

PAPER • OPEN ACCESS

## Vehicle detection using histogram of oriented gradients and real adaboost

To cite this article: D R Sulistyaningrum *et al* 2020 *J. Phys.: Conf. Ser.* **1490** 012001

View the [article online](#) for updates and enhancements.



**IOP | ebooks™**

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

# Vehicle detection using histogram of oriented gradients and real adaboost

**D R Sulistyaningrum<sup>1</sup>, T Ummah<sup>2</sup>, B Setiyono<sup>3</sup>, D B Utomo<sup>4</sup>, Soetrisno<sup>5</sup>, B A Sanjoyo<sup>6</sup>**

<sup>1,2,3,4,5,6</sup>Department of Mathematics, Institut Teknologi Sepuluh Novermber Campus ITS, Sukolilo, Surabaya, 60111, Indonesia

E-mail: dwiratna@matematika.its.ac.id

**Abstract.** The vehicle detection system is an important technology because it has many applications in traffic fields such as traffic monitoring, counting the number of vehicles passing, calculating the speed of an oncoming vehicle, and so on. Histogram Of Oriented Gradients (HOG) is a feature descriptor used for object detection. HOG describes features based on some local histogram of gradient orientation weighted by gradient magnitude. Real AdaBoost is a learning algorithm which combined weak classifier into a strong classifier that represents the final output of the boosted classifier. This research aims to detect vehicles in the static image using a Histogram Of Oriented Gradients methods and Real Adaboost. The steps of vehicle detection are pra-processing, feature extraction process with HOG, and classification process with Real AdaBoost. The testing result shows that the system can detect vehicles with an accuracy level of 91.78 % from 292 testing images, from 257 images of the vehicle and 35 images of not vehicle.

## 1. Introduction

Object detection is one of the research of the critical objects in computer vision, so it gets much attention in it. The rapid growth of high-powered computers, the increasing number of high-quality and economical cameras, and the increasing research in image analysis have made many researchers interested in searching for object detection algorithms.

Vehicle detection is a difficult task because the background conditions of the vehicle are very complex, and the intensity and color of the vehicle tend to appear to vary [1]. The appearance of objects observed in scenes often changes depending on camera orientation.

Many studies on object detection have been carried out, such as research conducted by Satish Madhogaria and friends entitled Car Detection by Fusion of HOG and Causal MRF. In that study, they introduced a two-stage algorithm to detect cars in aerial images using the HOG and SVM methods [2]. By using the Google Earth Data Set, the study succeeded in detecting cars with a detection rate of 78%. Then another study was conducted by Daisuke Aoki and Junzo Watada with the title Human Tracking Method Based on Improved HOG + Real AdaBoost [3]. They combined the features of a Histogram of Oriented Gradients (HOG) with two-stage boosting and the Real AdaBoost method. Testing uses three different test data patterns. In the first pattern, an image with a simple backlash obtained an error of 17%. The second pattern, with a more complex background, obtained an error of 49.5%. Whereas in the



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](#). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

third pattern with detected human changes, the error is 38.7%. Based on previous research, it was concluded that the combination of the HOG and AdaBoost methods with other methods has a relatively high degree of accuracy for object detection.

Histogram of Oriented Gradients was first introduced by Dalal and Triggs (2005) for human detection [4]. Nevertheless, in its development, research was conducted to detect diverse objects, such as vehicles, human faces, character recognition of human writing, and others. The HOG method as a feature extraction method is superior to other feature extraction methods because HOG operates in a local cell so that the shape of the object can be characterized by the distribution of the local intensity gradient or edge direction [4]. A geometric and photometric transformed image will not affect the HOG value. Furthermore, AdaBoost, as a machine learning method, has been widely used for data classification and object detection because of its strength and efficiency. One variant of AdaBoost is Real AdaBoost, which calculates a weak hypothesis by optimizing the upper limit of training errors so that it converges faster than AdaBoost and computing time is also faster [5,6].

Based on the explanation above, in this work, we have analyzed the performance of the HOG and Real AdaBoost methods for vehicle detection.

## 2. Method

Figure. 1 shows the proposed methods for vehicle detection using a Histogram Of Oriented Gradients (HOG) methods and Real Adaboost. It consists of several processing units: First, image acquisition. Second, pre-processing to enhancement image quality. Third, feature extraction with HOG. Next, classification using Real Adaboost. A detailed description of the scheme is described in the next discussion.

### 2.1. Preprocessing

At preprocessing, there are two processes. First, resizing the image so that the processed image will have smaller dimensions. It makes the feature extraction process, and the classification process runs faster, because the smaller the dimensions, the fewer features are produced. After the resizing process, a gray scaling process is carried out by changing the RGB input image to a grayscale image.

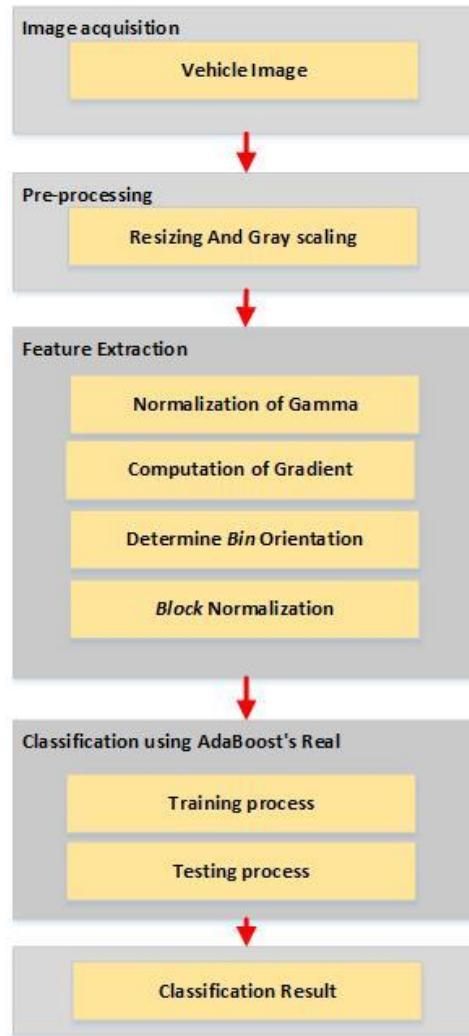
### 2.2. Feature Extraction with Histogram of Oriented Gradient (HOG)

Feature extraction is the process of transformation from a collection of input data, which is provided into a collection of features [7]. In this process, the components in the image are extracted and characterized. Features are individual attributes or properties of an object that are relevant in describing and recognizing objects and can distinguish them from other objects. The features of an object can be recognized based on texture, color, or shape, or a combination of the three. The Histogram of Oriented Gradients (HOG) method is a method used for feature-based extraction.

The histogram of Oriented Gradients (HOG) was introduced by Naveet Dalal and Bill Trigs in 2005. The steps of feature extraction with HOG are as follows [3] [8]:

#### 1. Normalization of Gamma or Square Root and Color

At the gamma normalization,  $\sqrt{p}$  (square-root normalization) or  $\log p$  (p) (gamma normalization) is calculated from each pixel p in the input image. Beside of gamma normalization, color normalization is also performed. RGB images will be processed into grayscale images to facilitate the calculation of image gradients.



**Figure 1.** The Proposed vehicle detection flowchart

## 2. Computation of Gradient

In the computation of gradient, vertical gradient  $f_x(x, y)$  from Equation (1) and horizontal gradient  $f_y(x, y)$  from Equation (2) will be calculated so that for the pixel value of edge image does not change. After getting gradient value, gradient orientation ( $\theta$ ) from Equation (4) will be calculated with magnitude ( $m$ ) from Equation (3) for every pixel of an image.

$$f_x(x, y) = I(x + 1, y) - I(x - 1, y) \quad (1)$$

$$f_y(x, y) = I(x, y + 1) - I(x, y - 1) \quad (2)$$

with  $I(x, y)$  is the intensity value in  $(x, y)$ .

$$m(x, y) = \sqrt{f_x(x, y)^2 + f_y(x, y)^2} \quad (3)$$

$$\theta(x, y) = \tan^{-1} \left( \frac{f_y(x, y)}{f_x(x, y)} \right) \quad (4)$$

### 3. Determine *Bin Orientation*

HOG extraction is a single-window approach, the image is divided into regions called blocks, and at the same time, each block is divided into smaller regions called cells. One histogram per cell is extracted and then gets one descriptor per block and combines them. Each histogram has the same number of the bin, which determines its accuracy. The bin represents the orientation of the gradient (angle) and must be equally placed at  $0^\circ - 180^\circ$  for the "unsigned" gradient or  $0^\circ - 360^\circ$  for the "signed" gradient. One histogram per cell is counted; each pixel in the cell contributes to the addition of the histogram where the histogram depends on the magnitude value associated with bin orientation; this weighting value is called vote.

### 4. *Block Normalization*

Cells are combined to form a block. Overlapping between blocks aims to ensure consistency throughout the image and reduce the influence of local variations. Then a large histogram is formed by combining all histograms created in a block. After being combined, the histogram is normalized with the following Equation (5) [9]:

$$v_n = \frac{v_i}{\sqrt{\|v\|_2^2 + \epsilon^2}} \quad (5)$$

where  $v_n$  is a normalized vector,  $v_i$  is a feature vector associated with a histogram combined for one block, the form of  $\|v\|_2$  is  $\sqrt{\sum_i x_i^2}$  so that  $\|v\|_2^2 = v_1^2 + v_2^2 + \dots + v_{36}^2$  if a block is  $2 \times 2$  cells and  $\epsilon$  a small constant is 1 to avoid division by zero. After that, the histogram for each block is collected to create a feature vector and then placed in a learning algorithm that will decide whether the vehicle is detected or not based on training data.

#### 2.3. Classification using AdaBoost's Real Algorithm

AdaBoost is a machine learning algorithm proposed by Y. Freund and R. Scaphire, which has a high level of detection ratio and easy to implement. Improving weak learner L performance is done by calling iterated training data distribution to find some weak classifier h. Furthermore, it combines weak classifier h into strong classifier H. In each iteration [5], AdaBoost invokes a simple learning algorithm (called the base learner / weak learner), which returns to a classifier, and gives a weight coefficient to the classifier. In this process, AdaBoost increases the weight of misclassified instances and reduces the weight of instances classified correctly by previous weak learners. In the next iteration, AdaBoost focuses on instances that are difficult to classify. Thus, AdaBoost tries to produce a new weak classifier, which is better for predicting samples that the previous weak classifier had poor performance. The final classification will be decided by the substantial "vote" of weak classifiers. The smaller the error of the weak classifiers, the higher the weight of the last vote. The results of the vote weak classifier are added together and taken signum (signum function is a mathematical function to get the real value) as a predicted label (-1 or 1) to be given to instance  $x_i$ , and magnitude  $|h(x)|$  as "confidence" (confidence) in predictions this. If the value of  $|h(x)|$  near zero, then the confidence of the prediction is low, whereas if the value of  $|h(x)|$  far from zero, the confidence of predictions is high. AdaBoost has binary outputs 1 and -1, while Real AdaBoost is an improved version of AdaBoost, where its weak learner is allowed to have real number  $f_m(x) \in R$  output.

Real AdaBoost algorithm is as follow [3] [9]:

1. Given image sample  $S = \{(x_1, y_1), \dots, (x_m, y_m)\}$ , weak classifier space  $h$ . In set  $S$ ,  $x \in X$  is a vector sample,  $m$  are many samples,  $y = \pm 1$  is a class label which related to training sample  $x_i$ , where if  $x_i$  is a vehicle image, then  $y = +1$ , and  $y = -1$  otherwise.
2. Initialize sample weight  $D_t(i) = \frac{1}{m}, i = 1, \dots, m$ .
3. Take weak learner, for  $t = 1, \dots, T$  ( $T$  is an amount of chosen weak learner):
  - a. For every weak classifier  $h$  do:
    - i. Divide vector sample  $X$  to  $x_1, \dots, x_n$ .

ii. With training sample weight  $D_t$ , count the probability of distribution density:

$$\begin{aligned} W_l^j &= P(x_i \in x_j, y_i = l) \\ &= \sum_{i:x_i \in x_j \wedge y_i = l} D_t(i) \end{aligned}$$

with  $l = \pm 1$ .

The probability of distribution density  $W_l^j$  generated by counting sample training features  $x_i$  and substituting training sample weight  $D_t(i)$  to the number of BIN  $j$  from the one-dimensional histogram related to the feature value.

iii. Set output from *weak classifier*

$$h_t(x) = \frac{1}{2} \ln \left( \frac{W_{+1}^j + \varepsilon}{W_{-1}^j + \varepsilon} \right)$$

with  $\varepsilon$  is a small positive constant = 0.0000001.

iv. Count the normalization factor

$$z = 2 \sum_j \sqrt{W_{+1}^j W_{-1}^j}$$

$z$  represents the degree of separation between distributions. A small value of  $z$  indicates the separation of numbers between positive and negative distribution classes. The smaller the value, the greater the separation. Therefore, the smallest value of  $z$  is used to select the weakest classifier in each cycle.

b. Choose  $h_t$  on *weak classifier* space to smaller the  $Z$

$$\begin{aligned} Z_t &= \min_Z \\ h &\in H \\ h_t &= \arg \min_Z \\ h &\in H \end{aligned}$$

c. Update weight of *training sample*

$$D_{t+1}(i) = \frac{D_t(i) \exp[-y_i h_t(x_i)]}{Z_t}$$

$Z_t$  is an initialization factor to make  $D_{t+1}$  become a PDF.

4. The last *strong classifier* is:

$$H(x) = \text{sign} \left[ \sum_{t=1}^T h_t(x) - \theta \right]$$

The *confidence* of  $H$  defined as:

$$\text{Conf}_H(x) = \left| \sum_t h_t(x) - \theta \right|$$

### 3. Experiments Design

This study aims to detect the presence of vehicles in the input image. The first step is pre-processing. The image pre-processing stage is done by resizing the image to a smaller image size so that the next process does not require a long time due to the large image dimensions. The next pre-processing stage is the conversion of input images in the form of color images (RGB) will be converted into grayscale images.

The classification process with Real AdaBoost is a classification process between vehicle images and non-vehicle images. There are two processes carried out in this process, namely:

- a. The training process is the process of training a system with input data that has been previously processed so that it can recognize when given new input. The model in the form of a set of weak classifier outputs is obtained, which will be used for decision making in the testing process later.
- b. The testing process is the decision making the process of the test image. The robust classifier is obtained from the output of the weak classifier. The output results are added together, and the signum of the total output is added to make a decision whether the tested image is a vehicle or not.

The data used in this system are vehicle images and non-vehicle images obtained from INRIA Car Dataset, searching on Google or Flickr, and manual retrieval with digital cameras. Data on this vehicle detection system consists of 974 images. Data is divided into training data and test data. The training data consists of 532 vehicle images and 150 non-vehicle images, while the test data is the remainder of the total amount of data that is not used for training data. The experiment using three groups data, namely vehicle image, Image with noise added and with box Object.

Calculation of accuracy process used Equation (6)

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{the amount of testing image}} \times 100\% \quad (6)$$

where TP is true positive, and TN is true negative.

#### 4. Result and Discussion

The following are the results of an experiment using three groups of experiment data

##### A. Vehicle Image

In this study, testing is done in cases when the road is busy and quiet. Busy road means more than one vehicle image, and a quiet road means there is only one vehicle image. Table 1 shows the testing result of image data with busy and quiet conditions.

**Table 1.** Testing Process Result

Detection Result	Input Image		
	Vehicle Image Busy	Vehicle Image Quiet	Non-Vehicle Image
Vehicle	135	100	2
Non-Vehicle	10	12	33

##### B. Image with noise added

To find out how strong the vehicle detection system, then try to test the testing image with noise added. In this study, it used Gaussian noise with a mean value of 0.02 and a variance value of 0.03. Examples of testing images that are added to noise are shown in Figure 2. This testing process only takes 18 images. Table 2 shows the results of the testing process with noise added.



**Figure 2** Example of testing image with noise added

**Table 2.** Testing Process Result with noise added

Detection Result	Input Image	
	Vehicle Image	Non-Vehicle Image
Vehicle	10	1
Non-Vehicle	2	5

### C. Image with box Object

Besides adding noise, non-vehicle image testing data is added with a box object image that is similar to a vehicle and a person image, shown in Figure 3. The results of the testing process with square objects and people can be seen in Table 3. The testing data used included ten non-vehicle images and added images of square objects such as office desks or boxes and pictures of people. Of the ten images tested, four images were detected incorrectly as vehicles, and six images were detected correctly as non-vehicles.

**Figure 3.** Example of testing image with box object and person added**Table 3.** Testing Process Result with box Object on non-vehicle image

Detection Result	Non-Vehicle Image with box Object and Person added
Vehicle	4
Non-Vehicle	6

Calculation of accuracy is performed on training data and testing data. The training data tested consisted of 532 vehicle images and 150 non-vehicle images. From the training data that has been tested, the detection results show that all images of both vehicle and non-vehicle images are detected correctly as vehicles or non-vehicles, or in other words, the training data tested has an accuracy of 100%. Table 4 shows the Accuracy Result from three group data.

**Table 4.** Calculation of Accuracy Result

Input Image	Accurate(%)
Training Data	100 %
Testing Data	91,78 %
Image with noise added data	83.33 %.

The reason for the low accuracy of vehicle detection systems in this study is the small amount of non-vehicle image training data. The difference in the amount of training data between vehicle images and large non-vehicle images causes the accuracy of the detection system to below.

## 5. Conclusion

Based on experiments and discussions on the results of tests that have been carried out on vehicle detection systems using the Histogram of Oriented Gradients and Real AdaBoost methods, then some conclusions can be drawn as follows:

1. This study has succeeded in detecting vehicles using the Histogram of Oriented Gradients and Real AdaBoost method, with the stages of the process, namely pre-processing, feature extraction with HOG, and the classification process with Real AdaBoost.
2. The Histogram of Oriented Gradients and Real AdaBoost methods can be used to detect vehicles with an accuracy rate of 91.78%. The results of testing data added by noise have a relatively good level of accuracy, which is 83.33%.
3. Image with a box object like a vehicle has a low level of accuracy because of the insufficient number and variety of training data images of non-trained vehicles.

## Acknowledgments

This research is supported by the Institute for Research and Community Development (Lembaga Penelitian dan Pengembangan Masyarakat/LPPM) Institut Teknologi Sepuluh Nopember Surabaya with Funding Agreement Letter number 1604/PKS/ITS/2019. We thank LPPM-ITS for giving us a chance to present this paper at the International Conference.

## References

- [1] Kandalkar P A and Dhok G P 2017 Review on Image Processing Based Vehicle Detection & Tracking System **3** 566–9
- [2] Madhogaria S, Baggenstoss P, Schikora M, Koch W and Cremers D 2015 Car detection by fusion of HOG and causal MRF *IEEE Trans. Aerosp. Electron. Syst.* **51** 575–90
- [3] Aoki D and Watada J 2015 *Human tracking method based on improved HOG+Real AdaBoost*
- [4] Dalal N and Triggs B 2005 Histograms of oriented gradients for human detection *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* vol 1 pp 886–93 vol. 1
- [5] Wu S and Nagahashi H 2015 Analysis of generalization ability for different AdaBoost variants based on classification and regression trees *J. Electr. Comput. Eng.* **2015**
- [6] Zhao Yuanshen A4 - Gong, Liang A4 - Zhou, Bin A4 - Huang, Yixiang A4 - Liu, Chengliang Y A-Z 2016 Detecting tomatoes in greenhouse scenes by combining AdaBoost classifier and colour analysis *Biosyst. Eng.* v. **148** 127-137–2016 v.148
- [7] Nixon M S and Aguado A S 2002 5 - Feature extraction by shape matching ed M S Nixon and A S B T-F E and I P Aguado (Oxford: Newnes) pp 161–216
- [8] Wu B, Ai H, Huang C, and Lao S 2004 Fast rotation invariant multi-view face detection based on real Adaboost *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.* pp 79–84
- [9] Chen P, Huang C, Lien C and Tsai Y 2014 An Efficient Hardware Implementation of HOG Feature Extraction for Human Detection *IEEE Trans. Intell. Transp. Syst.* **15** 656–62