# MARMARA UNIVERSITY

# FACULTY of ENGINEERING

# COMPUTER ENGINEERING DEPARTMENT

CSE4088 Introduction to Machine Learning

## Title of the Project

Image Based Car Detection

## Group Members

150116064 Furkan Kemal DÖNMEZ

150116056 Emre YİĞİT

## Lecturer

Prof. Dr. Çiğdem Eroğlu Erdem

# 1. Abstract

Looking at a place with the human eye, it is easy to perceive and identify the objects there. Cameras can also record the image but cannot identify the object by themselves. In this project, we tried to overcome this problem as much as we could.

Thanks to machine learning, it is possible for a machine to detect objects, and we built our research and work on it. And the object here is the car.

The main purpose of the project is to detect the images whether there is a car. We searched several different machine learning algorithms (svm, logistic, etc.) in the process. Among them, svm was the technique we chose and used for this project.

# 2. Overview

As written in the abstract section, we can summarize this project as car detection. As a result of our research, we decided that the most suitable technique for Object detection was svm among those mentioned and we used it.

## 2.1 Project Schedule

**Phase-1**

- **Task 1:** Searching Related Projects
- **Task 2:** Literature Search

**Phase-2**

- **Task 3:** Collecting Dataset
- **Task 4:** Checking Software Tools and Libraries
- **Task 5:** Preparation of Mid-term Report
- **Task 6:** Designing the Algorithms

**Phase-3**

- **Task 7:** Implementation

**Phase-4**

- **Task 8:** Preparation of Final Report

We did this project as a group of 2 people. Since the content of the project was new to us, we mostly worked together. Therefore, we cannot specify that only one person worked for any task. In addition, in Phase-1, we can state that we are progressing by doing our own research and telling each other later. Equal intensity was studied in all subtasks except that.

# 3. Project Accomplishment

## 3.1 Searching Related Works

We searched for works related to the subject we chose.

**Object classification for autonomous vehicle navigation of Stanford campus**

“We propose an object classification problem using camera images of objects an autonomous vehicle is likely to encounter. In particular, we bound the driving scenario to navigating Stanford campus around other vehicles and students. To accomplish this task, we will compare different supervised learning techniques with varying features for image recognition. One supervised learning technique we investigate is a support vector machine (SVM), which allows for high-dimensional feature kernels in case data is not linearly separable. Another supervised learning technique we propose to use is softmax regression because it is a generalized logistic regression algorithm and the output is a probability of the label. In addition to comparing these two learning techniques, we will discuss the impact of varying features to the learning algorithms. In particular, we will look at comparing the features of RGB values and gray-scale values. Lastly, we will discuss an alternative supervised learning approach using a Convolutional Neural Network, which does not require manual specification of features. In order to focus the problem on the goal of assisting autonomous vehicle navigation on campus, the project will be limited to classification of items most likely to be found around the Stanford campus.” [1]

**Whale Detection & Identification from Aerial Photography**

This project takes and processes aerial photographs and deals with whale detection and identification. This project uses SVM in the identification part. We loosely inspired from this project. In summary, this project detects the whale in the image and decides which pixels are whale and which pixels are ocean. [2]

**Vehicle Detection (tiny YOLO ver, HOG & SVM ver)**

The goal of this project is object detection like the others. In this project, two methods were used, HOG + SVM approach and YOLO approach. However, not all of them concern us. The part of the project that interests us is the part where the vehicle in the image is detected and how the relevant algorithms are used. [3]

## 3.2 Literature Search

As we mentioned in the project plan, we searched the literature. We examined the algorithms used in this subject and decided that svm is the most suitable algorithm for us.

Support Vector Machine is a classification algorithm similar to Logistic Regression. Both try to find the best line that separates the two classes. The algorithm allows the line to be drawn to be adjusted in two classes so that it passes from the furthest place to its elements. It is a classifier that takes no parameters. SVM can also classify linear and nonlinear data, but generally tries to classify the data linearly.

Advantages of SVM:

- They are effective in higher dimensional spaces.
- They are effective when the number of dimensions is greater than the sample size.
- A number of training points are used in the decision function ("support vectors"). Therefore, the memory is used efficiently.
- Versatile: Many different kernel functions can be used for the decision function.

## 3.3 Datasets

As a result of our research for this project, we decided to use the following dataset in both the train and test phases.

**KITTI Object Detection with Bounding Boxes**

This Datasets contains the Kitti Object Detection Benchmark, created by Andreas Geiger, Philip Lenz and Raquel Urtasun in the Proceedings of 2012 CVPR, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite". [4]

This dataset contains over 14,000 images 7,518 of images for testing and 7,481 of images for training. Some of them with bounding boxes labels but we are not interested in this part.

## 3.4 Software Tools and Libraries

We decided to use Python programming language and Numpy library.

### Python

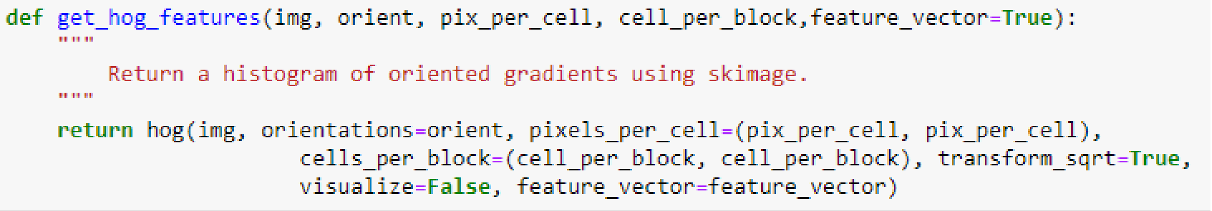
Python is a general purpose programming language which is created by Guido van Rossum, it is used for different platforms like mathematic, computer GUI, web and so many large scientific applications. We chose python because it is easy for coding and it has vast libraries for machine learnig especially numpy. Because of this features, python is most commonly used programming language for machine learning.

### Numpy

"NumPy forms the basis of powerful machine learning libraries like scikitlearn and SciPy. As machine learning grows, so does the list of libraries built on NumPy. TensorFlow's deep learning capabilities have broad applications — among them speech and image recognition, text-based applications, time-series analysis, and video detection." [5]

## 3.5 Design, Implement, Train and Test Models

We talked about advantages of SVM in section 3.2. Besides these advantages, SVM will be very slow during the train phase and so, we decided to use a feature descriptor by looking at the projects we inspired. Feature descriptor is an algorithm which takes an image as input and outputs feature vectors. The features can be shape, color, texture of the object. We decided to use Histogram of Oriented Gradient (HOG) as the feature descriptor. Because SVM have a good compatibility with HOG. HOG takes an image and divides into blocks. In these blocks there are some cells and finally in these cells there are some pixels. HOG extract the feature vector from them. In the training phase we performed HOG on a labeled training set.



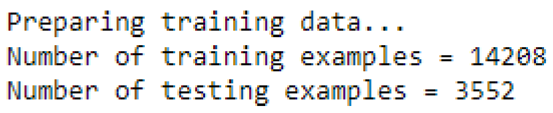
```
def get_hog_features(img, orient, pix_per_cell, cell_per_block,feature_vector=True):
    """
        Return a histogram of oriented gradients using skimage.
    """
    return hog(img, orientations=orient, pixels_per_cell=(pix_per_cell, pix_per_cell),
                       cells_per_block=(cell_per_block, cell_per_block), transform_sqrt=True,
                       visualize=False, feature_vector=feature_vector)
```

Figure 1: get_hog_fetures method

We used these parameters for this method:

orient = 9        pix_per_cell = 8        cell_per_block = 2

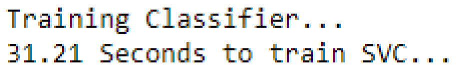
Here are the size of dataset that we used:



```
Preparing training data...
Number of training examples = 14208
Number of testing examples = 3552
```

Figure 2: number of examples

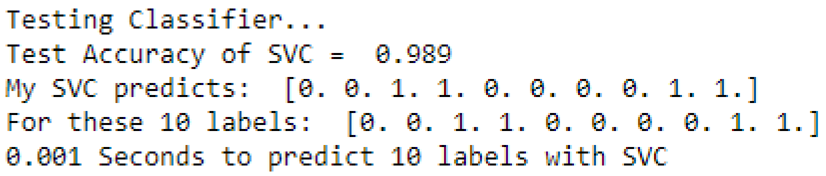
In the training part, there is a method named train_SVC. This method takes two parameters which are x_train and y_train. How long the train phase took is shown below figure.



```
Training Classifier...
31.21 Seconds to train SVC...
```

Figure 3: training time

In the testing part, there is a method named test_classifier. This method takes three parameters which are SVC (model of the svm), x_test and y_test. In this method firstly we calculated the accuracy of SVC using score function. And then predict the output. Lastly, result of the prediction and labels shown below figure.



```
Testing Classifier...
Test Accuracy of SVC =  0.989
My SVC predicts:  [0. 0. 1. 1. 0. 0. 0. 0. 1. 1.]
For these 10 labels:  [0. 0. 1. 1. 0. 0. 0. 0. 1. 1.]
0.001 Seconds to predict 10 labels with SVC
```

Figure 4: testing results

The outputs we got during the testing phase are explained in the images below.

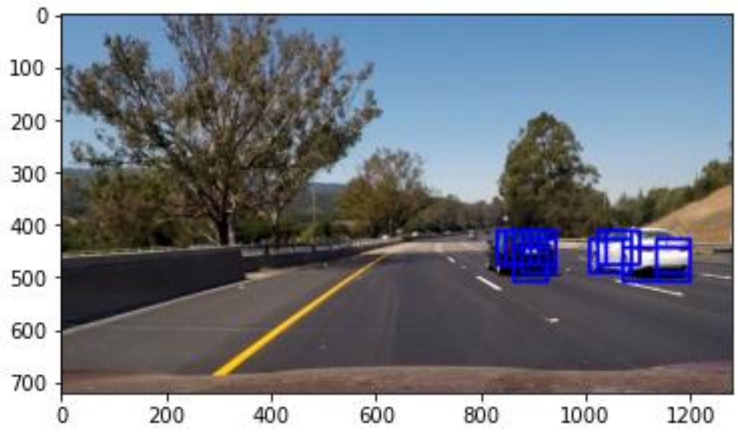


Figure 5: All detections

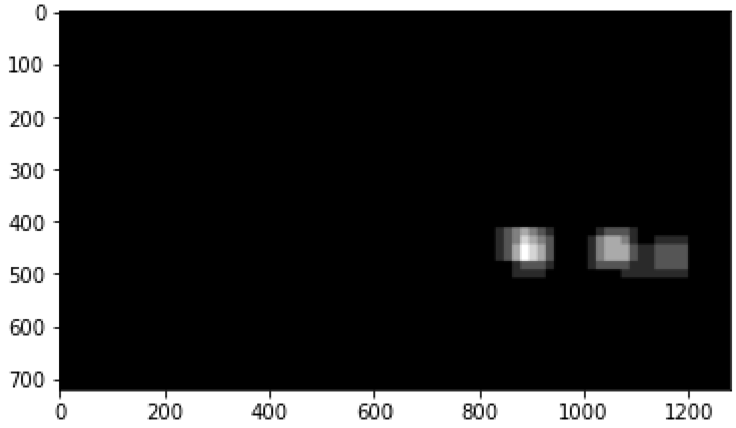


Figure 6: After applying heat

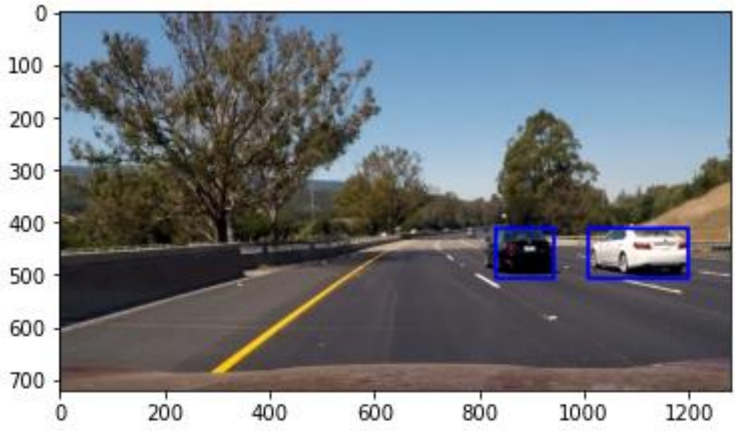
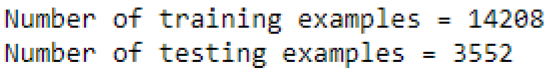


Figure 7: After treshold the final result

## 4.Summary

To summarize the project, the system takes a test image and if there are cars in this image, it detects and labels them. We see the size of the dataset used in the figure below.



```
Number of training examples = 14208
Number of testing examples = 3552
```

As a result, the object detection process took place and the cars in the image were detected. In the project, we only dealt with detecting cars, but as an advanced level of this work, it may be possible to detect tens, maybe hundreds of different objects in a single visual thanks to machine learning.

## 5.References

**1-** Heather Blendell and Sarah Thornton. “Object classification for autonomous vehicle navigation of Stanford campus”. Stanford University CS229 Machine Learning Final Project 2015.

**2-** Aditya Mahajan, Adrien Perkins. “Whale Detection & Identification from Aerial Photography”. Stanford University CS229 Machine Learning Final Project 2015.

**3-** Vehicle Detection (tiny YOLO ver, HOG & SVM ver), https://github.com/windowsub0406/Vehicle-Detection-YOLO-ver (Date of access: 11/12/2020)

**4-** https://www.kaggle.com/twaldo/kitti-object-detection

**5-** https://numpy.org/