IDENTIFYING EMPTY PARKING SPACES USING COMPUTER VISION

ABSTRACT:

This research projects investigates the use of computer vision to locate vacant spaces within a parking lot then alert a user with a mobile app. Specifically the effectiveness of a convolutional neural network to classify the difference between a car and an empty parking space, storing the locations in a cloud database, then making the information available within the app.

DATA PIPELINE:

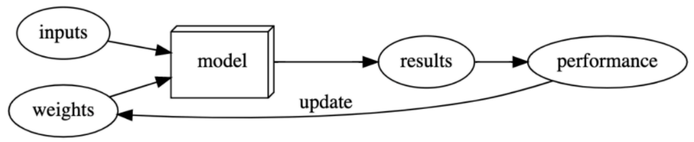
The most important part of any deep learning model is always going to be the data it is trained on. To obtain data of a parking lot a camera was placed outside the third-floor window of an academic building with a good view of tons of cars and parking spaces. Over the course of four months a picture was taken of the parking lot every 45 minutes. This was automated using a Python script and a cron job. All this data is great, but for it to be sent through a deep learning model it must be labeled. To help with the data labeling process another Python script was created to loop through all parking spaces, show a OpenCV window, allow the user to enter 'c' for car and 'o' for open, storing the image in the correct directory.

TRANSFER LEARNING:

When building a computer vision application instead of starting from scratch you can often make faster progress by using a pretrained model and transferring it to a new task. The research community has created lots of large datasets like ImageNet or MS COCO and trained deep learning models that are very accurate. It is possible to use these models and "freeze" the parameters and weights and then continuing training on new data. Because the pretrained models may have already been good at detecting lines, curves, or objects their knowledge can be transferred to a new task, in this case cars and parking spaces.

TRAINING AND DETECTION:

A neural network works by inputting parameters (which contain inputs, in this case a picture of a car or open spot with a label, and random weights) into a model. Output is then generated, and a loss function is used to calculate how good the output is doing. Gradient Descent is then performed which calculates how far off the output was, followed by back propagation which updates the weights. This process if continued until a satisfactory model is generated. Using transfer learning and a resnet18 pretrained on the ImageNet dataset the model was trained for 2 epochs (complete pass-through dataset) using the Cross Entropy loss function and Adam Optimizer for gradient descent, achieving 99% accuracy on the validation set. For detection, a multithreaded program was necessary to achieve real time detection, where one thread is constantly retrieving video frames, while the other grabs the most recent frame and runs it through the model to find which spaces are vacant.



CLOUD STORAGE AND MOBILE APP:

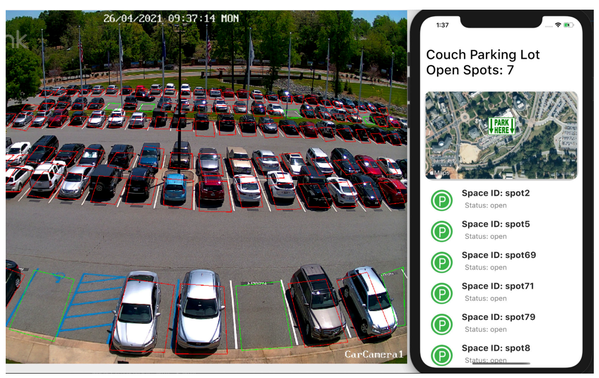
In order to make this information available to users in a mobile app, it must be sent to a cloud database. Using a NoSQL database each spot is stored in a document with a status of either open or taken. When the detection program is run, a snapshot of the database is taken and stored locally in a Python dictionary so it will only update spots whose status needs to be changed. This data is then read from a mobile app in real time and whenever the information is updated in the database a signal will be sent to the mobile app and it will display the changes, thanks to Google's Firebase Firestore and real time streams.

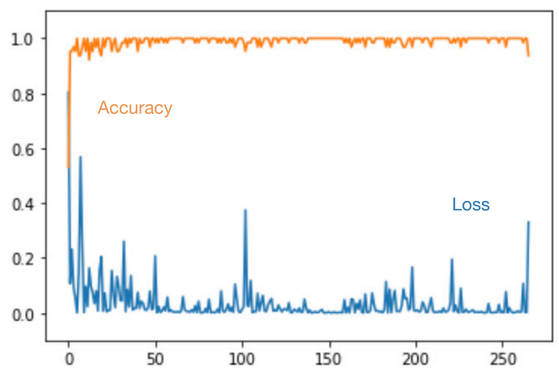
RESULTS:

After spending the past two years researching if this project was truly feasible as well as the best practices for implementation, it is safe to say it has been a major success. This project is proof that a mobile application can be created to show users where vacant parking spaces are in real time. Even with a relatively small dataset of around 3,000 cars and open spaces, the model is extremely accurate and will only continue to improve as more data is collected and labeled. Future goals for this project include moving from a classification to object detection model which will allow for even fast detection as well as creating a different model that can determine where parking spaces are in an image.

ACKNOWLEDGEMENTS:

Most practices and information in this project was obtained by taking both the Fast AI and Coursera Deep Learning Specialization coursers. A special thanks to Professor Shore who made this project possible through his guidance, motivation, and instruction.





URGW HIGH POINT UNIVERSITY

HIGH POINT UNIVERSITY
Webb School of Engineering

ANTHONY DEMATTOS
DEPARTMENT OF COMPUTER SCIENCE