Predicting Parking Availability with Ground-Level Imaging

Justin Sostre and Anushree Das

CSCI 631 Final Project

Rochester Institute of Technology

Rochester, NY 14623

{jos4907;ad1797}@rit.edu

Abstract

Parking can become a scarce resource at Rochester Institute of Technology (RIT) during busy times as more students become commuters and the school increases enrollment. Since investing in cameras which can provide live feeds can be expensive and require administration approval, we propose a Faster Region Convolution Network (R-CNN) to detect the amount of cars in a picture and predict if there are parking spots left. It can be easy for humans to accurately predict if there are spots left in a parking lot by taking a look at the lot and use surrounding information. We hypothesize that training a Faster R-CNN on pictures of numerous cars to detect how many cars are in a picture and feeding the amount of cars detected into a psuedo-probability model can reasonably approximate the probability that parking spots are still available in a parking lot. With an accurate Faster R-CNN, we can produce a robust model that advises drivers where parking spots are most likely to be and allow drivers to take risks such as attempting to find more prime spots such as nearer campus. We evaluate our model on the ability to properly detect cars in RIT parking lots from ground-level view and then approximate the availability of parking.

1 Introduction

Parking is sometimes very difficult to obtain especially during busy hours of the day. Specifically at Rochester Institute of Technology (RIT), it is difficult to find parking at busy hours such as noon. Even when busy hours are over, parking lots nearer campus or buildings of interest are likely to have less spots and you take a risk of wasting time attempting to find the spots. It can be difficult and stressful when risk of being tardy to work or class becomes a factor and can lead to making risky parking decisions which lead to fines from traffic enforcers.

Humans are capable of accurately predicting if there will be parking spots available at certain times in certain places by taking quick glimpses at parking lots and determining if the risk if worth it; however, the time to get to the lot to get a good glimpse can be costly on time and thus result in tardiness. We suggest an approach would be to fit cameras on poles in the parking lot that can give real-time data and detect empty parking spaces; however, it can become a hassle to convince administration to pay for and maintain a system such as this. It also can become difficult (and expensive) for this system to be powered appropriately and connect to a hub where it can share this data. With these hurdles, it can be seen as rather difficult to justify this system and without the metric of people actually using the system, it may be an almost impossible ask.

In this paper, we suggest an alternative to this solution for determining if there are parking spots still available in a lot. Such an alternative is to implement a Faster R-CNN capable of detecting cars in a parking lot using ground-level imaging and reporting the detected amount of cars into a statistical model which can then predict the likelihood of spots still existing in a lot while also considering time-of-day, weather, and lot capacity. With this, it is no longer a requirement that there are cameras taking a bird's-eye-view of the lot and instead relies on statistical properties to give an approximation to the likelihood an empty spot exists.

We also ascertain that this method can be used at many different parking lots that are not at RIT and predict that the use of a social media application can allow people to use this at highly used lots for others use. Moreover, this method can be amended to collect data to predict parking availability without the use of long-term imaging and just the use of imaging in the very beginning to understand the peaks of traffic. We are, however, focused on the using this method to eliminate the need for overhead views of parking lots.

1.1 Data

To use pre-trained Faster R-CNN (faster rcnn resnet152 v1 640x640) provided by TensorFlow Object Detection API and evaluate it against pictures of cars, we use a pre-annotated data set called the COCO data set located at `https://cocodataset.org/#home` Lin et al. [2015]. We compiled a selection of pictures with automobiles using tensor flow, `https://www.tensorflow.org`, a collection of Artificial Intelligence (AI) algorithms aimed at allowing people to easily access AI. We used this to evaluate the pre-trained Faster R-CNN model by detecting bounding boxes using the model and comparing those detections with the actual box. Afterwards, we used pictures of RIT parking lots, some of which were taken by us, and others scooped from the internet due to the lack of students commuting to class this year as a result of online classes. We annotated this data and used it to determine the accuracy of our model.

1.2 Project novelty

In past research, it has been a simple assumption that there would be a way to obtain overhead views of the entire parking lot which can easily allow for the artificial drawing of parking regions by humans which then can be used to classify whether the spot was empty or occupied by a car. This work would produce very good results as long as the classification was robust and accurate allowing a live feed of parking lot activity. Our work aims at providing a new perspective in which we cannot make an assumption we will have that technology to allow a live feed and an overhead view of the lot and introduce the new assumption that the person viewing where parking is is not near the lot such that the probability (even if there are 10 spots open) is 1 as it can be taken by other drivers. Moreover, the design of this project can be adapted with these assumptions to use probability. The probability in our work is not only to predict if spots are there but also to predict if they will be there in the next few minutes.

2 Past Related Work

Until now, some parking occupancy monitoring methods have been researched. First, in the proposed methodology of Choeychuen [2013], the object detection with automatic thresholding is performed to detect moving objects from static camera in the specific aerial parking view. The parked cars are distinguished from the moving cars to get the parking lot map. Finally, the parking lot map is used in detection of available parking spaces

The second paper, Cho et al. [2018], proposes a part-based and machine learning-based object detection algorithm to robustly classify the parking slot into vacant and occupied spaces. The proposed method processes images captured by wide-angle lens camera corrected by approximated lens correction algorithm. The classifier consists of a hierarchical combination of weak classifiers which detect parts of object. Baroffio et al. [2015] presented an approach that classifies the parking slot

Figure 1: An example of an acceptable picture (left) because it is getting a large view of the lot and an example of an unacceptable picture (right) because it is only getting a picture of two cars and you cannot see behind them.

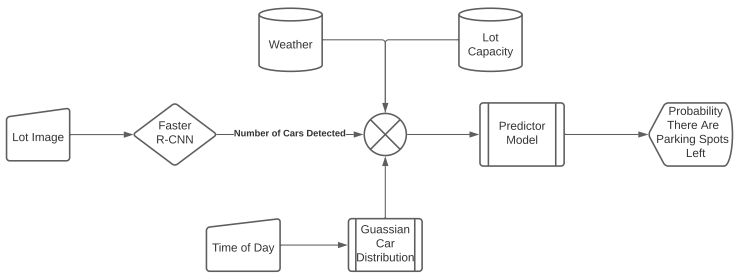

Figure 2: Flow Chart of the Model

according to the color histograms. Shih and Tsai [2014] presented another approach that calculates the pixel value difference between the parking slot and the registered slot image beforehand.

Lastly two applications developed live tracking of parking. Zack Banack et. al. developed a prototype called BlindSpot, that utilizes existing overhead security cameras to determine available parking spots in real-time. The live camera footage of RIT's parking lot is processed by machine learning algorithms. The algorithms are able to determine where and how many vehicles are in a given parking lot. Another developed application used OpenCV and houghlines to smooth out parking spots Dwivedi. One issue was the very far away imaging from the parking spots which caused the detection to detect multiple parking spots even when the spot was the same so Dwviedi assumed all parking spots were the same size to fix this and identified each individual parking spot.

While the past works in this field mostly involves detecting cars and finding empty parking spots via live feed of aerial view or wide angle view from high heights, our work detects cars from human eye view perspective and predicts the availability of empty parking space.

3 Proposed Approach

3.1 Assumptions

To make sure that our parking information makes sense, first we make a few assumptions about the pictures being put into the model. First, we assume that pictures are reasonably capturing of the parking lot such that it is slightly over eye level where you can see multiple cars behind other cars. This is important because if there are pictures only of a first row of cars, this will be a bad picture. We could possibly relax this requirement with a CNN that can classify bad or good pictures but that is discussed further in future work. We then assume that the function that describes the normal distribution of cars over the day is $c(t)$ where $c(t)$ is defined for $0 \leq t \leq 24$ and that the distribution does not depend on what lot. We justify this by knowing that most classes at RIT start around 11, 12, and 1 and thus there is a lot of parking during that time from students. We ignore the possibility of weekends and certain events such as a career fair. We also assume that harsh weather such as hard snowfall directly reduces the amount of students that attend class and thus the amount of parking is more likely. We also assume that weather will affect the turnout of students; for example, sunny weather will increase turnout and winter blizzards will decrease turnout in parking lots. Moreover, we assume that users attempting to find parking are not in the lot they are finding parking for. This allows us to have a more liberal probability that there is traffic because spots can be taken at any time since we do not have the ability to see every spot of the lot.

3.2 Method

We introduce a Faster R-CNN capable of detecting how many cars are in a picture of a ground-level view of cars in a parking lot. We train this Faster R-CNN on data annotated by the COCO data set using only a classification of car and not car ignoring the possibility of increasing our complexity with empty parking spaces. We justify this because it would not matter if we can see empty parking spaces because it is sometimes probable (dependent on the time of day) that the parking will not be available after the picture is taken and thus should be considered in the probability; moreover, we justify the prospect of non-stationary cars likewise in which the probability of these parking spots will be taken is more likely if there are cars driving around attempting to find spots.

Once the detection of all cars is performed for an image, the number of cars that are detected is fed into another model which takes in the following independent variables along with the detected cars: time-of-day, weather, and parking lot capacity. Moreover, parking lot capacity can change how likely it is for there to be parking. This metric is tricky because larger parking lot capacity makes the pictures less useful so we can also cut up parking lots into sections that are easier to photograph on the ground. This however then requires more pictures of lots so, for now, we will also assume that the pictures are sufficient. We derive a few equations to describe our model for predicting parking as follows:

$$c(t) = \frac{1}{2\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{t-12}{2})^2} \tag{1}$$

$$p(r, l, w) = \begin{cases} 1001, & r \leq 3, \\ \frac{l}{rwc(t)}, & otherwise \end{cases} \tag{2}$$

$$P(p) = \frac{p}{p + \alpha}, \tag{3}$$

where r is the detected cars from the Faster R-CNN, l is the capacity of the lot, α is constant to bound the probability between 0 and 1 and w is the weather coefficient which is assumed from this table of values:

Weather Coefficient Table	
Weather	Coefficient
Snow	0.80
Rain	0.95
Storm	0.90
Sunny	1.1
Any Other Weather	1

Equation 1 describes the distribution of cars over the day while equations 2 and 3 determine the probability and then bound it between 0 and 1 respectively. Note that these are psuedo-probability distributions because the area under the curve is not necessarily 1 which is a requirement for probability distributions. The α-value also should be changed when trained but since there is no real way to evaluate psuedo-probabilities, it would need to collect a lot data and then use previous predictions to change α (which are validated with statistical theory.) In this paper, we assume $\alpha = 4$.

We evaluate our Faster R-CNN by the time it takes to perform its classification, the accuracy of these classifications, and the regions proposed vs. the actual regions. The performance of our Faster R-CNN directly affects the probabilities in our statistical model since a more robust and accurate Faster R-CNN can produce better detection of cars and thus provide confidence to our statistical model. Our predictor model is harder to evaluate as it relies on guessing. As a result, our predictor model, for now, is fine-tuned based on our oversight which can be changed if we had more data. An unfortunate hurdle caused by the lack of cars and participation on campus due to online classes is the distinct lack of data we can collect ourselves. We evaluate our predictor model then on the reasonableness of the predictions and probabilities.

4 Evaluation, Experiments and Results

The pre-trained Faster RCNN model was evaluated using the tools provided by the TensorFlow Object Detection API. The precision and recall of object detection using Faster RCNN model is calculated using IoU(Interest of Union) between the actual bounding box and the detected bounding box. The precision and recall of the detected bounding boxes for cars, using the pre-trained Faster R-CNN model, for IoU in range 0.50 to 0.95 is as follows:

	Small Objects	Medium Objects	Large Objects
Mean Average Precision	0.119	0.446	0.606
Mean Average Recall	0.228	0.567	0.704

The small objects have area less than 32^2 pixels, area of medium sized objects lie in range of 32^2 pixels to 96^2 pixels and area of large objects lie in range of 96^2 pixels to 10000^2 pixels.

The error is calculated in terms of number of detection instead of precision of the bounding box for this project. From the figure, we get that the minimum error is around threshold value of 0.40. Therefore, for detecting cars, the decided threshold value for score of bounding box is 0.40.

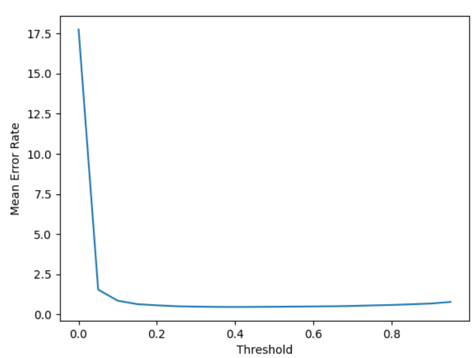

Figure 3: Mean error rate for corresponding threshold

The precision recall curve for our Object Detection model is displayed in figure 6 using our pictures taken from RIT parking lots and google images.

For using our model, currently it is in the state that it will be a csv file that contains, in order, name of picture, number of cars detected, time of the picture, weather coefficient, and lot capacity.

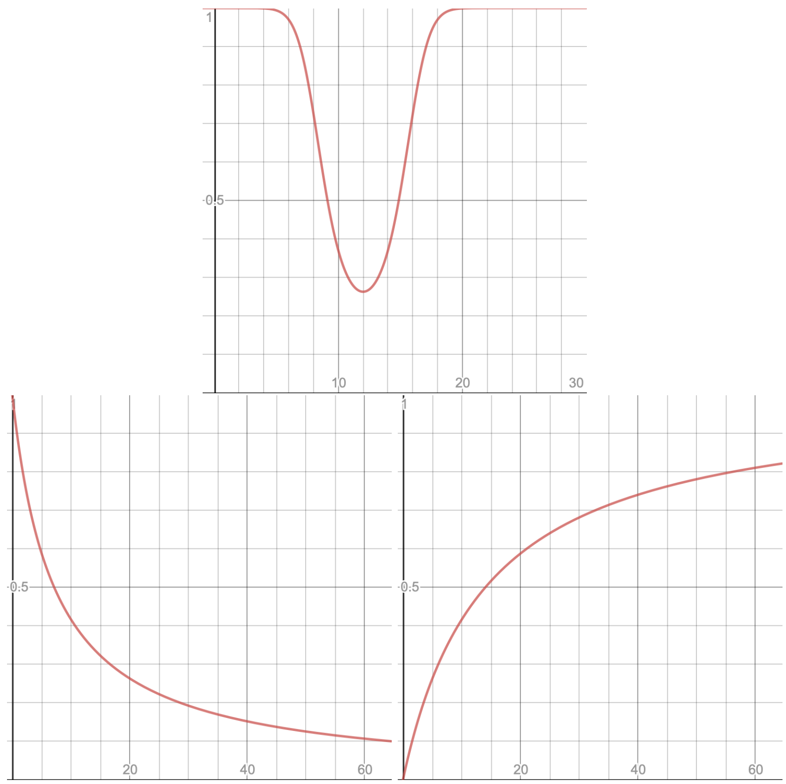

Figure 4: Graphs of the equation 1 (top) and graphs of the equation 3 as a function of detected cars, r (bottom left), and as a function of lot capacity l (bottom right)

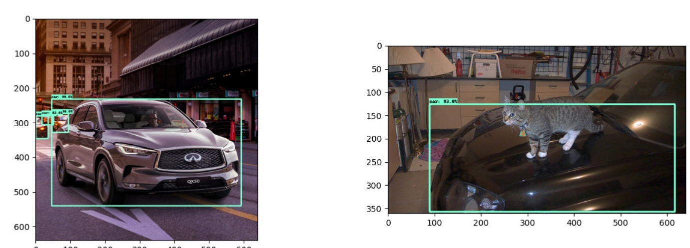

Figure 5: Two examples of car of object detection on cars.

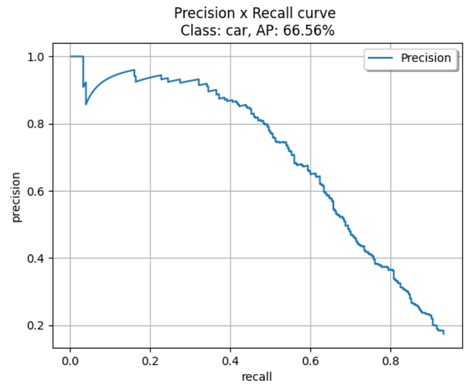

Figure 6: Precision-Recall Curve from classifying our parking pictures

4.1 Results and discussions

The mean average precision(mAP) seems low for COCO data set because the detected bounding box does not absolutely overlap the actual. To begin with, the mAP of this Faster RCNN model is 32% for the complete COCO data set. The mAP for larger objects is more when compare to mAP of small objects. Even though the mAP looks low for the given data set, but the model model detects most of the cars in a given image. The mAP is 66.56% for the decided threshold for our data set which contains pictures of RIT parking lots.

The detector seems to fail to distinguish images of cars that are blocked partially by other cars. This can cause issues when trying to get the amount of detected cars and should be considered poor. For example, we have found that taking pictures from ground-level will disallow the classifier to obtain cars that are behind, but visible to humans, another car. This will obviously make it difficult for this approach to work. One work around is using a custom data set (which we could not do due to the COVID pandemic which disallowed us to get pictures of RIT parking lots or most parking lots in general with a lot of cars) which trains the classifier on top of the already knowledge of what cars look like to also classify them when they are obstructed.

The probabilities also seem to be funky so ideally we need to tune the alpha value or possibly make the alpha value a function based on time or some other metric like the robustness of the Faster R-CNN because it is clear, if you check detections.csv, that a lot of the pictures have way more cars than are detected. This method would be very unreliable currently if we are just allowing the evaluation of the model just on how it performed alone on our parking spots ignoring the training data that we used to calculate the precision in recall in the first table with small, medium, and large objects.

We believe this can work if we make the model more robust and able to detect more cars which will require more training and we also believe that we can remove our current prediction model to use a neural net to approximate a probability in a non-linear way taking in all the inputs in the input layer of the neural network.

4.2 Member contributions

Justin Sostre:

- Wrote most of the write-up including finding the previous work to discuss
- Implemented the model that takes in data from the Faster R-CNN to predict where parking lots are
- Took pictures of RIT parking lots including examples of bad pictures
- Developed some figures for the write up
- Annotated about 20 pictures for a proof of concept

Anushree Das:

- Developed in Python for the Faster R-CNN to detect cars
- Proof-read and validated the write-up for submission
- Developed some figures for the write up
- Obtained data from the COCO data set online using Tensorflow.
- Contributed evaluation of our Faster R-CNN

5 Conclusion and Future Work

We have successfully created a system able to give an estimate about whether there is parking. However, we used a model that we justified to be able to account for the lack of data and parking pictures available instead of using another neural network to predict the availability of traffic. It would be useful to attempt implementing a neural network with an input layer to take in those parameters and produce classifications such as *no parking, some parking, a lot of parking, open lot*. It would require a lot of pictures over the years of lots for a neural network to figure out good features so we picked a simple mathematical model. It seems to give a good estimate knowing what we know about RIT parking and we also suspect tuning the model will allow us to obtain models for different lot environments such as a workplace.

One of the biggest issues was collecting pictures of full parking lots at RIT that exactly gave us what we wanted which was decent ground-level pictures. Since the use of online classes, there have been way less people in the parking lots even at the normal congestion hours of 11, 12, and 1. It really made the data processing hard so it would be necessary to obtain more pictures when more attendance is in person.

We failed to use different types of models and instead developed only one model to predict the availability of parking spots due to time constraints. This could be used as a baseline for future research to determine if another model performs better at predicting the availability of parking. Moreover, another thing we did not accomplish was filter very bad pictures such as pictures that do not encompass a huge amount of the parking lot but rather a few cars or pictures that are not good quality. This would reduce the robustness of the prediction as it would predict two cars as it is zoomed in on just two.

Another useful investigation is the use of coordinating different angles of the parking lot and training, as well, on empty parking spots. The empty parking spots can help increase the probability of spots left as the spots increase in the picture and with different angles, a computer can stitch together a useful image to predict and find more parking spots and calculate a more accurate probability parking spots will be there when the commuter arrives.

One possible application of this is an application of the mass deployment variety where people can take pictures of parking lots for others. It would simply require an app UI and this model as a backend development. Moreover, we predict this can be used for many different types of predictions farther than RIT parking or parking in general. We suggest that this can be also used for any phenomenon such as the amount of people at a party or event.

References

L. Baroffio, L. Bondi, M. Cesana, A. E. Redondi, and M. Tagliasacchi. A visual sensor network for parking lot occupancy detection in smart cities. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 745–750, 2015. doi: 10.1109/WF-IoT.2015.7389147.

Zack Banack BlindSpot. Fixing rit's parking problem in 24 hours. URL `https://medium.com/@zackbanack/fixing-rits-parking-problem-in-24-hours-a0aeb5113f8f`.

W. Cho, S. Park, M. Kim, S. Han, M. Kim, T. Kim, J. Kim, and J. Paik. Robust parking occupancy monitoring system using random forests. In *2018 International Conference on Electronics, Information, and Communication (ICEIC)*, pages 1–4, 2018. doi: 10.23919/ELINFOCOM.2018.8330608.

K. Choeychuen. Automatic parking lot mapping for available parking space detection. In *2013 5th International Conference on Knowledge and Smart Technology (KST)*, pages 117–121, 2013. doi: 10.1109/KST.2013.6512799.

Priya Dwivedi. Find where to park in real time using opencv and tensorflow. URL `https://towardsdatascience.com/find-where-to-park-in-real-time-using-opencv-and-tensorflow-4307a4c3da03`.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015.

S. Shih and W. Tsai. A convenient vision-based system for automatic detection of parking spaces in indoor parking lots using wide-angle cameras. *IEEE Transactions on Vehicular Technology*, 63(6):2521–2532, 2014. doi: 10.1109/TVT.2013.2297331.