Neural Network Real-Time Parking Monitor

Miranda Huddle
Electrical & Computer Engineering and Computer Science Department
Ohio Northern University
Ada, Ohio 45810
Email: m-huddle@onu.edu

Wayne Campbell
Electrical & Computer Engineering and Computer Science Department
Ohio Northern University
Ada, Ohio 45810
Email: w-campbell.2@onu.edu

Hayden Shenefield
Electrical & Computer Engineering and Computer Science Department
Ohio Northern University
Ada, Ohio 45810
Email: h-shenefield@onu.edu

Bryce Gray
Electrical & Computer Engineering and Computer Science Department
Ohio Northern University
Ada, Ohio 45810
Email: b-gray.1@onu.edu

Copyright © 2018, American Society for Engineering Education

Abstract

Neural networks can be used to monitor the spaces and the traffic in parking lots. Due to the high costs and problems handling unique situations such as motorcycles or a single car taking up two spaces, sensors are not the best solution for detecting and monitoring vehicles in parking lots. Neural networks can eliminate these issues by acting as a human brain and seeing things that sensors are not able to detect. Cameras will be mounted in the parking lot to capture live video feed to send across the network to a computing platform where the feed will be analyzed through deep learning and object detection models. This information will be sent to a client application where security and safety companies can monitor the parking lot remotely and be notified of any unusual activity such as vehicles parking in timed spaces for longer than the time limit or a vehicle taking up multiple spaces. The success of the project will be measured by the accuracy of parking space detection, the accuracy of vehicle detection, and the accuracy of the timing mechanism for each vehicle in the lot. It will also be measured by the satisfaction of Ohio Northern University Public Safety based on the ability to meet the needs they have indicated.

Problem Definition

The objective of the project is to design and develop an application that uses camera feedback in real time to record information about where empty parking spaces are present in parking lots as well as how long cars have been in their parking spots using deep learning and object detection. In order to take this research project straight on, we sought after a simulated client. Ohio Northern University Public Safety was approached to provide their perspective on what they would want out of a project like this. Security and safety companies are the potential customer base of this project especially those dealing with parking and parking lots.

The value of this project is twofold. For high volume parking lots or areas that experience high amounts of traffic, the application will allow users to see which spots are available and how long some of the cars have been parked, allowing them to better predict when parking spots will become available. This is especially helpful for people needing handicapped parking as they will be able to determine whether or not the spots most accessible to them are available or not. The second value of the project is for security companies or parking lot attendants. For parking spaces with time constraints, such as “15 Minute Parking Only,” the application will indicate which cars are not abiding by this rule, allowing personnel to respond accordingly.

The main factors that will be evaluated for competing solutions will be the accuracy of detecting the occupancy of a space, the accuracy of tracking how long vehicles have been parked, and the accuracy of vehicle detection. The accuracy of detecting the occupancy of a space will be quantified by the percentage of correctly detected open parking spaces. The higher

Copyright © 2018, American Society for Engineering Education

the percentage of correctly identified parking spaces, the better the solution will be evaluated to be. The time recorded for the time spent in a parking space will be counted to the second and will be evaluated by how closely the camera and software are to the actual time that the vehicle has been parked. The closer the application is to the actual time, the higher the application will be evaluated. The accuracy of detecting vehicles will be evaluated based on the percentage of correctly detected vehicles. The higher the number of correctly identified vehicles, the better the application will be evaluated to be.

Background

To our knowledge there are two types of existing products. One product uses some form of sensor, whether it is electromagnetic or infrared sensing, and tickets to detect when parking is at full capacity. The other is a solution that uses object detection through deep learning along with cameras to do the same; count existing vehicles to detect when a parking area is at full capacity. Neither of these achieve the goal we are trying to reach by creating an application that can show where available parking spaces exist based off of the information assessed through deep learning techniques.

Other advantages that our solution provides over current counting methods that use sensors are resistance to weather and more adaptability. Sensors are affected by weather, but our solution using outdoor, weatherproof cameras will be less affected by the weather conditions. A sensor would not know how to handle a situation where a vehicle is parked in the middle of two spaces because a sensor would just detect a vehicle in both spaces. Our solution using deep learning will essentially work as the human brain and be able to detect a vehicle taking up two spaces, sending this information to authority who can then take appropriate measures. Spaces occupied by motorcycles and other smaller vehicles could be shown as available since a sensor would most likely miss these vehicles. Once again, our solution acting as the human brain will be able to detect these smaller vehicles and accurately represent the spaces that they are occupying.

A research project by Zhang el al describes how to run a parking lot system using sensors[3]. These sensors used magnets to track cars in specific slots. Throughout the paper, they cite data explaining that they have almost reached 98% accuracy with using these magnetic nodes on the parking spots that would communicate with the main sensor node/router. There are other works in the field that have used other solutions including ultrasonic and shortest path solutions based on timing. In the conclusion, they say that the detection accuracy is better than 98%, but the only way to improve that rating is getting rid of the interference between the cars in adjacent parking spots. Our solution would get rid of that problem completely because we are working with standalone images instead of sensors. Furthermore, with deep learning our product will increase its accuracy overtime, not caring about what a vehicle would be composed of. Just

Copyright © 2018, American Society for Engineering Education

as in this research, our product will further increase the amount and quality of the vehicle we can sense without error. Furthermore, this provides us with an easier way to show real time video feed to our customers, rather than just showing them the parking spot number on a screen with data showing that there is a car parked in that current spot.

Other current measures to defend against vehicles overstaying their time limit in a parking space include law enforcement. Law enforcement cannot watch over every spot at every second, but a solution providing real-time video will be able to provide the information to law enforcement without them physically being there.

While there are advantages to our neural network application over current technologies, there are still potential failures that pose as a threat to the neural network application solution. For example, snow and other poor weather conditions could inhibit the camera from seeing the parking lines as well as vehicles that are parked in the spaces. To overcome this, we have created a user interface that allows the user to individually select the parking spaces that they wish to monitor, so the weather will not play a role in the application being able to see the parking spaces since the spaces are defined by the user. Also, the TensorFlow model was fed images of vehicles covered in snow, so the model knows how to classify those vehicles as well. The images will be taken in small time increments, so if one image is poor due to any weather conditions, the next image will be taken soon after for the model to assess and analyze. The application has been set up to work on a local area network to reduce the amount of possible IT failures and infrastructure restraints on the solution. The user interface design has been created based off of feedback from our security client, Ohio Northern University Public Safety. Feedback from a security company was needed to create a user interface that would sufficiently provide security companies with the information that they are looking for in a pleasing manner.

The only standard that seems to play a role in our application is privacy rights because there will be live video footage captured[2]. Our setup can be compared to surveillance cameras. Since the cameras will be in parking lots or parking areas, the surveillance cameras do not break any laws in any states because these are not places where people expect to have complete privacy. The people still have their rights to privacy until they are publicly notified that the video cameras are being used.

Solution

The solution to be implemented is completed by mounting multiple cameras around a parking lot to allow real-time video footage to be captured from multiple angles and eliminate any blockages that could happen if only one camera was used. The real-time feedback will be able to give accurate information down to the second rather than in intervals like single-frame timed images. Information can be gathered to identify vehicles, occupancy of parking spots, as well as the time each vehicle has been occupying each parking spot. The live video feed will be sent across the network to be processed by the deep learning software that has been trained to do object detection.

This potential solution will meet all of the aforementioned evaluation metrics. To create the most efficient system possible, we are using the Google MobileNet convolutional network. This network is re-trained on test images that we provide to TensorFlow. This network is also important because of potential clients requiring some sort of mobile application that can be used offline. Other networks that work on mobile require us to work with cloud solutions. This also allows our system to work offline, which is important in some security situations. MobileNet's are designed to instead of working with a conventional 3x3 convolution layer with batch norm and ReLu, it instead uses a 3x3 depthwise network and a 1x1 pointwise network.

Due to the use of the MobileNet, further testing of the model was needed to make sure the smaller size and speed of the network did not hurt the performance of the application. To compare this mobile fast network with a traditional network, we tested this against the inception_v3 model. This model uses the 3x3 sliding panel solution for the network creation. After finishing the re-training, the tradeoff looked to be in favor of the mobile network. We did not need to care about the training time, but comparing them hitting a threshold of 95%, the MobileNet hit that threshold 6 times faster (around 4 minutes of training). In that same about of time, we only lost ~.04 accuracy compared to the more robust model. Overall, since we used an even amount of images in the test data for our two classes of parking spaces and cars, the difference in accuracy does not matter as much as if we had a network with bigger classes. Then, the difference between the classifications would matter to our end user of our application. To improve the accuracy of our network, we used the 1.0 width multiplier and the 244 image resolution. Because we re-trained the model on a fairly powerful GPU, we used a high number of training steps.

The solution uses mounting materials to fixate multiple outdoor cameras on poles or other stationary objects that are already implemented in a parking lot to cover the field of view of the parking lot in its entirety. The cameras will be capturing live stream video footage that will be sent across the network through cable, a switch, and an outdoor router in order to send the data to the computing platform. The computing platform will be a computer constructed with an Intel Core i5 CPU and a 1080 8 GB GPU. The computing platform will utilize the models that are created with the code using the Tensorflow object detection libraries in order to train the algorithms to detect parking spaces and vehicles. The computing platform will store the occupancy of the spaces and the time that each vehicle occupies the space. This information will then be sent to a user application where the user can find the available parking spaces as well as

Copyright © 2018, American Society for Engineering Education

see how much time they have left to occupy their selected parking space. The information will also be sent to a client application to help security companies monitor if there are any parking violations occurring in the parking lot without them being present.

The two biggest things that impact our performance are the software tools and the hardware that we have selected, particularly the GPU which is responsible for the deep learning and object detection. After researching which GPU will perform best by looking into research with extensive testing, the best selection for the project's budget was selected and a proper goal was set based on the capabilities of the technology[1]. The other limiting factor for the performance prediction was the software. Because object recognition is based on confidence levels and not absolutes, the software cannot be 100 percent certain. Thus, the predictions are made in accordance with the degree of uncertainty as well as a degree of user-error due to the high learning curve of deep learning and neural networks.

Prototype

Our prototype will reflect our major goals in our project but on a much smaller scale. This will be to show to our client to prove our design has met our certain objectives that we have outlined in our initial prototype. Our initial prototype shows that the final application will simply detect cars from a live video feed and display on a screen a color around a space indicating if it is available or occupied as well as indicate the amount of time that has passed since the vehicle has been parked in that space. While this particular prototype will be simple, it will be easily scaled to coincide with our overall goals in the project. The first materialistic prototype will be produced from wood, paint, tape, miniature toy vehicles, and a small indoor IP camera. The wood, paint, and tape will be used to construct a small scale parking lot, and the toy vehicles will be the small scale vehicles in our parking lot. The small indoor IP camera will be positioned to collect data from the small scale parking lot to send to a computing platform for processing. Since we are working from our current knowledge of machine learning, it is important that we understand the basics on a simple, small-scale model before we move toward our greater objectives.

Implementation

Our system will be comprised of outdoor IP/Serial cameras connected to a network by an enclosed switch. The live video feed from these cameras will be sent and analyzed by deep learning algorithms for object detection on our base computer which will return a near real-time output with minimal delay to two applications that will give meaningful output to a client and users. The client application will show if there are any parking violations in the parking lot,

Copyright © 2018, American Society for Engineering Education

while the user application will show where the available parking spaces are located along with the amount of time the user has left to occupy that space.

The application level will feature several levels of applications. The first application will allow the user to select areas in which they want to test for car detection. This will allow our TensorFlow application to be accurate and allows easier car tracking for our main application. The main application will be more of a UI approach to our solution, allowing our research to be shown in a way that is understandable to an end user.

To look at the system analysis and the interactions of all of the parts, the IP cameras will have to be able to send information across the network through an enclosed switch, and this data will have to be sent to the computing platform. Once the data is received at the computing platform, the deep learning and object detection algorithms will need to be able to process the data and send the outputs to the applications in order to display the needed information for both the client application as well as the user application.

The implementation of the application is created using python. Python is used because of its usability with the OpenCV library and TensorFlow. OpenCV is used to handle a webcam input, display the camera feed, and print the output of our algorithms over this display. Because of the library that Tensorflow provides us, we can use these two to create our UI. The main UI of our program as shown in Figure 1 will be defined using the Tkinter library which will allow our program to have accessible ways to configure and view the results.

The user interface went through several design phases as we gave Ohio Northern University Public Safety different models and mockups of our proposed solution. After discussion and feedback, we decided on two main applications. One application as shown in Figure 2 is where the client can select and save the individual desired parking spaces to monitor, and another as shown in Figure 3 is where the user can see the output of the algorithm in real time which will consist of timers on each space along with a colored box indicating if the space is vacant, occupied, or past the time limit. Because of this being used in a security fashion, the client requested that there would be a type of log. This log would be used to determine and prove the legitimacy of our program if a driver would ask the security department for proof of how long the vehicle was occupying a spot.

Copyright © 2018, American Society for Engineering Education

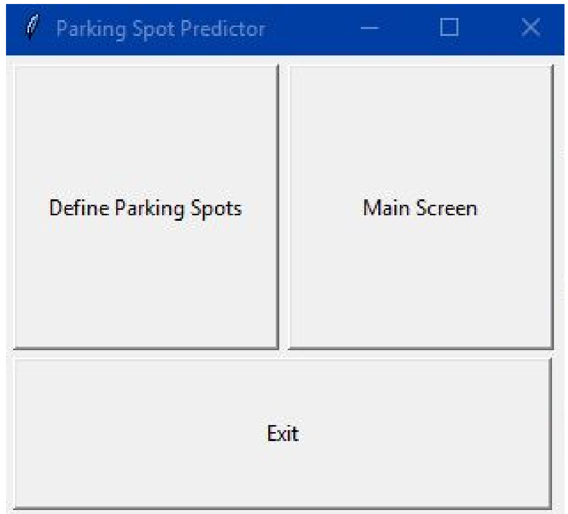

Figure 1: Main User Interface

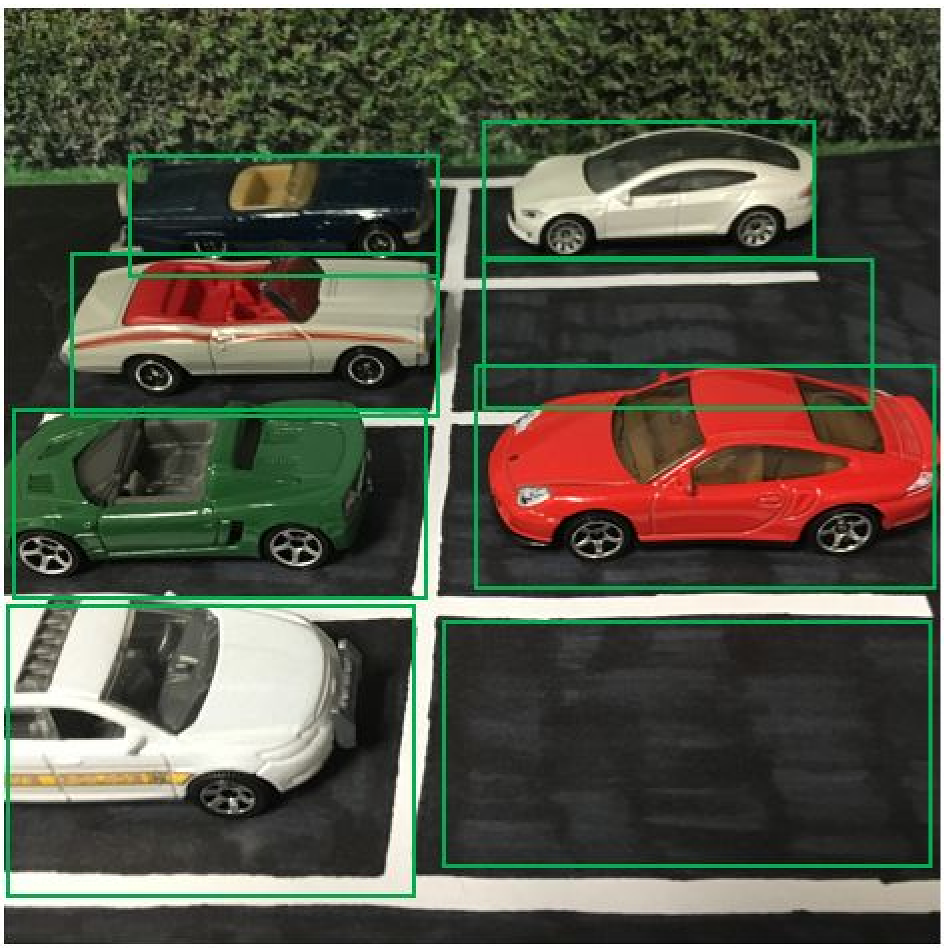

Figure 2: Defining the Individual Parking Spaces

Copyright © 2018, American Society for Engineering Education

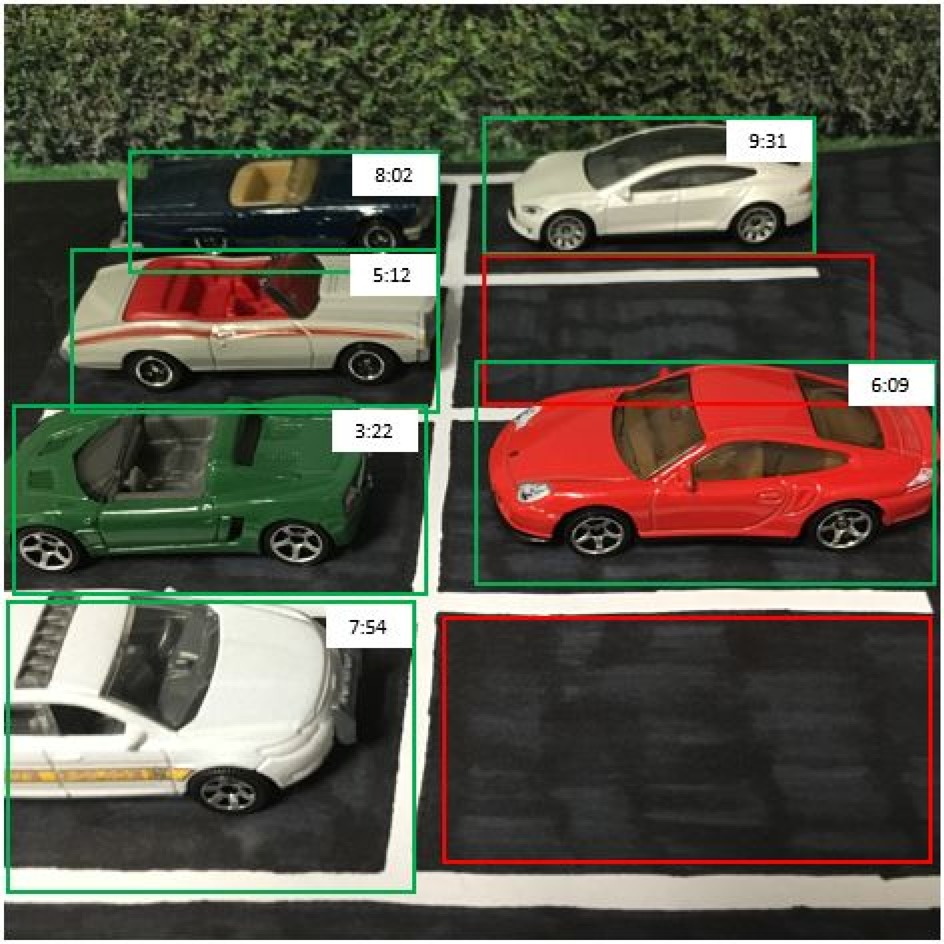

Figure 3: Output of the Algorithm with Timers and Colored Boxes

Testing

Testing our neural network involved making a sample program that allowed us to see the results of the model from a live webcam. Testing the TensorFlow model as a proof of concept before doing implementation was needed to ensure that the parking spaces and vehicles could be correctly detected on a small scale first. The first test image is shown below in Figure 4 where we received that it was 0.00% car and 100.00% space.

Copyright © 2018, American Society for Engineering Education

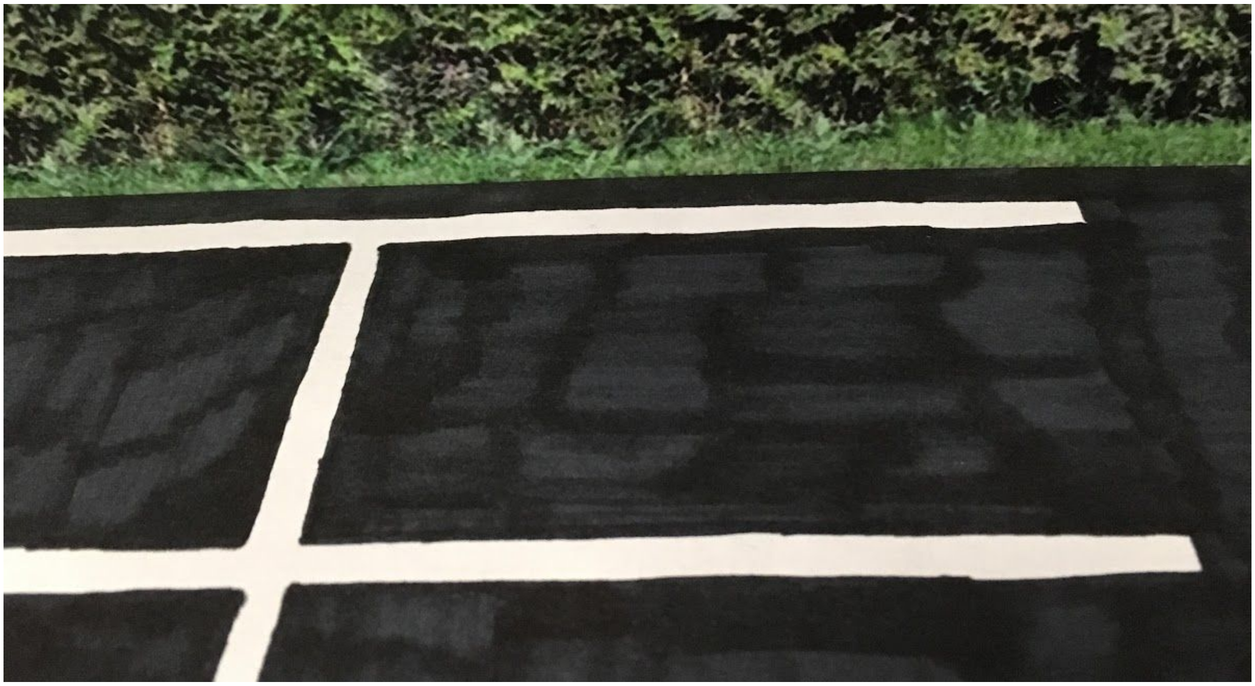

Figure 4: Test Image 1

The second test image is shown below in Figure 5 where we received that it was 98.13% car and 01.87% space.

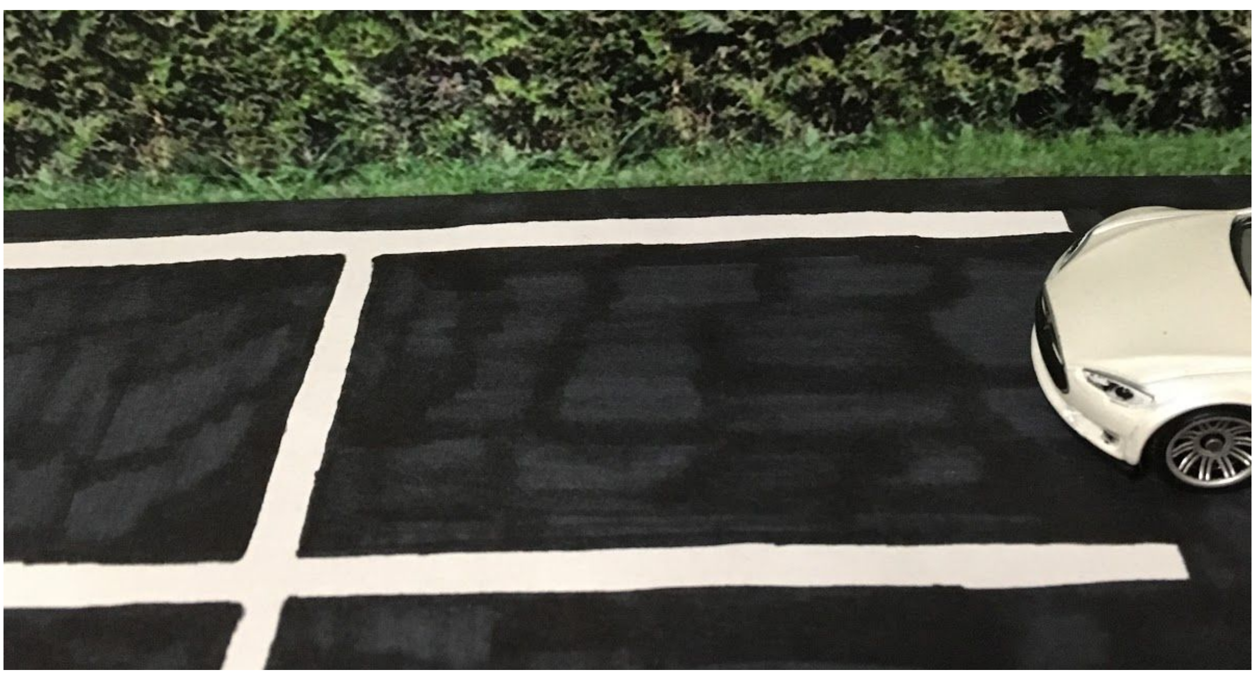

Figure 5: Test Image 2

The third test image is shown below in Figure 6 where we received that it was 100.00% car and 0.00% space.

Copyright © 2018, American Society for Engineering Education

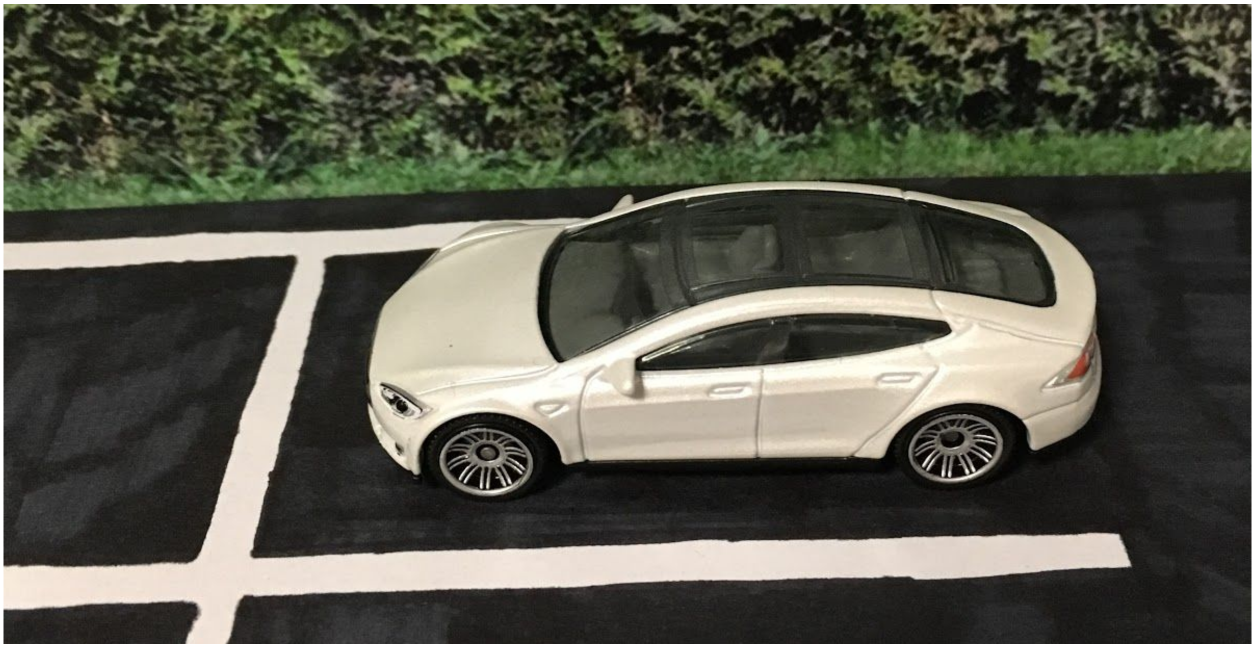

Figure 6: Test Image 3

From this output and output from many other similar tests, we were able to very that the TensorFlow model is accurate and precise enough for implementation and to meet the requirements of our solution that we had initially defined.

Conclusion

Our solution combines the two needs of an optimal parking lot security solution. The first is to research object detection through deep learning. Our solution uses cameras to capture live video feed to detect objects through the deep learning algorithms, so the first need is met. The second is to provide a real-time parking monitor that tracks the occupancy of parking spaces and the time that vehicles have been in the parking spaces. Our cameras take in live video feed in order to meet the real-time need, and the footage is processed on the computing platform to identify parking spaces, vehicles, the occupancy of the spaces, and the amount of time the vehicles have been occupying the spaces. This solution meets all of the needs of the project.

The user application portion of our project creates value because finding the available parking spaces is difficult in crowded areas with lots of cars as shown by the success of multiple parking applications that are currently in existence. With technology on the rise, having an app for parking simplifies the process of parking in crowded areas as well as makes it more convenient. The client application portion of our project creates value because it allows law enforcement to detect parking violations without being at the site of the violation at the exact time. Our project is also very valuable because we are testing and pushing the boundaries of the

Copyright © 2018, American Society for Engineering Education

capabilities of these deep learning techniques. We want to see how well this technology can do and in what areas this technology would be well suited. This is valuable information as deep learning is on the up and coming.

Bibliography

1. Dettmers, Tim (2017), Which GPU(s) to Get for Deep Learning. Retrieved from http://timdettmers.com/2017/04/09/which-gpu-for-deep-learning/

2. Oak Tree Press. (2015). *Video Surveillance Laws.* Retrieved from https://www.videouniversity.com/articles/video-surveillance-laws/

3. Zusheng Zhang, Xiaoyun Li, Huaqiang Yuan, Fengqi Yu (2013), A Street Parking System Using Wireless Sensor Networks. Retrieved from http://journals.sagepub.com/doi/full/10.1155/2013/107975

Copyright © 2018, American Society for Engineering Education