Alexis J. Renderos, William Lavelle, Rolf Anderson, Ritvik Maripally

Adaptive Visual Target Identification and Tracking through Convolutional Neural Network (CNN) based Machine Learning and Computer Vision

Abstract

The aim of this project is to identify a chosen object of interest at range using a camera feed and an image processing algorithm to assist in the detection, identification and tracking of the target. This solution was then implemented as a rail-mounted tactical scope attachment for a Nerf Blaster. The attachment provides the user with real-time feedback on their aim using a side mounted screen.

Data Collection and Analysis

To properly analyze incoming image data, we must first determine and analyze key characteristics of our target. The three chosen targets (Fig. 1) were photographed over 1,000 times each in varying lighting conditions and at varying angles, distances, and varying degrees of occlusion. This provided us with a set of testing data and training data for our machine learning based target detection and identification algorithm. We also analyzed the colors and markings of the targets in addition to the angle of the target relative to the camera and the lighting conditions. This was later used to create an object tracking algorithm.

Target Detection and Identification

Target detection and identification is primarily achieved through the implementation of a machine learning based algorithm. A *Convolutional Neural Network* (CNN) was trained on a dataset of nearly 3,000 photos of the three targets in order to create a comprehensive profile of what the targets looked in the eyes of a digital camera. The resulting model was then used to analyze incoming camera data to locate and identify any potential targets in its field of view.

Target Tracking

Target tracking is accomplished using a hybridized algorithm known as KCF, or *Kernelized Correlation Filters*. We chose this algorithm due to its high speed, exceptional accuracy, and excellent failure detection. Unfortunately, KCF does not recover well from full occlusion, but the target tracking algorithm is able to recover from this using the target detection and identification capabilities of the machine learning model.

Hardware Implementation

As a proof of concept, we took our two algorithms and implemented them as a camera-based, rail mounted tactical scope attachment for a Nerf Blaster. The computational side is supported by a Raspberry Pi 3 connected to a camera, while power is sourced from the batteries inside the Nerf Blaster. Attached to the barrel of the blaster is an RGB light ring to help light up the target.
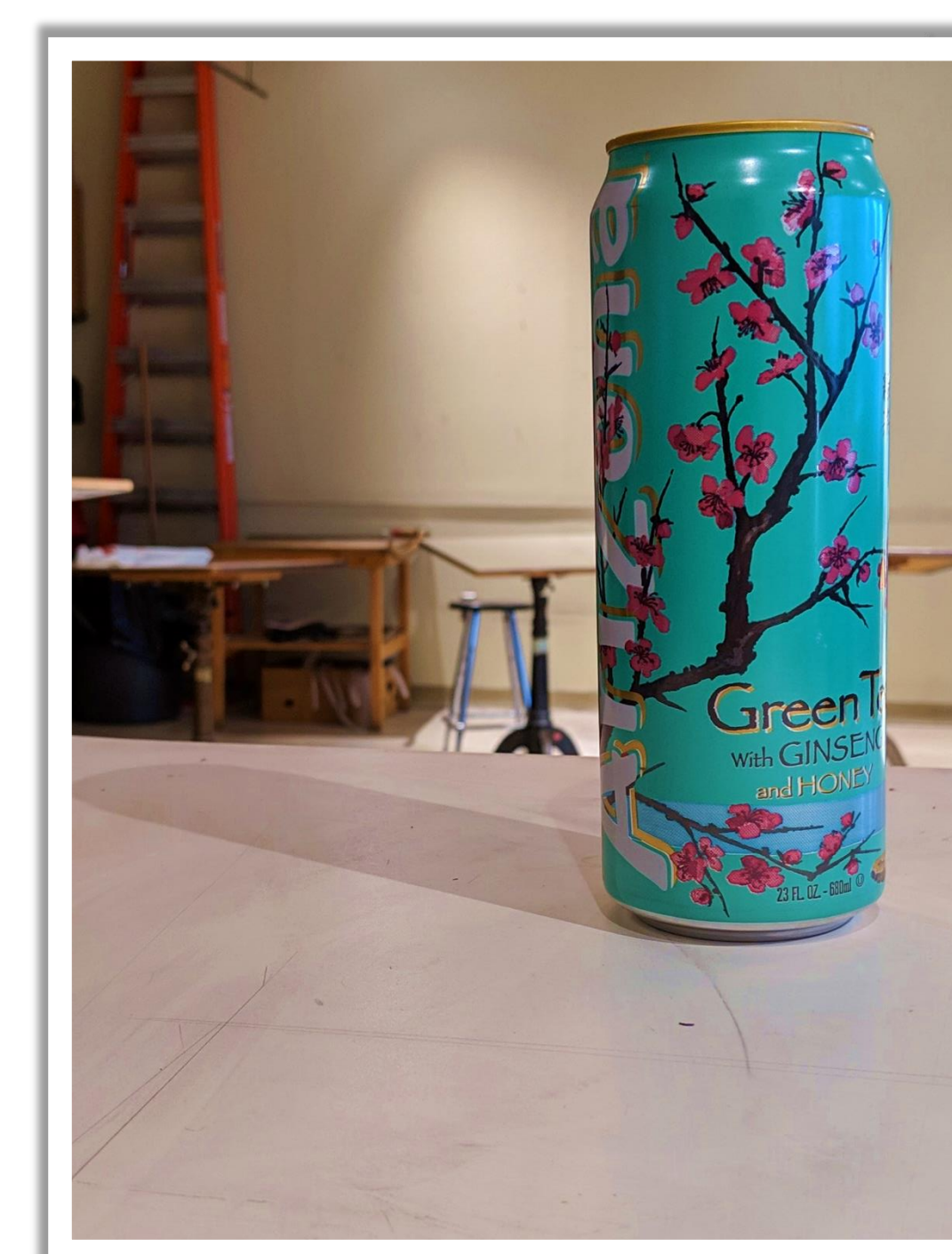
User Feedback

Mounted on the Nerf Blaster is a screen that displays the current camera feed to the user. A reticule representing the center of where the blaster should shoot is displayed on-screen in light blue while the center of the target is highlighted on-screen with a bright red circle. In addition to this, a dark blue rectangle is displayed to represent the shape and location of the target. Displayed from the center of the target to the center of the reticule is a green arrow indicating the error in aim. Figures 3 and 4 represent the tracking capabilities of the algorithm with a static target and a dynamic background, while figures 5 and 6 represent the object detection and identification properties of the algorithm with other targets.

Conclusion and Analysis

Overall, the CNN based algorithm performed poorly in comparison to the purely algorithmic KCF solution. We found that the CNN had many issues with identifying the cans primarily due to the low resolution of the target. The CNN overall did much better once the camera resolution was increased, but due to the low processing power available and the low native resolution of the screen used, the maximum resolution we could feasibly use was 640x480. In contrast, the algorithmic solution tracked its targets exceptionally well, but struggled to maintain focus on the target once the lighting conditions changed. We expect both algorithms to increase substantially in effectiveness once a different camera is implemented with either a higher native resolution or an optical zoom feature, in addition to a processor with more resources available.

Razzleberry Peace Tea (Fig. 3)   Razzleberry Peace Tea (Fig. 4)

Monster Juice Mango Loco (Fig. 5)   Arizona Green Tea (Fig. 6)

Objects of Interest (Fig. 1)

Convolutional Neural Network Framework (Fig. 2)

INPUT   CONVOLUTION + RELU   POOLING   CONVOLUTION + RELU   POOLING   ...   FLATTEN   FULLY CONNECTED   SOFTMAX

FEATURE LEARNING   CLASSIFICATION