Autonomous Parking Space Detection

Julien Nyambal
1439552

School of Computer Science and Applied Mathematics
University of the Witwatersrand, Johannesburg

Supervisor: Dr. Richard Klein

Master's Proposal

Overview

Introduction

Description

The project aims to automatically predict the availability of a parking spot given the feeds provided by a video camera monitoring the parking area.

Motivation

Hunting for parking brings no joy for students[1]

"...Students at Wits are struggling to find available parking spaces on campuses since..."
"...student X said that finding parking is the biggest issue she faces at Wits."

— Aarti Bhana, Wits Vuvuzela

Problems

Various factors can explain the parking space problems mostly in urban areas:

- Urban planning not following the quick growth of population dynamics
- Drivers not aware of available space due to lack of knowledge of the area or anything obstructing it from the driver
- Bad management of the parking area

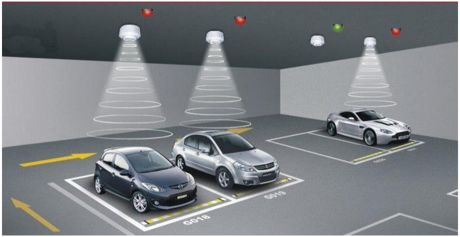
Aims

This project aims to:

- Perform comparative study between existing solutions (SVM based algorithms) and CNN,
- Use video cameras to allocate parking spaces to drivers
- Apply computer vision techniques to infer the status of a space (Background subtraction, Hough transform ...),
- Help the driver to locate a vacant spot using the dedicated web application,
- Run a quantitative analysis of the results produced by the system to predict parking habits of drivers.

Background and related work

Sensors Based Detection



Non-intrusive sensors, [2]

Intrusive sensors, [3]

Sensors Based Detection

Pros:

- Very accurate,
- Since individually installed per spot, the system can return the available number of spots [4],
- Almost no computation required

Cons:

- Expensive in installation in terms of number of devices to place per spot,
- Expensive in maintenance.

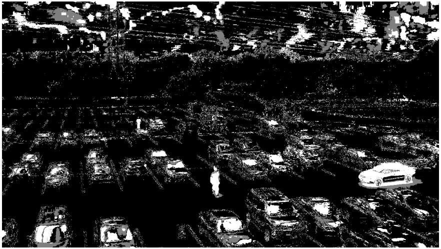
Definition

The background subtraction is a technique used in computer vision and image processing for detecting the foreground from a set a images of the same scenario (video sequence of a non-moving camera) for various types of detection problems.

$$pixel\_value = \begin{cases} 1, & |frame_i - frame_{i-1}| > T_h \\ 0, & \text{otherwise} \end{cases}$$

Video Based - Background Subtraction II

Original image

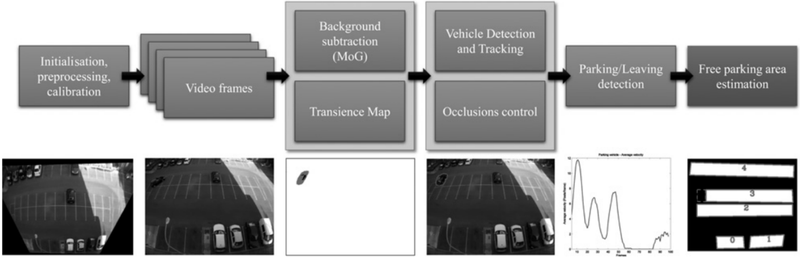


Background subtracted image

Video Based - Background Subtraction III

Background subtraction to:

- Create a transience map of the parking area or to model the parking area given the previous positions of vehicles [5]



Background subtraction method proposed by C. G. del Postigo et al.[5]

This method used the background subtraction and the Mixture of Gaussians to detect and track vehicles to infer the availability of remaining spots given a **threshold** on the transient value,

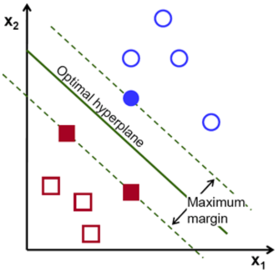
Video Based - Background Subtraction IV

- Use histograms of spatial features to map the spots of an unmanaged parking area but taking as object the group of pixels generated by vehicles, rejecting the non-vehicle objects by automatically adjusting the **threshold** [6].

Support Vector Machines I

Definition:
Supervised machine learning technique that is based on a decision boundary to separate data into different subsets (classes). Support vectors are the coordinates of data point in the data space that are the closest to the boundary lines, hyperplanes.
Support Vector Machines(SVM) can be used for **classification** and **regression** problems. The classification has been mostly used for the parking space detection.

Support Vector Machines II

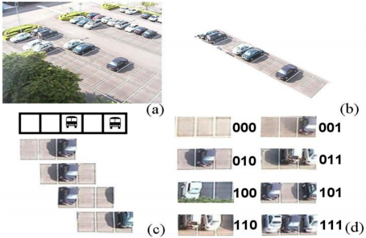


Support Vector Machines, OpenCVDocs

Support Vector Machines (SVM) have been widely used for the problem.
Usually associated with Histograms of Oriented Gradients (HoG), Difference of Gaussians (DoG), color histograms, Local Binary Patterns...

Support Vector Machines III

- To detect available parking spaces using both multi-class Support Vector Machines (SVMs) and a Markov Random Field (MRF) framework using the color histograms as features [7] with an accuracy of 93.52%,



Multi-class SVM, [7]

- To classify individual spot using DoG- features with an accuracy of 92.33% after a comparative study with the k-NN, LDA and HoG, RGB, HSV YUV (color spaces) as features [8]%

- As similar to the previous point,to classify individual spot using Local Binary Patterns and Local Phase Quantization as features with an accuracy of 99% [9].

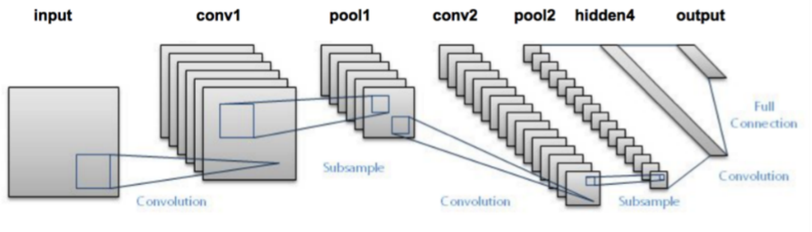
Segmented image, [9]

Deep Learning - Convolutional Neural Networks I

Definition

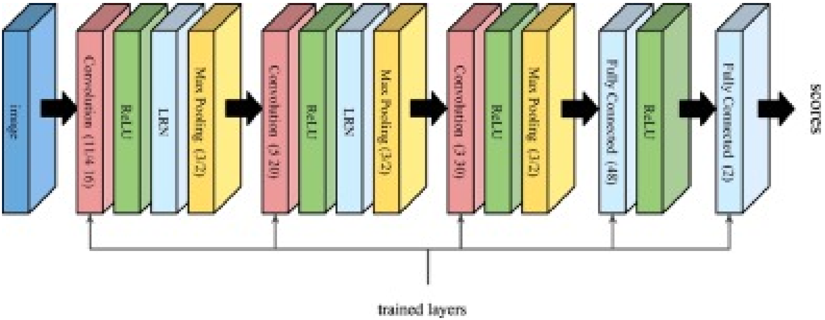
- Supervised learning branch of machine learning for pattern and object recognition,
- CNN learn extracted features,
- CNN are computationally expensive at the training phase especially when the dataset contains many classes with many objects to classify,
- But high accuracy compared to its direct competitor the SVMs.

Deep Learning - Convolutional Neural Networks II



Overview of a CNN: Lenet-5, [10]

Deep Learning - Convolutional Neural Networks III

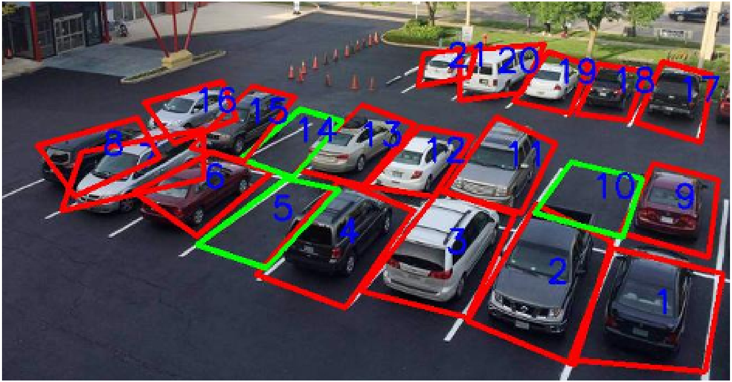


Modern CNN, mini-AlexNet, [11, 12]

Deep Learning - Convolutional Neural Networks IV

CNNs to bring a solution to parking management:

- The classification happens in a remote server that is fed with images coming from cameras facing a parking area.



Online classification, [13]

- The classification happens in individual customized cameras that contain Raspberry Pi's. Each camera runs a reduced version of AlexNet CNN [12].

Research Methodology

Research Hypothesis

- **Hypothesis1:** Hough Transform for lines detection which implies spot presence (on the first frame) + CNN for classification,
- **Hypothesis2:** Hough Transform for lines detection + Background Subtraction to the incoming frame from the camera to automatically segment the image for parking spaces through their lines + SVM * (Histograms of Oriented Gradients OR Difference of Gaussians OR Local Binary Patterns ...),
- **Hypothesis3:** RCNN for spots detection + (SVM OR Normal-CNN) OR (trained-Normal-CNN AND CNN-learned-features-SVM) for spots classification.

Research Questions

- Do the background subtraction and the Hough transform associated with the Convolutional Neural Networks and the Support Vector Machines efficiently return automatically the status of all the parking spaces?
- Will the RCNN be better at automatically detect the spots without human intervention?
- How could the union of the features generated by a trained CNN and the power of the SVM improve the classification accuracy and credibility of the parking space management?

Background Subtraction

Method 1:

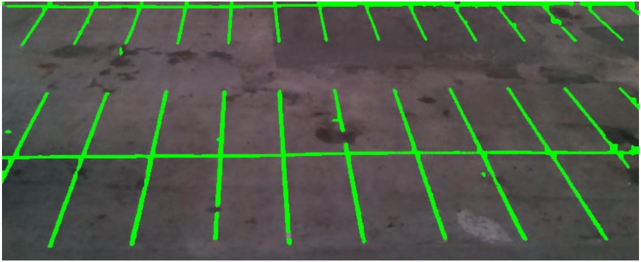
- To automatically create a mapping of all the parking spaces given a parking lot as illustrated in Section 5 when using the transience map [5],
- From the map, feed the cropped spot to either the CNNs or SVMs.

Method 2:

- To use GrabCut on one image of the parking space to crop the background from the foreground, create a map (ground-truth) out of GrabCut
- And classify the results using CNNs or SVMs.

Hough Transform

Parking spots contain lines that can be detected using the Hough Transform. The end-points of those lines will joint altogether by the Convex Hull algorithm to create rectangle of spots to build a map (ground-truth) of the parking area and classify each spot per frame.



Result of Hough Transform

Convolutional Neural Networks: CNN

- **Method 1:** To reduce the depth of the successfull AlexNet, since 2 classes in the project, instead of 1000,
- **Method 2:** A new CNN will be created using state-of-the-art rules for CNN design. CNN design is an art that follows some rules to have both speed and specially accuracy.

Region-based Convolutional Neural Networks: R-CNN

The algorithm will be ran on one frame of the parking area of interest to return the coordinates of the potential parking spots. If all the spots are not mapped, manual labeling will take place to create a full map of the parking area. After all the spots being mapped, the cropped patches will get classified by either the CNNs or SVMs.

Sample Result of RCNN

SVMs

Have been broadly used previously. The hyper-parameters of the SVMs, C, γ, kernel will be computed using grid search.

The Data: From A to Z I

Data Collection

- Mostly collected from the 4th, 3rd, Postgraduate Students parking next the Chamber of Mines , Wits
- With a GoPro Hero 4 camera,

Data Preparation

- GoPro camera = Distortion $\implies$ distortion removal mainly,
- Reduction of the image size.

Data Labeling

- **Detection**: Create ground-truth of all spots appearing in the collected pictures in XML format, The input is the original image and the output a tuple $t = \{image.jpg, image.xml\}$
- **Classification**: Generate patches of individual spots from all the pictures collected, regardless the state of the patch.

The Data: From A to Z II

Datasets

- **Training**: Generate 2 folders corresponding to the states of interest: Vacant and Occupied.
- **Testing**: PKLot dataset will be used, and some other scenarios of parking spaces different from the training ones.

Training & Testing I

Detection:

- Caffe implementation of the Region-based CNN (RCNN)
- Different solvers type (Stochastic Gradient Descent Adaptive Gradient, Nesterovs Accelerated Gradient ...) and learning rate, learning rate policy for better choice of accuracy,

CNN Classification:

- Caffe or hdf5 (Keras) model for classifying the crop image of the parking spot,
- Feeding the CNN (no matter what model used) with the patches of parking lots collected from the dataset (vacant & occupied folders).

SVM Classification:

- Using the winning feature extractors HOG, SURF, SIFT ...
- Using grid search to determine the optimal hyper-parameters of the proposed SVM model generated,

Training & Testing II

- Using k-fold cross-validation to efficiently train and test and avoid leaks of the training set to the testing set compared to using a static training-testing phase.

How Will We Evaluate The Model? I

Confusion Matrix: Table or matrix to visually evaluate the performance of a model. This research project is a supervised learning binary classification problem. The resulting confusion matrix will produce four outcomes:

- **T**rue **N**egatives (**TN**): Actual negative outcomes that are predicted negatives,
- **T**rue **P**ositives (**TP**): Actual positive outcomes that are predicted positives,
- **F**alse **N**egatives (**FN**): Actual positive outcomes that are predicted negatives,
- **F**alse **P**ositives (**FP**): Actual negative outcomes that are predicted positives.

How Will We Evaluate The Model? II

		Predicted Classes	
		Occupied	Vacant
Actual Classes	Occupied	**TN**	**FP**
	Vacant	**FN**	**TP**

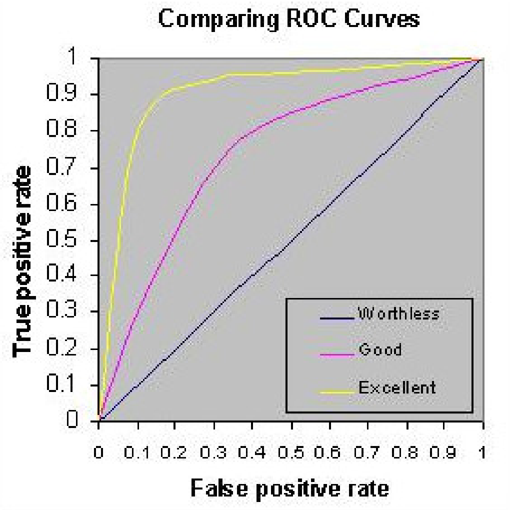
Confusion Matrix: By convention, the true classes are on the left-hand side of the matrix and the predicted classes are on the top of the matrix.

- Accuracy $= \frac{\text{Total number of correct classication}}{\text{Total number of cases}} = \frac{\mathbf{TP} + \mathbf{TN}}{\mathbf{TP} + \mathbf{TN} + \mathbf{FN} + \mathbf{FP}}$
- Recall $= \frac{\text{Number of True Positives}}{\text{Total number of actual positives}} = \frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FN}}$
- Precision$= \frac{\text{Number of True Positives}}{\text{Total number of predicted positives}} = \frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FP}}$

How Will We Evaluate The Model? III

- **Receiver Operating Characteristic curve (ROC) curve** $=$ **TPR** vs **FPR** $= \frac{\textbf{TP}}{\textbf{TP} + \textbf{FN}}$ (y-axis) vs $\frac{\textbf{FP}}{\textbf{FP} + \textbf{TN}}$ (x-axis): In that graph, the goal is to obtain a curve growing towards the upper right corner, reducing the number of false negatives and false positives as illustrated in Figure 14.

How Will We Evaluate The Model? IV

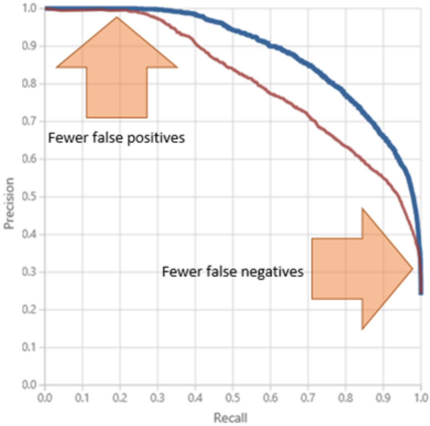


Three different cases of ROC curves, [14]

How Will We Evaluate The Model? V

- **Precision-Recall Curve** = **Precision** (y-axis) vs **Recall** (x-axis): : In that graph, the goal is to obtain a curve growing towards the upper right corner, reducing the number of false positives as illustrated in Figure 15.

How Will We Evaluate The Model? VI



Different cases of precision-recall curves, [15]

Research Plan

Research Phases and Time Plan

Phase	**Duration**	**# of weeks**
Data collection	June 2017- August 2017	12 weeks
Data Preparation	August 2017- October 2017	12 weeks
Training & Testing	October 2017- February 2018	16 weeks
Deployment	February 2018- April 2018	12 weeks
Analysis	March 2018- June 2018	16 weeks
Write-up	July 2018- September 2018	12 weeks

Table: Research Phases and Time Plan

Potential Issues and Risk Management

- **Data Collection Issues**:
  The data will be collected by the researcher if there is no, or not usable footage available,
- **Delays on Defined Time Plan**:
  The design of the different components may take longer to produce, that will generate a delay on the delivery time,

End

References I

A. Bhana, "Hunting for parking brings no joy for students," *Wits Vuvuzela*, p. 2, 2017.

computerstationco, "Overhead sensors." `http://computerstationco.net/let/images/imagedata/pms_img3.jpg/`, 2017.
(Accessed on 04/03/2017).

eurocarparks, "Ground sensors." `https://www.eurocarparks.com/custom/uploads/2009/12/sensor_parking.jpg/`, 2017.
(Accessed on 04/03/2017).

S. Lee, D. Yoon, and A. Ghosh, "Intelligent parking lot application using wireless sensor networks," in *Collaborative Technologies and Systems, 2008. CTS 2008. International Symposium on*, pp. 48–57, May 2008.

C. G. del Postigo, J. Torres, and J. M. Menndez, "Vacant parking area estimation through background subtraction and transience map analysis," *IET Intelligent Transport Systems*, vol. 9, no. 9, pp. 835–841, 2015.

References II

K. Choeychuen, “Automatic parking lot mapping for available parking space detection,” in *2013 5th International Conference on Knowledge and Smart Technology (KST)*, pp. 117–121, Jan 2013.

Q. Wu, C. Huang, S. y. Wang, W. c. Chiu, and T. Chen, “Robust parking space detection considering inter-space correlation,” in *2007 IEEE International Conference on Multimedia and Expo*, pp. 659–662, July 2007.

M. Tschentscher, C. Koch, M. Knig, J. Salmen, and M. Schlipsing, “Scalable real-time parking lot classification: An evaluation of image features and supervised learning algorithms,” in *2015 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, July 2015.

P. R. de Almeida, L. S. Oliveira, A. S. B. Jr., E. J. S. Jr., and A. L. Koerich, “{PKLot} a robust dataset for parking lot classification,” *Expert Systems with Applications*, vol. 42, no. 11, pp. 4937 – 4949, 2015.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” in *Proceedings of the IEEE*, pp. 2278–2324, 1998.

References III

A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.

G. Amato, F. Carrara, F. Falchi, C. Gennaro, C. Meghini, and C. Vairo, "Deep learning for decentralized parking lot occupancy detection," *Expert Systems with Applications*, vol. 72, pp. 327 – 334, 2017.

S. Valipour, M. Siam, E. Stroulia, and M. Jagersand, "Parking-stall vacancy indicator system, based on deep convolutional neural networks," in *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pp. 655–660, Dec 2016.

Columbia, "Evaluating risk prediction with roc curves." `https://www.mailman.columbia.edu/research/population-health-methods/evaluating-risk-prediction-roc-curves`, July 2017.
(Accessed on 07/14/2017).

References IV

Microsoft, "Using roc plots and the auc measure in azure ml." https://blogs.msdn.microsoft.com/andreasderuiter/2015/02/09/using-roc-plots-and-the-auc-measure-in-azure-ml/, July 2017. (Accessed on 07/14/2017).