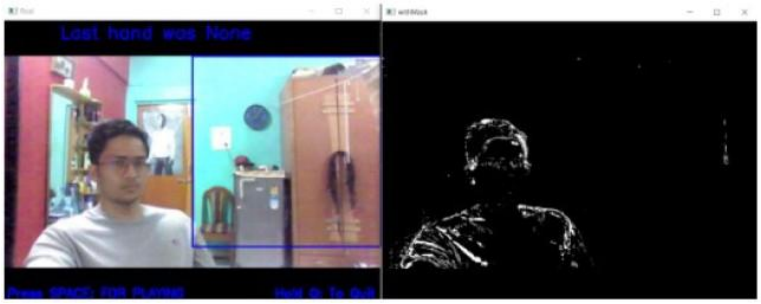

Figure 5. Continuous background subtraction

2. We perform the gesture we wish to do and the program captures this image (see Figure 6).

Figure 6. Gesture made by user

3. We subtract the current image with the saved image. This gives use the required format of the image (see Figure 7).

Figure 7. Background subtraction with hand

Volume 3, Issue 1, pages 55–64

p-ISSN 2655-8564, e-ISSN 2685-9432

4. We cut out the part of the image we need (hand) which is then fed to the model (see Figure 8).

Figure 8. Cut out of hand

The previously encountered software issues involved difficulty in extracting just the hand from the image in such a way that it resembles the images that the model was trained on. This is necessary to reduce any real time errors that we may encounter. All these steps had to be performed in order to get our real time images to be fed to the model, as close to the data set as possible. Only then would we be able successfully use our trained model.

6 Conclusion

The proposed algorithms show the functionality of our AI Pet Robot. We have used VGG16 and background subtraction to implement a simple game of stone, paper and scissors and integrated it in out AI Pet Robot with various other features. The use of background subtraction has helped us solve the issues we faced while taking a cut out of the action performed by the user. This game can be played by children to pass time and have some kind of entertainment without being exposed to the internet/mobile devices at a young age.

Volume 3, Issue 1, pages 55–64

p-ISSN 2655-8564, e-ISSN 2685-9432

References

[1] Healthy Pets, Healthy People. U.S. Department of Health & Human Services. https://www.cdc.gov/healthypets/healthbenefits/index.html (Accessed: 02/03/2020).

[2] E. Paul Cherniack, MD and Ariella R. Cherniack, "Assessing the benefits and risks ofowning a pet." *Canadian Medical Association Journal*, 2015.

[3] The life and death of Tamagotchi and the virtual pet, https://wellcomecollection.org/articles/WsT4Ex8AAHruGfWb (Accessed: 13/08/2020).

[4] S. Aibo: The Dog and Personal Assistant of the Future, https://www.forbes.com/sites/moorinsights/2019/05/01/sony-aibo-the-dog-and-personal-assistant-of-the-future (Accessed: 05/03/2020).

[5] pibo, https://pibo.circul.us (Accessed: 05/03/2020).

[6] Simonyan, K. Zisserman and Andrew. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv 1409*.1556, 2014.

[7] Step by step VGG16 implementation in Keras for beginners, https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c (Accessed: 08/03/2020).

[8] Using Background Subtraction, https://docs.opencv.org/master/d1/dc5/tutorial background subtraction.html (Accessed: 10/03/2020).