Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2014, Article ID 876451, 9 pages
http://dx.doi.org/10.1155/2014/876451

*Research Article*

# A Multistep Framework for Vision Based Vehicle Detection

**Hai Wang**[1] **and Yingfeng Cai**[2]

[1] *School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China*
[2] *Automotive Engineering Research Institute, Jiangsu University, Zhenjiang 212013, China*

Correspondence should be addressed to Yingfeng Cai; caicaixiao0304@126.com

Received 1 May 2014; Revised 17 July 2014; Accepted 11 August 2014; Published 27 August 2014

Academic Editor: Yantao Shen

Vision based vehicle detection is a critical technology that plays an important role in not only vehicle active safety but also road video surveillance application. In this work, a multistep framework for vision based vehicle detection is proposed. In the first step, for vehicle candidate generation, a novel geometrical and coarse depth information based method is proposed. In the second step, for candidate verification, a deep architecture of deep belief network (DBN) for vehicle classification is trained. In the last step, a temporal analysis method based on the complexity and spatial information is used to further reduce miss and false detection. Experiments demonstrate that this framework is with high true positive (TP) rate as well as low false positive (FP) rate. On road experimental results demonstrate that the algorithm performs better than state-of-the-art vehicle detection algorithm in testing data sets.

## 1. Introduction

Advanced driver assistant system (ADAS) is developed to improve driver safety and comfort which require comprehensive perception and understanding of on-road environment. For example, stable and dynamic obstacles on road need to be detected accurately in real time so that safe driving space of ego vehicles can be determined. Recently, with the fast development of optical devices and hardware, vision based environment sensing has drawn much attention from researchers. Vehicle detection with in vehicle front-mounted vision system is one of the most important missions for vision based ADAS.

Robust vision based vehicle detection on the road with low miss detection rate and low false detection rate is with many challenges since highways, urban and city road are dynamic environment, in which the background and illuminations are dynamic and time vary. Besides, the shape, color, size and appearance of vehicles are with high variability. Making this task even difficult, the ego vehicle and target vehicles are generally in motion so that the size and location of target vehicles captured in image are diverse.

For robust vehicle detection, a two-step strategy proposed by Sun et al. [1] including candidate generation (CG) and candidate verification (CV) is often applied by many researchers. In CG step, image blocks in which vehicles might exist will be generated with easy and low time cost algorithms. In CV step, the candidate vehicles generated in CG will be verified with another relatively complex machine learning algorithm.

In this work, by mainly following the two-step strategy, a multilevel framework for vision based vehicle detection is proposed. In the first step, for vehicle candidate generation, a novel geometrical and coarse depth information based method is proposed. In the second step, for candidate verification, a deep architecture of DBN for vehicle classification is trained. The last step, to further reduce miss and false detection, a temporal analysis method based on the complexity and spatial information will be used. The overall framework proposed by our work is shown in Figure 1.

Our main contribution is mainly in three aspects: (1) a novel geometrical and coarse depth information based vehicle candidate generation method is proposed, which dramatically reduces the number of candidates needed to be verified thus speeding up the whole vehicle detection process. (2) The DBN based deep vehicle classifier is designed. (3) To deal with occasionally appeared miss detection and false detection, a temporal analysis method based on the complexity and spatial information is further proposed.

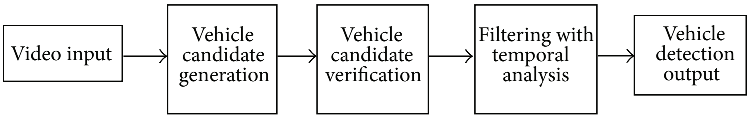
Video input
Vehicle candidate generation
Vehicle candidate verification
Filtering with temporal analysis
Vehicle detection output

Figure 1: Overall framework of vehicle detection.

## 2. Vehicle Candidate Generation

In vehicle candidate generation step, traversal search such as sliding window method is often used traditionally. Since traversal search will generate a huge number of candidates, it is not able to meet real-time requirement. Another CG method based on ground plane assumption is proposed by Kim and Cho, which dramatically decreases the candidate numbers [2]. However, this method performs poorly when the camera orientation changes or the road surface is a curve. In [3], Felzenszwalb and Huttenlocher use symmetry and edge based candidate generation method which cannot handle the partial vehicle occlusion situation.

In our work, a novel geometrical and coarse depth information based vehicle candidate generation method by integrating bottom and intermediate layer information of image is proposed. In bottom layer information extraction, original image is transformed to super-pixel form according to the pixel approximation. Then in intermediate layer information extraction, road image scene is mainly divided into three parts which are horizontal plane, vertical plane, and sky by geometrical information extraction. Meanwhile a coarse depth information estimation method is applied to get approximate depth information of single image. Finally, by combining the acquired geometrical and coarse depth information of single image, super pixels are clustered so that vehicle candidate can be generated. The vehicle candidate generation process is shown in Figure 2.

*2.1. Bottom Layer Information Extraction.* The bottom message is the image information which can be directly maintained without further process such as image pixel value, color, and so forth. Super pixel is an important expression of image underlying information. Super pixel is essentially an over-segmentation method which aggregated neighboring pixels with similar features into a group and name as one super pixel. Because the use of super-pixel segmentation method is able to better reflect the significance of human perception, it has a unique advantage in the field of object recognition tasks. Since firstly proposed in [3], there is now a large number of machine vision approaches that uses super pixels instead of pixels in a wide range of applications such as image segmentation [4], image analysis [5], image targeting [6], and other areas. In our application, the SLIC method described in [3] is applied to segment road image into super-pixel format (Figure 2(b)).

*2.2. Intermediate Layer Information Extraction*

*2.2.1. Geometrical Information Extraction.* Through statistical analysis of a large number of road pictures, it is found that flat plane in image usually belongs to road surface area, vertical plane in image tends to be the surface of on-road objects such as vehicles, trees, fence, and so forth, and the sky is often presented in the upper part of the image. If each image pixel can be identified as flat plane, vertical plane, or sky, it will provide rich information for vehicle candidate generation. Luckily, by characterizing each super pixel with color, position, and perspective effect and putting them into a pretrained regression AdaBoost classifier, Hoiem et al. successfully maintain the category of each super pixel [7]. Following Hoiem's contribution, in our work, images are divided into the aforementioned road pavement, vertical object, and sky. As shown in left bottom picture of Figure 2(c); road pavement is marked with green color, sky is marked with blue color, and vertical object is marked with brown as well as black "*X*".

*2.2.2. Coarse Depth Information Extraction.* To maintain depth information in images, there are many mature methods based on stereo vision but they are not suitable for our work with just one front-mounted camera. Recently, single image based depth information acquirement has received widespread attention. Literature [8, 9] have proposed accurate depth information obtaining algorithms based on single image, but these two methods need several seconds to achieve this which is difficult to meet the real time requirements.

Different from the application that needs accurate depth information, vehicle candidate generation tasks only need coarse depth information in the image. For this, a SVM classifier that can get coarse depth of image objects is proposed. A large number of training images is collected with stereo vision and marked with depth information. The depth information $d$ is roughly divided into three groups:

$$d_i = \begin{cases} \text{near}: \ 0 \sim 100\,\text{m} & i = 1 \\ \text{middium}: \ 100 \sim 150\,\text{m} & i = 2 \\ \text{far}: \ 150\,\text{m} \sim \infty & i = 3. \end{cases} \quad (1)$$

For training, all the sample images are divided into $6 * 6$ small grid and the following three features of image samples are extracted to form feature vector: (1) mean value and histogram of three channels in HSV image space, (2) gabor features based texture gradient feature, and (3) category of grids (sky, ground, etc.). The classifier $C$ is trained with libsvm toolbox [10] and the output of $C$ is depth category $d_i$. As shown in right bottom picture of Figure 2(c), three different coarse depths are marked with different colors.

Based on the coarse depth information, the range in pixel level of vehicle candidate can be seen in Table 1.

*2.3. Vehicle Candidate Generation Strategy.* In Section 2.2, images are segmented as super pixels. Meanwhile geometrical

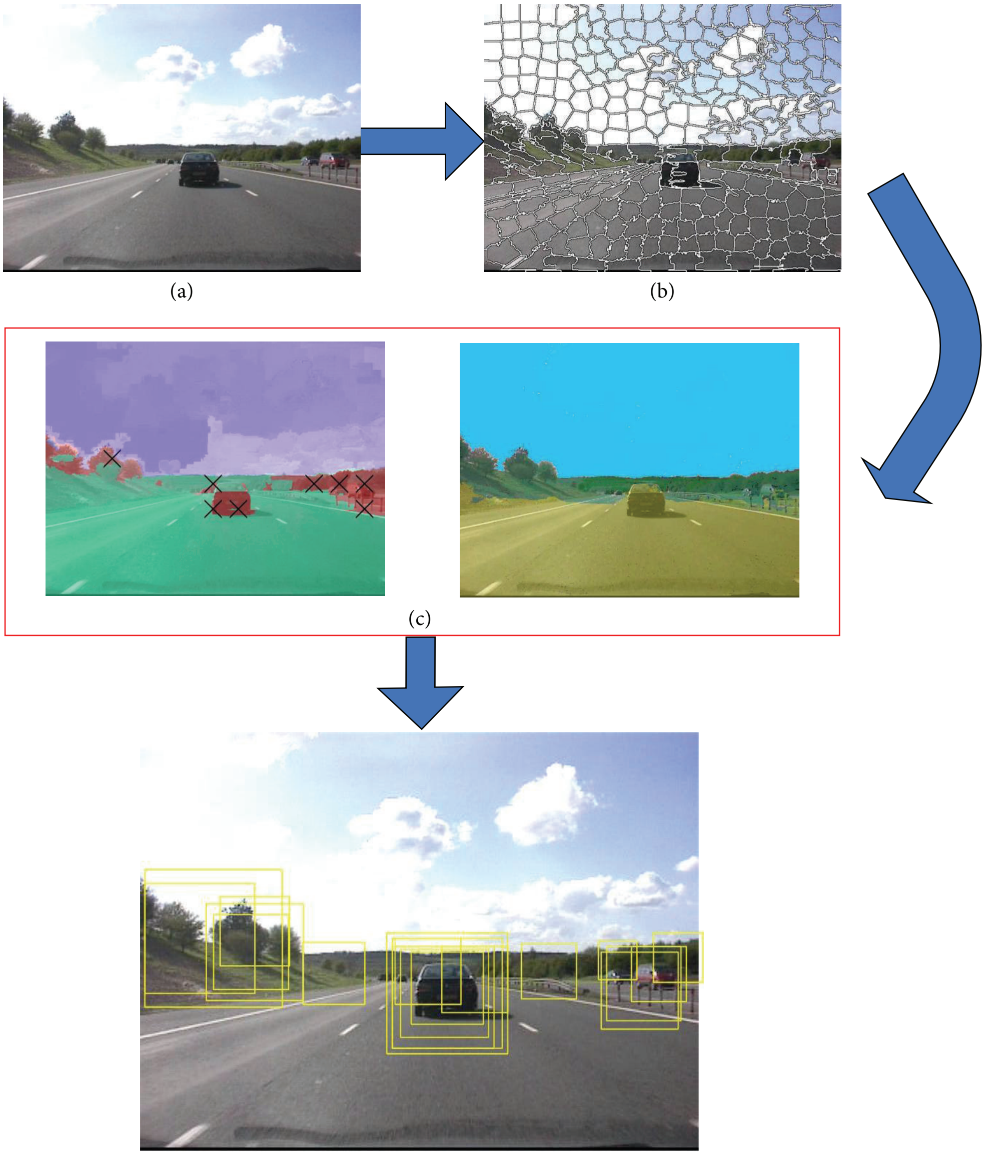

Figure 2: One sample of vehicle candidate generation.

Table 1: Vehicle size range in pixel level.

| Distance | Min size | Max size |
|---|---|---|
| 0~100 m | 50 × 50 | 200 × 200 |
| 100~150 m | 25 × 25 | 70 × 70 |
| 150 m~ ∞ | 10 × 10 | 30 × 30 |

and coarse depth information of images is made. In this section, firstly, the super pixels that satisfy the requirements of the vehicle will be picked out. Then, the vehicle candidate will be generated with the selected super pixels by clustering.

The super pixel selection is mainly based on the priori knowledge as follows.

(1) Vertical constraints: the super pixel that might belong to vehicle should be present in or adjacent to the vertical plane.

(2) Ground constraints: the super pixel that might belong to vehicle should connect to ground area.

(3) Depth constraints: the super pixel that might belong to the same depth can be clustered together.

(4) Size constraints: the super pixel that might belong to vehicle should not be out of the range of vehicle size in image.

By applying the four rules, a clustering strategy is proposed so that proper super pixels can be selected and grouped as vehicle candidate. The clustering method is as follows.

*Algorithm 1* (Clustering Method for Vehicle Candidate Generation). Consider the following steps:

*Step 1.* Start from super pixel group $M$ that belongs to vertical plane and $M$ connecting to ground area.

*Step 2.* For $m \in M$, $N$ is super pixel group that is near $m$ and belongs to vertical plane. If $N$ is empty or all $m$ are processed, STOP. Otherwise, For $n \in N$, find the super pixel pair $m, n$ with minimal Euclidean distance $d(m, n)$.

*Step 3.* If $m, n$ satisfies (a) $m, n$ is with the same depth range and (b) $m \cup n$ is not out of the range of vehicle size in image, then $m, n$ will merge into a new super pixel $m = m \cup n$.

*Step 4.* Remove $n$ from $N$.

*Step 5.* If $m$ satisfies the constraint of image vehicle pixel size range shown in Table 1, jump to Step 2. If $m$ do not satisfy the constraint, $m$ is considered as a vehicle candidate and jump to Step 2.

Figure 3 shows an example of the proposed clustering Method for vehicle candidate generation. Figure 3(a) is the original image for processing and Figures 3(b)–3(e) are the clustering process.

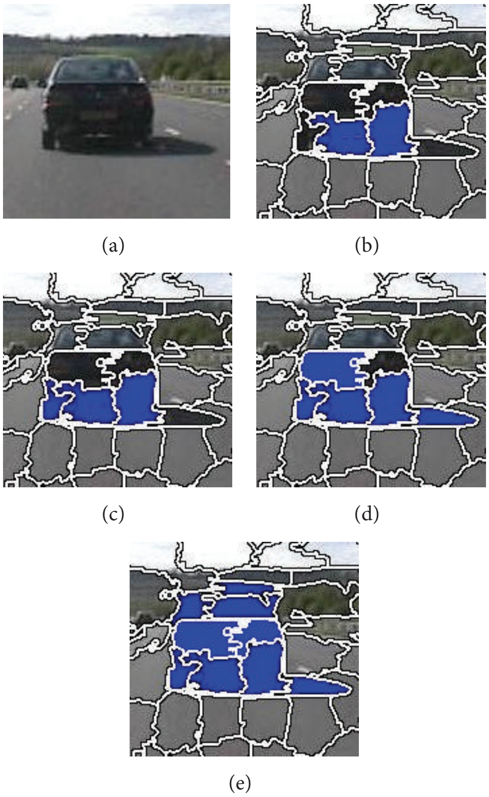

Figure 3: An example of the proposed clustering method.

## 3. Vehicle Candidate Verification

In this section, a deep belief network (DBN) based vehicle candidate verification algorithm is proposed.

Machine learning is proved to be the most useful method for vehicle candidate verification task. Support vector machines (SVM) and AdaBoost are the most two common classifiers that are used for training vehicle detector [11–20]. However, classifiers such as SVM and AdaBoost all belong to shallow learning model because both of them can be modeled as structure with one input layer, one hidden layer, and one output layer. On the other hand, deep learning is another class of machine learning technique, where hierarchical architectures are exploited for representation learning and pattern classification. Superior to those shallow models, deep learning is able to learn multiple levels of representation and abstraction of image data.

There are many subclasses of deep architecture in which deep belief networks (DBN) modal is a typical deep learning structure which is first proposed by Hinton et al. [21]. DBN has demonstrated its success in MNIST classification. In [22], a modified DBN is developed in which a Boltzmann machine is used on the top layer, which is used in a 3D object recognition task. In our work, by using DBN, a classifier is trained for vehicle candidate verification tasks.

In Section 3.1, the overall architecture of the DBN classifier for vehicle candidate verification will be introduced. In Sections 3.2 and 3.3, the training method of the whole DBN for vehicle candidate verification will be deduced.

### 3.1. Deep Belief Network (DBN) for Vehicle Candidate Verification.

Let $X$ be the set of data samples including vehicle images and nonvehicle images. Assuming that $X$ is consisting of $K$ samples which is shown as follows:

$$X = [\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_k, \ldots, \mathbf{X}_K]. \tag{2}$$

In $X$, $\mathbf{X}_k$ is training samples and is in the image space $\mathbf{R}^{I \times J}$. Meanwhile, $Y$ means the labels are corresponding to $X$, which can be written as

$$Y = [y_1, y_2, \ldots, y_k, \ldots, y_K]. \tag{3}$$

In $Y$, $y_k$ is the label vector of $\mathbf{X}_k$. If $\mathbf{X}_k$ is belonging to vehicles, $y_k = (1, 0)$. On the contrary, $y_k = (0, 1)$.

The ultimate purpose of vehicle candidate verification task is to learn a mapping function from training data $X$ to the label data $Y$ based on the given training set, so that this mapping function is able to classify unknown images between vehicle and nonvehicle.

Based on the task described above, DBN architecture is applied to address this problem. Figure 4 shows the overall architecture of DBN. A fully interconnected directed belief network including one visible input layer $V^1$, $N$ hidden layers $H^1, \ldots, H^N$ and one visible label layer La at the top. The visible input layer $V^1$ maintains $I \times J$ neural and equal to the dimension of training feature which is the original 2D image pixel values of training samples in this application. An the top, the La layer just has two units which is equal to the classes this application would like to classify. Till now, the problem is formulated to search for the optimum parameter space $\theta$ of this DBN.

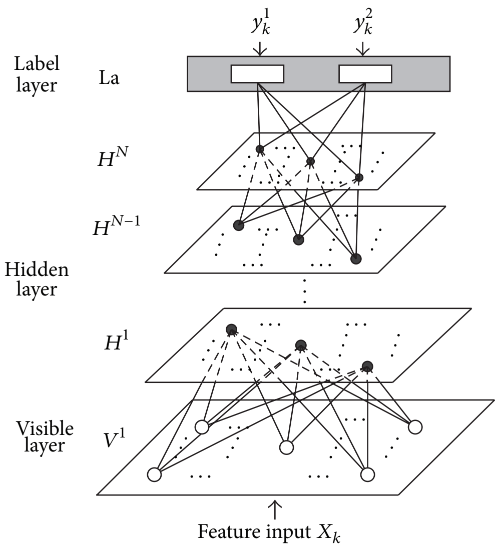

FIGURE 4: Proposed DBN for vehicle candidate verification.

The main learning process of the proposed DBN is with two steps.

(1) The parameters of the two adjacent layers will be refined with the greedy-wise reconstruction method. Repeat Step 1 till all the parameters of hidden lays are fixed. Here, Step 1 is so-called pretraining process.

(2) The whole DBN will be fine-tuned with the La layer information based on back propagation. Here, Step 2 can be viewed as supervised training step.

*3.2. Pretraining with Greedy Layer-Wise Reconstruction Method.* Assume that the size of the upper layer is $P \times Q$. In this subsection the parameters of the two adjacent layers will be refined with the greedy-wise reconstruction method proposed by Hinton et al. [21]. To illustrate this pretraining process, we take the visible input layer $V^1$ and the first hidden layer $H^1$ for example.

The visible input layer $V^1$ and the first hidden layer $H^1$ contract a restrict Boltzmann machine (RBM). $I \times J$ is the neural number in $V^1$ and $P \times Q$ is that of $H^1$. The energy of the state $(v^1, h^1)$ in this RBM is

$$\begin{aligned} E\left(\mathbf{v}^1, \mathbf{h}^1, \theta^1\right) &= -\left(\mathbf{v}^1 \mathbf{A} \mathbf{h}^1 + \mathbf{b}^1 \mathbf{v}^1 + \mathbf{c}^1 \mathbf{h}^1\right) \\ &= -\sum_{i=1,j=1}^{i\le I, j\le J} \sum_{p=1,q=1}^{p\le P, q\le Q} v_{ij}^1 A_{ij,pq}^1 h_{pq}^1 \\ &\quad - \sum_{i=1,j=1}^{i\le I, j\le J} b_{ij}^1 v_{ij}^1 - \sum_{p=1,q=1}^{p\le P, q\le Q} c_{pq}^1 h_{pq}^1, \end{aligned} \tag{4}$$

in which $\theta^1 = (\mathbf{A}^1, \mathbf{b}^1, \mathbf{c}^1)$ are the parameters between the visible input layer $V^1$ and the first hidden layer $H^1$, $A_{ij,pq}^1$ is the symmetric weights from input neural $(i, j)$ in $V^1$ to the hidden neural $(p, q)$ in $H^1$, and $b_{ij}^1$ and $c_{pq}^1$ are the $(i, j)$th and $(p, q)$th bias of $V^1$ and $H^1$. So this RBM is with the joint distribution as follows:

$$P\left(\mathbf{v}^1, \mathbf{h}^1; \theta^1\right) = \frac{1}{Z} e^{-E(\mathbf{v}^1, \mathbf{h}^1; \theta^1)} = \frac{e^{-E(\mathbf{v}^1, \mathbf{h}^1; \theta^1)}}{\sum_{v^1} \sum_{h^1} e^{-E(\mathbf{v}^1, \mathbf{h}^1; \theta^1)}}, \tag{5}$$

where $Z$ is the normalization parameter, and the probability that $\mathbf{v}^1$ is assigned to $V^1$ of this modal is

$$P\left(\mathbf{v}^1\right) = \frac{1}{Z} \sum_{h^1} e^{-E(\mathbf{v}^1, \mathbf{h}^1; \theta^1)} = \frac{\sum_{h^1} e^{-E(\mathbf{v}^1, \mathbf{h}^1; \theta^1)}}{\sum_{v^1} \sum_{h^1} e^{-E(\mathbf{v}^1, \mathbf{h}^1; \theta^1)}}. \tag{6}$$

After that, the conditional distributions over visible input state $\mathbf{v}^1$ in layer $V^1$ and hidden state $h^1$ in $H^1$ are able to be given by the logistic function, respectively:

$$\begin{aligned} p\left(\mathbf{h}^1 \mid \mathbf{v}^1\right) &= \prod_{p,q} p\left(h_{pq}^1 \mid \mathbf{v}^1\right), p\left(h_{pq}^1 \mid \mathbf{v}^1\right) \\ &= \sigma\left(\sum_{i=1,j=1}^{i\le I, j\le J} v_{ij}^1 A_{ij,pq}^1 + c_{pq}^1\right) \\ p\left(\mathbf{v}^1 \mid \mathbf{h}^1\right) &= \prod_{i,j} p\left(\mathbf{v}_{ij}^1 \mid \mathbf{h}^1\right), p\left(\mathbf{v}_{ij}^1 \mid \mathbf{h}^1\right) \\ &= \sigma\left(\sum_{p=1,q=1}^{p\le P, q\le Q} h_{pq}^1 A_{ij,pq}^1 + b_{ij}^1\right), \end{aligned} \tag{7}$$

where $\sigma(x) = 1/(1 + \exp(-x))$.

At last, the weights and biases are able to be updated step by step from a random Gaussian distribution value $A_{ij,pq}^1(0)$, $b_{ij}^1(0)$ and $c_{pq}^1(0)$ with contrastive divergence algorithm [23], and the updating formulations are

$$\begin{aligned} A_{ij,pq}^1 &= \vartheta A_{ij,pq}^1 + \varepsilon_A \\ &\quad \times \left(\left\langle v_{ij}^1(0)\, h_{ij}^1(0)\right\rangle_{\text{data}} - \left\langle v_{ij}^1(t)\, h_{ij}^1(t)\right\rangle_{\text{recon}}\right) \\ b_{ij}^1 &= \vartheta b_{ij}^1 + \varepsilon_b \left(v_{ij}^1(0) - v_{ij}^1(t)\right) \\ c_{pq}^1 &= \vartheta c_{pq}^1 + \varepsilon_c \left(h_{pq}^1(0) - h_{pq}^1(t)\right), \end{aligned} \tag{8}$$

in which $\langle\cdot\rangle_{\text{data}}$ means the expectation with respect to the data distribution, $\langle\cdot\rangle_{\text{recon}}$ means the reconstruction distribution after one step, and $t$ is step size which is set to $t = 1$, typically.

Above, the pretraining process is demonstrated by taking the visible input layer $V^1$ and the first hidden layer $H^1$, for example. Indeed, the whole pretraining process will be taken from low layer groups $(V^1, H^1)$ to up layer groups $(H^{n-1}, H^n)$ one by one.

*3.3. Global Fine-Tuning.* In the above unsurprised pretraining process, the greedy layer-wise algorithm is used to learn the DBN parameters. In this subsection, a traditional back propagation algorithm will be used to fine-tune the parameters $\theta = [\mathbf{A}, \mathbf{b}, \mathbf{c}]$ with the information of label layer La.

Since good parameters initiation has been maintained in the pretraining process, back propagation is just utilized to finely adjust the parameters so that local optimum parameters $\theta^* = [\mathbf{A}^*, \mathbf{b}^*, \mathbf{c}^*]$ can be achieved. In this stage, the learning objection is to minimize the classification error $[-\sum_t \mathbf{y}_t \log \hat{\mathbf{y}}_t]$, where $\mathbf{y}_t$ and $\hat{\mathbf{y}}_t$ are the real label and output label of data $\mathbf{X}_t$ in layer $N$.

## 4. Temporal Analysis Using Complexity and Spatial Information

After the vehicle candidate verification step proposed in last section, most vehicles in frames will be detected. However, due to search window scale sparsity and inadequate detector performance, there may be some miss detections and false detections in some frames. By observation, it is found that most miss detections and false detections appear just occasionally, so that a temporal analysis using complexity and spatial information is proposed to refine vehicle detect results.

The temporal analysis is with two compositions. First, for detected target $i$ in frame $t$, if it is judged that the target $i$ also appeared in $t-1$ and $t-2$ frames by using a similarity function, this target $i$ is considered as true positive, otherwise it is considered as false detection and needs to be eliminated. Then, for verified detected target $j$ in frame $t-1$, also take advantage of the similarity function to determine whether there is a corresponding target in frame $t$. If not, miss detection is considered and the area having the highest similarity and is above a threshold is regarded as the detection target.

The similarity between any two targets $S({v_t}^{(i)}, {v_{t+1}}^{(j)})$ is defined as follows:

$$S\left({v_t}^{(i)}, {v_{t+1}}^{(j)}\right) = 0.3 S_C\left({v_t}^{(i)}, {v_{t+1}}^{(j)}\right) + 0.7 S_S\left({v_t}^{(i)}, {v_{t+1}}^{(j)}\right), \tag{9}$$

in which $S_C({v_t}^{(i)}, {v_{t+1}}^{(j)})$ means the complexity similarity between target $i$ in frame $t$ and target $j$ in frame $t+1$. $S_S({v_t}^{(i)}, {v_{t+1}}^{(j)})$ means the spatialsimilarity. Complexity similarity and spatialsimilarity are defined as follows.

*(a) Complexity Similarity.* There are multiple ways to calculate image complexity, in which edges proportion is a good measurement. Let $n$ be pixel number of an image and $\tilde{n}$ be the number of pixels belonging to edge. Then image complexity is defined as

$$C = \frac{\tilde{n}}{n}. \tag{10}$$

The complexity ratio of two images is considered as similarity function. The complexity similarity between target $i$ in frame $t$ and target $j$ in frame $t+1$ is

$$S_C\left({v_t}^{(i)}, {v_{t+1}}^{(j)}\right) = \frac{\min\left({C_t}^{(i)}, {C_{t+1}}^{(j)}\right)}{\max\left({C_t}^{(i)}, {C_{t+1}}^{(j)}\right)}. \tag{11}$$

*(b) Spatial Similarity.* Vehicle movement in video is a continuous process and the time interval between two consecutive frames is very short (around 40 ms) so the vehicle motion does not usually change dramatically in such a very short period of time and vehicles in two consecutive frames are with very small displacement and deformation. So the detection window size and centroid coordinates are used to build spatialsimilarity measurement function

$$\begin{aligned} S_S&\left({v_t}^{(i)}, {v_{t+1}}^{(j)}\right) \\ &= \exp\left\{-\left[\left(\frac{\left(x_i - x_j\right)^2}{\sigma_x^2} + \frac{\left(y_i - y_j\right)^2}{\sigma_y^2}\right)\right.\right. \\ &\qquad \left.\left. + \left(\frac{\left(H_i - H_j\right)^2}{\sigma_H^2} + \frac{\left(W_i - W_j\right)^2}{\sigma_W^2}\right)\right]\right\}, \end{aligned} \tag{12}$$

in which $x, y$ is centroid coordinates of detection window, $W, H$ means detection window size, and $\sigma_x, \sigma_y, \sigma_W$, and $\sigma_H$ are constant factors.

## 5. Experiments and Analysis

The proposed multistep vehicle detection framework is tested on PETS2001 dataset and many video clips captured on the highway by our project groups. In the experiment, the hidden number of the DBN is set to 2, the neural number of two hidden layers are $24 \times 24$ and $16 \times 16$, respectively, and $\vartheta$ is set as 0.8. Besides, image samples for training are all resized to $32 \times 32$. The frames are all $640 \times 480$ resolution and the experiment is made in our Advantech industrial computer.

Some of the vehicle candidate generation effects are shown in Figure 5 where (a) is the original image, (b) is the sky in the vertical plane, the ground plane geometry information extracted from the original image, (c) is the depth information by using pretrained classifier, and (d) is generated candidates.

Based on the vehicle candidate generation results, the DBN based vehicle candidate verification method is applied further. The vehicle candidate verification results are shown in Figure 6. The left column is generated vehicle candidates marked with yellow while the right column is the verified vehicles marked with blue rectangle.

Finally, by applying temporal analysis, some miss detection can also be detected. As shown in Figure 7, the blue window means normal detected vehicles by the DBN vehicle detector while the dash green window means the miss detected vehicles but is corrected by the temporal analysis.

Our method is tested in more than 10 road captured videos including different times and different weather conditions. The overall vehicle detection effects are shown in Table 2 as well as some state-of-the-art vehicle detection effects.

From the results shown in Table 2, it can be seen that the proposed vehicle detection framework maintains the lowest false positive (FP) rate while reaching the second highest true positive (TP) rate which is 0.24% lower than that of Southall's stereo vision based method. Meanwhile, by using monocular vision, the processing speed of our method is much faster than Southall's.

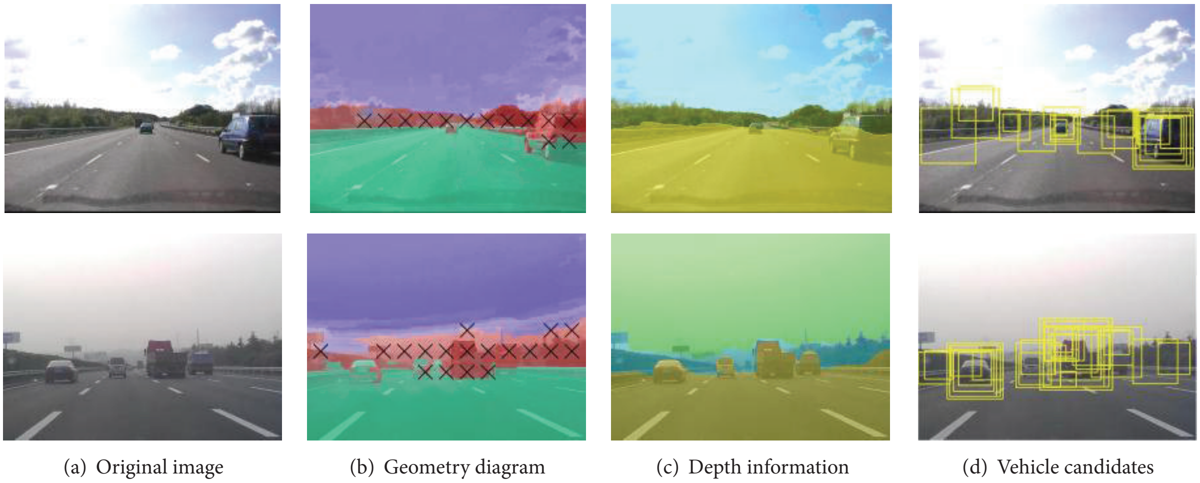
(a) Original image
(b) Geometry diagram
(c) Depth information
(d) Vehicle candidates

Figure 5: Two vehicle candidate generation results.

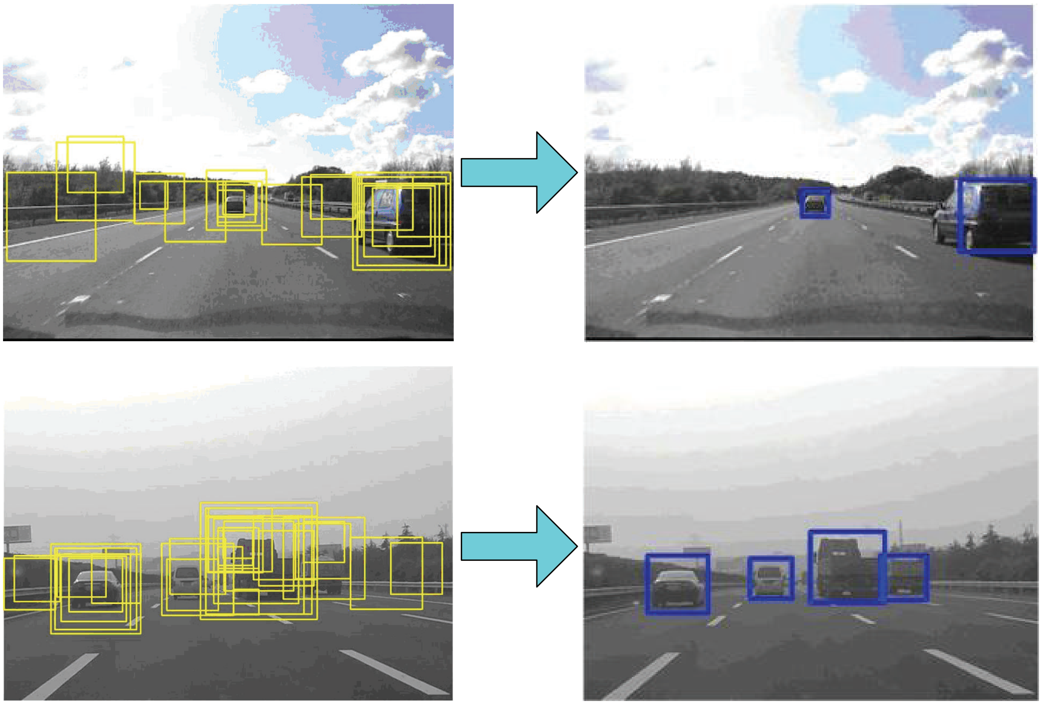

Figure 6: Vehicle candidate verification results.

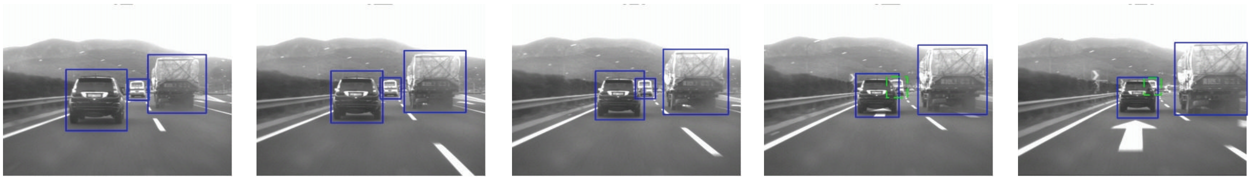

Figure 7: Vehicle filtering with temporal analysis.

Table 2: Overall vehicle detection effects comparison.

| Authors | TP rate | FP rate | Notes |
|---|---|---|---|
| Bergmiller et al. [24] | 81.48% | 16.7% | Monocular vision; |
| Alonso et al. [25] | 92.63% | 3.68% | Monocular vision; |
| Sun et al. [12] | 98.5% | 2% | Monocular vision; For their most effective classifier |
| Southall et al. [26] | **99.37%** | 1.77% | Stereo vision; 16 HZ frame rate; |
| **Our Framework** | 99.13% | **0.78%** | Monocular vision; 27 HZ frame rate; |

## 6. Conclusion

In this work, a multistep framework for vision based vehicle detection is proposed. In the first step, for vehicle candidate generation, a novel geometrical and coarse depth information based method is proposed. In the second step, for candidate verification, a deep architecture of DBN for vehicle classification is trained. In the last step, a temporal analysis method based on the complexity and spatial information is used to further reduce missed and false detection. Experiments demonstrate that this framework is with high TP rate as well as low FP rate.

## Conflict of Interests

The authors declare that they have no conflict of interests.

## Acknowledgments

This research was supported in part by the National Natural Science Foundation of China under Grants 51305167, 61203244 and 61403172 Information Technology Research Program of Transport, Ministry of China, under Grant 2013364836900, Natural Science Foundation of Jiangsu Province under Grant BK20140555, Jiangsu Province Colleges and Universities, Natural Science Foundation, under Grant 13KJD520003 and Jiangsu University Scientific Research Foundation for Senior Professionals under Grant 12JDG010, 1291120026.

## References

[1] Z. Sun, G. Bebis, and R. Miller, “On-road vehicle detection using evolutionary Gabor filter optimization,” *IEEE Transactions on Intelligent Transportation System*, vol. 6, no. 2, pp. 125–137, 2005.

[2] G. Kim and J. Cho, “Vision-based vehicle detection and inter-vehicle distance estimation,” in *Proceedings of the 12th International Conference on Control, Automation and Systems (ICCAS ’12)*, pp. 625–629, October 2012.

[3] P. F. Felzenszwalb and D. P. Huttenlocher, “Efficient graph-based image segmentation,” *International Journal of Computer Vision*, vol. 59, no. 2, pp. 167–181, 2004.

[4] X. He, R. Zemel, and D. Ray, “Learning and incorporating top-down cues in image segmentation,” in *Proceedings of the 9th European Conference on Computer Vision*, pp. 338–351, 2006.

[5] J. Tighe and S. Lazebnik, “Superparsing: scalable nonparametric image parsing with superpixels,” in *Proceedings of the 11th European Conference on Computer Vision: Part V (ECCV ’10)*, pp. 352–365, 2010.

[6] B. Fulkerson, A. Vedaldi, and S. Soatto, “Class segmentation and object localization with superpixel neighborhoods,” in *Proceedings of the International Conference on Computer Vision*, pp. 670–677, 2009.

[7] D. Hoiem, A. A. Efros, and M. Hebert, “Recovering surface layout from an image,” *International Journal of Computer Vision*, vol. 75, no. 1, pp. 151–172, 2007.

[8] A. Saxena, M. Sun, and A. Y. Ng, “Make3D: learning 3D scene structure from a single still image,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 824–840, 2009.

[9] B. Liu, S. Gould, and D. Koller, “Single image depth estimation from predicted semantic labels,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR ’10)*, pp. 1253–1260, June 2010.

[10] C. Chang and C. Lin, “LIBSVM: a Library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, article 27, 2011.

[11] W. Liu, X. Wen, B. Duan, H. Yuan, and N. Wang, “Rear vehicle detection and tracking for lane change assist,” in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV ’07)*, pp. 252–257, June 2007.

[12] Z. Sun, G. Bebis, and R. Miller, “Monocular precrash vehicle detection: features and classifiers,” *IEEE Transactions on Image Processing*, vol. 15, no. 7, pp. 2019–2034, 2006.

[13] S. Sivaraman and M. M. Trivedi, “Active learning for on-road vehicle detection: a comparative study,” *Machine Vision and Applications*, vol. 25, no. 3, pp. 599–611, 2014.

[14] S. S. Teoh and T. Bräunl, “Symmetry-based monocular vehicle detection system,” *Machine Vision and Applications*, vol. 23, no. 5, pp. 831–842, 2012.

[15] R. Sindoori, K. S. Ravichandran, and B. Santhi, “Adaboost technique for vehicle detection in aerial surveillance,” *International Journal of Engineering and Technology*, vol. 5, no. 2, pp. 765–769, 2013.

[16] J. Cui, F. Liu, Z. Li, and Z. Jia, “Vehicle localisation using a single camera,” in *Proceedings of the IEEE Intelligent Vehicles Symposium (IV’10)*, pp. 871–876, June 2010.

[17] T. T. Son and S. Mita, “Car detection using multi-feature selection for varying poses,” in *Proceedings of the IEEE Intelligent Vehicles Symposium*, pp. 507–512, June 2009.

[18] D. Acunzo, Y. Zhu, B. Xie, and G. Baratoff, “Context-adaptive approach for vehicle detection under varying lighting conditions,” in *Proceedings of the 10th International IEEE Conference on Intelligent Transportation Systems (ITSC ’07)*, pp. 654–660, Seattle, Wash, USA, October 2007.

[19] C. T. Lin, S. C. Hsu, J. F. Lee et al., “Boosted vehicle detection using local and global features,” *Journal of Signal & Information Processing*, vol. 4, no. 3, 2013.

[20] O. Ludwig Jr. and U. Nunes, “Improving the generalization properties of neural networks: an application to vehicle detection,” in *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems (ITSC ’08)*, pp. 310–315, December 2008.

[21] 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.

[22] V. Nair and G. E. Hinton, “3D object recognition with deep belief nets,” in *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems (NIPS ’09)*, pp. 1339–1347, December 2009.

[23] F. Wood and G. E. Hinton, “Training products of experts by minimizing contrastive divergence,” Tech. Rep., Brown University, 2012.

[24] P. Bergmiller, M. Botsch, J. Speth, and U. Hofmann, “Vehicle rear detection in images with generalized radial-basis-function classifiers,” in *Proceeding of the 2008 IEEE Intelligent Vehicles Symposium (IV ’08)*, pp. 226–233, Eindhoven, The Netherlands, June 2008.

[25] D. Alonso, L. Salgado, and M. Nieto, “Robust vehicle detection through multidimensional classification for on board video based systems,” in *Proceedings of the 14th IEEE International Conference on Image Processing (ICIP ’07)*, pp. IV321–IV324, September 2007.

[26] B. Southall, M. Bansal, and J. Eledath, “Real-time vehicle detection for highway driving,” in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 541–548, June 2009.