Initial Project Proposal

Year: 2023 Semester: Spring

Project Name: Parking Tracking System

Creation Date: ­November 1, 2022 Last Modified: , 2022

Team Members (#1 is Team Leader):

Member 1: Heath Aaron Lovell Email: hlovell@purdue.edu

Member 2: Jonathan Doorn Email: jdoorn@purdue.edu

Member 3: Daniel English Email: englishd@purdue.edu

Member 4: Frederick Kepler Email: fkepler@purdue.edu

1.0 Description of Problem:

One problem we are trying to solve is to better inform the parker of the parking availability of the lot that they are trying to park in before they enter. Anyone that parks in a parking lot that utilizes our solution will benefit from this. This problem impacts these parkers every day by forcing them to spend extra and unnecessary time throughout their day trying to find a parking spot. Solving this problem would not only reduce the time spent trying to find a parking spot, but it would also reduce the number of students and faculty showing up late to class.

2.0 Proposed Solution:

Our solution will track when cars enter and leave the parking garage. As objects appear in the camera feed, a neural network will process the image data and identify the object in the picture. If the object is identified to be a car, then the software will determine whether the car is leaving or entering the garage. This information will be sent to a central controller, which will calculate how many available spaces are currently in the garage at any given moment. Additionally, the use of cameras and microcontrollers will allow our solution to run without causing any inconvenience to the customer and give our solution flexibility to be adapted to other parking structures. Lastly, more cameras and individual processing units can be added to the system to accommodate garages with multiple entrances.

3.0 ECE477 Course Requirements Satisfaction

3.1 Expected Microcontroller Responsibilities

One microcontroller, connected to the camera, will analyze the camera feed provided from the entrance of the garage to perform a two-step object identification and analysis. First, the microcontroller software will identify if the object is a car and then determine if it is leaving or entering. Once a car leaving or entering the garage has been identified, the microcontroller will communicate the results to the central microcontroller. The central microcontroller will update the number of available parking spots accordingly and send out the corresponding display data to the customers.

3.2 Expected Printed Circuit Responsibilities

Our printed circuit board will include two microcontrollers, on-board memory, power subsystem, USB interfaces, and a 7-segment display.

4.0 Market Analysis:

Even in the largest US cities, more than half of residents own cars [2]. In large cities, residents may pay up to $300 each month for a parking space [3]. From personal experience, I have paid $20/day to park in Chicago. In a city like Chicago, which supports more than three million jobs [4], the revenues from parking in the city itself could be well over $100M each month. Purdue employs roughly 17,000 staff, of which 10,000 work full-time [5]. There are likely more than 10,000 cars which must enter and exit Purdue parking facilities each day, which would be a good population to test a product on. A good parking product could be sold to players in the city parking industry to take a share of the large monthly revenue.

5.0 Competitive Analysis:

5.1 Preliminary Patent Analysis:

Thorough research was performed to protect our project against patent infringement claims. A collection of patents that contain relevant intellectual property in the field are described in this section.

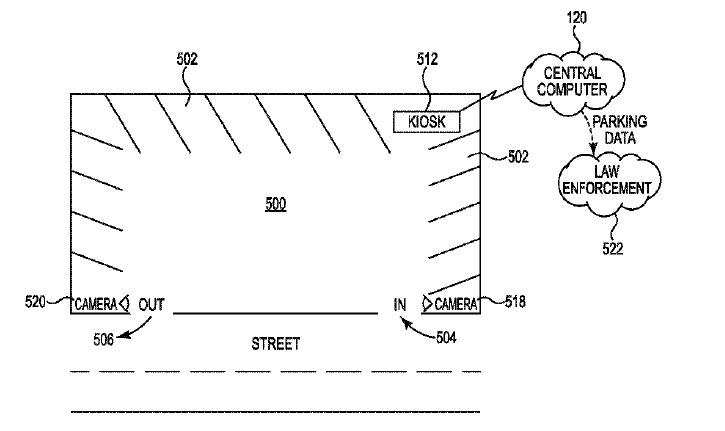
5.1.1 US Patent Application US10121172 B2:

Patent Title: “Parking Lot Monitoring System”

Patent Holder: Municipal Parking Services INC.

Patent Filing Date: March 15th, 2014

This patent [13], assigned to Municipal Parking Services INC., is similar to our proposed solution in that they utilize a system of cameras to track cars entering and exiting a parking lot, as seen in Figure 5.1.1.1. The images of the car and the time the car entered or left the lot are recorded and sent to a remote network. This remote network takes the image data and identifies the vehicle that entered or exited the lot. The system also determines if a car should be issued a parking violation based on the time of entry, the time of exit, and the parking time paid for that vehicle at a kiosk machine located in the lot.



**Figure 5.1.1.1**

The advantage of this system, compared to our solution, include the ability to charge vehicles for parking violations without any intervention from a worker, thus reducing man hours and increasing revenue. The disadvantages of this system, compared to our solution, include the extra computing power and complexity necessary for vehicle identification (as opposed to only detection in our proposed solution), the negative implications of an error or mistake in the software, and the lack of information provided to the drivers about the number of spaces available in a lot or garage without individual meters.

5.1.2 US Patent Application US10043307 B2:   
Patent Title: “Monitoring parking rule violations”

**Patent Holder:** Ubicquia IQ LLC  
**Patent Filing Date:** August 7th, 2018

This patent [17], assigned to Ubicquia IQ LLC discusses a system to aggregate data of where and for how long cars have parked to show law enforcement agencies vehicles which violate parking zone rules. This system relies on cameras located in parking zones to collect data on cars, then transmits the collected data to a remote location to be analyzed for potential violations. This patent describes an algorithm in detail to determine if a parking space is filled.

5.1.3 US Patent Application US9852629 B2:

Patent Title: “Methods and systems for tracking a vehicle’s position using a plurality of light sensors”

Patent Holder: Signify Holding BV

Patent Filing Date: July 9th, 2013

This patent [9], assigned to Signify Holding, discusses a system that has the ability to track a vehicle going into a parking area, guide it to an unoccupied spot, and know whenever that vehicle has left the spot. The proposed method of doing this is to utilize a plurality of light sensors where they have the potential to transmit locally (to neighboring light sensors) the vehicle identifier, or, where the vehicle currently is located. Through this, the light sensors have the vehicle identifier travel with the vehicle until it is in its final location.

5.2 Commercial Product Analysis:  
After doing research into the products that currently exist to do a similar functionality to what ours will accomplish, there are some shortcomings that our project will be able to handle. For example, other product alternatives are a high-cost, bulky alternative to our system. Along with this, some of them only have the ability to show the number of spots available, rather than the specific spots that are available in the lot.

5.2.1 intuVision VA Parking by intuVision:

The intuVision VA Parking software custom package by intuVision is an adaptable parking counting system that utilizes video analytics to perform a variety of parking computations [14]. With their unique state of the art algorithms, the software can effectively determine available parking spots in an open lot or count vehicles entering or exiting a garage. The live analytical results and current count number can be accessed through their web-based API (Application Programming Interface), as seen in Figure 5.2.1.1.



Figure 5.2.1.1

In addition to the software, intuVision also sells a variety of hardware packages that contain the necessary CPUs and GPUs to run their software based on the number of cameras in the system. Advantages of this product, compared to our proposed solution, include modularity based on use case, flexibility with implementation, and highly accurate video analytical computations. The disadvantages of this product, compared to our proposed solution, include high-cost, bulky hardware units that cannot withstand cold temperatures below 40°F, and the need to separately buy and set up compatible cameras.

5.2.2 OpenSpace by Parking Logix

The OpenSpace system by Parking Logix [10] is a parking lot counter system that functions by placing sensors at the entrance(s) and exit(s) of the parking lot, an example of a sensor can be seen in Figure 5.2.2.1. It is then displayed over the cloud to show potential customers for the lot how many spaces are available. Parking space availability can either be hosted on a message sign outside of the lot, on the client’s website, or on any integrated applications. Along with this, it also has the ability to display data for another parking lot nearby, directing cars to a lot that has more space if the current one is full. The pros of this product compared to ours is that it is very thoroughly integrated with systems already available. One con, however, that our idea will do better with, is the ability to show which individual spots are available, rather than only how many spots are available in a certain lot.



Figure 5.2.2.1

5.2.3 Commercial Product #3: Parkeagle StreetEagle Sensors

The StreetEagle Sensors are a wireless system that monitors the number of vehicles passing through a monitored street, the number of vehicles that are parked on that street, and the traffic on that street. This is done using four sensors placed on the four corners of the boundaries that you want to enclose and monitor. These sensors are shown in figure 5.2.3.1 and 5.2.3.2. These sensors collect data on the vehicles that are entering and exiting the barriers and share it with the StreetEagle Cloud [8]. The data is then analyzed and transformed into more useful information such as parking occupancy, traffic flows, and vehicle types by the StreetEagle Cloud. This information is presented in the Parkeagle management system. This commercial product also pairs with the StreetEagle App. This app is intended to give insights on where the user can park their car.



Figure 5.2.3.1



Figure 5.2.3.2

5.3 Open Source Project Analysis:

After doing a deep dive into the field, similar projects conducted for non-commercial purposes were discovered that could prove to be advantageous to our product. Some useful examples of these projects are described in the subsections below.

5.3.1 Parking Slot Detection:

Parking Slot Detection is an application made in Python that uses a deep learning approach [12]. It first divides the problem into two parts – detecting where the parking spots are located, and then detecting if the spot is occupied. This approach uses camera footage to detect where the spots are, prior to cars being parked there. The first part of the problem is solved using MaskRCNN & YOLO, while the second part uses RESNET & VGG classifiers to detect if those spots are occupied. A pro of this system is that it has the ability to automatically detect where spots are located, before determining if they are full or not. However, one major con of this system is that it does not have numbered spots or the ability to host the available spots online, where it would be viewable by potential users of the lot.

5.3.2 PKSpace Parking Space Occupancy Detection:

PKSpace is an open-source solution for parking space occupancy detection that utilizes multiple cameras and processing units that are connected locally to a server [15]. All the computation is done locally on each processing unit, and its data can be accessed from an outsider by connecting to the server using SSH, as seen by Figure 5.3.2.1.

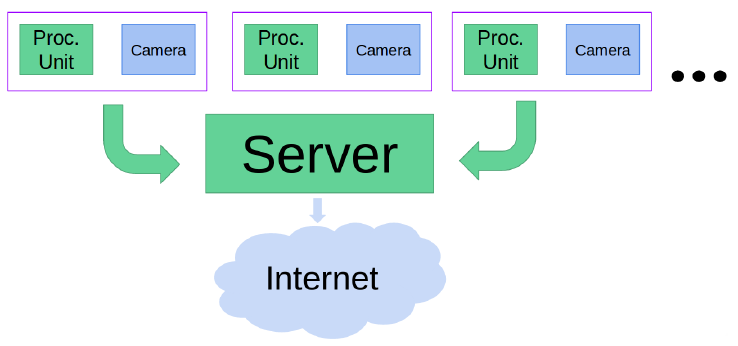


Figure 5.3.2.1

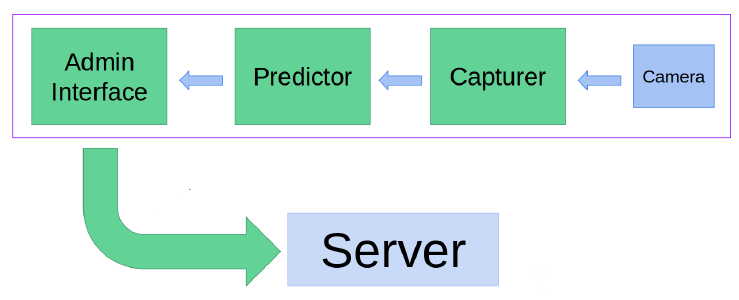


Figure 5.3.2.2

Utilizing Python and JavaScript, the processing is broken into three stages: a Capturer, a Predictor, and an Admin Interface, as seen in Figure 5.3.2.2. The Capturer takes the images from the camera and performs some simple operations to prepare them for further analysis. The Predictor determines whether a spot is occupied for each spot in the parking lot based on the trained model. The vacant or occupied results are sent to the Admin Interface, which summarizes the data to be displayed on a visual interface. Also included in this interface are settings for the previous two stages along with tools for the user to customize the model to their own, specific situation by providing personalized datasets to use for training. An example of the live feed is shown in Figure 5.3.2.3.



Figure 5.3.2.3

5.3.3 Smart Park

Smart Park is an application made in Python that maintains empty parking spots using a real-time vehicle detection algorithm called YOLOv3. This application uses video processing to analyze a parking lot and calculate how many empty spots are currently available within the image it is analyzing. However, this software comes with some limitations [6]. For instance, the resolution of the video along with the different angles at which the parking spots and cars are oriented in can affect the accuracy of the application. This project is currently available on GitHub, and an example of an implementation of it is shown in figure 5.3.3.1. A pro of this system is that it detects the number of parking spots available in most cases. This would be valuable to our system. However, a con to this system is the inaccuracy that comes with lower resolution and different angles of parking spots. We might have to adapt this car detection from video for our project, however we would need to correct the inaccuracies.

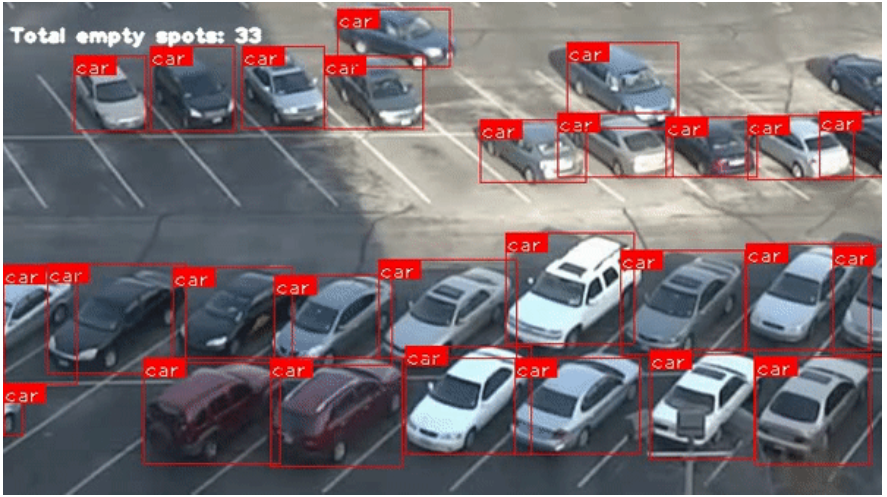


Figure 5.3.3.1

6.0 Sources Cited:

[1] ReportLinker, “Parking lots and garages global market report 2022,” *GlobeNewswire News Room*, 16-Feb-2022. [Online]. Available: https://www.globenewswire.com/news-release/2022/02/16/2385932/0/en/Parking-Lots-and-Garages-Global-Market-Report-2022.html.

[2] M. Maciag, “Vehicle ownership in U.S. cities data and map,” *Governing*, 14-Jun-2021. [Online]. Available: https://www.governing.com/archive/car-ownership-numbers-of-vehicles-by-city-map.html.

[3] “The state of parking in Chicago - no spaces anywhere,” *CooperatorNews Chicagoland, The Condo, HOA & Co-op Monthly*. [Online]. Available: https://chicago.cooperatornews.com/article/no-spaces-anywhere.

[4] “Chicago area employment - August 2022 : Midwest Information Office,” *U.S. Bureau of Labor Statistics*, 30-Sep-2022. [Online]. Available: https://www.bls.gov/regions/midwest/news-release/areaemployment\_chicago.htm.

[5] “Faculty/Staff Headcount,” *Faculty/Staff Headcount - Institutional Data Analytics + Assessment - Purdue University*. [Online]. Available: https://www.purdue.edu/idata/Products/Data/SelfServiceReporting/sfcounts.php.

[6]ankit1khare, “Ankit1khare/smart-park-with-yolo-V3: Maintaining empty parking spot count using Yolo real-time vehicle detection. code readily runnable in Google Colab.,” *GitHub*. [Online]. Available: <https://github.com/ankit1khare/Smart-Park-with-YOLO-V3>.

[7] K. Ancion, “How does Streeteagle differentiate from other smart parking products?,” *Parkeagle*, 27-May-2019. [Online]. Available: <https://www.parkeagle.com/2019/05/02/streeteagle-product-differentiation/>.

[8] “Streeteagle product,” *Parkeagle*, 17-Mar-2020. [Online]. Available: <https://parkeagle.com/streeteagle-product/>.

[9] Dirk Engelen and Bernt Meerbeek, “Methods and Systems for Tracking a Vehicle’s Position Using a Plurality of Light Sensors,” U.S. Patent 9 852 629 B2, Jul. 9, 2013. Available: <https://patents.google.com/patent/US9852629B2/en>.

[10] “OpenSpace system,” *Parking Logix - Intuitive Parking Counting*, 05-Jan-2021. [Online]. Available: https://parkinglogix.com/openspace-parking-guidance/.

[11] V. Meel, “Yolov3: Real-time object detection algorithm (guide),” *viso.ai*, 22-Aug-2022. [Online]. Available: <https://viso.ai/deep-learning/yolov3-overview/>.

[12] Visualbuffer, “Visualbuffer/Parkingslot: Automated parking occupancy detection,” *GitHub*. [Online]. Available: https://github.com/visualbuffer/parkingslot.

[13] Thomas G. Hudson *et al.,* “Parking Lot Monitoring System,” U.S. Patent 10 121 172 B2, Mar. 15, 2014. Available: <https://patents.google.com/patent/US10121172B2/en>

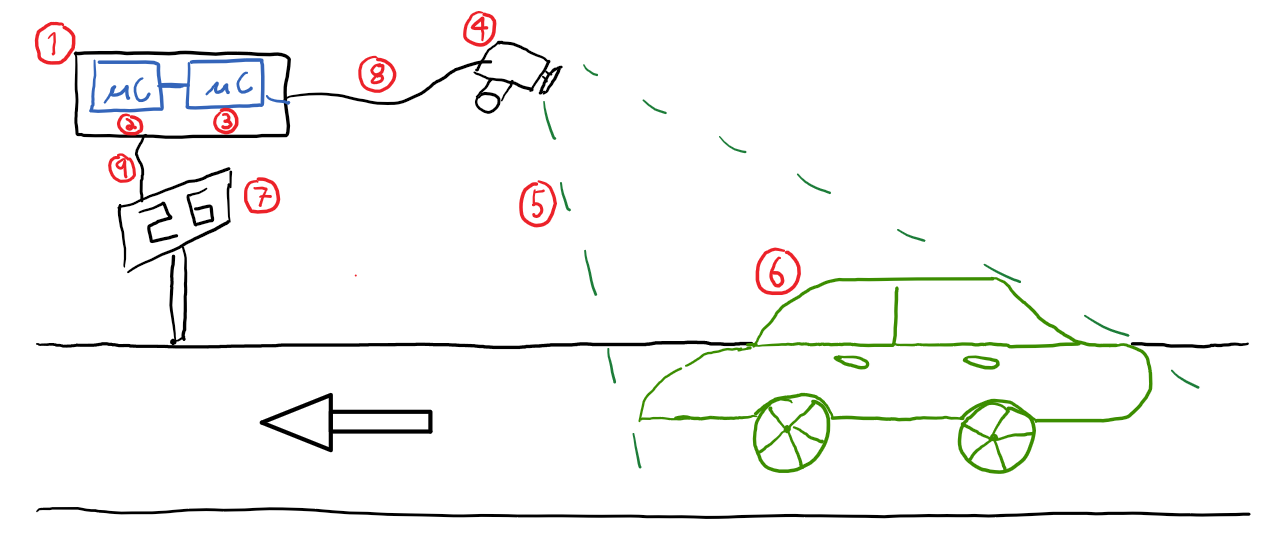
[14] “intuVision® Parking.” intuVision. <https://www.intuvisiontech.com/intuvisionVA_solutions/intuvisionVA_parking> (accessed Nov. 23, 2022).

[15] R. Števanák, A. Matejov, O. Jariabka, M. Šuppa, “PKSpace: An Open-Source Solution for Parking Space Occupancy Detection”, *Proceedings of the 21st Central European Seminar on Computer Graphics*, CESCG 2017. Available: <https://cescg.org/cescg_submission/pkspace/>

[16] “Patent images | innovation is more than words.” [Online]. Available: <https://patentimages.com>

[17] Lokesh Babu Krishnamoorthy *et al.,* “Monitoring Parking Rule Violations,” U.S. Patent 10 043 307 B2, Aug. 7, 2018. Available: <https://patentimages.storage.googleapis.com/d9/8a/a8/bd79db1ca9e0a0/US10043307.pdf>

Appendix 1: Parking Tracking System Concept Overview



1. Electrical Housing
2. Central Microcontroller
3. Image Processing Microcontroller
4. Camera
5. Field of View of Camera
6. Car entering the Garage
7. Sign Displaying Number of Available Spots
8. Connection Cord between Camera and Microcontroller
9. Connection Cord between Microcontroller and Display

