Using Unmanned Aerial Vehicles and Machine Learning to Identify Vacant Parking Spaces

Ty Grisham, Sam McGee, Halcyon Morgan, Tanner Woolbright
Mississippi State University, Mississippi State, MS 39762 USA

This project aims to mitigate the campus parking issues at Mississippi State University which are becoming more substantial by introducing an unmanned aircraft system capable of detecting empty parking spaces. The group has acquired the resources for a drone and the necessary flight-testing equipment. After considering multiple options, the team decided that it was best to implement machine learning algorithms using instance segmentation for vehicle detection. Two test flights have been conducted, capturing images for future mapping and proving that the drone has the ability to fly autonomously. Further work must be done to obtain a fully functioning machine learning model, achieve autonomous flight, and achieve smooth data transfer.[1]

Nomenclature

CMOS = Complementary Metal Oxide Semiconductor
GCP = Ground Control Point
GPS = Global Positioning System
GRI = Geosystems Research Institute
HPCC = High Performance Computing Collaboratory
IS = Instance Segmentation
LiDAR = Light Detection and Ranging
ML = Machine Learning
MSU = Mississippi State University
PCA = Principal Component Analysis
RFRL = Raspet Flight Research Center
UAS = Unmanned Aerial System

I. Introduction

THE enrollment at Mississippi State continues to climb, and so has the number of commuters. Although the university consistently attempts to solve this problem, the increase in drivers has coincided with a sharp decrease in available parking spaces throughout the day. This project aims to eliminate this search by implementing an unmanned aircraft system (UAS) equipped with an optical system capable of detecting empty spaces. This UAS will fly to a predestined point above the parking lot of interest and capture an aerial photo of the area. Once the scan is completed, the data will be processed, and empty parking spaces will be identified. This technology not only has the potential to decrease traffic congestion on campus and lower stress for users, but it could also be applied to various other urban infrastructures.

[1] Halcyon Morgan (Undergraduate) Student Member 1538011, Tanner Woolbright (Undergraduate) Student Member 1595335, Ty Grisham (Undergraduate) Student Member 1604897, Sam McGee (Undergraduate) Student Member 1604887

II. Methodology

In order to capture aerial photographs of a parking lot, a drone had to be acquired. Multiple routes were discussed, but after speaking with our faculty advisor, Jimmy Cook, a new resource was found in the Geosystems Research Institute (GRI). Mr. Cook pointed the team towards a friend of his, named Daniel McCraine who currently works in aerial surveying of crop fields to observe growth patterns and detect irregularities. After meeting with Mr. McCraine and his team, a path forward was chosen. This path included using one of the GRI's drones to capture the necessary images – specifically the DJI Phantom 4 Pro V2 quadcopter. The drone chosen comes equipped with a 20 Megapixel (MP), Complementary Metal Oxide Semiconductor (CMOS) image sensor with mechanical shutter. Because of these specifications, this drone was ideal for the project and will have no issues capturing high quality images with virtually no distortion.

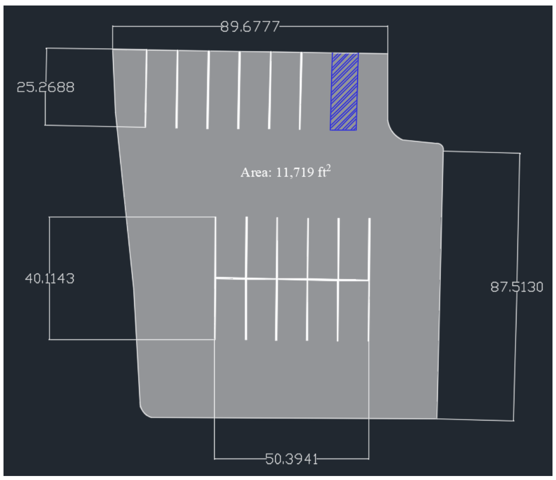

Figure 1: A scaled computer aided design (CAD) model of the Rose Garden's parking lot in feet

Initially, the plan was to implement the flight test above a parking lot on Mississippi State's campus, but due to concerns over campus regulations for UAVs, this proved to be an issue. After reviewing the documentation to fly on campus and discussing options, the group decided it would be more reasonable to test at an off-campus parking lot and the Veterans Memorial Rose Garden lot in Starkville, MS was chosen (Figure 1). Furthermore, the group's decision to team up with Mr. McCraine from the MSU HPCC for test flights not only aids the group in avoiding concerns about regulations but provides the group with all the necessary equipment for a flight test. Were this proof of concept to be implemented on campus, the regulations would certainly be acquired by the operating party and adhered to. Acquiring the images of a parking lot alone solves no problems regarding parking spaces, so that is where the data analysis comes into play. At first, a Light Detection and Ranging (LiDAR) sensor seemed like the most accurate way to find cars using a drone. LiDAR measures range by sending thousands of lasers that reflect off of objects and back to the sensor. The time it takes for the laser to return is measured and distance to various objects is calculated. Because of the high precision of LiDAR, it would be very easy to distinguish between natural terrain and a vehicle, which has surfaces above ground level. After many discussions, LiDAR was ultimately discarded due to lengthy processing times and high cost of operation.

Instead of LiDAR, the team opted to utilize a Computer Vision Machine Learning (ML) algorithm to detect vehicles. There are three categories of ML algorithms for computer vision. Object classification models identify objects and return the class (user-defined label) of that object. Since the goal of this project is to locate empty parking spaces, this option was not considered since it cannot locate objects.

The two ML algorithms that can locate objects are Object Detection and Image Segmentation. Object detection involves identifying and locating specific objects within an image, providing bounding boxes around them, and assigning class labels as seen in Figure 2. On the other hand, image segmentation classifies objects and detects all pixels belonging to that object's class. Unlike object detection, image segmentation assigns a label to every pixel in an image, outlining the precise boundaries of objects. In summary, object detection deals with recognizing and localizing objects, whereas image segmentation focuses on pixel-level classification of object boundaries within an image.

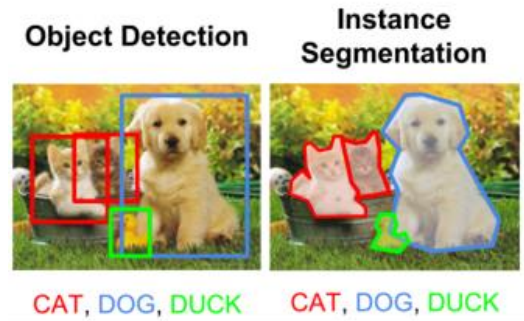

Figure 2: A figure illustrating the differences between Object Detection and Image Segmentation

Figure 3: Object Detection bounding boxes vs. Image Segmentation masks for cars parked in angled spaces.

Initially, the group chose to use object detection since vehicles can be approximated as rectangles. This decision was quickly reversed as it was discovered that the bounding box information would not be sufficient to decide whether a parking space was empty or occupied. Cars parked in a way that is angled within the image captured by the drone would prove that vehicles cannot be approximated as rectangles as seen in Figure 3. In order to obtain accurate vehicle boundary information, it is necessary that an image segmentation model be used. Specifically, Mask R-CNN is the architecture of choice since it is essentially the industry standard.

It is helpful to think of the computer vision program as two parts. The image segmentation part will identify, locate, and provide accurate boundary data for each vehicle and the reference object. The next part of the code defines a new coordinate system and transforms the pixel data obtained in the previous part into the new real unit (feet) coordinate system. From there, the car's boundaries can be compared to the predefined parking spot location data to determine whether or not a spot is empty.

Defining a new coordinate system may seem arbitrary, but this is necessary for two reasons. Since the drone being used is not Global Positioning System (GPS) enabled, the location from which the images are captured will be inexact and slightly varied each time. The other reason is that this will allow a human (not an ML algorithm) to predefine the boundaries of each parking space. This is important because many parking lots have faded lines which would be difficult for an algorithm to detect. The new coordinate system will be oriented about and originate from a reference object chosen by the group. The reference object must have a fixed location and orientation so that all parking spot location coordinates are constant with respect to this object. The object must have a fixed width so that a scale factor from pixels to real units (feet) can be determined from it. The object must be low to the ground to avoid any occlusion error. Lastly, the reference object must be easily identifiable because the algorithm will need to locate it each time the drone captures an image for processing.

Transforming from the photo's coordinate system to the coordinate system originated by the reference object is a simple process. The image segmentation model will return the matrix of pixel coordinates that belong to the reference object. From there the centroid of the object will be determined and each pixel coordinate pair belonging to a vehicle will be subtracted from the centroids. This defines a consistent origin for each image. In this way, the drone can be slightly translated longitudinally or laterally, and the program will still be able to compare the vehicles' coordinates to that of the predefined parking spots.

Figure 4: An image where PCA has been applied. The reference object in this case is the access aisle between the two handicap spots. Notice the major (x) axis (blue) is running lengthwise across the reference object while the minor (y) axis (red) runs parallel to the width.

Orientation of the reference object (and the coordinate system itself) will be determined by conducting Principal Component Analysis (PCA). This technique uses eigenvectors to determine the major and minor axis of an object as seen in Figure 4. Whenever an image is processed, the coordinates of each segmented instance will be multiplied by a rotation matrix, with the angle of rotation being determined by PCA. This corrects for any rotational attitude the drone may have had while capturing the image.

The scale factor is determined by dividing the reference object's physical length in feet by the reference object's length in pixels within the image. This will be important because each image taken for processing may be taken from slightly varying altitudes (effectively zoom levels). Obtaining the image's scale factor will correct that. An image of a predefined parking space grid (those of Figure 3) that has undergone all coordinate transformations can be seen in Figure 5.

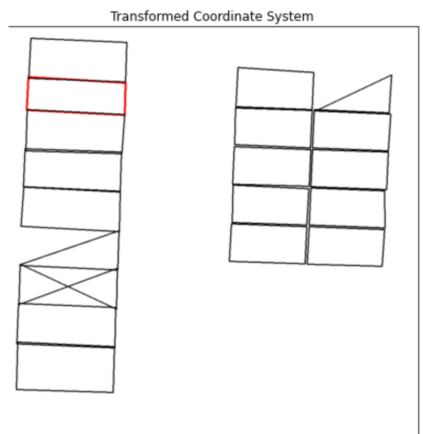

Figure 5: A visual representation of the transformed coordinates of the spaces in Figure 3.

After the coordinate system of the vehicle segmented within the image being processed is transformed, the vehicles' coordinate pairs are ready to be compared to that of the predefined parking spaces. This will be accomplished by a simple "if" statement. If there are any vehicle coordinates within the parking spot's boundaries, the spot is occupied. If there are not any vehicle coordinates within the space, the space is unoccupied. This part of the program has yet to be coded.

While the computer model was being created, the test flight plan was also in the works. November 7th, 2023, was chosen for the first flight test, and the goal for this test was to capture images to test the ML model with, and verify autonomous flight was possible. To fly a drone autonomously, a flight software is required, and, for this project, the DJI GS Pro iOS application was used. This application allows the user to plot waypoints either by flying to them manually or selecting points from a certain previously established map. At each waypoint, an action can be selected, in this case the action was set to capture an image.

The first flight was conducted manually, and the drone was guided above the parking lot to determine a proper altitude for the drone to autonomously navigate to during later flights. After capturing an image at this location, a waypoint was saved. The next flight path would be capturing images around the perimeter of the parking lot. First, the drone was flown manually to four points surrounding the parking lot and returned to the ground. At each of the four points, a waypoint was set for use in the first automated flight test. The drone flight was initiated, the aircraft flew to a set altitude of 100 feet, and navigated to each of the four points where an image was captured. Each image captured by the drone was saved to an SD card, and upon reaching ground the SD card was removed and every picture uploaded to a laptop. Once images were collected and a flight plan verified, the first test flight was complete.

The second flight test was conducted on February 23rd, 2024, with Daniel McCraine and Sean Carpenter. The primary goal of this test was to confirm autonomous takeoff to desired location, image capture, and autonomous landing. Another variation from the first flight test was that we placed two Ground Control Points (GCP) at opposite corners of the parking lot, as seen in Figure 6. The GCPs were meant to help create a coordinate system to map the parking lot, but this method proved ineffective when testing the computer model. Instead, an ArUco marker was implemented digitally and successfully generated a scaled coordinate plane, in which the parking spaces can be defined.

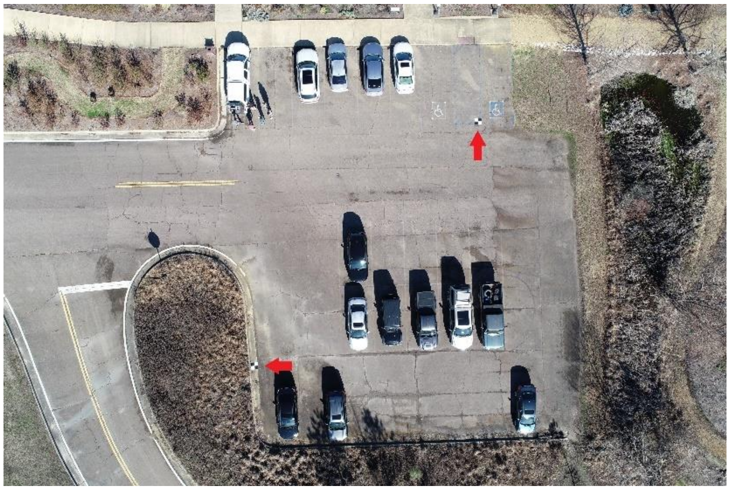

Figure 6: Image taken during second flight test. GCPs indicated by red arrows.

III.Discussion/Results

Once the image from the second test flight was input to the program, the digital ArUco marker was recognized, and the user defined parking spaces were displayed, as seen in Figure 7. Next, a Mask R-CNN . The cars will then be assigned a location relative to the coordinate system created with the ArUco marked for the final step. Lastly, a comparison is made between vehicles and the defined parking spaces in order to determine which spaces are truly occupied. This project was ultimately done as a proof of concept. Other implications of this drone system could be used in construction management when managing logistics of material deliveries and material usage. Other industry practices could also be affected by this type of system like traffic management, state transportation departments, large event parking, etc.

Figure 7: An image with a detected ArUco tag and outlined parking spots

IV. Conclusion

There are still multiple parts of this project to be completed, with the primary hurdles being the verification of ML model accuracy and seamless data transfer from drone to ground station. Once each of these are accomplished, the steps and issues encountered will be documented. The end goal is still to construct a fully automated system that will relay a map to the user depicting open spots versus occupied spots.

The next step that will be taken is testing the completed computer model. Simultaneously, the team will work with the GRI team to understand better how to fully automate the data transfer to a proper ground station. Once these tasks are completed, it will be time for the next flight test. During this test, the goal will be to press a button that starts the flight, the drone will proceed to the chosen location, capture an image and transmit it to the ground station, then return home. If this test is successful, the only tasks left will be to operate this system on a timer and to develop an easily readable user interface. The initial progress has been rewarding in the information we have gained about image mapping onto existing structures and programming techniques used. This will later be used in further development.

Appendix

Acknowledgments

Many thanks to our faculty advisors, Mr. Calvin Walker and Mr. Jimmy Cook for helping to point us in the right direction at each step we take. Mr. Calvin has provided great insight into the realm of UAVs, while Mr. Cook has aided in our decision on sensing technology.

We could not complete any test flights without the help of Daniel McCraine and Sean Carpenter from the GRI, who did not hesitate to go out of their way to complete flights with our team.

References

Tirado, Guilleum Budia, Semwal, Sudhanshu, “Autonomous Parking Spot Detection System for Mobile Phones using Drones and Deep Learning”, International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, 2021

Ravindra Savaram, “TensorFlow Object Detection”, MindMajix, 2021, April 22

Op 79.11: Unmanned Aircraft Systems/Model Aircraft – Policies

Neitzel, F. and Klonowski, J.: MOBILE 3D MAPPING WITH A LOW-COST UAV SYSTEM, Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XXXVIII-1/C22, 39–44, https://doi.org/10.5194/isprsarchives-XXXVIII-1-C22-39-2011, 2011.

Madawalagama, Sasanka, et al. "Low cost aerial mapping with consumer-grade drones." 37th Asian Conference on Remote Sensing. 2016.