Hindawi
Journal of Sensors
Volume 2017, Article ID 5627281, 10 pages
https://doi.org/10.1155/2017/5627281

Research Article

Vehicle Detection Based on Deep Dual-Vehicle Deformable Part Models

Yingfeng Cai,[1] Ze Liu,[2] Xiaoqiang Sun,[1] Long Chen,[1] Hai Wang,[2] and Yong Zhang[3]

[1] *Institute of Automotive Engineering, Jiangsu University, Zhenjiang 212013, China*
[2] *School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China*
[3] *School of Automotive and Traffic Engineering, Nanjing Forestry University, Nanjing 210037, China*

Correspondence should be addressed to Yong Zhang; zy.js@163.com

Received 8 June 2017; Revised 29 September 2017; Accepted 10 October 2017; Published 5 December 2017

Academic Editor: Mi Zhang

Vehicle detection plays an important role in safe driving assistance technology. Due to the high accuracy and good efficiency, the deformable part model is widely used in the field of vehicle detection. At present, the problem related to reduction of false positivity rate of partially obscured vehicles is very challenging in vehicle detection technology based on machine vision. In order to address the abovementioned issues, this paper proposes a deep vehicle detection algorithm based on the dual-vehicle deformable part model. The deep learning framework can be used for vehicle detection to solve the problem related to incomplete design and other issues. In this paper, the deep model is used for vehicle detection that consists of feature extraction, deformation processing, occlusion processing, and classifier training using the back propagation (BP) algorithm to enhance the potential synergistic interaction between various parts and to get more comprehensive vehicle characteristics. The experimental results have shown that proposed algorithm is superior to the existing detection algorithms in detection of partially shielded vehicles, and it ensures high detection efficiency while satisfying the real-time requirements of safe driving assistance technology.

1. Introduction

Nowadays, vehicle traffic accidents cause about 12 million casualties and 1–3% of the total global GDP loss of social property. The major causes of road accidents are related with the subjective factors of drivers. Therefore, it is imperative to improve road safety and help drivers to anticipate and avoid traffic accidents. In recent years, more and more scholars began to work on vehicle testing and development of driving support technology. The vehicle vision detection based on machine vision is a hotspot in the field of computer vision and safe driving aids. At present, many scholars have applied pattern recognition, image processing, and machine learning to the field of vehicle detection and have achieved good results that have played an important role in basic research and engineering application [1–5].

Currently, researchers use more general and robust features such as HOG features and Haar characteristics to detect vehicles. The HOG feature is an interpreted image feature that can be used to confirm the attitude of the vehicle. However, the extraction process of these features is time consuming and the feature dimension is large, which often leads to longer training time and slower detection speed. In this paper, the HOG algorithm proposed by Porikli [6, 7] and the pyramid HOG algorithm proposed by Bosch et al. [8] are combined and HOG feature dimension is effectively reduced and detection is accelerated. Maji et al. [9] studied the improvement of HOG feature classifier and proposed an additional kernel support vector machine (AKSVM), which is superior to linear kernel support vector machine. In 2000, Papageorgiou and Poggio [10] proposed Haar wavelet concept. The Haar feature is not only suitable for detection of horizontal, vertical, and symmetrical structures, but also it uses the integral map for feature extraction, which can be used in real-time calculation. Viola and Jones [11] introduced the concept of integral graphs to speed up the extraction of Haar features. Xing [12] proposed an algorithm that was first presented by Haar and AdaBoost and then

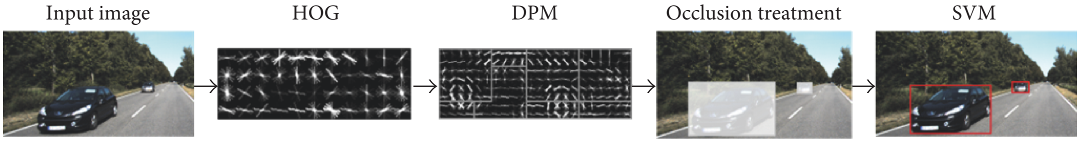


Figure 1: The vehicle detection based on deep model.

retested using HOG and libSVM to ensure the same detection accuracy. Zhang et al. [13] designed Haar-like features that have good robustness to occlusion. In 2008, Felzenszwalb et al. [14] proposed deformable part model (DPM), which achieved the best detection of multiple targets and modelling using the implicit variable support vector machine (LSVM) training, and the introduction of cascade greatly improved the algorithm detection speed. In 2010, Park et al. [15] proposed a multiresolution model to achieve better detection results than traditional DPM. In 2012, Ouyang and Wang [16] proposed a method for simultaneous detection of two human body targets, which reduced the target missing rate. In 2013, Yan et al. [17] proposed an algorithm for detection of multiresolution targets. There are also some scholars [18, 19] that used the scale-invariant feature transform (SIFT) to detect the tail of vehicle.

At present, the most popular feature extraction method is based on the deep learning method, wherein the features are extracted using the model trained with a large amount of data. The deep learning model proposed in [20] received much attention from computer vision research community. In 2012, Krizhevsky who applied the deep learning in the International Image Recognition Challenge (ILSVRC) for the first time, wherein the deep learning was used for image classification and target positioning, got the results that were much better than the second rated ones. By 2014, in almost all ILSVRC, the method in deep learning framework was used. The DeepID1 project is of the Chinese University of Hong Kong and the DeepID2 project achieved excellent results in the Face Recognition in the Wild (LFW) database based on the deep learning approach. The DeepID2 improved the recognition rate to 99.15%, which is better than all current algorithms and human recognition rates. Accordingly, the characteristics extracted by the deep learning method are better than the ones extracted by the traditional artificial design methods; thus, the deep learning has great potential in future research field. Thus, a lot of researcher began to apply deep learning to target detection. In 2013, Sermanet et al. [21] used a convolutional sparse coding to learn the characteristics from the pixel data of image. In 2014, Luo et al. [22] proposed a switchable deep network (SDK) and achieved good detection results. Although in the field of vehicle detection the deep learning has not yet reached the impact it has in other areas of recognition, it still represents the future research trend.

After several years of research and development of vehicle detection, a great progress in terms of detection accuracy, detection speed, and detection stability has been achieved. However, the main difficulties in its development process is the obstruction of vehicle target and multifeature fusion problem. The main goal of this study is to analyze and solve the problem of high false detection rate in current vehicle detection algorithms. Therefore, a vehicle detection algorithm based on the dual vehicle deformable depth model is proposed to improve the vehicle detection characteristics. The dual-vehicle deformable part model relates to the bottom layer, and the dual vehicle deformation depth model relates to the deep learning characteristic. In this paper, two models are combined to achieve vehicle detection, and their advantages are used to improve detection rate.

2. Deep Model and Vehicle Detection

2.1. Detection Algorithms. The vehicle detection based on the deep model [23] is shown in Figure 1, wherein model training and model testing are two separate processes. In this paper, model training and model testing process are the same, and they include four steps: feature extraction, deformation processing, occlusion, and classifier training. In the model testing process, through the input image and depth model parameters to get the output category, that is to determine whether the detection window contains vehicle targets.

2.2. Deep Model. The structure of the deep model is shown in Figure 2, wherein it can be seen that deep model includes input layer, feature extraction layer, feature mapping layer, component detection layer, deformation processing layer, and visualization reasoning and classification layer.

The functions of the layers are as follows:

(1) Input layer: first, the input image is scaled to 84×28 pixels, and then it is preprocessed to obtain a 3-channel image data.

(2) Feature extraction layer: a 64-feature map is obtained by convolution of image data using 64 filters with the size of $(9 \times 9 \times 3)$ pixels.

(3) Feature mapping layer: a 4×4 filter is used to average the feature map in order to obtain the final vehicle characteristics.

(4) Component detection layer: first, the vehicle is divided into multiple parts and levels according to the size as it is shown in Figure 3, wherein a white background denotes the actual part, a black background denotes the part that is not seen at that time, and arrows to the large-size parts denote a combination of small-size parts. The size of the corresponding parts filtered by vehicle convolution operation is used to get the corresponding mapping of each part. In general, the filter of convolution layer has a fixed size, but due to different sizes of vehicle

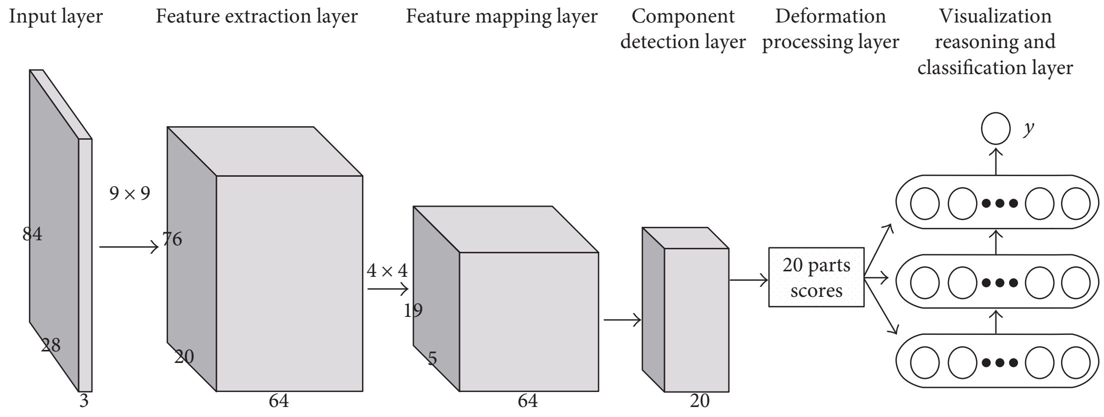

Input layer
Feature extraction layer
Feature mapping layer
Component detection layer
Deformation processing layer
Visualization reasoning and classification layer
y
9 × 9
84
28
3
76
20
64
4 × 4
19
5
64
20
20 parts scores

Figure 2: The structure of deep model.

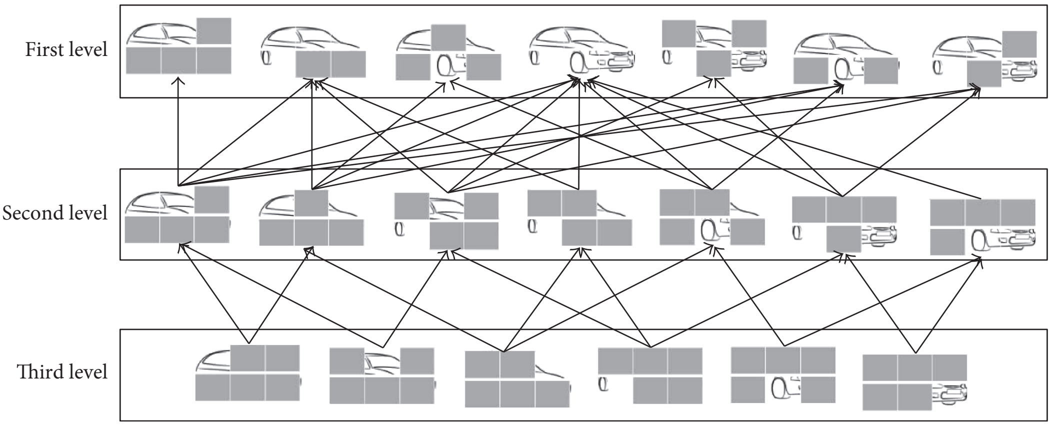


Figure 3: The parts of vehicles.

parts, filters with different sizes are needed in the component detection layer.

(5) Deformation processing layer: the degree of part matching is calculated by deformation degree of certain parts, Figure 4. The sum of mapping sums B_p of distortion map and component detection map is as follows:

$$B_p = M_p + \sum_{n=1}^{N} c_{n,p} D_{n,p}, \tag{1}$$

where p represents the number of parts ($p = 20$), M_p denotes the corresponding mapping of part p, $D_{n,p}$ represents the nth deformation map that corresponds to part P, which is predefined, $c_{n,p}$ represents the $D_{n,p}$ corresponding weight, N denotes the number of deformed maps, and $N = 4$. The matching degree of part p is equal to the value of global max pooling determined by the following:

$$s_p = \max_{(x,y)} b_p^{(x,y)}, \tag{2}$$

where $b_p^{(x,y)}$ represents the element value that corresponds to (x, y) in B_p.

The position of detection part can be calculated by the following:

$$(x, y)_p = \arg\max_{(x,y)} b_p^{(x,y)}. \tag{3}$$

The degree of part deformation can be represented by the quadratic function (4), wherein the subscript o is omitted.

$$\begin{gathered} B = M + c_1 D_1 + c_2 D_2 + c_3 D_3 + c_4 D_4 + c_5 L, \\ b^{(x,y)} = m^{(x,y)} + c_1 d_1^{(x,y)} + c_2 d_2^{(x,y)} + c_3 d_3^{(x,y)} + c_4 d_4^{(x,y)} + c_5, \\ d_1^{(x,y)} = (x - a_x)^2, \quad d_2^{(x,y)} = \left(y - a_y\right)^2, \quad d_3^{(x,y)} = x - a_x, \\ d_4^{(x,y)} = y - a_y, \quad c_5 = \frac{c_3^2}{4c_1} + \frac{c_4^2}{4c_2}, \end{gathered} \tag{4}$$

where L denotes the matrix, $d_n^{(x,y)}$ represents the element value that corresponds to (x, y) position in D_n, $m^{(x,y)}$ indicates the element value that corresponds to (x, y) position in the part detection map M, (a_x, a_y) represents the ideal position where the part p is preset, and lastly, c_1, c_2, c_3, and c_4 represent the deformation parameters.

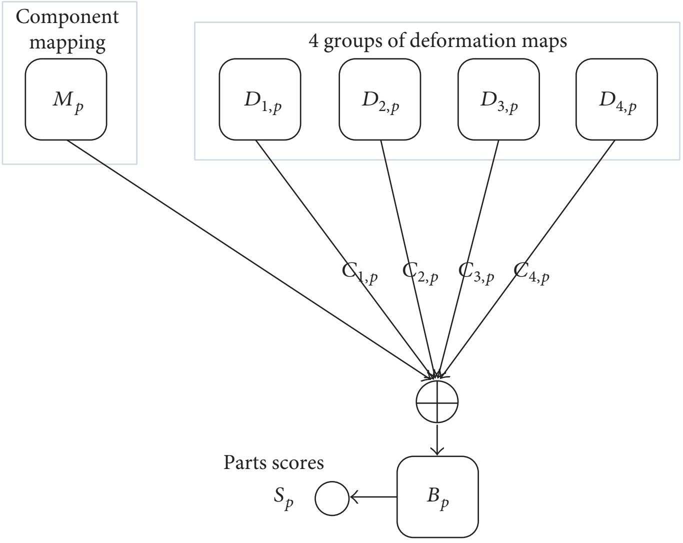


Figure 4: Deformation processing layer.

(6) Visualization reasoning and classification layer: the components are divided into multiple layers by size, and a convolution neural network shown in Figure 5 is formed. As it can be seen in Figure 3, most parts correspond to multiple parent parts and multiple subparts, and two visual parts in the same hierarchical layer can be common parent parts of a part in the lower hierarchical layer. The correlation between parts is shown by an arrow in Figure 5.

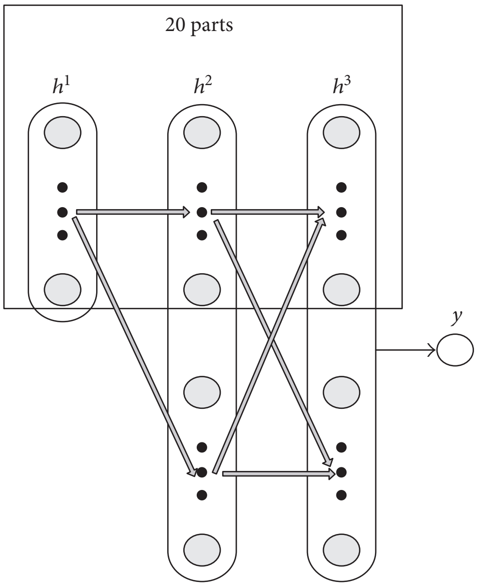


Figure 5: Visualization reasoning and classification layer.

Then, the BP algorithm is used for iteration:

$$\begin{aligned} h_i^l &= \sigma\left(c_i^l + g_i^l s_i^l\right), \\ h_i^{l+1} &= \sigma\left(\left(h^l\right)^T W_{*,i}^l + c_i^{l+1} + g_i^{l+1} s_i^{l+1}\right), \quad l = 1, 2, \\ y &= \sigma\left(\left(h^3\right)^T W^{cls} + b\right), \end{aligned} \tag{5}$$

where l represents the number of network layers, i represents the part number, s_i^l represents the matching degree of the ith component of layer l, h_i^l is the visualization of the part and here it is also the unit of convolutional neural network, $\sigma(t) = (1 + \exp(-t))^{-1}$ represents the excitation function, g_i^l denotes the weight of s_i^l, c_i^l denotes the offset term, W^l denotes the correlation value between h^l and h^{l+1}, $W_{*,i}^l$ is the ith row of W^l, W^{cls} denotes the implied unit h^3 linear classifier, b denotes the offset term, and y denotes the estimated value of the detection tag.

2.3. Model Training and Matching Process. The main steps of deep model training are as follows:

(1) Image preprocessing: the training input is an image that is preprocessed to obtain 3-channel image data. First, the image is converted from RGB to YUV color space, with Y channel as the first channel of image data. Then, half of the YUV image is reduced and the 3 channels of the YUV image are concatenated into a 84 × 56 size image as the second channel of the image data, and the blank space is padded with 0. Finally, the Sobel operator is used to detect the three channels of YUV image, and the image is numbered as 84 × 56 size as the third channel of image data, and the blank is filled with zero. The preprocessed image data contains different resolution images and original edge information, and the light changes are processed better by performing zero mean and unit variance operations on the data for each channel.

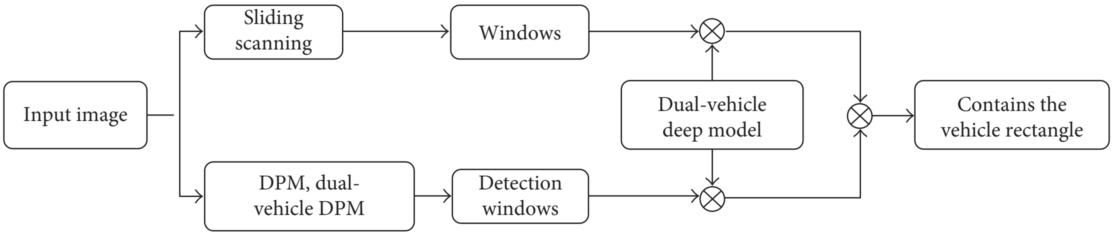


Figure 6: The flowchart of vehicle detection based on dual-vehicle deep model.

(2) Extraction of vehicle characteristics: first, 64 of the $9 \times 9 \times 3$ filters were used for convolution on the image data to obtain 64 feature maps, and then using the 2×2 filter on the 64 feature map for the average pool to get the final vehicle characteristics.

(3) Deformation processing: first, the vehicle is divided into multiple parts of different sizes and into multiple grades by size, and then the matching degree of each part is calculated using the part deformation degree.

(4) Occlusion: the number of parts divided into multiple levels determines the number of layers of convolution neural network. Each part represents a neuron in the network layer. The degree of part matching and output value is used to estimate the existence of vehicle targets.

(5) Classifier training: after establishment of convolution neural network, the parameters between the last hidden layer and the output layer can be regarded as classifier parameters. It is important to note that the selection of several key parameters is critical to the training of the model before the network is trained. The first key parameter, learning rate, determines the speed of the weight update. If the setting is too large, the result will exceed the optimal value. If the setting is too small, the descent speed will be too slow. Only by relying on human intervention to adjust, the parameters need to constantly modify the learning rate, so the latter three parameters are based on proposed adaptive ideas and solutions. The following three parameters are weight decay, momentum, and learning rate decay. The use of weight decay is in order to neither improve the convergence accuracy nor increase the convergence rate; the ultimate goal is to prevent overfitting. In the loss function, weight decay is placed in front of a regularization coefficient; regularization generally refers to the complexity of the model, so the role of weight decay is to adjust the complexity of the model on the loss of the function; if the weight decay value is large, the value of the complex model loss of function is also large. Momentum is a commonly used acceleration technique in the gradient descent method. It is derived from Newton's law. The basic idea is to find the optimal effect of adding "inertia." When there is a flat area in the error surface, the stochastic gradient descent can learn faster. Learning rate decay is to improve the search ability of the stochastic gradient descent, specifically to reduce the size of the learning rate every iteration.

Since the model matching process is the same as the model training process, after the deep model training is completed, it is possible to judge whether the detection window contains the vehicle target by observing the output value after matching the input image by the deep model.

3. Vehicle Detection Algorithm Based on Dual-Vehicle Depth Model

3.1. Overview of Detection Algorithms. Although the dual-vehicle deformable part model has a better performance in detecting multivehicle targets, wherein vehicles are close to each other, the artificial design of vehicle feature extraction method will be always imperfectly local [24]. In order to further improve the accuracy of vehicle detection, this paper proposes a vehicle detection algorithm based on the dual-vehicle depth model. The algorithm, which includes model training and confirmation window, is shown in Figure 6.

The dual-vehicle depth model is used to train the dual-vehicle depth model. The resulting dual-vehicle depth model includes input layer, feature extraction layer, feature mapping layer, component detection layer, deformation processing layer, and visualization reasoning and classification layer. However, due to different training sets, the specific parameters and structures of training process are different. In addition, the extraction of filtered vehicle parameters, vehicle split into parts, and construction of convolution neural network structures are not the same.

In this paper, the dual-vehicle depth model that differs from the traditional depth model is introduced. After model training, the model is used to test the detection window obtained by the dual-vehicle deformable part model and the window generated by sliding scanning of input image. The output is used to determine whether the window contains the vehicle target. In other words, the dual-vehicle depth model is combined with the dual-vehicle deformable part model to achieve vehicle detection, and use of the dual-vehicle deformable part model for rough detection can greatly reduce the number of windows and improve the detection rate.

3.2. Model Training

3.2.1. Dual-Vehicle Split and Parts Combination. Similarly to the depth model training, after vehicle extraction, the vehicle

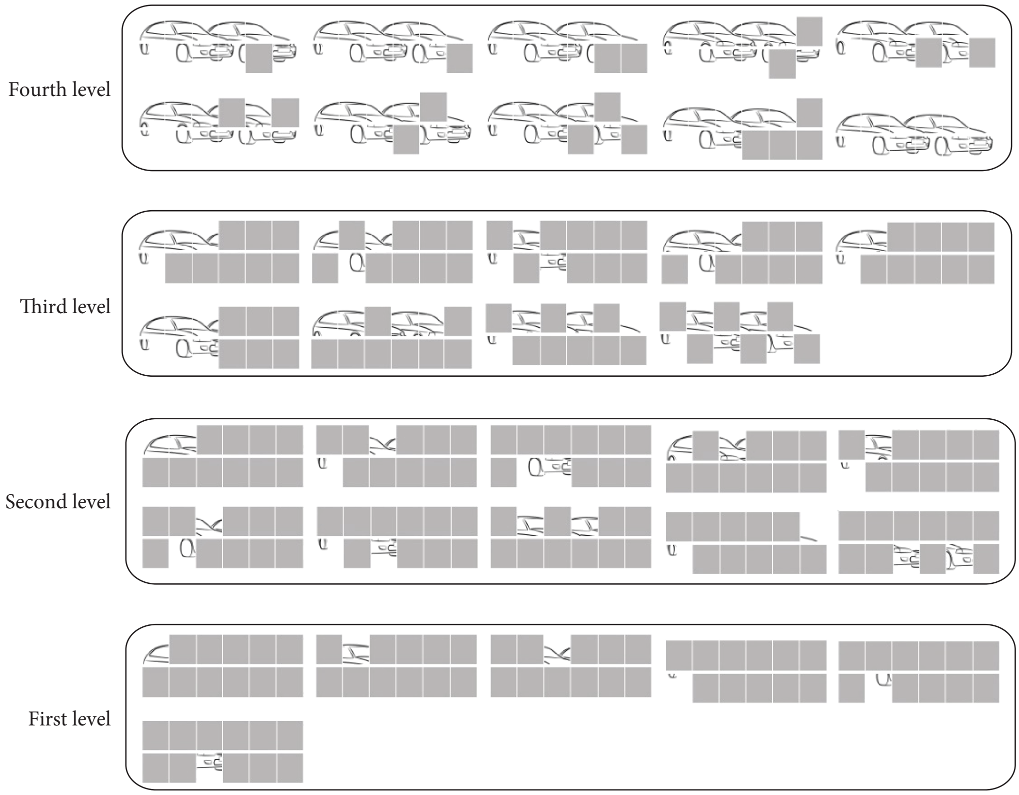


Figure 7: The model based on parts.

target is split into multiple parts. Here, two vehicles are split into 63 parts with different sizes, and sizes are divided into four levels in order to facilitate browsing and saving of image part (Figure 7). In Figure 7, a white background denotes the existence of a part and a black background denotes that the part is not present at that time. In the first stage in Figure 7, 12 parts of the smallest size are shown, wherein parts that are symmetrical to six parts are not shown; in the second stage, 17 parts of the medium size are shown, and parts that are symmetrical to the first seven parts are not shown; in the third stage, 15 parts of the medium size are shown, where parts that are symmetrical to the first six components are not displayed; and in the fourth stage, 19 parts of the largest size are shown, wherein the first nine symmetrical parts are not displayed.

3.2.2. Convolution Neural Network Structure. After two vehicles are split into multiple parts, each part score is calculated by (2). Then, a convolutional neural network is constructed with the same visualization reasoning and classification layer as the depth model. Since the parts are divided into four levels as shown in Figure 7, the network is better than the depth model presented in Figure 5, where the parameter $l = 1, 2, \text{and } 3$ in (5) is derived. Finally, the final dual-vehicle depth model is obtained by iterative learning through BP algorithm.

3.3. Window Confirmation and Identification. After training, the dual-vehicle depth model was tested in order to confirm whether the window contained the vehicle target. The window confirmation was divided into two categories.

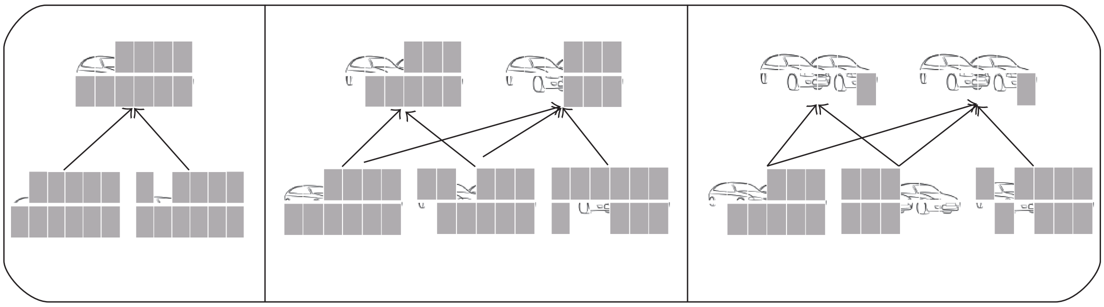
In the first type of window, a vehicle detection algorithm was used to check whether window contained a vehicle target based on a vehicle detection algorithm of a dual-vehicle deformable part model.

In the second type of window, the window of sliding scan of input image, namely, a series of detection windows, were generated in the area of image pyramid wherein the input image was not obtained by the sliding scanning method.

Combining these two window types, while reducing the threshold, we increased the number of the first-class windows that eventually returned, and these windows contained most of the vehicle targets in the image, which greatly reduced the number of missing cases. In addition, the dual-vehicle depth model was used to confirm that the two-part window can overcome the shortcomings of the dual-vehicle deformable part model and the advantages of the dual-vehicle depth model.

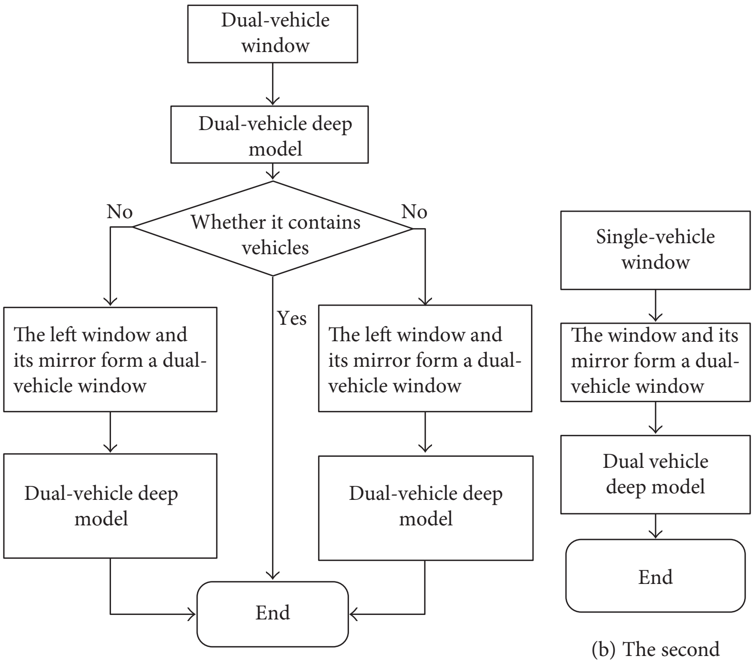
Since the window that is detected is present in a window containing a single vehicle and a window containing two vehicles, the input of the dual-vehicle deformable depth model can only be a window containing two vehicles, thus giving a method pair two cases of the window to confirm; the process is shown in Figure 9.

The window confirmation process was based on two overlapping (Figure 10(a)) or close vehicles (Figure 10(b)) that were directly combined into a dual-vehicle window, and then, the output was observed to determine whether



(a) First level to second level (b) Second level to third level (c) Third level to fouth level

Figure 8: The lower-level parts compose an upper-level part.


Dual-vehicle window
Dual-vehicle deep model
Whether it contains vehicles
No
No
Yes
The left window and its mirror form a dual-vehicle window
The left window and its mirror form a dual-vehicle window
Dual-vehicle deep model
Dual-vehicle deep model
End
Single-vehicle window
The window and its mirror form a dual-vehicle window
Dual vehicle deep model
End

(a) The first condition

(b) The second condition

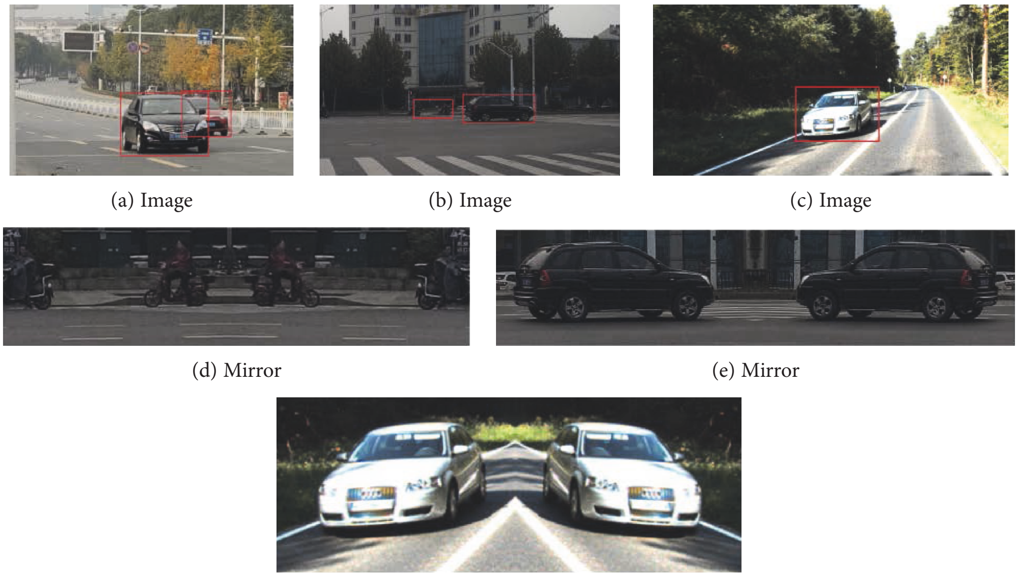
Figure 9: The flowchart of window confirmation.

the window contained two vehicles. The principle is as follows. When the window contains two vehicles, then the left and right subwindows of the window contain the vehicle target and the process ends; otherwise, the confirmation process continues. In Figure 10(e), the subwindow of Figure 10(b) is shown, wherein two windows are not divided into two subwindows, thus they are mirrored with their own dual-vehicle window. In that case, the observation of output continues in order to determine whether to include dual vehicles or not. If the judge shows that the description of the subwindow that contains the vehicle target is not included, the confirmation process ends. For a single detection window (Figure 10(c)), the window is directly mirrored to form a dual-vehicle window, and the vehicle is judged by observing the output. If it is included, it indicates that the window contains the vehicle target and process ends.

The presented method is used to determine whether each window contains a vehicle target and to achieve full advantage of dual-vehicle depth model in detecting a plurality of vehicle targets that are close to each other and to further reduce the leakage rate and false detection rate.

4. Experimental Results and Analysis

In order to verify the vehicle detection algorithm based on the dual-vehicle depth model, the algorithm was validated on the KITTI dataset. The experimental images were from the KITTI standard dataset. The KITTI training set contained 7481 images, which contained about 35,000 vehicles; KITTI test set contained 7518 images, which contained about 27,000 vehicles. The experiments were divided into two groups. In each experiment, 300 pictures are randomly selected for the KITTI standard dataset. The first group of experiments related to traditional vehicle detection algorithm, single-vehicle deformable part model, and dual-vehicle deformable part depth model, which were used



(a) Image

(b) Image

(c) Image

(d) Mirror

(e) Mirror

(f) Mirror

Figure 10: Ordinary images and mirror images.

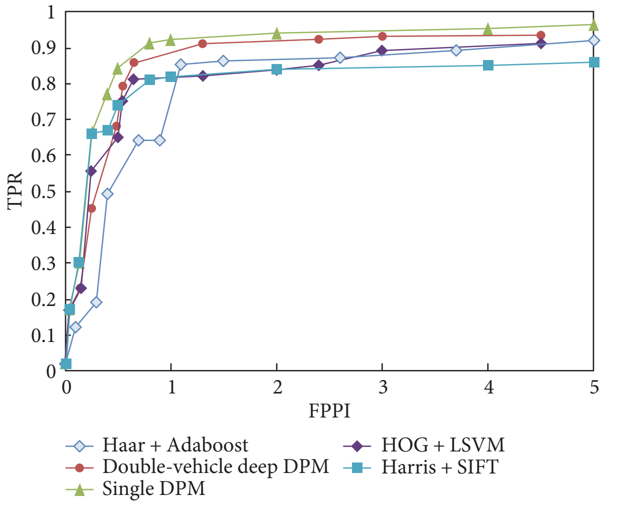


Figure 11: The relationship between the false positive rate (false positive per image (FPPI)) and the real rate (true positive rate (TPR)) for a single vehicle detection.

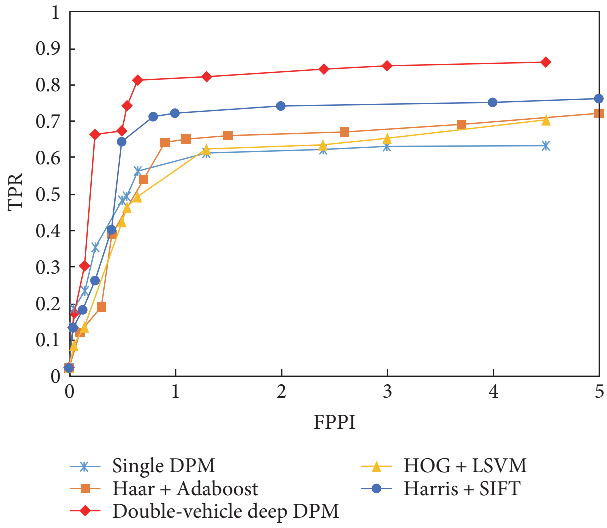


Figure 12: The relationship between the false positive rate (false positive per image (FPPI)) and the real rate (true positive rate (TPR)) for a multivehicle detection.

to compare the detection effect of a single vehicle without a shielded vehicle in the sample bank. The second group of experiments related to the comparison between traditional vehicle detection algorithm, single-vehicle deformable part model, and dual-vehicle deformable part depth model in order to examine the detection effect of multivehicle in the sample dataset that contained the vehicle. In particular, the traditional vehicle detection algorithm were the Haar and Adaboost classifier [13], the HOG and LSVM classifier [14], and the Haaris and SIFT algorithm [18]. The experimental platform consisted of Intel Core 2 Duo 2.67G processor, 4G memory, and we used the operating system Windows 7, the

Table 1: Comparison of KITTI standard dataset test results.

Method	The number of correctly identified vehicles	TPR (%)	Time (ms)
Haar and Adaboost	256	70.9	13.7
HOG and LSVM	309	85.6	12.5
Haaris and SIFT	281	77.8	10.1
Single DPM	293	81.1	9.2
Ours	**342**	**94.7**	**8.1**

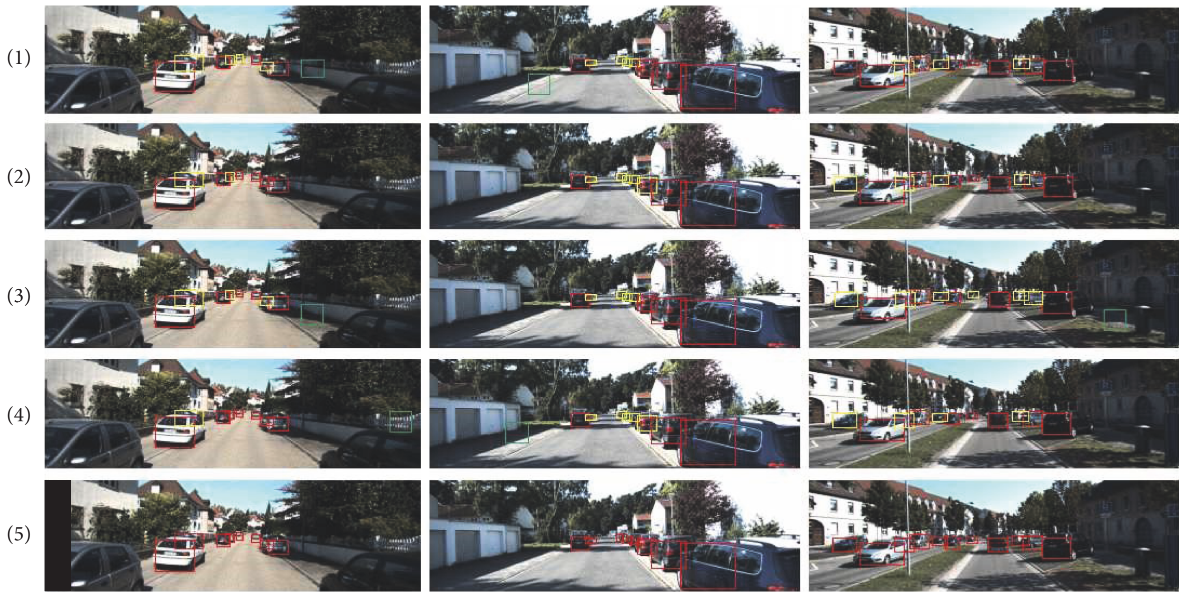


Figure 13: Vehicle detection when vehicles are occluded.

programming software Microsoft Visual 2013, and MATLAB 2015b. In the presentation of experimental results, green rectangular boxes denoted wrong vehicles, yellow rectangular boxes denoted the missing vehicles, and a red rectangular box denoted the target vehicles.

In addition, the ROC curve was used as a performance evaluation index for each vehicle detection. The above two groups of experiments were used to determine the relationship between the false positive rate (false positive per image (FPPI)) and the real rate (true positive rate (TPR)).

4.1. The First Experiment. In this experiment, the dual-vehicle deformable part depth model, the single-vehicle deformable part model and the traditional vehicle detection algorithm were compared in terms of the detection rate of a single vehicle in the sample dataset. The experimental results are shown in Figure 11, wherein it can be seen that FPPI is equal to 1 and detection rates of the dual-vehicle deformable component depth model, the single-vehicle deformable part model, the model presented in [13], the model presented in [14], and the model presented in [18] are 91.58%, 94.75%, 90.87%, 89.62% and 84.37%, respectively.

4.2. The Second Experiment. In this experiment, the dual-vehicle deformable part depth model designed in this paper was compared with the traditional single-vehicle deformable part model and the traditional vehicle detection algorithm using the KITTI standard dataset to achieve the partially blocked multivehicle detection situation. The experimental results are shown in Figure 12, wherein it can be seen that FPPI is equal to 1 and detection rates of the dual-vehicle deformable component depth model, the single-vehicle deformable part model, the model presented in [13], the model presented in [14], and the model presented in [18] are 86.37%, 61.30%, 71.34%, 67.45%, and 72.78%, respectively.

In the detection time, the performance of different algorithms are slightly different. The following table lists the traditional vehicle detection algorithm, single-vehicle deformable part model, and dual-vehicle deformable part depth model using 300 detection images to correctly identify the number of vehicles, the real rate, and the total time spent (as shown in Table 1).

In addition, in order to facilitate the comparison, the experimental results for KITTI standard dataset and vehicle detection are shown in Figure 13.

In Figure 13, it can be seen that the single-vehicle deformable part model and the traditional classifier have higher false detection rate and higher false alarm rate of vehicle detection if the vehicle is occluded. In the first group of experiments, the traditional detection algorithm and the single-vehicle deformable part model missed the left side white color blocked car, and the dual-vehicle deformable part model can be accurately detected. Likewise, in the second and third groups of experiments, traditional algorithms and single-vehicle deformable part models wrongly detected white walls and roadside debris as a vehicle, while the dual-vehicle deformable part depth model can effectively detect the obstructed vehicle in a plurality of perspectives in a multivehicle road condition; thus, the false detection rate is greatly reduced.

5. Conclusion

In this paper, two main problems of the vehicle detection algorithm are studied deeply. The false detection can easily occur in detection of multiple vehicles with close proximity or mutual occlusion. The depth vehicle detection algorithm is proposed to overcome mentioned problem. The experimental results have proven the effectiveness of proposed vehicle detection algorithm based on the dual-vehicle deformable part depth model, which uses a dual-vehicle depth model to convert the window obtained by vehicle detection algorithm and the window generated by sliding scanning of input image. Thus, by combining the advantages of dual-vehicle deformable part model and dual-vehicle depth model, a vehicle target with more severe occlusion can be detected without affecting the detection speed.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work has been supported by The National Natural Science Foundation of China (U1664258, U1564201, 61601203, and 61403172), Key Research and Development Program of Jiangsu Province (BE2016149), and Natural Science Foundation of Jiangsu Province (BK20140555).

References

[1] L. W. Tsai, J. W. Hsieh, and K. C. Fan, "Vehicle detection using normalized color and edge map," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 850–864, 2007.

[2] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 530–549, 2004.

[3] Z. Zhang, Y. Xu, J. Yang, X. Li, and D. Zhang, "A survey of sparse representation: algorithms and applications," *IEEE Access*, vol. 3, pp. 490–530, 2015.

[4] X. Wen, L. Shao, W. Fang, and Y. Xue, "Efficient feature selection and classification for vehicle detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 508–517, 2015.

[5] Y. Cai, Z. Liu, H. Wang, and X. Sun, "Saliency-based pedestrian detection in far infrared images," *IEEE Access*, vol. 5, pp. 5013–5019, 2017.

[6] F. Porikli, "Integral histogram: a fast way to extract histograms in Cartesian spaces," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 829–836, San Diego, CA, USA, 2005.

[7] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2, pp. 1491–1498, New York, NY, USA, 2006.

[8] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, pp. 401–408, Amsterdam, Netherlands, 2007.

[9] S. Maji, A. C. Berg, and J. Malik, "Efficient classification for additive kernel SVMs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 66–77, 2013.

[10] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.

[11] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, pp. I-511–I-518, Kauai, HI, USA, 2001.

[12] P. Viola, M. J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *International Journal of Computer Vision*, vol. 63, no. 2, pp. 153–161, 2005.

[13] S. Zhang, C. Bauckhage, and A. B. Cremers, "Informed Haar-like features improve pedestrian detection," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 947–954, Columbus, OH, USA, 2014.

[14] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, Anchorage, AK, USA, 2008.

[15] D. Park, D. Ramanan, and C. Fowlkes, "Multiresolution models for object detection," in *European Conference on Computer Vision*, pp. 241–254, Springer, Berlin, Heidelberg, 2010.

[16] W. Ouyang and X. Wang, "Single-pedestrian detection aided by multi-pedestrian detection," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3198–3205, Portland, OR, USA, 2013.

[17] J. Yan, Z. Lei, L. Wen, and S. Z. Li, "The fastest deformable part model for object detection," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2497–2504, Columbus, OH, USA, 2014.

[18] J. Y. Choi, K. S. Sung, and Y. K. Yang, "Multiple vehicles detection and tracking based on scale-invariant feature transform," in *2007 IEEE Intelligent Transportation Systems Conference*, pp. 528–533, Seattle, WA, USA, 2007.

[19] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," in *2013 IEEE International Conference on Computer Vision*, pp. 2056–2063, Sydney, NSW, Australia, 2013.

[20] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[21] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: integrated recognition, localization and detection using convolutional networks," 2013, http://arxiv.org/abs/1312.6229.

[22] P. Luo, Y. Tian, X. Wang, and X. Tang, "Switchable deep network for pedestrian detection," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 899–906, Columbus, OH, USA, 2014.

[23] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, "Vehicle detection in satellite images by hybrid deep convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 10, pp. 1797–1801, 2014.

[24] K. Wang, Z. Huang, and Z. Zhong, "Simultaneous multi-vehicle detection and tracking framework with pavement constraints based on machine learning and particle filter algorithm," *Chinese Journal of Mechanical Engineering*, vol. 27, no. 6, pp. 1169–1177, 2014.

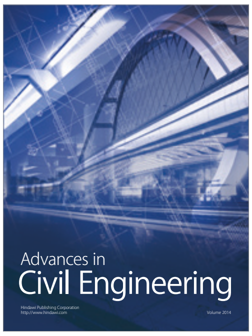
Journal of
Engineering

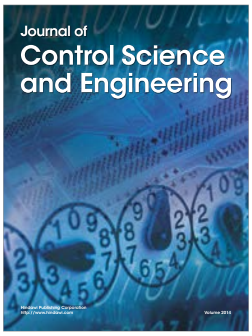
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

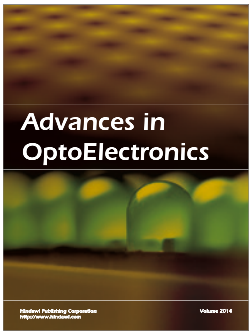
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

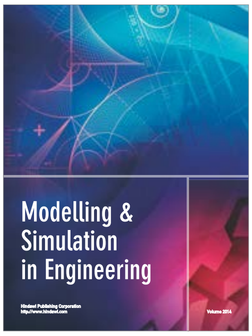
Journal of
Robotics

Journal of
Electrical and Computer Engineering

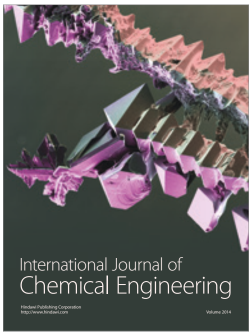

Advances in
OptoElectronics

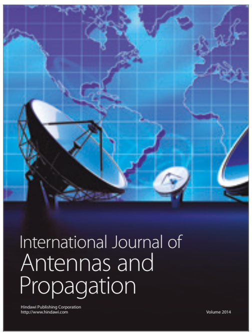
VLSI Design

International Journal of
Navigation and Observation

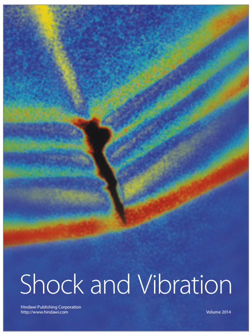

Modelling &
Simulation
in Engineering

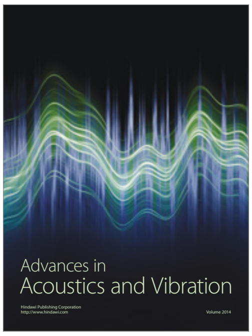
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration