International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 8 (2015) pp. 19633-19646

http://www.ripublication.com

Automatic Vehicle Classification System

Almehmadi Tarig Saeed S, Zaw Zaw Htike

Department of Mechatronics Engineering
Faculty of Engineering, IIUM, Malaysia

Abstract

The automatic vehicle classification system has emerged as an important field of study in image processing and machine vision technologies' implementation because of its variety of applications. Despite many alternative solutions for the classification issue, the vision-based approaches remain the dominant solutions due to their ability to provide a larger number of parameters than other approaches. To date, several approaches with various methods have been implemented to classify vehicles. The fully automatic classification systems constitute a huge barrier for unmanned applications and advanced technologies. This project presents software for a vision-based vehicle classifier using multiple Viola-Jones detectors, moment invariants features, and a multi-layer perceptron neural network to distinguish between different classes. The results obtained in this project show the system's ability to detect and locate vehicles perfectly in real time via live camera input.

Keywords: Automatic vehicle classification, Viola-Jones detection, Moment invariants, Neural network

1. Introduction

Traffic accidents kill an estimated 1.27 million people a year globally [12] and violation of traffic rules causes 22% of these fatalities, according to the Insurance Institute for Highway Safety [1]; these figures necessitate intelligent traffic systems. In the meanwhile, research is moving toward fully automatic systems that collect the data required, analyse it, and achieve the results wanted in the operation. In modern transportation systems, a fully automated system approach requires vehicle detection and classification.

Automatic vehicle classification systems have passed through several stages, which can be divided into two main techniques: hardware-based classification and software-based classification [13]. In recent years, software systems have undergone significant development, especially in vision-based vehicle classification. Vision-based classification is easier to install and maintain, offers lower operation cost, and provides a large amount of information which can be manipulated in many different ways. Although vision-based classification systems have weak points, such as long processing and slow response, they are much better and more advantageous than hardware-based systems. Since vision-based classification systems have offered the best solution for vehicle classification to date, the problem specified involves the image processing point of view. Vehicle classification operation can be divided into three phases: vehicle detection and segmentation, features extraction, and vehicle classification.

However, due to the number of challenges surrounding the classification issue, it seems wise to minimize the obstacles associated with the problem by introducing some assumptions. The following assumptions were adopted to enhance the focus on the main classification system elements:

1. Weather and illumination effects are neglected.
2. Images are taken from a side view of the vehicles.
3. No occlusion occurs and only one vehicle appears in each image.

The following part of the article comprises four sections. The first discusses related work conducted in the field of machine vision, especially machine learning, image processing, and computer vision, by comparing the methods and algorithms proposed in those fields for vehicle classification. The following section covers the method implemented to develop the system's program. In addition, it provides a clear discussion of each technique used in this section. The section also contains an experimental trial and results of analysis to ensure the validity of the method proposed to solve the problem. Finally, in the last section, the paper concludes with a summary of the achievements and analyses of the results obtained; recommendations are also provided.

2. Related Work

In this section, the work related to the project will be discussed. The structure mainly follows the procedures in each article due to the importance of the operations sequence to describe the systems. The techniques used for detection, segmentation, features extraction, and classification are also illustrated to achieve an acceptable level of understanding and analysis of solutions and to define the critical points which need further investigation or development.

Chung-Cheng Chiu used a statistical algorithm established in earlier work for background extraction. Background extraction was used for vehicle detection in this work; then the boundary box was obtained using the connected component labelling method [2]. The shape characteristics obtained from the horizontal edge detection and the outline measurements for the vehicles were used for occlusion segmentation. Since each vehicle was separated in its own boundary box, the tracking method could be used. The tracking method depends on the measurement of distance between the reference point and the predictive point, the vehicles' colour average intensity, and the visual length of the roof. The classification process divides the vehicles into the categories of mini truck, van utility vehicle, sedan, and bus/truck. Classification and recognition was done based on the visual length and width, with a recognition rate of 98%.

The system employed by Jun Yee Ng and Yong Haur Tay [6] uses scale-invariant feature transform (SIFT), Canny's edge detector, K-means clustering, and Euclidean distance matching techniques and divides the classification procedure into two stages: inter-class and intra-class. At the beginning, features were extracted through several steps. The accuracy calculation showed that the inter-class classification rate, car and minivan, was 98.5% and the intra-class classification rates for sedan and taxi were 93% and 99.23%, respectively.

However, due to the complicated techniques used, the systems developed were not able to fulfill the real-time application requirements. Furthermore, the features extracted and the classifiers used did not provide a satisfactory level of robustness. Thus, more advanced techniques must be investigated to improve performance. Additionally, the Viola-Jones detection algorithm was not implemented in fully designed systems which involve detection, feature extraction, and classification.

3. Methodology

The methodology passed through several stages of development to achieve the final design to be used to solve the automatic classification problem.

The preliminary stages of the methodology development started with the idea of developing a Viola-Jones detector to work as a classifier. Each class was to be detected in different detectors and combined to form the system. The main obstacle to applying this algorithm for classification issues is that the algorithm is built to detect objects, which means that its output reflects a binary decision; either the target exists or it does not. Such a problem can be solved by introducing the multi-level Viola-Jones algorithm. The classifier resulted in duplicate detection of vehicles due to the similarity of the vehicles' shapes in different models.

In the following stage, the Viola-Jones detector was applied normally as one detector instead of a multi-level detector. Furthermore, the number of classes increased and a neural network classifier was applied to the detector's output. However, the similarity in vehicle shape minimized the efficiency of the detection to an unacceptable range. Also, the neural network was assigned to classify the images directly; this resulted in poor classification as the images were fed whole into the neural network.

In the final stage of the methodology development, the system requirement was narrowed to provide a solution for automating the toll plaza counters in Malaysia. The system design required identifying the classes based on the toll system. However, for prototype design, only the first class, passenger vehicles, and second class, trucks, were considered in this project.

3.1 Viola-Jones Object Detection Algorithm

The Viola-Jones Object Detection Algorithm was used for vehicle detection from each image in the video sequence. This algorithm consists of several techniques which were implemented to detect an object in the input image, including the integral image, Haar-like features, Adaboost, and cascade classifier techniques. Integral image represents the first step in implementation of this algorithm. The pixel value in the integral image represents the summation of the pixels' values above it and to the left. This can be represented by the equation:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \qquad (1)$$

where ii(x,y) represents the integral image and i (x′,y′) represents the original image. As a result, the sum of pixels in any rectangle in the integrated image can be represented by the value of its corners:

$$Sum = (LRC - URC - LLC + ULC) \qquad (2)$$

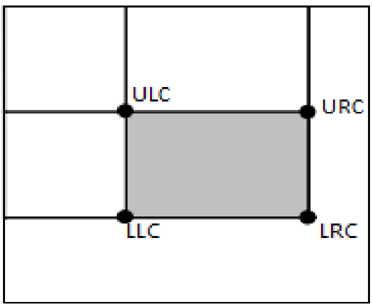

Figure 1. Integral Image

where LRC is the lower right corner, URC is the upper right corner, LLC is the lower left corner, and ULC is the upper left corner [10].

To appreciate the integral image idea and understand its importance, the Haar-like features technique should be employed first. Haar-like features are the summation of the pixel intensities inside adjacent rectangular regions at a certain location in the detection window and are used to calculate the difference between the sums [7]. The algorithm adopts three types of features – two-rectangular feature, three-rectangular feature, and four-rectangular feature – with base resolution of 24X24 for the detector. The importance of the integration image is clearer now. When allowing for all possible sizes and positions of the features, a total of approximately 160,000 different features can be constructed [10]. The computation time for features extraction drops significantly when the extraction is applied on the integrated image instead of the normal intensity image.

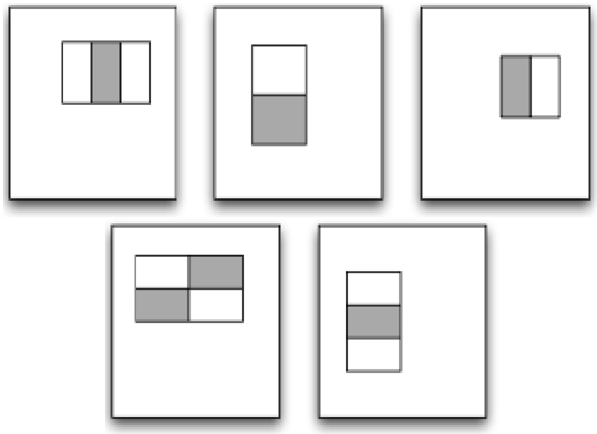

Figure 2. Haar-like features

The following technique is the Adaboost Learning Algorithm (Adoptive Boosting Learning Algorithm), which is a machine learning algorithm designed mainly to enhance the classification performance. It combines the weak classifiers obtained to form a stronger one as a linear combination of the weak learners. The misclassified examples' weight is increased in the next round to emphasize them until the algorithm achieves a sufficient set of weak learners to be defined as a stronger classifier.

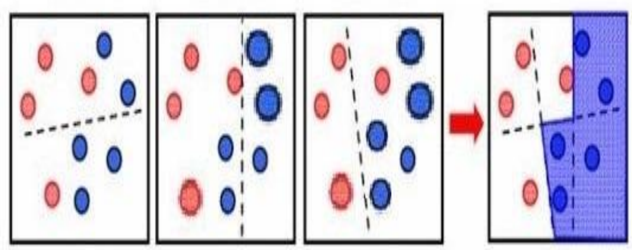

Figure 3. Adaboost procedures visualization [9]

However, in this approach, the Adaboost is implemented for the best relevant features selection and for training the classifier [3].

The cascade classifier is used as the final technique in the Viola-Jones detection algorithm. The cascade classifier is a multistage classifier which uses the ensemble method to obtain better performance. It provides the overall form of the detection process through a degenerate decision tree. At each stage, a strong classifier distinguishes between the negative sub-windows and the positive sub-windows. It dismisses the negative sub-windows and passes the positive ones to the following stage. By finding non-target sub-windows first and discarding them, the time consumed for classification is reduced. Initially, the classifiers in the first stages reject most of the negative sub-windows with a high false positive rate which is minimized over the stages to obtain the lowest false positive rate and the highest detection rates.

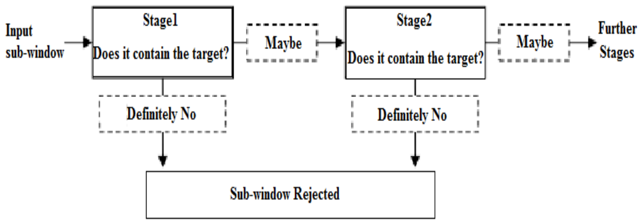

Figure 4. Schematic depiction of a detection cascade

3.2 Features extraction technique

Features extracting was applied using the moment invariants technique. The invariant global technique is a set of moments and their functions calculated for the images to define the moment features, as Hu illustrated in his article [4]. However, since infinite moments exist, Hu defined what is necessary for machine recognition. The seven moment values are computed by normalizing the central moment. They are translation invariant, rotation invariant, and scale invariant [5]. The moment invariants can be applied after changing the image into a grayscale image and then finding the following values:

1. The 2D moments m_{pq} of the gray image f(x,y)

$$m_{pq} = \sum_{x=0}^{x=M-1} \sum_{y=0}^{y=M-1} (x)^{p} (y)^{q} f(x, y) \qquad (3)$$

2. The central moments and its centroid

$$\bar{x} = \frac{m_{10}}{m_{00}} \; and \; \bar{y} \; \frac{m_{01}}{m_{00}}$$

$$\mu_{pq} = \sum_{x} \sum_{y} (x - \bar{x})^{p} . (y - \bar{y})^{q} f(x, y) \qquad (4)$$

3. The value of the central moments should be normalized using

$$\eta_{pq} = {\mu_{pq}} / {\mu_{00}}^{\gamma} ; \gamma = [\frac{p+q}{2}] + 1 \qquad (5)$$

4. Finally the seven moments are given as

$$
\begin{aligned}
M_1 &= (\eta_{20} + \eta_{02}) \\
M_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
M_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
M_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
M_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) \\
&\quad \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] \\
&\quad + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \\
&\quad \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \\
M_6 &= (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
M_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) \\
&\quad \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] - (\eta_{30} + 3\eta_{12}) \\
&\quad (\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]
\end{aligned} \tag{6}
$$

3.3 Neural network classifier

The multilayer perceptron (MLP) is a feedforward artificial neural network which uses a supervised learning technique called backpropagation, and it is the most suitable neural network for classification issues. In the MLP neural network, activation of the neuron nodes is nonlinear as it models the biological brain in its frequency of action. The MLP consists of an input layer, an output layer, and one or more hidden layers. A certain weight is used to connect each node to every node in the next layer.

The moment invariants technique is applied to the training set to generate the input training samples for the neural network. Also another matrix is formed to verify the target class for each input in the training. Thus, the number of input neurons corresponds to the number of the features extracted, which is seven neurons. The number of output neurons represents the number of classes required in the system, which is two nodes. For the hidden layer, the most mysterious part of neural network training is choosing the number of neurons in the hidden layer. Several rules limiting the range of values should be considered, but no single equation can provide the exact value for the number of neurons. However, the optimal choice for the hidden neurons can be identified using trial and error; three neurons were identified in this project. Finally, the backpropagation learning procedures were performed.

3.4 Programing and graphical user interface development

Finally, to implement these techniques together, a program was developed to receive the live camera output and apply the necessary modifications in the input before each process. However, the user does not transact with the source code directly, so a graphical user interface (GUI) was introduced in this project. The GUI helped to start and stop the program and contained four windows. The main window was used to display the main videos from the live cameras after the classification procedure for general monitoring. The three side windows were used to display the vehicles in the booths in case the system could not classify them correctly. The user is asked to verify the class of the vehicle appearing in the side windows through a popup input dialog window.

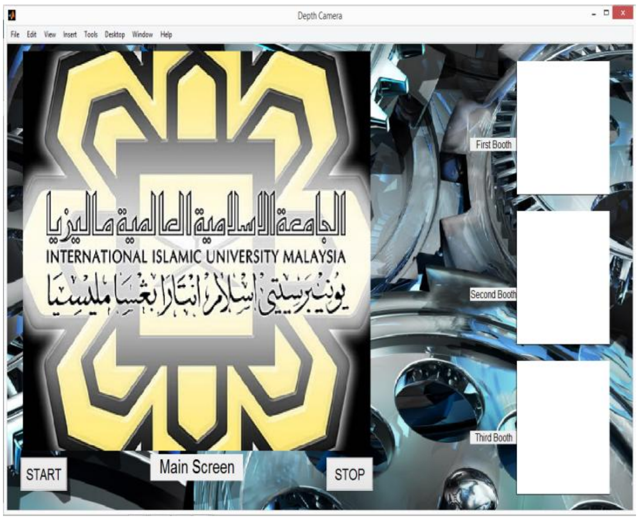

Figure 5. Program GUI initial screen

4. Experimental Trial and Results Analysis

This section describes the implementation of the program algorithm discussed to ensure its validity for vehicle classification. In the experiment, the categories were limited to two, Class1 and Class2, to show the ability of the algorithm to distinguish between classes. The computer used in this experiment had the specifications listed in table 1 and we used the Matlab 2013 version.

Table 1. Laptop Specifications

Type	Laptop
Processor	Intel® Core™ i7-3630QM (2.4GHz up to 3.4GHz, 6MB L3 Cache)
Hard Drive	500GB SATA HDD 5400RPM
RAM	16Gb DDR3
Graphic Processor	NVIDIA GeForce GT750M 2GB DDR5
Operating System	Windows 8.1

4.1 Experiment Procedures

The procedures start with the collection of positive samples and negative samples. First, 800 images collected for passengers' vehicles represent the first class and truck vehicles represent the second class, with 400 images for each class.

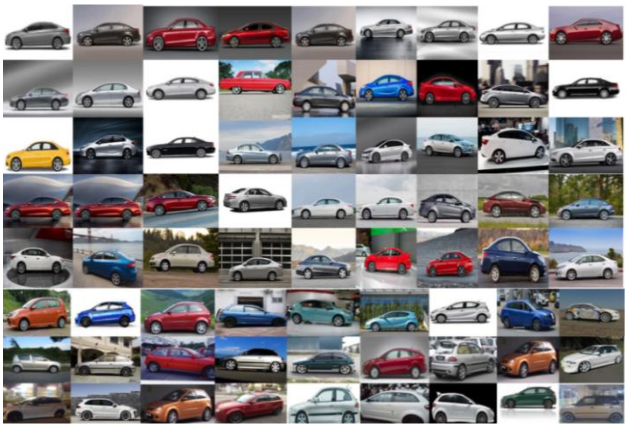

Figure 6. Example of the positive samples for first class

For the negative samples, 1600 images were collected and represent the possible backgrounds found in the test images. Moreover, after deleting the region of interest, the positive samples were used as negative samples, equalling another 400 images for each type.

Figure 7. Example of the deleted region of interest negative samples

In the second step, images were loaded to define the region of interest (ROI) for each image manually. The information extracted from each image, the ROI and the image location, was saved into a Matlab file to use for the cascade object detector training.

Figure 8. Example of the ROI locating

In the following step, the false alarm rate per stage, true positive rate per stage, number of stages, and features extraction method were the parameters used for training the cascade object detector. In this experiment the parameters were defined as follows:

False alarm rate per stage: 0.5	True positive rate per stage: 0.995
Number of stages:14	Features extraction: Haar

Thus, the overall false alarm rate is 0.5^{14}= 0.00006 and the overall true positive rate is 0.995^{14}=0.9322.

The cascade detector training takes place in this step to obtain the XML file. The XML file contains information about the detection target object, such as the above mentioned parameters, ROI boundaries, and the features type that is used for detection. The previous steps were repeated for the second class to obtain the XML file for the second class as well.

At this step, the positive samples were normalized and resized to extract the features for training the neural network classifier. Then, the training input set for the MLP was constructed in a matrix form and the training target was set. This was followed by selection of the test set, validation set, and test set for the MLP from the input and target matrices. The hidden layer neurons were defined and the training was performed.

At this stage, the system was ready to test. To test the system, the input video was fractionated into a sequence of images and then converted to grayscale. The detection algorithm was applied and outputs obtained from the detector were used to define the vehicle position in the image. The detected vehicles were cropped and separated from the main image. Several operations were performed on the vehicles' new images, such as resizing, normalizing, and changing the data type from uint8 to double. The outputs of the previous step were fed to the moments invariants extractor function. The values of the moments obtained were fed to the MLP to classify the vehicles in the images. The output of the neural network is a certainty percentage vector whose elements are equal to the number of output neurons. A threshold was applied to the certainty to see whether the class was classified or not. The largest level of certainty above the threshold identified the class classification. Moreover, if all the levels of certainty were below the threshold, then user input was required to classify the vehicle.

4.2 Results

The overall performance of the system was examined in several ways. First, the Viola-Jones detection algorithm was tested separately on two sets of data: the first set from the database and the second set not in the database. The MLP classifier was tested separately in the same manner. Forty test vehicles were selected, included and not included in the training datasets, to perform the test for each technique individually.

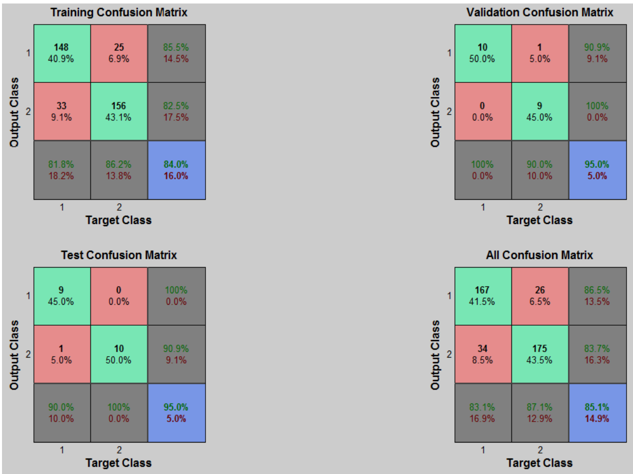

Figure 9. MLP confusion matrices

Table 2. Viola-Jones detectors test

	From The Database		Not In The Database	
Detectors	Detected	Not Detected	Detected	Not Detected
Detector for Class1	20/20	0/20	20/20	0/20
Detector for Class2	20/20	0/20	18/20	2/20

Table 3. Overall system test

	Classified as class1 automatically	Classified as class2 automatically	Classified as class1 manually	Classified as class2 manually	Misclassified
Class 1	23/25	0/25	2/25	0/25	0/25
Class 2	0/25	18/25	1/25	6/25	1/25

4.3 Results Analysis

The results illustrate that the overall performance of the system results in good ability to detect the vehicles and classify them. The angle from which the vehicles were captured may cause mis-detection in the detector. The illumination effects could not be eliminated from the system. However, Figure 9 shows that the detectors achieved 100% accuracy on the images used to train the detectors. Moreover, the test set detection accuracy varied from one detector to another since the first-class detector achieved 100% while the second-class detector achieved 90%. The misclassification in the second detector could be caused by the large degree of variation in the second class. The MLP neural network provides a satisfactory level of performance as the numbers in the green boxes are much higher than those in the red boxes. In the overall confusion matrix, the accuracy obtained was 85.1%. In the system as whole, employing user input to provide the classes may improve the classification results much more than in evaluation performed on the MLP alone. Finally, the system evaluation showed 98% accuracy in the overall system performance.

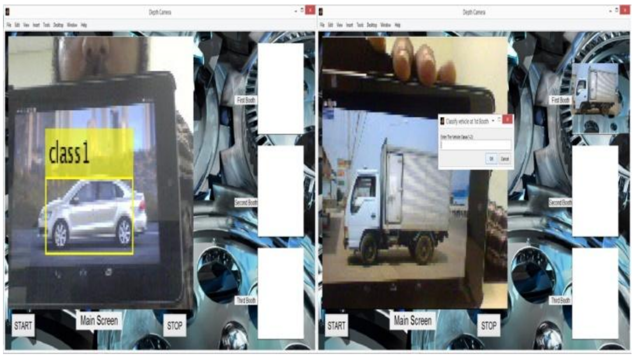

Figure 10. Positive classification example and input dialog example

5. CONCLUSION AND RECOMMINDATION

5.1 Conclusion

In this project, a general understanding of image process and machine learning were developed to counter the problem of developing vehicles classification software. The software design was developed to achieve no man in loop classification systems; therefore, the Viola-Jones algorithm was adopted because of its ability to detect objects and its accuracy. In addition, the multi-layer perceptron neural network was used for pattern recognition to classify vehicles with the help of moment invariants features. Moreover, the software has a very short run time so it can be used for real-time application. This project achieved the intended objectives, which were designing classification software, selecting proper parameters to achieve the classification tasks, and real-time implementation capability.

5.2 Limitations

In this project, the limitations introduced in the assumptions section (1.5) were minimized as much as possible. However, the illumination issues and the occlusion could not be overcome and it is still a problem in the system. The results obtained could not be further improved due to limitations in resources, such as computing power and the quantity of good-quality images. The fast movement of the vehicles could cause errors in detection due to the limitations of the camera's ability. Moreover, while the system requests user verification of the class, the entire system will be on hold, which can lower system efficiency.

5.3 Recommendations

Fully automatic classification systems are necessary for future development of the vision system, which plays an important role in overcoming the drawbacks of the current traffic control and other systems. This should provide an incentive for mechatronics system designers to propose more radical and creative solutions based on today's advanced technologies. Furthermore, the design of this project can be further modified to accommodate more classes. Note that the detector should not be trained to detect more than one class since variation minimizes detection efficiency dramatically.

However, the classes with a large degree of variation similar to the second class should be divided into two or more detectors. In addition, the incremental learning techniques can be considered for future work to achieve perfection in the design. Also, due to the lack of information in the image coordinates about the real-world coordinates, the features extracted cannot distinguish between the classes with precision, especially with overlapping in the shape designs. The area represented by each pixel can be investigated and calculated based on the distance between the camera and the detected object. As a result of this solution, the exact physical measurement can be defined accurately. Additionally, the review paper on the development of vehicle detection based on vision systems [8] concluded that future work could involve a combination of stereo vision techniques and monocular vision techniques.

References

[1] Aarth R., Arun K. and Raghesh K., "Automatic Isolation and Classification of Vehicles in a Traffic video," *IEEE,* pp. 357-361, 2011.

[2] Chiu C., Ku M., Hong S., Wang C., Wu J. and Li H., “Automatic Traffic Surveillance System for Vision-Based Vehicle detection and Traking,” *Journal of Information Science and Engineering,* vol. 26, pp. 611-629, 2010.

[3] Freund Y. and Schapire R., "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Journal of Computer and System Sciences,* vol.55, pp. 119-139, 1997.

[4] Hu M., "Visual Pattern Recognition by Moment Invariants," *IEEE,* vol.8, pp. 179-187, 1962.

[5] Mercimek M., Gulez K. and Mumcu T., "Real object recognition using moment invariants," *IEEE,* pp. 765 - 775, 2005.

[6] Ng J. and Tay Y., “Image-based Vehicle classification system,” *The 11th Asia-Pacific ITS Forum and Exhibition,* 2011.

[7] Papageorgiou C., Oren M. and Poggio T., "A general framework for object detection," *Sixth International Conference on Computer Vision,* pp. 555-562, 1998.

[8] Sivaraman S. and Trivedi M., "A Review of Recent Developments in Vision-Based Vehicle Detection," *IEEE,* pp. 310-315, 2013.

[9] Solà M., "Automatic Labeling Application Applied to Food-Related Object Recognition," *Facultat de Matemàtiques Universitat de Barcelona*, 2013.

[10] Viola P. and Jones M., "Robust Real-Time Face Detection," *International Journal of Computer Vision* , vol. 57, pp. 137-154, 2004.

[11] Wang H., Mohamad D. and Ismail N., “An Efficient Parameters Selection for Object Recognition Based Colour Features in Traffic Image Retrieval,” *The International Arab Journal of Information Technology,* vol. 11, pp. 308-314, 2014

[12] World Health Organization, http://www.who.int/mediacentre/news/releases/2009/road_safety_report_20090615/en/, 20 OCT 2014.

[13] Yousaf K., Iftikhar A. and Javed A., "Comparative Analysis of Automatic Vehicle Classification Techniques: A Survey," *International Journal of Image, Graphics and Signal Processing* , vol. 4, pp. 52-59, 2012.