Playing Catch with Boston Dynamics' Spot

Raphael Napinas
Department of Computer Science
Cal Poly Pomona
rcnapinas@cpp.edu

Edward Zhang
Computer Science Department
Mount San Antonio College
jzhang165@student.mtsac.edu

Dr. Fang Tang
Department of Computer Science
Cal Poly Pomona
ftang@cpp.edu

***Abstract*—Human-robot interaction is a futuristic ideal where robots can be of benefit to humans in daily life. Cooperation can be difficult due to robots' often limited knowledge and mobility. In this research, we investigate the extent of a quadruped robot's ability to interact with humans in a way mirroring living dog. To do so, we demonstrate an object detection model trained using transfer learning [1], in tandem with a motorized payload design to enable Spot to play "catch" with its owners.**

Keywords—human-robot interaction, payload, transfer learning, object detection

I. Introduction & Research Goal

Boston Dynamics' Spot [2] is a quadruped robot intended to autonomously survey and inspect situations that are potentially hazardous to humans. Spot's accompanying arm paired with its extensive cameras and sensing suite unlock dexterous manipulation capabilities. Its vast functionalities have been used in several industries, such as handling toxic materials or investigating possible weaponized threats. However, this research aims to examine a more entertainment-focused application and test Spot's potential for commercial use. In pursuing this, it is our hope to expand Spot's capability to more daily use to the average citizen and advance human-robot interaction to increase quality of life.

This research aims to construct a lightweight and compact payload specifically tailored to launch racquetballs. Additionally, transfer learning object detection models will be trained for Spot to detect balls and humans to deliver the ball via the payload. The concept is a twist on the classic game of "fetch" played with dogs, turned into a form of "catch", such that a user would throw the ball onto the ground for Spot to detect, locate the user, and assume a propulsion sequence to "throw" the ball to the user.

II. Method

A. Payload Design

This work is inspired by Tennibot [3], an autonomous tennis assistant designed to automatically pick up balls on the court. In addition, the latest iteration of the propulsion system is heavily influenced by Scott Kraemer's "Project Lazy Ball" [4]. Mounted atop Spot's railings (see Figure 1), it uses a dual flywheel mechanism to propel balls out a pipe towards its target. The motors are controlled with a DBH-12 H-Bridge motor driver connected to an ESP32 microcontroller. The whole system is powered through the pins on Spot's General Expansion Payload.

Figure 1. (Left) Spot Payload Prototype
(Right) General Expansion Payload/Power Supply

B. Object Detection

The software components are largely derived from Boston Dynamics' tutorial "Playing Fetch with Spot" [5], apart from some optimizations done to the model training and changes to the robot manipulation to turn fetch into catch. The TensorFlow 2 Object Detection API is used to create an object detection model. Using the EfficientDet model, a pre-trained model able to detect a wide variety of generic objects, we exploit its already learned abilities to improve its detection of our specific ball.

We begin by taking pictures on Spot's front, side, and rear cameras. Over 750 pictures capture a variety of scenarios: with and without the ball in view, inside, outside, on grass, on cement, with varying amounts of shade, angles, and distances. Next, we use the LabelImg program to annotate the location of the ball in each of the images (if in view). Once annotated, we conducted training of a new model via transfer learning. By passing in our new dataset, our model creates new neural network layers for it to distinguish our ball from other objects. After model training, we export it to a TensorFlow SavedModel configuration for evaluation and use on Spot in real time.

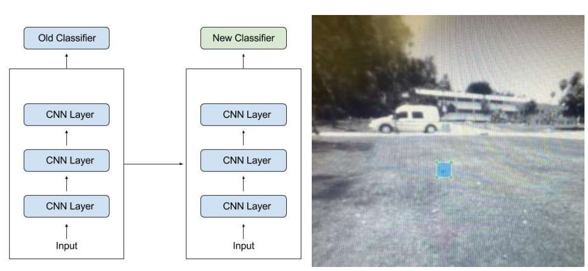

Figure 2. (Left) Breakdown of Transfer Learning Process [3]
(Right) Example Image Annotation

C. *Robot Manipulation and "Catch"*

The Python script for Spot to play catch is adapted from Boston Dynamics' script to play fetch. Through Spot's built-in Wi-Fi network, we utilize the Network Compute Bridge [6], a server used for our computers to communicate to and from Spot. We can then broadcast the SavedModel for Spot to use and detect balls in the area. Additionally, we broadcast a pre-trained Resnet model, already able to detect humans in view. With both models available to Spot, we can conduct the catch script. The function makes use of the Boston Dynamics' API to capture bounding boxes of the ball in real time and manipulate Spot's movements.

The script's loop procedure is as follows:

1. Take control of Spot and connect to models.
2. Continuously search for ball using all cameras.
3. Once found, navigate to the target and pick up ball.
4. Search for the nearest human in view.
5. While carrying the ball, navigate to the found human.
6. Get within two feet of the human, turn 180 degrees, then walk five feet forward.
7. Move arm above Spot's and release ball into payload.

When the human is detected, a ROS2 publisher sends out a message telling the ESP32 to begin the launch sequence and slowly accelerate the motors. This system takes advantage of Spot's onboard network and the ESP32's Wi-Fi capabilities.

III. Experiments & Results

After 40,000 training steps, we observed our model's loss value decrease from around 8.5 to near 0.2, showing increasing accuracy in our model's predictions. The models mean average precisions were around 0.6 and 0.35 for medium and small objects respectively. What we could infer is that larger objects (when the ball is closer to the cameras) are more likely to be recognized.

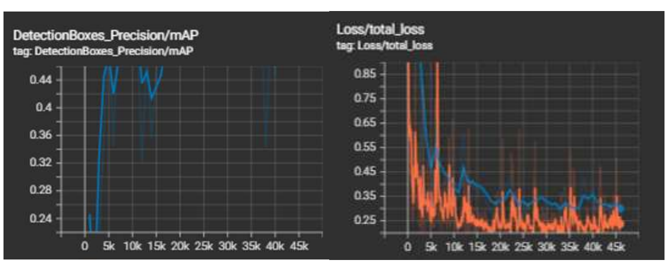

Figure 3. Mean Average Precision and Loss Value Graphs

In practice, we used Spot's ML Model Viewer utility to test our detection models live without conducting catch. Through Spot's controller, we could see bounding boxes around the ball with confidence levels between 85 and 95%. When running Boston Dynamics' premade fetch script, Spot was successfully able to pick up the ball and bring it to the nearest person.

We originally had only trained our model with pictures of the ball taken inside. When testing detection outside, we found Spot had trouble detecting the ball, and if it did it had very little confidence. This was likely due to model overfitting, which prompted us to retrain our model with added images taken outside.

IV. Conclusion & Future Work

We have presented a functional object detection model, which when used on Spot, can pick up and bring racquetballs to the nearest person and prepare for propulsion. However, this work is still in progress, ROS2 needs to be implemented for Spot-payload communication, and model optimization is still being done. Current progress shows the feasibility of the research goal, which will hopefully be completed in the coming months.

Figure 4. Successful Ball Detection and Pickup

This experiment was done to hopefully bridge the gap between humans and robots a little more. While our application was purely for entertainment, the work could be diversified and used for other household objects. Future research can also be oriented towards assisting the disabled and other groups with daily tasks. Recently, Boston Dynamics showcased integration with ChatGPT [7], allowing Spot to analyze human speech and respond vocally using artificial intelligence. A potential application of this would be assisting the elderly living alone. Ultimately, we foresee a future with citizens petting, playing, and speaking with robotic dogs in the same way as ones today.

References

[1] "What is transfer learning? exploring the popular deep learning approach.," Built In, https://builtin.com/data-science/transfer-learning (accessed Aug. 20, 2023).

[2] "Spot," Boston Dynamics, https://bostondynamics.com/products/spot/ (accessed Aug. 18, 2023).

[3] "How does it work?," Tennibot, https://www.tennibot.com/how-it-works/ (accessed Aug. 18, 2023).

[4] "Project lazy ball - a tennis ball thrower," YouTube, https://www.youtube.com/watch?v=wwKdipPjfz8 (accessed Aug. 20, 2023).

[5] Playing Fetch with Spot - https://dev.bostondynamics.com/docs/python/fetch_tutorial/fetch1 (accessed Aug. 20, 2023).

[6] Machine Learning Bridge and External Compute https://dev.bostondynamics.com/docs/concepts/network_compute_bridge (accessed Aug. 20, 2023).

[7] A. Mok, "These robot dogs can't bark - but they can speak in full sentences thanks to chatgpt," Business Insider, https://www.businessinsider.com/boston-dynamics-robot-dogs-can-speak-full-sentecnes-ai-chatgpt-2023-5 (accessed Aug. 18, 2023)