Seminar: Technology of Autonomous Vehicles

Camera Based Vehicle Tracking and Detection

Abhishek D Jagtap, Lakshith Nagarur Lakshminarayana Reddy

Contents

1 Introduction **1**

2 Methods and Algorithms **2**
2.1 Histogram of Oriented Gradients 2
2.2 Data Preprocessing 3
2.3 Support Vector Machines 3
2.3.1 SVM with Kernel function 4
2.4 Sliding Window 4
2.5 Heatmap 5

3 Results **6**

4 Conclusions **7**

Abstract

The main motive of this paper is to detect vehicles in real time driving scenario based on the video data obtained from a camera in the windshield of a vehicle. Using Histogram of Oriented Gradients to extract features and Support Vector Machines for classification. The Dataset used for training the classifier is KITTI and GTI vehicle dataset. The accuracy achieved is 96. percent to increase the accuracy of training introduced grid parameters and achieved accuracy of 99.2 percent. The concept of Sliding-Window and Heatmap is used to detect the vehicles in the video based on trained classifier.

Keywords

Histogram of Oriented Gradients, Support Vector Machines, KIITI dataset, Sliding-Window, Heatmap.

1 Introduction

Road accidents and fatality rates are increasing every year and coming up with the solution for this is a non-trivial task and are mainly caused because of various human factors [1]. From past many years various Research and Development organisations working on developing autonomous systems in vehicles to avoid road accidents and to create driver safety environment. Autonomous vehicle includes several stages in development, the main technology behind this is Artificial Intelligence and the main surrounding environment information is gathered using sensors like RADAR, LIDAR and cameras [2].The advancement in Machine Learning and computer vision algorithms made vehicle detection is more practical using camera sensors.Vision based object detection can be applied in various scales, orientations and lighting conditions. There are different feature detectors, Histogram of Oriented Gradients[HOG] this was used by Dalal and Triggs for human detection in the year 2005 from then on HOG was used in various object detection [3]. For our implementation using HOG for extracting features and Linear Support vector machine(SVM) for classification as both are efficient and can be used in real time applications [4].

Figure 1: vehicle dection in real time

2 Methods and Algorithms

2.1 Histogram of Oriented Gradients

HOG is one of the popular technique in computer vision for detecting object in an image. In HOG the gradient magnitudes and directions are computed at each and every pixels in an image and they are binned into small group of bins.The way this orientation binning works is each pixel votes for a orientation based on the nearest bin available that is 0 to 180 if we ignore negative directions.This vote is selected based on the magnitude. There are around five steps involved in computing HOG. Firstly, the image is normalized and then it is divided into small grid of cells and these cells are grouped into a overlapping blocks where cells may belongs to more than one block. In the following step horizontal and vertical gradients are computed [3] [5] [6]. With these gradients Histogram of orientations are computed for every cell. The size of image used is 64 pixels width and 64 pixels height.

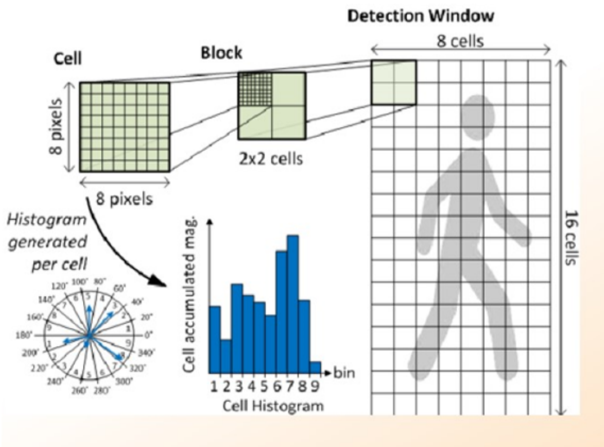

Figure 2: Cell Histogram

At each pixel, Gradient and Magnitude are computed by

$G = \sqrt{GX^2 + GY^2}$

$\theta = \arctan \frac{GX}{GY}$

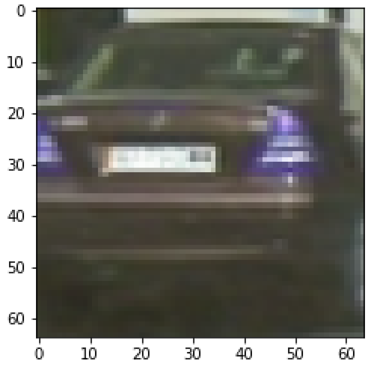

Figure 3: Vehicle image

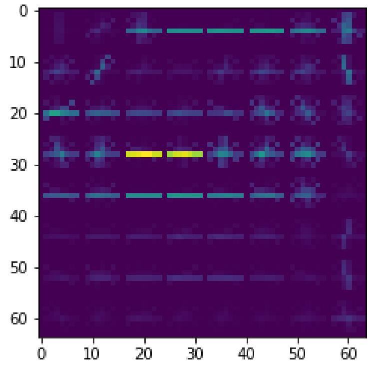

Figure 4: HOG Feature

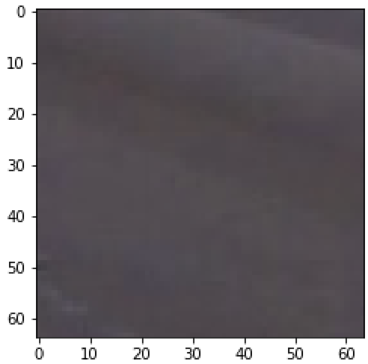

Figure 5: Non vehicle image

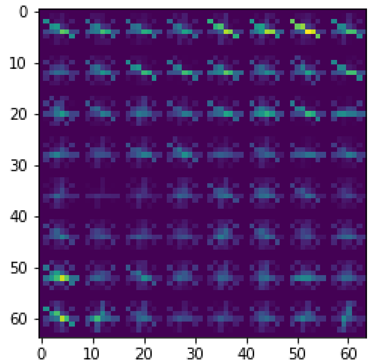

Figure 6: Hog feature

In our alogorithmic implementation used the following HOG parameters:
PixelsPerCell = $8 * 8$
CellsPerBlock = $2 * 2$
Orientations = 9
Implemented HOG on vehicles and non vehicle images to extract features of images with an without objects .

2.2 Data Preprocessing

The dataset used for training the classifier is KITTI and GTI vehicle dataset [7].The features of both vehicle and Non vehicles are archived by using the previous step. Now this dataset is normalized and split into training and test phase in the ratio of 80:20. Data normalization is usually done to change the values of numeric columns in the dataset to a common scale, without distorting differences in the ranges of values.There are different methods to normalize the data, for our purposes we have normalized by removing mean and scaling it to unit variance.

2.3 Support Vector Machines

Support Vector Machines is a supervised learning algorithm that are used for both classification and regression. The basic intuition behind it is to find a hyperplane that separates our data. The optimal hyperplane is found by optimization techniques such as lagrange multipliers.
A simple example of how SVM works is shown in below figure 7 black circles and white circles represent positive and negative samples. The SVM constructs three hyperplanes H1, H2, H3 denoted in the below figure. These hyperplanes tries to classify our given data. It is obvious from the figure

that H1 does not classify the given data and is not considered as good hyperplane.H3 classifies the data in the best possible way as the distance is more compared to H2, thus SVM is also referred as Max-margin separator.

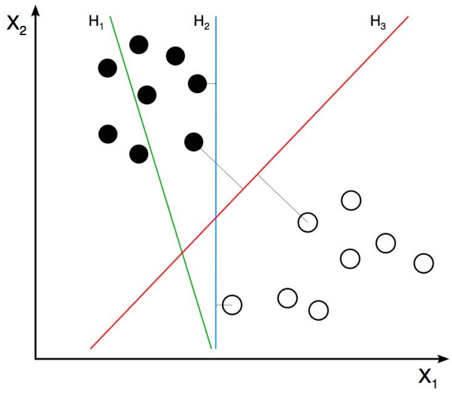

Figure 7: SVM Classifier

The main objective of our paper is to classify images into vehicle and non vehicles in the best possible way, thus support vector machines is most suitable for our task. For our purposes we have used linear support vector machines(Binary classifier) for classifying the two different classes.Accuracy of our model using Linear classifier was 96.8 percent with computational time 2.1s.

2.3.1 SVM with Kernel function

The basic intuition behind using SVM with kernel functions is for large input datasets linear support vector machines does not yield good accuracy [8].
There are different types of Kernel functions

Polynomial Kernel - $k(x, x_i) = (< X, x_i > +1)^2$

Sigmoidal Kernel - $k(x, x_i) = tanh(< X, x_i > +1)$

Radial basis function Kernel - $k(x, x_i) = exp^{-\Gamma||x-xi||^2}$.

For our model we have used radial basis function as a kernel function [9]. In rbf we have to define several parameters such as gamma which defines how smooth the classfier, this specifies the area of influence of support vectors. c as a regularization parameter, which controls the trade-off between achieving a low error on the training data.But rbf is very sensitive to the gamma chosen and that may lead to over fitting. so we use k-fold cross validation method to choose the best gamma and c value which minimizes the error [10]. the accuracy achieved after performing svm using rbf as kernel function is 99.2 percent with computational time of 50 minutes this might be a drawback.

2.4 Sliding Window

Classifier has been tried now could able predict vehicles and non vehicles. To predict the location of the vehicle in the image used the concept of sliding window. sliding window runs through the image with the defined size of the window and by applying the classifier image data whether it could able to predict vehicle or not.

But its not a feasible idea to perform the sliding window throughout the image as searching for the vehicle in sky. So can ignore at-least top quarter part of the image 8 . But the size of the vehicle varies because of different distances from the viewing camera, to overcome this issue, used three different window sizes and overlapping of window to detect the vehicle at different distances in the image. Trained our model on feature extracted from 64*64 pixels image,so for the window with different size have to be resized to 64*64 in order to keep the features same.

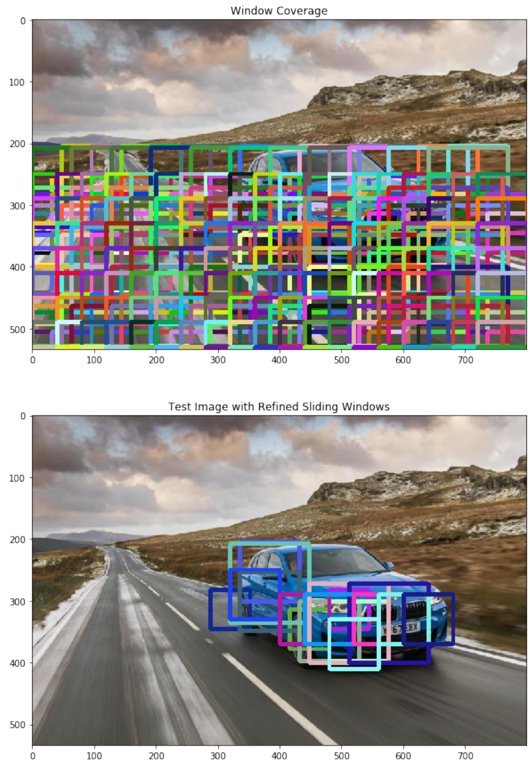

Figure 8: Sliding window

2.5 Heatmap

It is possible to predict the vehicles in the image using the sliding window method but there are many overlapping windows for one vehicle. To draw the final bounding box the concept of heatmap by applying threshold is used. to draw the final bounding box for the predicted vehicle using the sliding window method. we create a blank black image with the same size of the original image.Adding the pixel values by one for the whole region of identified refined windows. The intensity with the common region is being highly intense, by applying threshold on the final clipped image, could able to get the final coordinates of the final bounding box.

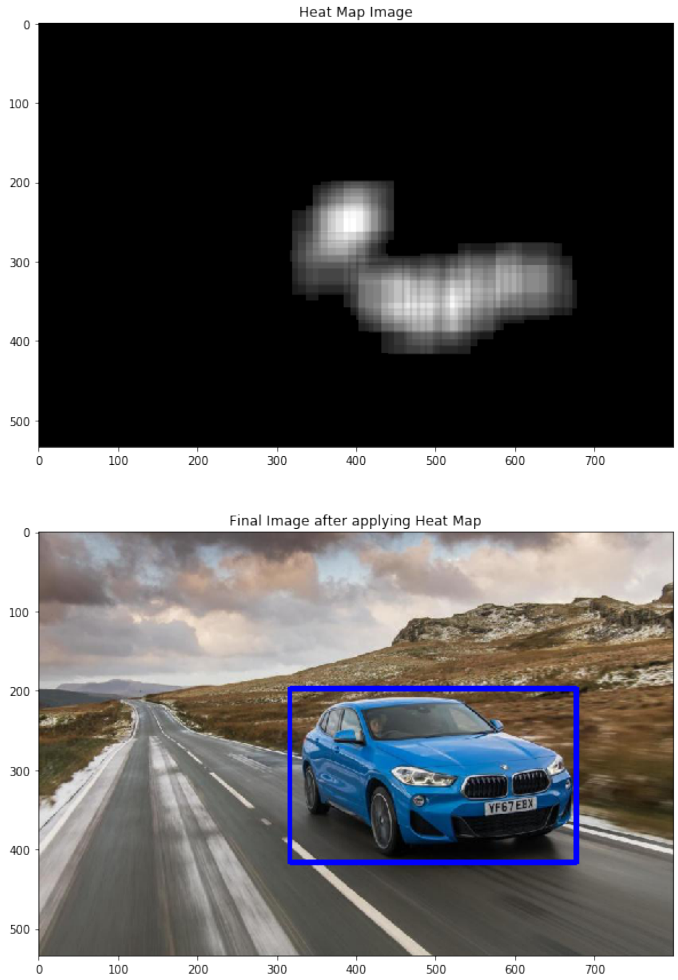

Figure 9: Heatmap

3 Results

This paper was mainly focused on tracking and detecting vehicles in real time. we have used HOG and Linear classifier to train the model.The sample images used for training the classifier are the combination of GTI vehicle database and KITTI vision benchmark. HOG features are obtained using 9 orientation bins, 8*8 pixels per cells and 2*2 cells per block.Using linear classifier we achieved around 96 percent. To increase the model accuracy we make use of SVM with kernel function,we have introduced hyperparameters c, gamma and radial basis kernel function to increase the accuracy of the model and we could able to achieve 99.2 percent of the accuracy.

The below figures illustrates the vehicle detected while testing new images.

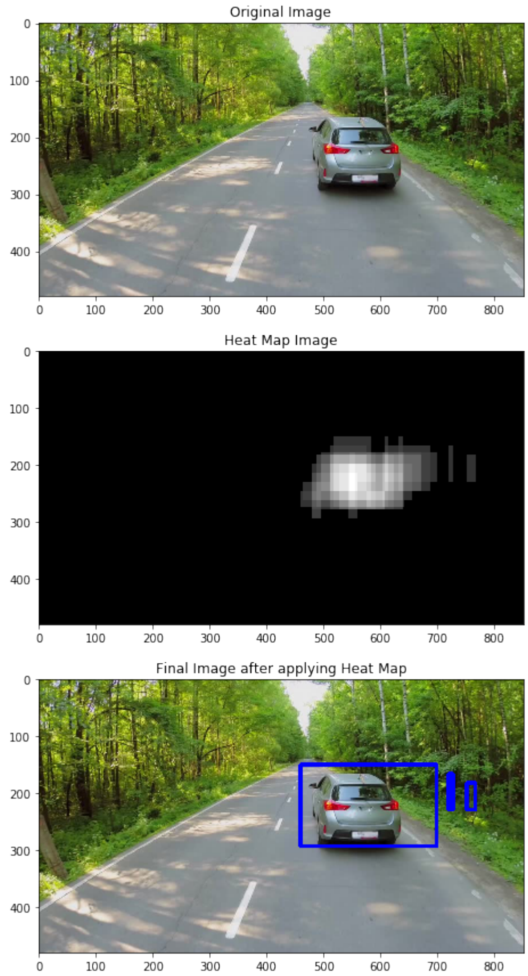

Figure 10: Test results

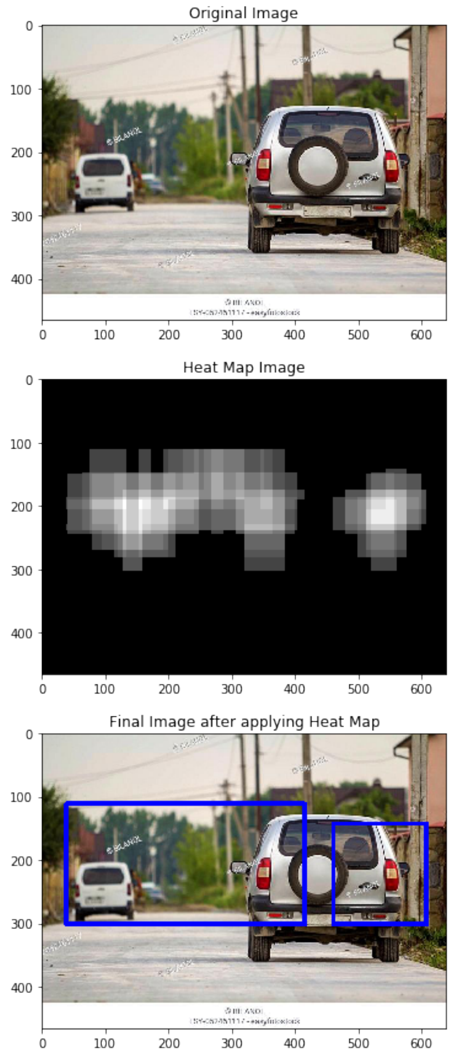

Figure 11: Test results

4 Conclusions

The vehicle detection and tracking algorithm pipeline was introduced. Histogram of Oriented Gradients for extracting the features and Linear classifier to classify vehicles and non vehicles. To increase the accuracy of the our model we have used hyper-parameters such as C, gamma and Radial basis kernel function. Sliding window and Heatmap concept used to detect the exact position of the vehicle in the image and to get the final bounding box for the detected vehicle.

References

[1] I. C. for Road Safety International, "Icorsi's comments on who's draft global targets for road safety risk factors," *International Journal of Injury Control and Safety Promotion*, pp. 1–6, 2020.

[2] E. Guizzo, "How google's self-driving car works," *IEEE Spectrum Online*, vol. 18, no. 7, pp. 1132–1141, 2011.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. IEEE, 2005, pp. 886–893.

[4] B. Morris and M. Trivedi, "Improved vehicle classification in long traffic video by cooperating tracker and classifier modules," in *2006 IEEE International Conference on Video and Signal Based Surveillance*. IEEE, 2006, pp. 9–9.

[5] Y. Zakaria, H. Munim, M. Ghoneima, and S. Hammad, "Modified hog-based on-road vehicle detection method," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 18, pp. 3277–3285, 2018.

[6] S. Nawaz, "Hog-svm car detection on an embadded gpu," 2016.

[7] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "The kitti vision benchmark suite," *URL http://www. cvlibs. net/datasets/kitti*, 2015.

[8] T. Howley and M. G. Madden, "The genetic kernel support vector machine: Description and evaluation," *Artificial intelligence review*, vol. 24, no. 3-4, pp. 379–395, 2005.

[9] B. Yekkehkhany, A. Safari, S. Homayouni, and M. Hasanlou, "A comparison study of different kernel functions for svm-based classification of multi-temporal polarimetry sar data," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 40, no. 2, p. 281, 2014.

[10] T. Fushiki, "Estimation of prediction error by using k-fold cross-validation," *Statistics and Computing*, vol. 21, no. 2, pp. 137–146, 2011.