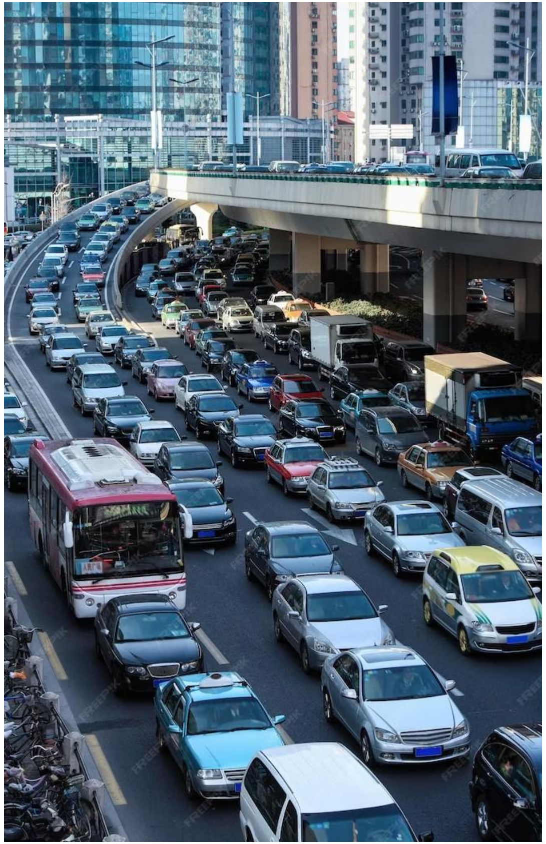


Enhancing Vehicle Detection Accuracy using Linear SVM Classifier and HOG Features

Introduction

In this presentation, we will explore the concept of enhancing vehicle detection accuracy using a Linear SVM Classifier and HOG Features. We will discuss the importance of accurate vehicle detection in various applications and the challenges faced in achieving high accuracy. We will also delve into the fundamentals of Linear SVM Classifier and HOG Features and how they can be leveraged to improve detection performance.

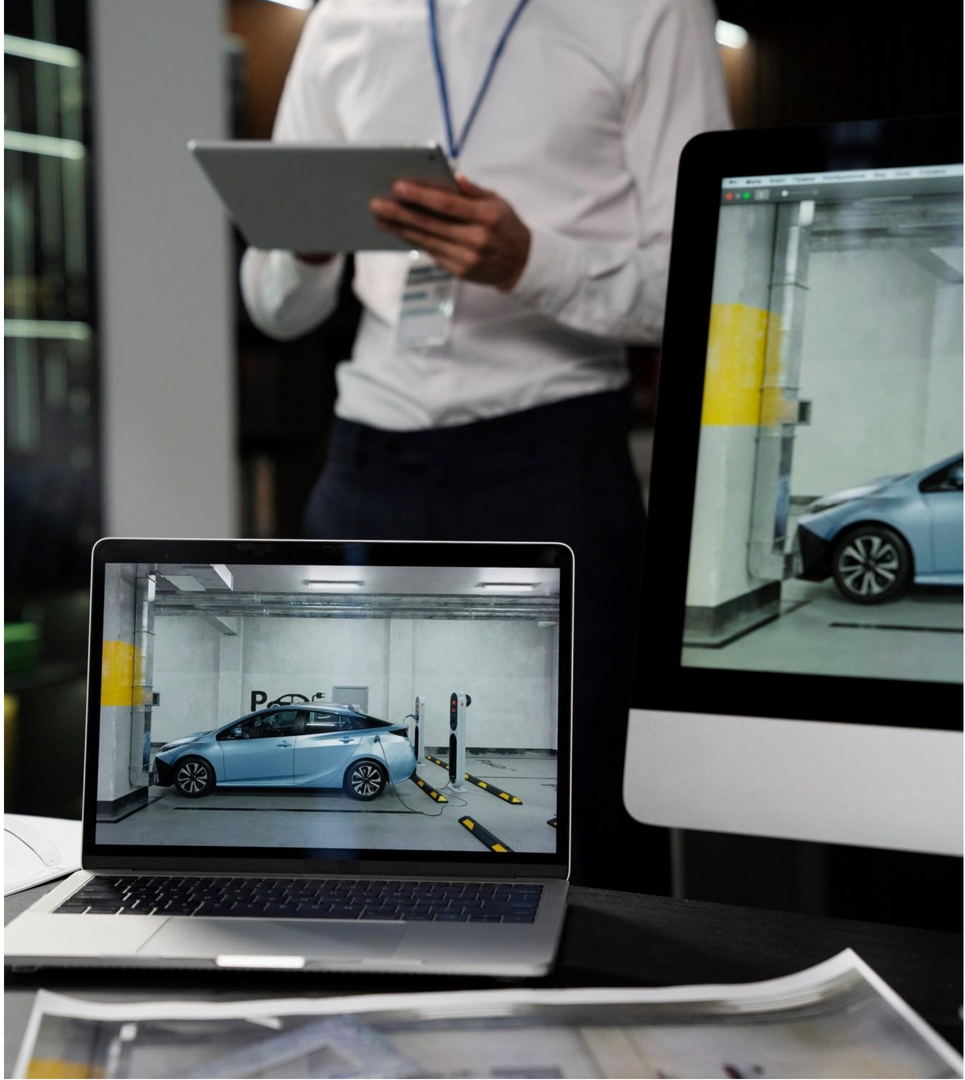
Importance of Vehicle Detection

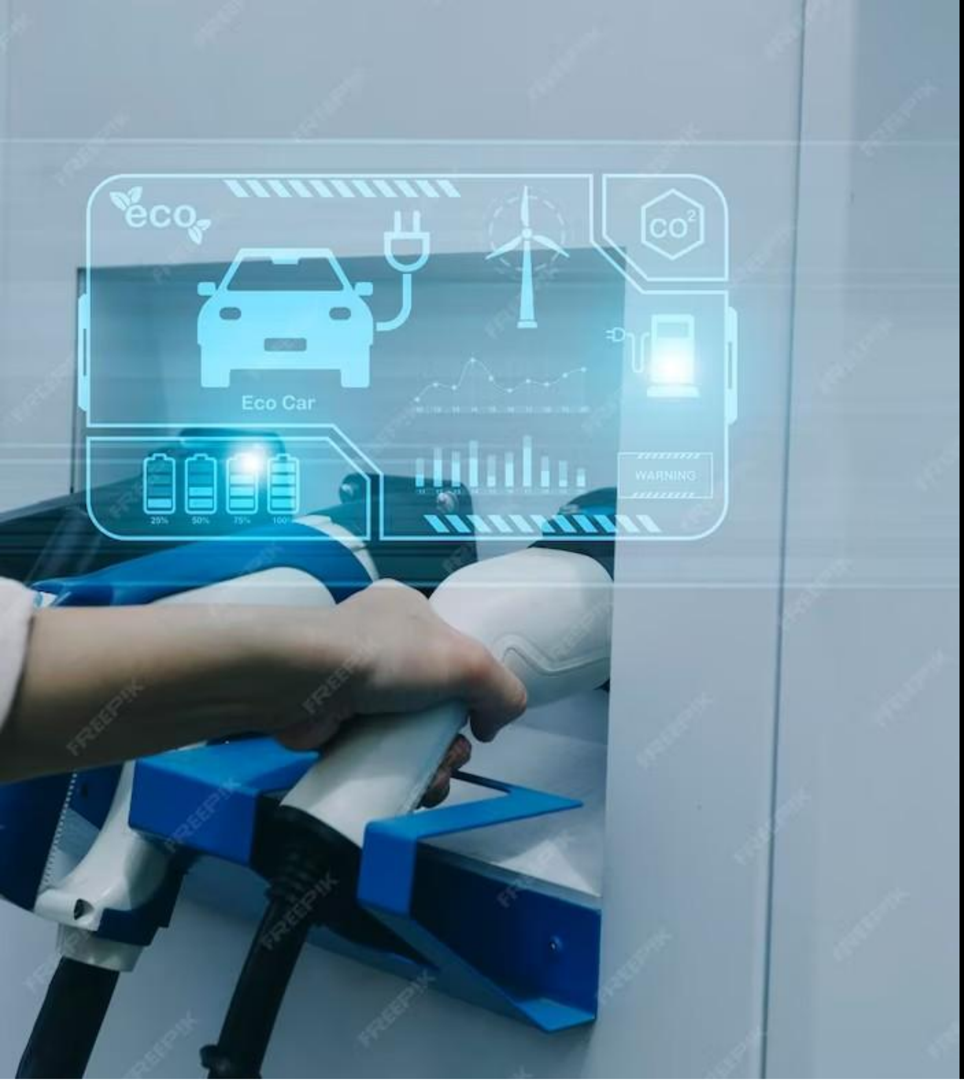
Accurate vehicle detection plays a crucial role in numerous applications such as traffic surveillance, autonomous driving, and parking management. Reliable detection enables efficient traffic flow management, enhanced safety measures, and improved navigation systems.

However, achieving high accuracy in vehicle detection is challenging due to varying lighting conditions, occlusions, and complex backgrounds. In this presentation, we will explore techniques to overcome these challenges.

Linear SVM Classifier

A Linear SVM Classifier is a powerful machine learning algorithm that can be used for binary classification tasks. It works by finding an optimal hyperplane that separates the data points into different classes. In the context of vehicle detection, the Linear SVM Classifier can be trained to distinguish between vehicles and non-vehicles based on a set of carefully crafted features. This enables accurate classification of vehicle regions in an image.





HOG Features

HOG (Histogram of Oriented Gradients) is a popular feature descriptor used in computer vision for object detection. It captures the local shape and appearance of objects by analyzing the distribution of gradient orientations in an image. HOG features are particularly effective in representing the structural characteristics of vehicles, making them well-suited for vehicle detection tasks.

By combining HOG features with a Linear SVM Classifier, we can significantly enhance detection accuracy.

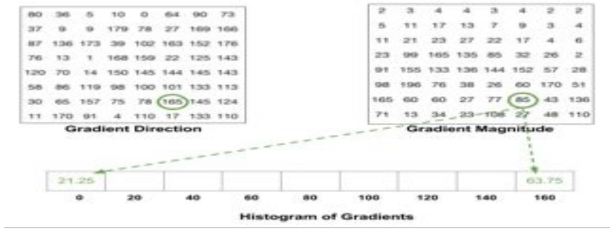
Enhancing Vehicle Detection Accuracy

By leveraging the power of a Linear SVM Classifier and HOG Features, we can enhance vehicle detection accuracy. The combination of these techniques allows us to effectively capture the discriminative information required for accurate classification of vehicles. Through proper training and parameter tuning, we can achieve robust and reliable vehicle detection performance, even in challenging scenarios. In the following slides, we will explore experimental results and real-world applications of this approach.

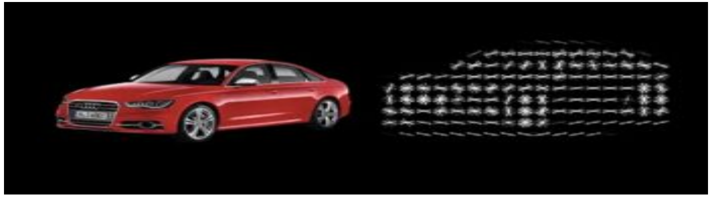
LITERARY REVIEWS

1. HOG (Histogram of Oriented Gradients) functions are essential for car detection as they capture localized intensity gradients or edge directions, providing information about object appearance and shape.

2. Support Vector Machines (SVMs) are effective for solving various problems quickly, but they require a substantial number of input examples, resulting in large matrix calculations. SVMs are computationally efficient for classification but demand significant time for kernel parameter regularization.
3. To achieve car detection, a combination of HOG descriptors and SVM classification is employed, leveraging the strengths of both techniques to efficiently and accurately identify cars in images.

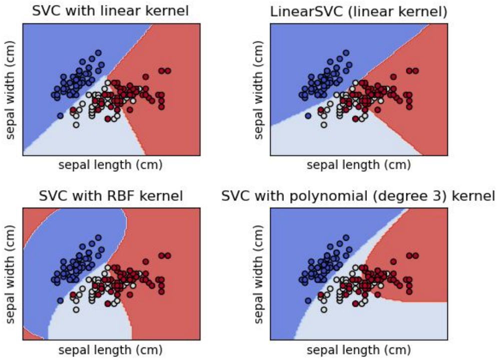


1.3 allotting bins

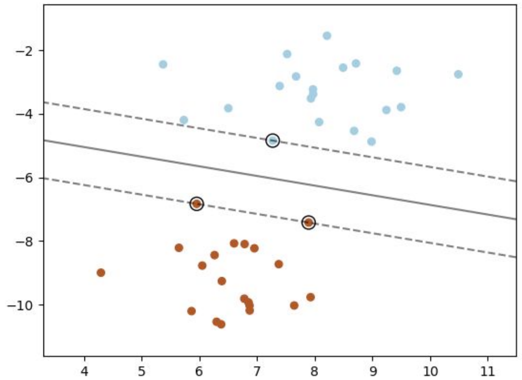


1.4 Histogram of Oriented Objects

Souce: S. Bougharriou, F. Hamdaoui and A. Mtibaa, "Linear SVM classifier based HOG car detection," 2017 18th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA), Monastir, Tunisia, 2017, pp. 241-245, doi: 10.1109/STA.2017.8314922.



Different SVM Library options



Linear SVM Representation

Source: https://scikit-learn.org/stable/modules/svm.html

Support Vector Machine:

A support vector machine constructs a hyper-plane or set of hyperplanes in a high or infinite dimensional space, which can be used for classification, regression or other tasks. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training data points of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier. The figure below shows the decision function for a linearly separable problem, with three samples on the margin boundaries, called “support vectors”.

SVM is much more accurate classifier as compared to its counterparts like random forest, Decision Tree etc, upto 20% more accurate and with no additional time lags.

Source:KMN Syed Ali Fathima1 DR.K. Merriliance, ”SVM with Hog Based on Classification Using Vehicle’s Different Viewpoints “

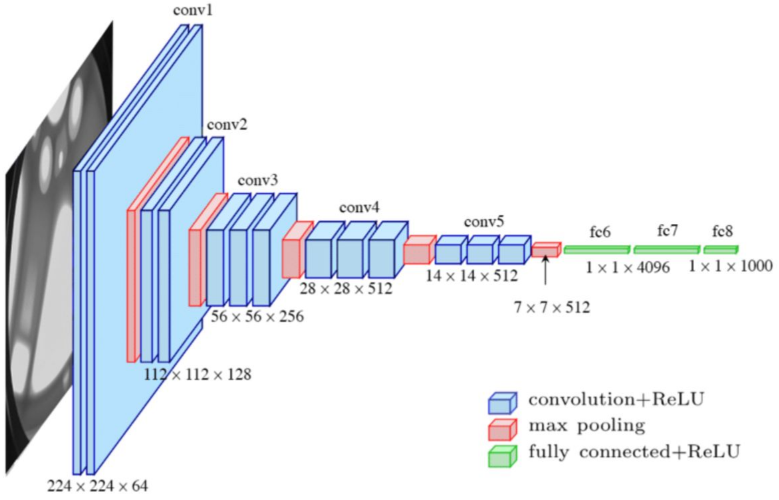
Raw Image Detection classification

1. Object detection is a crucial task in systems like real-time surveillance cameras, smartphones, advanced driver assistant systems (ADAS), and unmanned aerial vehicles (UAV).

2. Object detection algorithms can be categorized into hand-crafted feature-based and convolutional neural network (CNN)-based approaches, with CNNs offering superior detection performance.

3. CNN-based systems tend to be computationally and memory-intensive, which can lead to increased power consumption, making them less efficient than hand-crafted feature-based methods in terms of power usage.

4. When applying a HOG-based detector designed for color images to raw images, both false-negative and false-positive errors increase, highlighting the need for a dedicated HOG algorithm optimized for raw images for efficient and accurate object detection.



4. Recent research has shown that specific stages in the Image Signal Processor (ISP) pipeline, such as gamma correction and demosaicing, are critical for vision tasks, making raw images suitable for tasks like edge detection, feature map extraction in CNNs, and HOG+SVM-based detectors.

5. To overcome frame-rate limitations, a partially parallel processing architecture has been proposed, which integrates an image sensor and processor for efficient object detection.

6. Co-designing the ISP and detection algorithm at the system level can significantly enhance the efficiency and performance of the detection system without compromising accuracy.

7. The main contributions of this work include the development of an efficient HOGR (Histogram of Oriented Gradients) dedicated for raw Bayer images, a partially parallel architecture synchronized with the image sensor for scalability, and the ability to support various primitives. HOG is known for its ability to capture edge orientations and distribution, which are essential for object detection.

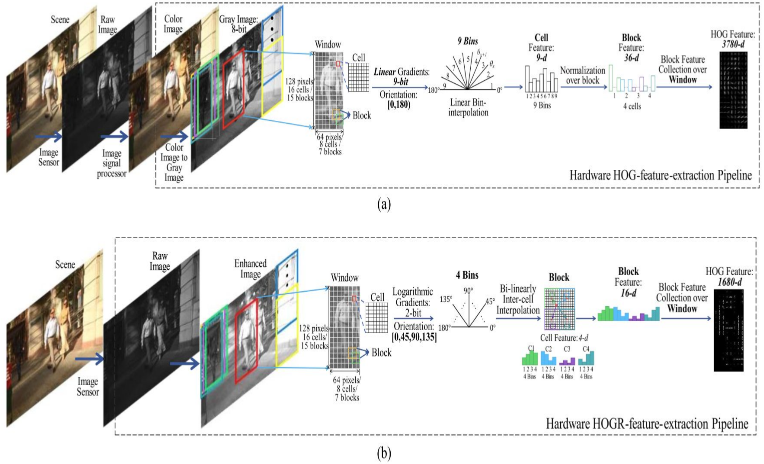


Fig. 4. (a) Conventional color image-based hardware HOG-feature-extraction pipeline, extracting *3780-d* features and (b) the proposed raw image-based approach, extracting *1680-d* features. The C1, C2, C3, C4 are the abbreviation of cell 1, cell 2, cell 3, and cell 4, respectively. The four adjacent cells make up one block.

Source: X. Zhang, L. Zhang and X. Lou, "A Raw Image-Based End-to-End Object Detection Accelerator Using HOG Features," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 69, no. 1, pp. 322-333, Jan. 2022, doi: 10.1109/TCSI.2021.3098053.

FLOATING POINT HOG IMPLEMENTATION FOR REAL-TIME MULTIPLE OBJECT DETECTION

1. Object detection and localization in video streams is a critical requirement for various vision systems. This article presents a design embedded in a reconfigurable device that utilizes Histogram of Oriented Gradients (HOG) for feature extraction and Support Vector Machine (SVM) classification to detect multiple objects. Achieving superior accuracy, the implementation uses single-precision 32-bit floating-point values throughout all image processing stages and is fully pipelined, eliminating the need for external memory.

2. While the use of floating-point values on FPGA-based systems is uncommon, previous implementations have shown their potential to compete with CPUs and GPUs in terms of floating-point computing performance, with some achieving data throughputs as high as 160 GFLOPS in radar applications.

3. To handle multiple objects, different instances of SVM classifiers are employed, each processing the image, dividing it into cells and blocks, and computing feature vectors. This approach enables the detection of various objects, such as humans, heads (in cases of partial occlusion), and bicycles, making it a versatile system for object detection in video streams.

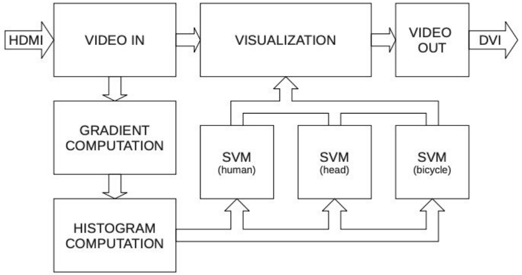


Fig. 3. Classification system

Source: M. Komorkiewicz, M. Kluczewski and M. Gorgon, "Floating point HOG implementation for real-time multiple object detection," 22nd International Conference on Field Programmable Logic and Applications (FPL), Oslo, Norway, 2012, pp. 711-714, doi: 10.1109/FPL.2012.6339159.

Multi-vehicle detection algorithm through combining Harr and HOG features

1. A two-step detection algorithm is proposed to enhance the performance of multi-vehicle target detection and tracking in complex urban environments. The algorithm combines Haar and Histogram of Oriented Gradients (HOG) features, leveraging the descriptive power of HOG features and the use of Harr features along with an SVM classifier to extract regions of interest (ROI). This approach improves operational efficiency while remaining adaptive to complex environments.

2. The first step involves selecting Harr feature prototypes and setting system parameters. Eight basic feature types, such as edge and line features, are defined. The sample size is 24x24, the step size is 10, the scale factor is 1.2, and there are three classifier series. This step is crucial for defining templates and model parameters.

3. The algorithm utilizes gray images for ease of computation and applies Gamma filtering to mitigate the effects of light variations, shadows, and image lighting. Additionally, it incorporates multi-scale scaling to account for the diverse positions, sizes, and perspectives of the targets within images. Integral Image Calculation further streamlines the complex Harr feature extraction process by efficiently calculating corner coordinates based on the integral image.

Souce: Yun Wei, Qing Tian, Jianhua Guo, Wei Huang, Jinde Cao, Multi-vehicle detection algorithm through combining Harr and HOG features, Mathematics and Computers in Simulation,2019, Pages 130-145, ISSN 0378-4754

Object Detection using Histogram of Oriented Gradients

1) Precision detection is required for solving recognition. Background subtraction is widely used as a technique for detecting moving objects.
2) This paper works on reducing the matching speed for the images.
3) 3 different procedures were proposed: pixel matching, cell matching, and Block matching.
4) In this paper, we presented three methods to speed up human detection using Histogram of Oriented Gradients. The pixel matching and the cell matching speeded up, but there was a bad detection rate. The block matching succeeded in shortening the matching time and keeping a good detection rate.
5) THe above-proposed method responds to scaling. But if scale invariant feature transformation (SIFT) is coupled with hog, this can help us bring about robust speed-efficient algorithms to detect objects.

Source: Terayama, Masahiro & Shin, Jungpil & Chang, Won-Du. (2009). Object Detection using Histogram of Oriented Gradients.

YOLOv4 Algorithm for Multiple Object Detection in Image and Video Dataset using Deep Learning

1. YOLOv4, a state-of-the-art object detection algorithm, is employed in this work for detecting and localizing multiple objects in images and videos for traffic surveillance applications. It is capable of classifying objects and providing precise annotations.
2. YOLOv4 is an improvement over the YOLOv3 algorithm, featuring a CSP darknet-53 classifier and spatial pyramid pooling that connects the YOLOv3 head. This results in high detection accuracy, accurate bounding box positioning, and fast computations, particularly when using multi-scale prediction techniques.

3. The research utilized an NVIDIA GTX 1080Ti GPU, which, while achieving high accuracy and excellent performance across confusion matrix parameters, is associated with a substantial cost. This highlights the need for further research to optimize the approach and reduce the GPU-related expenses.

4. Hardware and software requirements for implementing the YOLOv4 algorithm include a personal computer or desktop with a minimum of 8GB RAM, 500GB ROM, a 64-bit processor, and a GPU. These specifications are essential for ensuring faster computation rates when working with the algorithm.

Source: U. P. Naik, V. Rajesh, R. K. R and Mohana, "Implementation of YOLOv4 Algorithm for Multiple Object Detection in Image and Video Dataset using Deep Learning and Artificial Intelligence for Urban Traffic Video Surveillance Application," 2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT), Erode, India, 2021, pp. 1-6, doi: 10.1109/ICECCT52121.2021.9616625.

Object Detection using Haar Cascade Classifier

1) Haar cascade is a machine learning–based approach where a bunch of positive and negative images is used to train the classifier. Since the system's accuracy is high, Haar Cascade is a superior technique for object detection
2) With 20 % more precision than other classifiers
3) The main advantage of using Haar features as classifiers in object detection is their ability to quickly and accurately identify objects in an image.
4) Haar features are simple, rectangular features that can be computed efficiently, making them ideal for real-time applications
5) During each stage, the classifier checks for a specific set of features and only passes on the image to the next stage if it meets the necessary criteria. This helps to reduce the number of false positives and increases the overall speed and accuracy of the detection process.

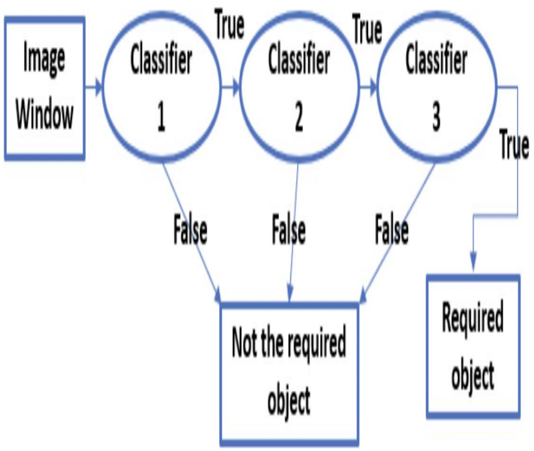
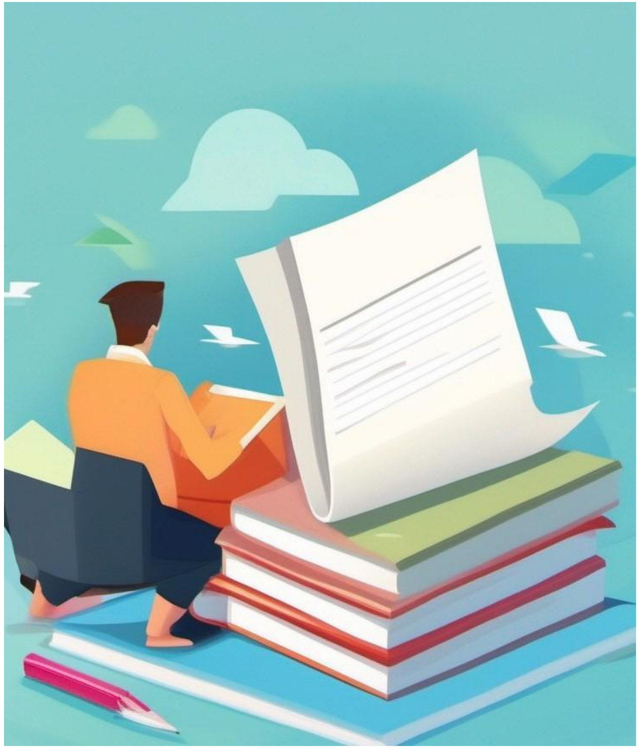


Fig. 2. Working of Cascade Classifiers

Source: Yun Wei, Qing Tian, Jianhua Guo, Wei Huang, Jinde Cao, Multi-vehicle detection algorithm through combining Harr and HOG features, Mathematics and Computers in Simulation,2019, Pages 130-145, ISSN 0378-4754

Methodology..



To develop a solution, we followed a two-step approach. Firstly, we extensively studied research papers on hog and SVM. Secondly, we utilized a dataset containing vehicle and non-vehicle samples. This approach allowed us to devise an effective solution.

Dataset and Data Preprocessing Details:

For our SVM training, we utilize the GTI vehicle image database, a collection comprising both positive (cars) and negative (non-car) samples. The dataset specifics are as follows:

- Number of Car Images: 8792
- Number of Non-car Images: 8968
- Image Shape: (64, 64, 3)

This dataset contains 8792 car images and 8968 non-car images, all of which are of size 64 pixels by 64 pixels with 3 color channels (RGB). These images serve as the basis for training our Support Vector Machine (SVM) for vehicle detection, where the positive samples represent cars, and the negative samples depict objects other than cars.

Methodology: Feature Extraction and Combination

Binned Color Features: The function bin_spatial resizes images to a defined size, converting them to feature vectors by flattening them.

Color Histogram Features: The color_hist function computes histograms for each color channel and concatenates them to create a feature vector capturing color distribution.

Histogram of Oriented Gradients (HOG): get_hog_features extracts HOG features from image channels, olering insights into local gradient directions for object shape representation.

Feature Integration: The extract_features method combines spatial, color histogram, and HOG features, allowing for a comprehensive feature set. It converts images to the specified color space, then extracts spatial, color histogram, and HOG features per image, combining them into a single feature vector.

Iterative Image Processing: This methodology iterates through a list of images, transforming them to a chosen color space and extracting various features, including spatial, color histogram, and HOG features, ultimately creating feature vectors for each image

Methodology: Feature Extraction and Prediction

Image Preprocessing and Feature Extraction:

- The find_cars function conducts a multi-scale search by resizing the image and preparing it for feature extraction, including color space conversion to YCrCb and scaling for diIerent window sizes.
- Utilizes Histogram of Oriented Gradients (HOG) by computing features for individual color channels and extracting HOG features for each sliding window to capture the local shape and gradient information.

Feature Combination and Classification:

- Extracts color and spatial features using functions like bin_spatial and color_hist, followed by combining these features with the HOG features.
- Scales the extracted features using X_scaler and feeds them into the trained SVM (svc) for making predictions on whether a car is present within each sliding window.

Detection and Visualization:

- If specified, it visualizes the bounding boxes for detected regions, indicating potential car positions in the image. Detected areas are outlined with red rectangles.

Output:

- The function returns a list of bounding box positions representing the detected regions, an annotated image with marked detections, and, if enabled, a list of box positions for visualization purposes.

Parameters and Image Processing

Parameters Definition:

- Specifies parameters such as spatial, histogram bins, colorspace, orientation, pixel and cell per block size, spatial size, heat threshold, HOG channel, and multi-scale window configurations to optimize the detection process.

Image Processing:

- Resizes the image to 1280x720.
- Utilizes a subsampled Histogram of Oriented Gradients (HOG) window method (find_cars) to identify potential detections within specified sliding windows based on the defined parameters.
- Integrates previous detection history and applies heat mapping to smooth out detections and generate a heatmap from these detections.

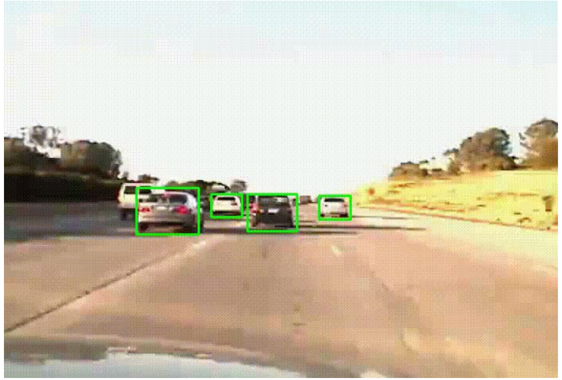
Refinement of Vehicle Detection:

- Processes the heatmap using a labeling function to identify and isolate individual car regions according to the accumulated
- detection "heat." Draws bounding boxes around the identified car areas for clearer visualization..

Additional Image Enhancement::

- Specifies that the image has been scaled by a factor of 3 and processed
  with Gaussian filtering using a 3 by 3 kernel and a standard deviation of 0 to refine the image quality for better analysis and detection accuracy.

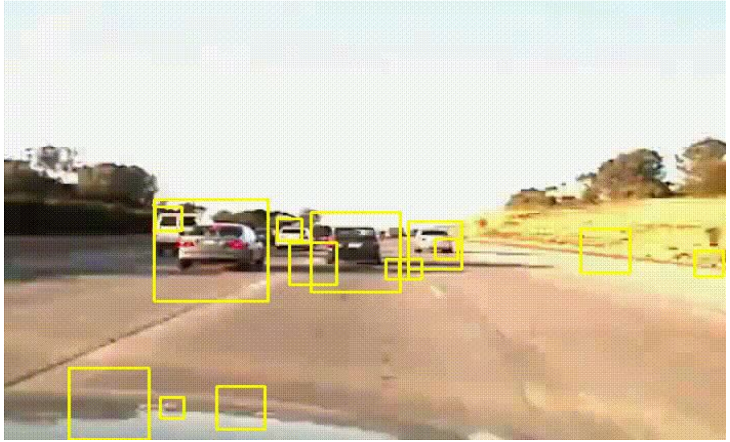
Results.....



Ground Truth

HOG+SVM

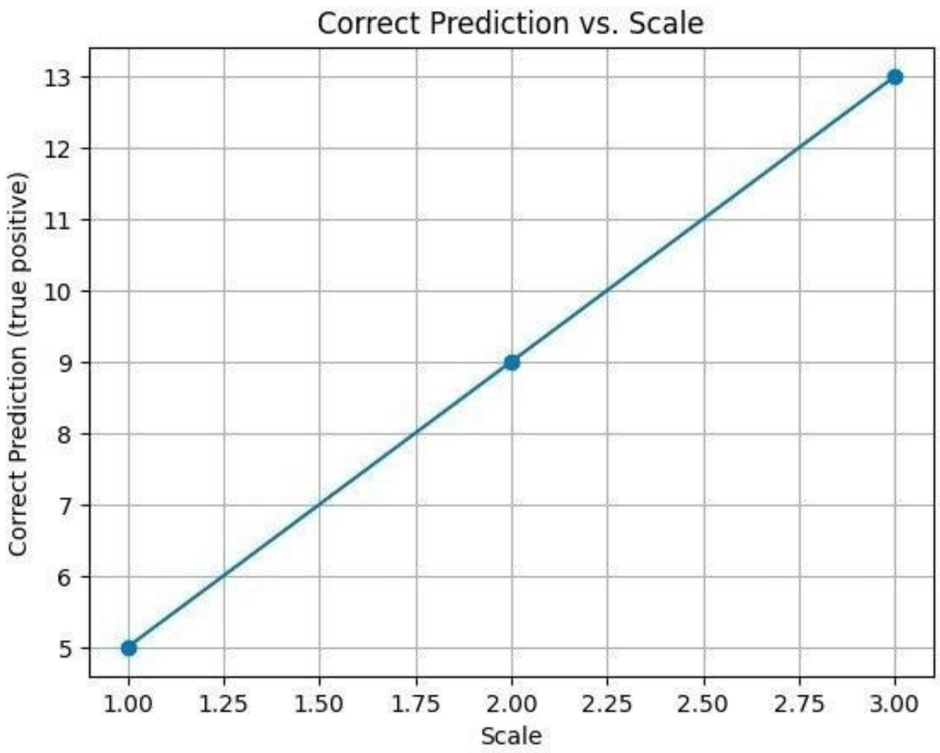
YOLO



Har
Cascades

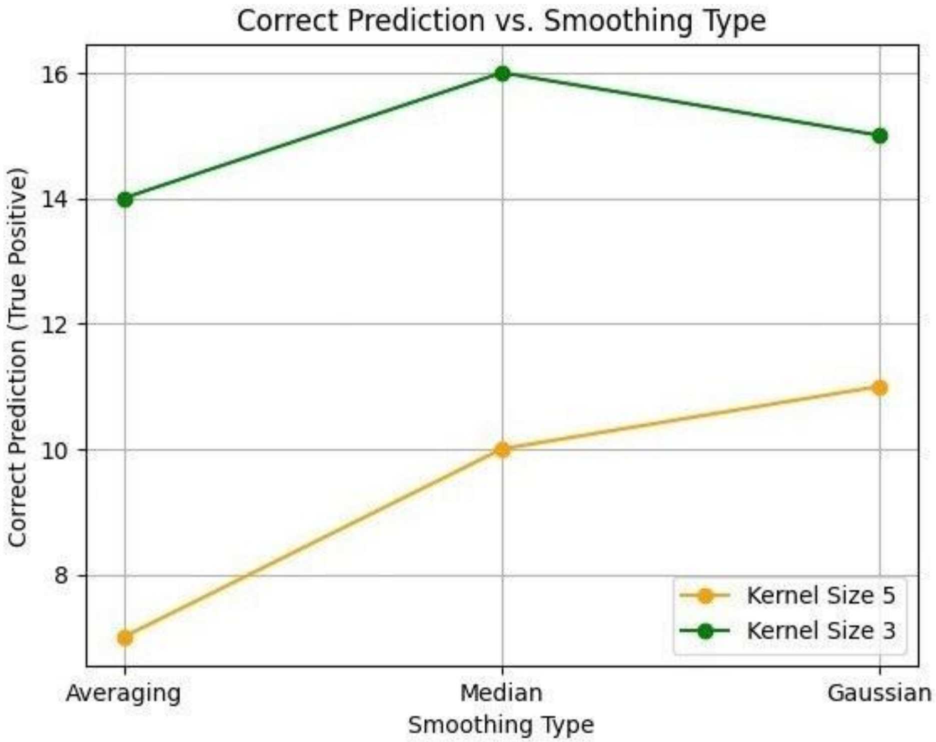
Effect on scaling

Scale	Correct Prediction (true positive)
1	5
2	9
3	13



Effects of soothning:

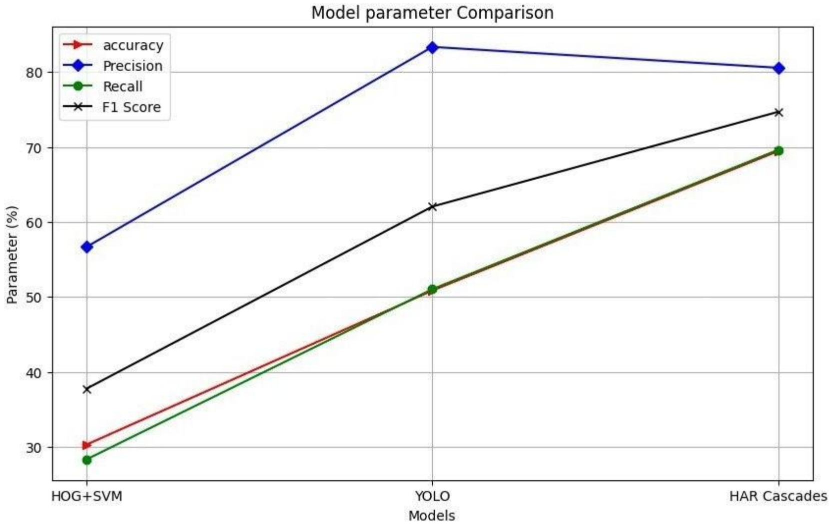
Scale	Correct Prediction (true positive)
1	5
2	9
3	13



Comparison to other Models:

Comparison to other Models:

	HOG+SVM	YOLO	HAR Cascades
Accuracy	**30.30%**	**50.84%**	**69.43%**
Precision	**56.66%**	**83.33%**	**80.55%**
Recall	**28.33%**	**51.02%**	**69.60%**
F1	**37.76%**	**62.04%**	**74.67%**



Drawbacks…

Low Accuracy of Models:

- Despite various available approaches,the models being used lack accuracy.
- One significant bottleneck in achieving higher accuracy is the inadequate size of datasets used for training.

Challenge of Scaled Videos:

- Scaling videos to specific sizes results in significant changes in the dimensions of objects within them.
- This leads to poor accuracy in object detection or recognition.

Future prospect…

Addressing the Issue with SIFT:

- Many research papers acknowledge this issue and propose the use of Scale-Invariant Feature Tracking (SIFT) in combination with HOG + SVM for object detection.
- The plan is to leverage SIFT to address the challenge of scaled videos and improve the accuracy of the model.

Proposed Approach:

- The intended strategy involves integrating SIFT into the existing model's workflow to enhance accuracy.
- SIFT is expected to provide scale-invariant features, allowing for more robust object detection despite variations in object dimensions caused by video scaling.

Conclusion

The fusion of Linear SVM Classifier and HOG Features presents a robust solution for improving vehicle detection accuracy, overcoming challenges in various applications. While deep learning models such as YOLO achieve notably higher accuracy, they require resource-intensive machines, contrasting the efficiency of HOG+SVM, which can operate on low-resource devices like the Raspberry Pi. When appropriately optimized, HOG + SVM can attain up to 80% accuracy on modest devices, offering a viable alternative with competitive speeds compared to deep learning models, indicating promise for practical and cost-effective implementations in real-world scenarios.

Thanks!

Do you have any
questions?

Drumil H ved
Punith B Nayak