A novel vehicle detection system

Yehia Zakaria[1], Mohamed Ashraf Ali[1], Hossam E. Abd El Munim[2], Ahmed Hassan Yousef[2,3]
Maged Ghoneima[1], and Sherif Hammad[1]
[1]Mechatronics Engineering Department, Faculty of Engineering, Ain Shams University, Cairo, Egypt
[2]Computer and Systems Engineering Department, Faculty of Engineering, Ain Shams University, Cairo, Egypt
[3]School of Information Technology and Computer Science, Nile University, Giza, Egypt

***Abstract*—Histogram of oriented gradient (HOG) feature has been widely used in vehicle detection. In this paper, a modified version of HOG is proposed by introducing compass gradient into the HOG calculation. Three different versions of the modified HOG features are used as an input for linear and nonlinear support vector machine (SVM). The modified HOG variants proved to have better classification performance than that of the standard HOG. The classification results of modified HOG and nonlinear SVM are compared to the classification results of YOLO object detector. Finally, a vehicle detection system based on the best performing classifiers is introduced.**

***Keywords*—Vehicle detection, Histogram of Oriented Gradient, HOG, Support Vector Machines, SVM, CNN, YOLO.**

I. Introduction

In the last few years, making driving safer has been a target for the automotive industry. Since 1990, over a million people die annually because of road related accidents, most of them are in the age range of 15 to 49 years [1]. These numbers grab the attention of all concerned entities to improve traffic systems to decrease the number of causalities. Improving infrastructure such as providing higher quality roads, introducing better traffic systems can help in deaths reduction. While it was reported that most of the road crashes are due to human factors [2]. As an example, driving under the influence (Driving while drunk) causes 10-37 % of road fatalities [3]. Humans can also be easily distracted specially with all kinds of technology surrounding us. Due to all these reasons and in pursuit of safer roads, vehicles manufacturers try to increase computer intervention in the driving process. Advanced Driver Assistance Systems (ADAS) provide the necessary data to the drivers in order to take proper actions. Recently ADAS has become one of the basic components of modern vehicles. A long-term target is to fully automate the driving process with an autopilot that can drive vehicles safely from place to another.

Although autonomous driving is a complex task, it's composed of three main processes: Sensing, processing, and acting. Autonomous driving can be described as a continuous cycle of these three stages. At the sensing stage, the vehicle uses its sensors to perceive the environment. The vehicle should have information about the surrounding obstacles, whether it's static such as roads and traffic signs or dynamic as the vehicles, pedestrians, or any other object that exists on the road. The vehicle should be able to localize itself according to a certain map and use the information collected from all sensors to plan its path to the destination and more importantly, to avoid collision with the surrounding objects. All these processes should be performed in realtime as the response time is a limiting parameter for the vehicle speed.

Vision is at the center of autonomous driving, it's a primary task in the sensing stage. Using its vision module, the vehicle should be able to detect its lane, recognize different traffic signs, and detect dynamic objects in the surrounding. Vehicle detection is one of the main tasks of the vision system of autonomous driving or ADAS systems. One approach to tackle vehicle detection problem is to perform it on two stages: hypothesis generation and hypothesis verification. In the hypothesis generation, regions of interest (ROI) where vehicles are most likely to exist are determined. The output of the hypothesis generation process is then verified using different combinations of features and classifiers to confirm the vehicle existence in the suggested window.

One of the features that is widely used in object detection is histogram of oriented gradient (HOG). The HOG feature was firstly introduced by Dalal and Triggs [4] with an application on pedestrian detection but it had been used in various object detection applications including vehicle detection [5], [6], and [7]. Since then efforts had been made to improve the discriminative power of HOG introducing different changes on Dalal and Triggs HOG version [8], [9], [10], [11]. However, all the efforts maintained the same gradient calculation method as in original HOG.

Convolution Neural Network (CNN) [12] is recently used in object detection problems. Modified versions of CNN are introduced in [13] and [14]. You Only Look Once (YOLO) is a recent object detection system that based on CNN. Introduced in [15], YOLO handles object detection as a single regression problem from the input image to the object bounding boxes. An enhanced version of YOLO was introduced in [16] to improve the detection rate, localization, and speed.

The goal of this paper is to introduce a new HOG variant using compass gradient mask in the calculation of HOG that is proven to increase the discriminative power of the original HOG on vehicle detection application. The support vector machines (SVM) [17] is used for classifiers training. The combination of SVM and HOG is widely used in literature [18], [9]. The nonlinear SVM classifiers are then compared to the results obtained from YOLO on the same dataset.

The following parts of this paper are organized as follows: The proposed HOG variant along with an overview of the YOLO object detector are described in section II. The Experiments and results of the comparison between the classifiers based on modified HOG and nonlinear SVM with YOlO object detector are described in section III. Finally, the conclusion and future directions are in section IV.

II. Methods

A. Compass HOG

In the conventional HOG, the gradient of the image calculated in two directions (vertical and horizontal), this will result in losing data from the image and hence a less discriminative feature. On the other side, the proposed compass HOG uses gradient calculated from the change in all eight compass directions, which make it more descriptive for the image and hence more discriminative feature. Compass gradient is used to create three different HOG variants, the first one is obtained by concatenating the HOG features calculated using the four compass gradient.

$$compHOG = [\text{HOG1 HOG2 HOG3 HOG4}]^T$$

At which HOG1, HOG2, HOG3, and HOG4 are the HOG features calculated using the gradient in four directions. The length of the feature vector, in this case, is four times the length of the feature vector of the original HOG. Another HOG variant is formed by averaging the gradients in four compass directions. The third HOG variant is formed by taking the direction of the gradient that has the maximum magnitude. The later two HOG variants have the same vector length of the conventional HOG.

B. Support Vector Machines (SVM)

SVM is one of the supervised machine learning approaches that's widely used in binary classification problems. It depends on mapping the input observations into n-dimensional space where n is the length of the feature vector. The next step is to find a hyperplane that separates the two groups of observations. Support vectors are the nearest observations to the hyperplane from both groups. The distance between the support vectors and the hyperplane is called the margin. Finding the hyperplane is an optimization problem at which the maximum margin that achieves minimum error is found.

In some cases at which the data is not linearly separable, nonlinear SVM is applied. In nonlinear SVM, another dimension is added to feature space by applying a kernel to the data. In this paper nonlinear SVM with Gaussian kernel is used in classification.

C. YOLO Object detector

You Only Look Once (YOLO) is an object detection architecture, which gets the input image and divides it into SXS cells then each cell enters the Yolo's architecture and uses the Convolutional Neural Network (CNN) architecture shown in figure 1 to predict the confidence of each class for each grid. The default weights provide pre-trained weights which had been trained on PASCAL dataset or MS COCO dataset. These weights can detect 20 classes. Another detector

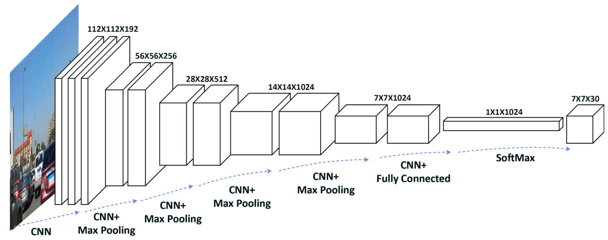

Fig. 1. The architecture: the YOLO's detection network consists of 24 convolutional neural networks and then two fully connected networks with CNN window size 3X3 and the final product of the network is 7X7X30 tensors which contain the predictions of the input image

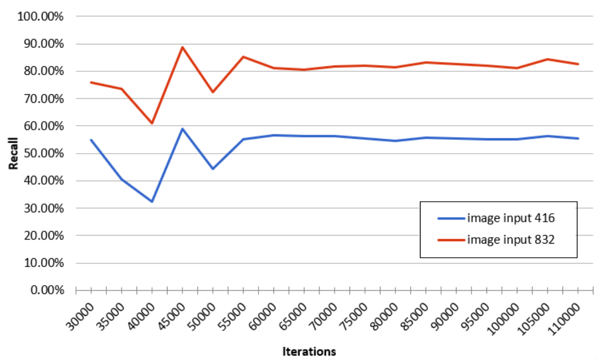

Fig. 2. Recall percentage for different training iterations

for vehicle class only is trained for the comparison with nonlinear SVM and compass HOG detector. KITTI dataset is used to train YOLO. Some changes were made in the training configurations. The Batch (number of input images in one iteration) is given a value of 2. The subdivision (number of divides in one iteration) and the number of classes (number of objects to be detected) is set to 1. The problem is, when can we stop YOLO's training? That's why we need to test every 5000 iterations to choose the best weights file before over-fitting happens. Figure 2 shows the recall percentage of each iteration.

We found that the recall percentage is very low for the default configuration for detection, so the resize function was changed in YOLO to resize the image to 832 instead of 416 to make better detection for small objects.

III. Experiments

The KITTI dataset was used to train both HOG+SVM detector and YOLO detector. Geiger et al. [19] first introduced KITTI dataset because it was noticed that the datasets used in research are less complex than the real-life case. KITTI dataset contains data acquired by driving around Karlsruhe city in Germany. It contains lifelike situations that any vision module in the autonomous driver or ADAS should handle.

For training and testing, KITTI frames are split into two groups. Each group contains half the images provided by KITTI for training. Using the ground truth of the dataset,

TABLE I
NUMBER OF IMAGES USED IN TRAINING AND TESTING FOR EACH VEHICLE CLASS

	Back class	Side class	Back/Side
Training	2382	1201	1801
Testing	10226	18405	18401

TABLE II
TRUE POSITIVE RATE, TRUE NEGATIVE RATE, AND AREA UNDER THE CURVE OF NONLINEAR CLASSIFIERS FOR VEHICLE'S REARVIEW CLASS

	Single HOG	compHOG	compHOG/Max	compHOG/Avg
TPR	88.11%	89.56%	89.32%	88.47%
TNR	92.89%	92.55%	92.65%	93.59%
AUC	0.9792	0.9814	0.9805	0.9804

vehicles with no occlusion are extracted from the training and testing groups. These samples will work as positive training and verification samples. Vehicles are visually different according to their viewpoint from the ego vehicle. They were divided based on their aspect ratio into three different classes. The first class contains the rear view vehicles. The second class contains the side view vehicles and the third class contains vehicle at which both rear and side of the vehicles are seen by the ego vehicle.

Different HOG variant features calculated for the samples and used to train linear and nonlinear SVM classifiers. Gaussian was used as a kernel for nonlinear SVM. On the other hand, the same images were used in YOLO training. The number of images used for training of each class is shown in table I.

A. Testing on Back class

Linear and nonlinear SVM classifier for vehicle rearview detection are tested using 1657 vehicle sample and 10113 non-vehicle sample. To compare the performance of the classifiers, ROC curve is drawn for each classifier by setting ascending threshold values for the classifier confidence level. As shown in the ROC curve (figure 3)the nonlinear group of classifiers has higher Area Under Curve (AUC) values which indicate better performance than that of linear classifiers. It's noticed also that the performance of the classifiers based on compass HOG exceeds that of the convenient HOG, The compHOG classifier shows the best performance followed by the compass HOG with maximum gradient selection. The true positive rate (TPR), true negative rate (TNR), and AUC for nonlinear classifiers are shown in table II. The YOLO classification results of the same class are shown in table III. The results are for a different number of training iterations. It's shown that the best TPR achieved is 72.12% and the best TNR is 94.67% , however, these results can not be achieved together as they belong to different training iterations.

B. Testing on Side class

The classifiers trained to detect the side view of the vehicles are tested using 583 positive test sample and 13893 negative samples. As shown in the ROC curve (figure 4), the classifiers show close performance, however, the AUC values show that the classifiers based on compass HOG have higher AUC than the conventional one. The best performance achieved by nonlinear SVM based on compass HOG classifier shows a true positive rate of 87.31 % and true negative rate of 97.99 % with AUC equals to 0.9881. Table IV shows a comparison between the performance of different HOG variants classifiers. The YOLO classification results for the same class are shown in

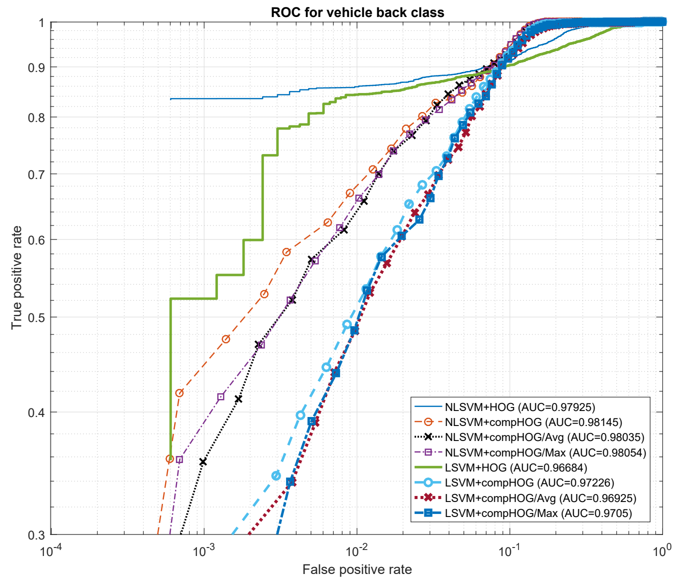

Fig. 3. ROC curve for compared classifiers in the vehicle back class

TABLE III
YOLO CLASSIFICATION RESULTS FOR VEHICLE'S REAR VIEW CLASS

Iterations	45000	55000	70000
TPR	72.12%	65.24%	60.89%
TNR	91.32%	91.93%	94.67%

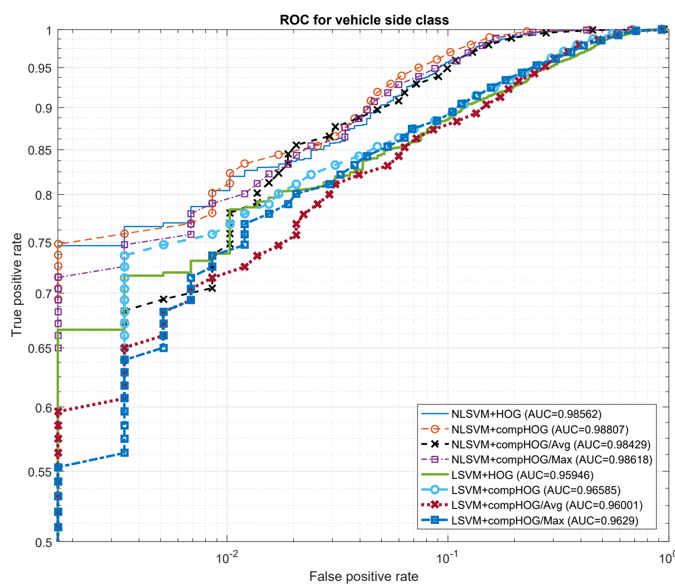

Fig. 4. ROC curve for compared classifiers in the vehicle side class

TABLE IV
TRUE POSITIVE RATE, TRUE NEGATIVE RATE, AND AREA UNDER THE CURVE OF NONLINEAR CLASSIFIERS FOR VEHICLE'S SIDE CLASS

	Single HOG	compHOG	compHOG/Max	compHOG/Avg
TPR	87.14%	87.31%	85.76%	83.53%
TNR	97.03%	97.99%	97.93%	98.23%
AUC	0.9856	0.9881	0.9862	0.9843

TABLE V
YOLO CLASSIFICATION RESULTS FOR VEHICLE'S SIDE CLASS

Iterations	45000	55000	70000
TPR	92.97%	93.14%	90.91%
TNR	93.75%	94.1%	95.61%

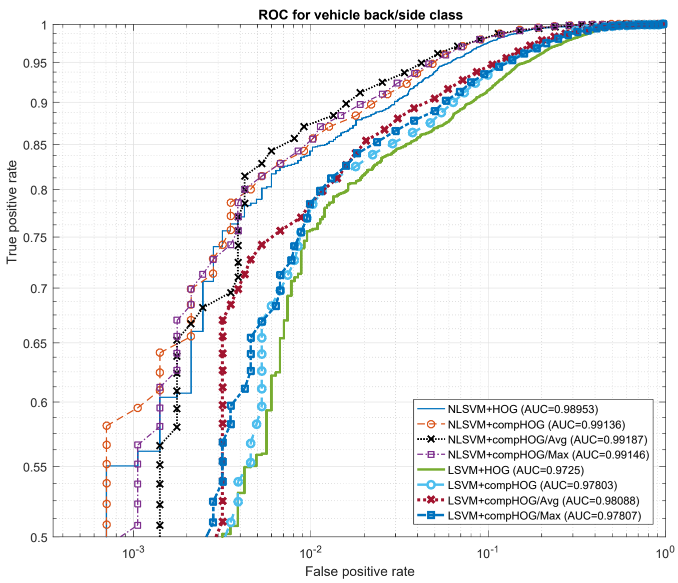

Fig. 5. ROC curve for compared classifiers in the vehicle back/side class

table V. Compared with the HOG+SVM classifier, YOLO's TPR rate is higher with the average percentage of 92.3 %, on the other hand, the TNR is lower with an average percentage of 94.5 %. This indicates a higher vehicle detection chance but at the same time a high number of false detection.

C. Testing on back/side class

The classifiers trained to detect the side view of the vehicles are tested using 2837 positive test sample and 10334 negative samples. As shown in the ROC curve (figure 5), all the classifiers based on compass HOG introduced better performance than the conventional HOG. However, the difference in performance decreased in the nonlinear classifier, but the compass HOG variants still outperforming the conventional one. As been noticed before in testing on the vehicle rear class, the average performance of the nonlinear classifiers is higher than that of the linear SVM classifier. The best performance achieved by nonlinear SVM based on compass HOG classifier shows a true positive rate of 92.92 % and true negative rate of 97.26 % with AUC equals to 0.9919. A comparison between different nonlinear classifiers is shown in table IV. Table VII shows the YOLO's results. Both TPR and TNR are lower than the nonlinear SVM classifiers.

TABLE VI
TRUE POSITIVE RATE, TRUE NEGATIVE RATE, AND AREA UNDER THE CURVE OF NONLINEAR CLASSIFIERS FOR VEHICLE'S BACK/SIDE CLASS

	Single HOG	compHOG	compHOG/Max	compHOG/Avg
TPR	91.86%	92.35%	92.63%	92.92%
TNR	96.57%	97.21%	97.21%	97.26%
AUC	0.9895	0.9914	0.9915	0.9919

TABLE VII
YOLO CLASSIFICATION RESULTS FOR VEHICLE'S BACK/SIDE CLASS

Iterations	45000	55000	70000
TPR	88.72%	85.13%	81.81%
TNR	91.63%	92.36%	94.36%

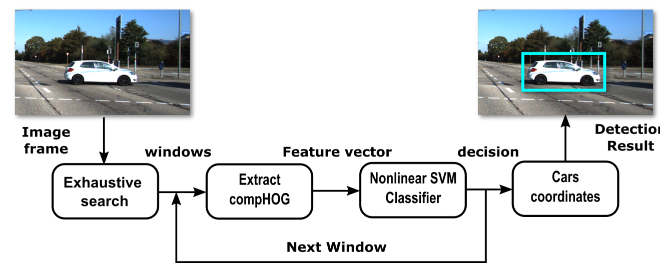

Fig. 6. Vehicle detection system overview

D. Vehicle detection

In this section, A vehicle detection system based on the best performing classifier in each class is introduced. An overview of the detection system is shown in figure 6. The input to the system is the image frame, then the exhaustive search algorithm is applied to check the whole image. A sliding window with different sizes scans the image frame and provides input for the classifier. Feature extraction is performed on each window, then the feature vector corresponding to each window is checked by the classifier to determine its class. The bounding boxes of the detected vehicles are recorded. Non-maximum suppression (NMS) is used to remove overlapping detected windows and make only one bounding box around the detected object. The algorithm of the vehicle detection system is illustrated in algorithm 1. Samples for vehicle detection are shown in figure 7, the results show that the detection algorithm is effective in detecting vehicles with near and medium distance range, which is the range covered by the sliding windows used in detection, also the detector shows robustness in vehicle detection of different shapes, colors and at different lighting conditions.

IV. CONCLUSION

The modification introduced to the HOG feature increased its discriminative power. Compass HOG feature in combination with nonlinear SVM was proven to have better classification performance than both conventional HOG and YOLO object detector. A vehicle detection system based on compass

Algorithm 1 Vehicle detection algorithm

Inputs: Frame **I** of height **H** and Width **W**, **windowHeights**, **aspectRatios**, and **step**
Output: detectedCars: array of bounding boxes

for each frame I **do**
  for each **windowHeight** in **windowHeights do**
    for each **aspectRatio** in **aspectRatios do**
      windowWidth $\leftarrow$ **windowHeight*aspectRatio**
      $\mathbf{x} \leftarrow 0, \mathbf{y} \leftarrow 0$
      while x $\leq$ **W - windowWidth do**
        while y $\leq$ **H - windowHeight do**
          window$\leftarrow$ I(**y**,**x**,**windowHeight**,**windowWidth**)
          v $\leftarrow$computeCompHOG(**window**)
          o $\leftarrow$NLSVMClassifier(**v**)
          if o $==$ *car* **then**
            Add **window** to **detectedWindows**
          end if
          y $\leftarrow$ **y** + **step**
        end while
        $\mathbf{x} \leftarrow \mathbf{x} +$ **step**
      end while
    end for
  end for
  detectedCars $\leftarrow$ NMS(**detectedWindows**)
end for

Fig. 7. Samples of vehicle detection results on KITTI frames

HOG feature was introduced and applied on KITTI dataset. Although the exhaustive search technique is computationally expensive, it provides an extensive testing for the detection system. For realtime implementation, another hypothesis generation technique should be used to reduce the number of windows to be checked by the classifier in each frame. GPU implementation of the proposed technique as well as involving an object tracking approach will accelerate the process to achieve realtime performance.

References

[1] "https://ourworldindata.org/causes-of-death#road-incidents."

[2] "ITF, Alcohol-Related Road Casualties in Official Crash Statistics, https://www.itf-oecd.org/alcohol-related-road-casualties-official-crash-statistics."

[3] "ITF, Road Safety Annual Report 2018, https://www.itf-oecd.org/road-safety-annual-report-2018."

[4] 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, Jun. 2005, pp. 886–893 vol. 1.

[5] L. Mao, M. Xie, Y. Huang, and Y. Zhang, "Preceding vehicle detection using Histograms of Oriented Gradients," in *2010 International Conference on Communications, Circuits and Systems (ICCCAS)*, Jul. 2010, pp. 354–358.

[6] J. Arróspide, L. Salgado, and M. Camplani, "Image-based on-road vehicle detection using cost-effective Histograms of Oriented Gradients," *Journal of Visual Communication and Image Representation*, vol. 24, no. 7, pp. 1182–1190, Oct. 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1047320313001478

[7] G. Yan, M. Yu, Y. Yu, and L. Fan, "Real-time vehicle detection using histograms of oriented gradients and AdaBoost classification," *Optik - International Journal for Light and Electron Optics*, vol. 127, no. 19, pp. 7941–7951, Oct. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0030402616305423

[8] J. Zhang, K. Huang, Y. Yu, and T. Tan, "Boosted local structured HOG-LBP for object localization," in *CVPR 2011*, Jun. 2011, pp. 1393–1400.

[9] M. Cheon, W. Lee, C. Yoon, and M. Park, "Vision-Based Vehicle Detection System With Consideration of the Detecting Location," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1243–1252, Sep. 2012.

[10] J. Kim, J. Baek, and E. Kim, "A Novel On-Road Vehicle Detection Method Using pi-HOG," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3414–3429, Dec. 2015.

[11] P. H. Kassani, J. Hyun, and E. Kim, "Application of soft Histogram of Oriented Gradient on traffic sign detection," in *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Aug. 2016, pp. 388–392.

[12] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, Dec 1989.

[13] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *CoRR*, vol. abs/1311.2524, 2013. [Online]. Available: http://arxiv.org/abs/1311.2524

[14] R. B. Girshick, "Fast R-CNN," *CoRR*, vol. abs/1504.08083, 2015. [Online]. Available: http://arxiv.org/abs/1504.08083

[15] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015. [Online]. Available: http://arxiv.org/abs/1506.02640

[16] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," *CoRR*, vol. abs/1612.08242, 2016. [Online]. Available: http://arxiv.org/abs/1612.08242

[17] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995. [Online]. Available: https://link.springer.com/article/10.1023/A:1022627411411

[18] H. T. Niknejad, A. Takeuchi, S. Mita, and D. McAllester, "On-Road Multivehicle Tracking Using Deformable Object Model and Particle Filter With Improved Likelihood Estimation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 748–758, Jun. 2012.

[19] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 3354–3361.