

# Wheeler-Classified Vehicle Detection System using CCTV Cameras

Pratishtha Gupta  
Assistant Professor: Computer  
Science  
Banasthali University  
Jaipur, India

G. N. Purohit  
Professor: Computer Science  
Banasthali University  
Jaipur, India

Saroj Kumari  
M.Tech Scholar Information  
technology  
Banasthali University  
Jaipur, India

## ABSTRACT

Wheeler-Classified Vehicle Detection System is the implementation to check whether vehicle is two-wheeler or four-wheeler from the ordinary traffic images captured through CCTV cameras installed at crossings along a particular road at an instant which is an important task in Intelligent Transportation Systems (ITS).

The key contribution of this work are the introduction of methodology to reflect type of vehicle (two or four-wheeler) using MATLAB. Firstly, the region containing the vehicle is identified which is then cropped automatically from the traffic image and vehicle type is resulted using the approach of total number of corners obtained through corner detection of individual vehicle. If total number of corners exceeds the defined limit then output is reflected as four-wheeler otherwise two-wheeler.

## Keywords

MATLAB, CCTV, Wheeler-Classified Vehicle Detection, Corner Detection.

## 1. INTRODUCTION

Wheeler-Classified Vehicle Detection System concentrate on vehicle classification (two or four-wheeler), uses the most common feature among all vehicles, ubiquitous wheel. Vehicle classification newest application and is important especially for ITS including Toll Collection and Traffic Security Measures. This document presents a new effective method to check whether vehicle is two-wheeler or four-wheeler from the traffic images captured through CCTV cameras installed at crossings along a particular road at an instance for real time traffic signal control. In recent years, this is one of the applications of image and video processing. This experiment has been performed in MATLAB Environment.

For detecting vehicle type, two source images have to be given as input to the system, one is blank road image and another is corresponding road image with vehicles. Taking the advantage of static background, background subtraction techniques have been used for separating static background (blank road image) with a moving foreground (corresponding image with vehicles). So that area where all type of road vehicles such as motorcycles, cars, buses etc. are present can be determined accurately. Exact location of vehicle is detected by computing intensity changes. This is done while traffic image is scanned from top to bottom, and correspondingly RGB intensity of each pixel is computed. If change in intensity is observed means there is a probability of occurrence of vehicle. Coordinates where change in intensity is detected are used for cropping individual vehicles. Corner

detection method is applied on the traffic image for detecting corners. Now, count (total number of corners) corners of each cropped vehicle. If total number of corners exceeds the defined limit then output is reflected as four-wheeler otherwise two-wheeler. In this way, classification of type of vehicle can be obtained.

The rest of paper is organized as follows: Section 2 describes literature survey. Section 3 introduces the proposed methodology; Section 4 and 5 present implementation of developed system on two different real-time images. Section 6 presents the experimental results. Section 7 provides concluding remarks.

## 2. LITERATURE SURVEY

The ubiquitous wheel feature was used for developing wheel detector system by Achler O. et al. [1] [2] responses from the filter bank are applied to precomputed set of principle components; these are compared to a Gaussian mixture model of wheels and non-wheels. Where vehicle location was inferred using background segmentation. An efficient method was developed by Leung, M.K. et al. [3] for detecting the wheels of a vehicle in stereo image consists of two steps i.e. geometrical transformation and circle extraction. The circle extraction algorithm consists of template matching and the Hough transform.

Different camera-based systems have been developed in the past that aimed at classifying the vehicle or detecting regions of interest in traffic images captured at the crossings. Applications of these systems are mainly in the area of ITS. These ITS applications make use of many techniques. Recently, some authors have reported the use of techniques proposed by Aragon A. C. et al. [4] was computer vision technique and machine learning algorithm. This technique is useful for the visual detection of parts of interest i.e. bumpers, door handles, windows, wheels, lateral mirror, windshield, center, roof, head lights and rear lamp on the outer surface vehicles. Paul v. et al. [5] proposed a new technique called Cascade of Boosted Classifiers (CBC) for visual detection parts of interest. CBC combines pool of weak Classifiers which quickly discard area where the features of interest are not present. Complex classifiers work on most promising regions of the image. Classifiers use very simple features that are easy and fast to evaluate; this technique is useful for real time, not only detect one pattern in a scene but more than ten different patterns.

Seong-Min K. et al. [6] and Agarwal S. et al. [7] used Template Matching to locate regions of interest. Semantic information about the problem domain is stored in Template [8]. Similarity measures are used to determine correct position of a template over the image. These similarity measures are

based on the Euclidean distance, probabilistic functions and Mean Square Errors cross-correlation functions.

### 3. PROPOSED METHODOLOGY

MATLAB is most commonly used tool while dealing with problems related to video and image processing. So, MATLAB is used to acquire unique and good solution to the selected problem i.e. “Wheeler-Classified Vehicle Detection”. This experiment has been performed using MATLAB 7.14. Output is obtained both on the command window and on the figure window. Main objective is to determine whether vehicle is two-wheeler or four-wheeler. The system works with both color and gray scale traffic images taken at an instant by CCTV cameras installed along particular road. The particular wheeler detection system would work on specific dimensions of traffic images. The size of traffic images should be 701 pixels wide and 209 pixels high. In future this system can be extended for different orientation and dimensions.

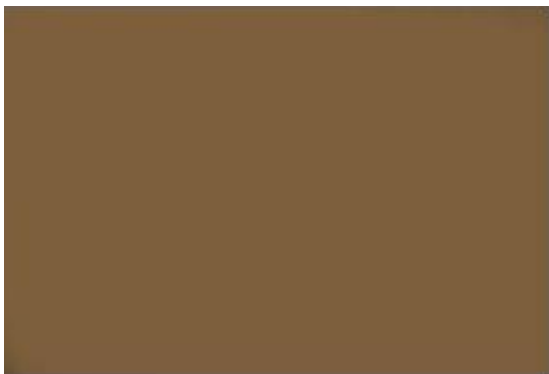
The Procedure, discussed below can be used for classifying vehicles according to their type i.e. vehicle is whether two-wheeler or four-wheeler.

The procedure used in wheeler-classified vehicle detection system been first experimented on dummy traffic images. After successful implementation on dummy traffic images, the system is used on real traffic images.

#### Step 1: Image Capture

##### a) Blank Road Image Capturing

Initially, image is captured at the junction through CCTV cameras installed along a particular road when there is no traffic on the road (Figure 1).



**Fig 1. Blank Road Image**

##### b) Image with Vehicles

Now, capture corresponding image with vehicles on the same road from CCTV Camera installed at junction (Figure 2).



**Fig 2. Image with Vehicles on the Road**

#### Step 2: Background Subtraction

Background subtraction technique is an effective way for separating a static background with a moving foreground. For traffic related applications, background subtraction is used in multiple conditions. Background subtraction alone creates a mask of where can be a vehicle in the scene.

The procedure is as follows: When background image is subtracted from the second image i.e. image with vehicles to obtain the foreground dynamic objects. If the absolute value of subtraction is greater than threshold, it is marked as foreground.

Remaining processing is performed on the image i.e. obtained after applying subtraction (Figure 3).



**Fig 3. Subtracted Image**

#### Step 3: Black and White Image through Comparison

Black and white image is obtained (Figure 4) by comparing the image with vehicles on road (Figure 2) with background image (Figure 1). While comparing two images pixel by pixel, at particular pixel if pixel value is same in both the images then that pixel value will be replaced by 1 otherwise 0.



**Fig 4. Black and White Image through Comparison**

#### Step 4: Fill Unwanted Holes

Holes are filled by computing area; if area computed is below the specified area limit then intensity value of all the pixels in that area is set as 0 as shown in Figure 5.



**Fig 5. Image after Removing unwanted Holes**

**Step 5: Coordinate Detection**

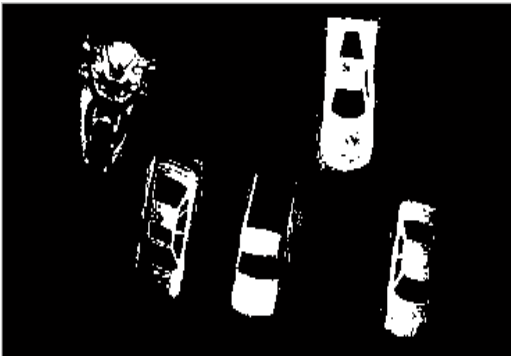
Scan above image (Figure 5) and compute intensity pixel wise from top to bottom.

Compute starting and ending coordinates of individual vehicles with respect to row and column in the image obtained after filling unwanted holes. Coordinate value is reflected when there is sudden change in RGB intensity means positive value of RGB is obtained.

Now, find minimum coordinate value with respect to row and column from where object starts in the traffic image.

**Step 6: Conversion to Binary Image**

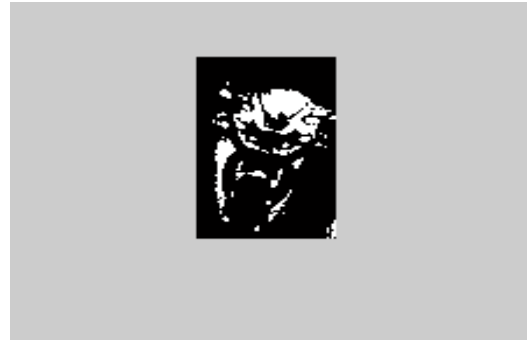
The system works with both color and gray scale traffic images taken at crossings by CCTV cameras. If traffic image is colored or gray scale, it should be converted to binary image (Figure 6), So that corners can be detected efficiently.



**Fig 6. After Conversion to Binary Image**

**Step 7: Cropping**

Automatic cropping of vehicles is performed (Figure 7, 8, 9, 10 and 11) after converting traffic image to binary image using inbuilt function with the help of coordinates computed above.



**Fig 7. Cropped Image of First Vehicle**



**Fig 8: Cropped Image of Second Vehicle**



**Fig 9. Cropped Image of Third Vehicle**



**Fig 10. Cropped Image of Fourth Vehicle**



**Fig 11. Cropped Image of Fifth Vehicle**

**Step 7: Corner Detection**

Detect corners of individual objects cropped from traffic image (Binary Image) using inbuilt function.

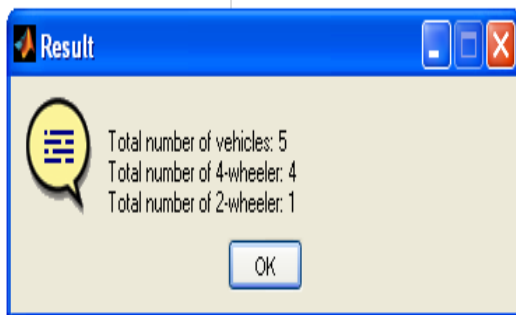
**Table 1. Corners of Vehicles**

Objects	Corners
Object 1 (2-Wheeler)	24
Object 2 (4-Wheeler)	4
Object 3 (4-Wheeler)	20
Object 4 (4-Wheeler)	12
Object 5 (4-Wheeler)	2

**Step 8: Wheeler Detection**

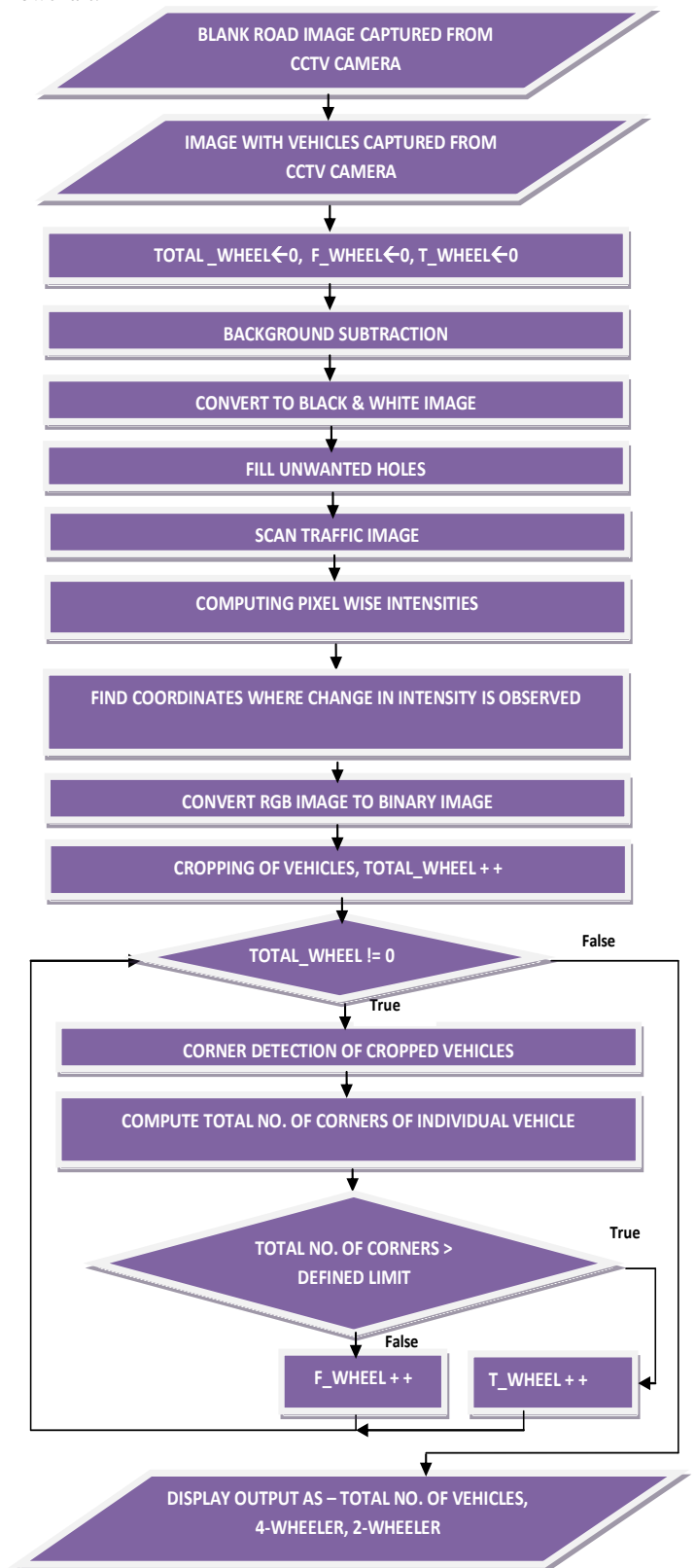
Now, deal with individual cropped vehicles. Compute total number of corners present in the first vehicle. According to total number of corners obtained through corner detection of individual vehicle. If total number of corners exceeds the defined limit then output is reflected as four-wheeler otherwise two-wheeler. Same process is also performed with remaining vehicles.

To display total four or two wheeler a count (initialized with 0) variable is maintained which is updated (incremented by 1) when total number of corners exceed the defined limit.



**Fig 12. Total No. of Vehicles, No. of Four-Wheeler and Two-Wheelers**

Above discussed procedure can be summarized below in the flowchart:



**Fig 13. Flowchart for Wheeler-Classified Vehicle Detection**

The above developed wheeler-classified vehicle detection system will be tested by applying it on real time traffic images captured at an instance so that more efficient results will be

obtained. System will be tested on two different real traffic images captured on the same background.

#### **4. PRACTICAL EXPERIMENT 1**

##### **Step 1: Image Capture**

*Blank Road Image Capturing*



**Fig 14. Blank Road Image**

*Image with Vehicles*



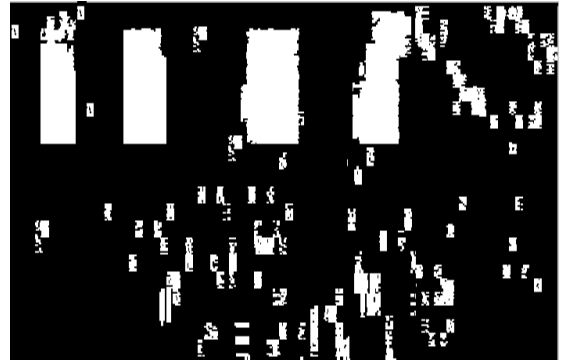
**Fig 15. Image with Vehicles on the Road**

##### **Step 2: Background Subtraction**



**Fig 16. Subtracted Image**

##### **Step 3: Black and White Image through Comparison**



**Fig 17. Black and White Image through Comparison**

##### **Step 4: Fill Unwanted Holes**



**Fig 18. Image after Removing Unwanted Holes**

##### **Step 5: Conversion to Binary Image**



**Fig 19. After Conversion to Binary Image**

##### **Step 6: Cropping of Vehicles**



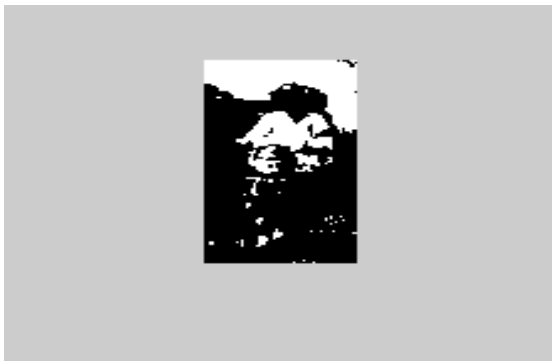
**Fig 20. Cropped Image of First Vehicle**



**Fig 21. Cropped Image of Second Vehicle**



**Fig 22. Cropped Image of Third Vehicle**



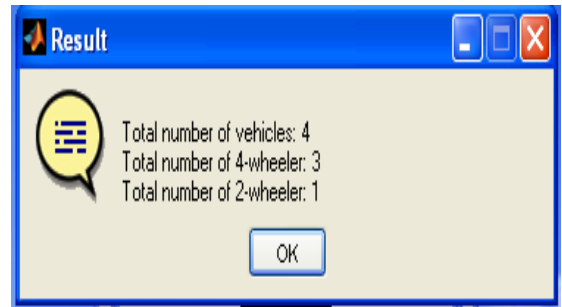
**Fig 23. Cropped Image of Fourth Vehicle**

**Step 7: Corner Detection of cropped vehicles**

**Table 2. Corners of Vehicles**

Objects	Corners
Object 1 (4-Wheeler)	9
Object 2 (4-Wheeler)	2
Object 3 (4-Wheeler)	5
Object 4 (2-Wheeler)	24

**Step 8: Wheeler Detection**



**Fig 24. Total no. of Vehicles, no. of four-wheeler and two-wheelers**

**5. PRACTICAL EXPERIMENT 2**

**Step 1: Image Capture**

*Blank Road Image Capturing*



**Fig 25. Blank Road Image**

*Image with Vehicles*



**Fig 26. Image with Vehicles on the Road**

**Step 2: Background Subtraction**



**Fig 27. Subtracted Image**

**Step 3: Black and White Image through Comparison**



**Fig 28. Black and White Image through Comparison**

**Step 4: Fill Unwanted Holes**



**Fig 29. Image after Removing unwanted Holes**

**Step 5: Conversion to Binary Image**



**Fig 30. After Conversion to Binary Image**

**Step 6: Cropping of Vehicles**



**Fig 31. Cropped Image of First Vehicle**



**Fig 32. Cropped Image of Second Vehicle**



**Fig 33. Cropped Image of Third Vehicle**



**Fig 34. Image of Fourth Vehicle**



**Fig 35. Cropped Image of Fifth Vehicle**



**Fig 36. Cropped Image of Sixth Vehicle**



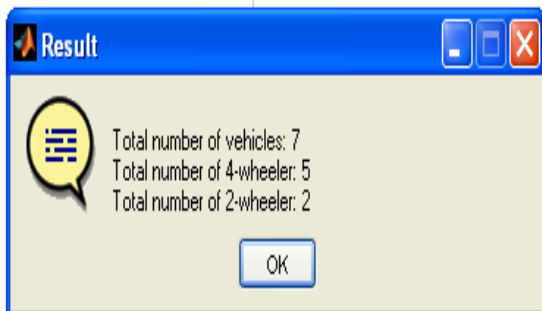
**Fig 37. Cropped Image of Seventh Vehicle**

**Step 7: Corner Detection of cropped vehicles**

**Table 3. Corners of Vehicles**

Objects	Corners
Object 1 (4-Wheeler)	17
Object 2 (4-Wheeler)	9
Object 3 (4-Wheeler)	10
Object 4 (4-Wheeler)	4
Object 5 (4-Wheeler)	4
Object 6 (2-Wheeler)	23
Object 7 (2-Wheeler)	21

**Step 8: Wheeler Detection**



**Fig 38. Total No. of Vehicles, No. of Four-Wheeler and Two-Wheelers**

**6. RESULT**

Vehicle is whether four-wheeler or two-wheeler was recognized properly. Experimental result is shown in Table IV.

**Table 4. Table for Results**

Technique-MATLAB	Dummy Image	Real-Time Images 1	Real-Time Images 2
Number of vehicles	5	4	7
Number of 4-wheeler	4	3	5
Number of 2-wheeler	1	1	2
Result accuracy	100%	100%	99%

**7. CONCLUSION**

An efficient image processing approach has been suggested and analyzed for computing the wheeler-classified detection of vehicles from traffic images captured through CCTV cameras along a particular junctions or crossings at particular instance. This is an important application in Intelligent Transportation System. Experiments performed on dummy image and two real time traffic images show that wheelers of vehicles (Four-wheeler or Two-wheeler) were properly recognized with 99% accuracy. Experiments were performed using MATLAB 7.14 environment.

This work proves to be a significant part of Intelligent Traffic System. As a future scope, it needs to be implemented using Simulink model Blockset, so that it can be executed on hardware and made real-time.

**8. REFERENCES**

- [1] Bowman, M., Debray, S. K., and Peterson, L. L. 1993. Reasoning about naming systems. .
- [2] Ding, W. and Marchionini, G. 1997 A Study on Video Browsing Strategies. Technical Report. University of Maryland at College Park.
- [3] Fröhlich, B. and Plate, J. 2000. The cubic mouse: a new device for three-dimensional input. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems
- [4] Tavel, P. 2007 Modeling and Simulation Design. AK Peters Ltd.
- [5] Sannella, M. J. 1994 Constraint Satisfaction and Debugging for Interactive User Interfaces. Doctoral Thesis. UMI Order Number: UMI Order No. GAX95-09398., University of Washington.
- [6] Forman, G. 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. 3 (Mar. 2003), 1289-1305.
- [7] Brown, L. D., Hua, H., and Gao, C. 2003. A widget framework for augmented interaction in SCAPE.
- [8] Y.T. Yu, M.F. Lau, "A comparison of MC/DC, MUMCUT and several other coverage criteria for logical decisions", Journal of Systems and Software, 2005, in press.
- [9] Spector, A. Z. 1989. Achieving application requirements. In Distributed Systems, S. Mullender.