Hindawi Publishing Corporation Journal of Sensors Volume 2016, Article ID 3403451, 8 pages http://dx.doi.org/10.1155/2016/3403451



# Research Article

# Night-Time Vehicle Sensing in Far Infrared Image with Deep Learning

# Hai Wang, Yingfeng Cai, Xiaobo Chen, and Long Chen

<sup>1</sup>School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China <sup>2</sup>Automotive Engineering Research Institute, Jiangsu University, Zhenjiang 212013, China

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

Received 27 May 2015; Accepted 3 August 2015

Academic Editor: Fadi Dornaika

Copyright © 2016 Hai Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The use of night vision systems in vehicles is becoming increasingly common. Several approaches using infrared sensors have been proposed in the literature to detect vehicles in far infrared (FIR) images. However, these systems still have low vehicle detection rates and performance could be improved. This paper presents a novel method to detect vehicles using a far infrared automotive sensor. Firstly, vehicle candidates are generated using a constant threshold from the infrared frame. Contours are then generated by using a local adaptive threshold based on maximum distance, which decreases the number of processing regions for classification and reduces the false positive rate. Finally, vehicle candidates are verified using a deep belief network (DBN) based classifier. The detection rate is 93.9% which is achieved on a database of 5000 images and video streams. This result is approximately a 2.5% improvement on previously reported methods and the false detection rate is also the lowest among them.

## 1. Introduction

On average, at least one person globally dies in a vehicle crash every minute. Auto accidents also injure at least ten million people each year, with two or three million of these people seriously injured [1]. To address this problem, Advanced Driver Assistant Systems (ADAS) are used more and more often to provide assistance and supplementary information for drivers. Current ADAS developments include many functions such as lane departure warning, forward collision warning, parking assistance systems, and night vision enhancement [2].

The vision sensor is one of the most popular sensors in ADAS and many algorithms are designed to use it including vehicle detection, lane detection, pedestrian detection, and traffic sign recognition [3–6]. Among these methods, vehicle detection is a popular research area. Monocular vision is often used in this task. The Sivaraman model considers vehicle detection as a two-class classification problem and trains an Adaboost classifier with a Haar feature [7]. Stereo vision is another common method. Hermes et al. use stereo vision and propose a vehicle detection method based on

density map clustering [8]. Additionally, motion based algorithms have also been used for overtaking vehicle detection [9].

The current reality is that a large number of road accidents occur during times of low visibility such as at night time. Statistical data also demonstrates that more than half of accidents causing fatalities occur at night. However, most existing vehicle detection systems and algorithms are more focused on daylight vehicle detection with visible spectrum cameras. Although some researchers have put effort into night-time vehicle detection with vehicle lamp detection, the detection results are often affected by many factors such as low illumination, light reflection on rainy days, and the camera exposure time [10]. To compensate for these limitations of visible spectrum cameras, far infrared (FIR) sensors are becoming more and more frequently employed for night-time vehicle detection tasks. They do not require any illumination source and rely purely on heat signatures from the environment to produce a gray scale image. Therefore, FIR sensors can detect the infrared heat signatures generated by vehicle parts such as the engine, wheels, and exhaust pipe. A group of typical images captured at night



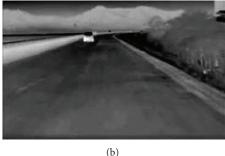


FIGURE 1: Night views using visible spectra (a) and FIR spectrum cameras (b).

by a visible spectrum camera and FIR camera is shown in Figure 1.

Compared with the visible spectra camera, the image captured by the FIR spectrum camera lacks color and detailed information. Up until now, not much work has been done specifically on vehicle detection in FIR images. In 2000, Andreone (University of Parma) designed an FIR camera based vehicle detection prototype car and detected vehicles by mainly relying on size, shape, and the content distribution of the image high intensity area [11]. In 2010, Besbes et al. introduced a machine learning framework to this task and designed a vehicle detection method with SURF-based features and an SVM classifier [12]. There are also many researchers that focus on the similar problem of detecting pedestrians in an FIR image. Most of the proposed methods use a two-class classification framework and use different features such as HOG and different classifiers such as SVM and Adaboost [13, 14].

A review of the approaches proposed in the literature finds that vehicle detection in an FIR image is most commonly performed using a two-class classification framework. However, a newly proposed approach in pattern recognition named deep learning has also been used in rare studies. Classifiers such as SVM and Adaboost are all shallow learning models because they can be modeled as a structure with one input layer, one hidden layer, and one output layer. Deep learning refers to a class of machine learning techniques which exploit hierarchical architectures for representation learning and pattern classification. In contrast to shallow models, deep learning has the ability to learn multiple levels of representation and abstraction that enables better understanding of the image data. Deep belief networks (DBN) model is a typical deep learning structure which is first proposed by Hinton and has demonstrated its success in simple image classification tasks of MNIST [15].

In this work, a deep learning based vehicle detection algorithm for FIR images is proposed. In Section 2, the vehicle candidate generation method in FIR image will be described. Candidate shape segmentation will then be performed with a contour generation method in Section 3. In Section 4, vehicle verification will be implemented using shape feature vectors and a deep belief network. The experiments and conclusion will be presented in Sections 5 and 6 separately.

#### 2. Vehicle Candidate Generation

For vehicle detection in an image, two steps are usually performed which are vehicle candidate generation (VCG) and vehicle candidate verification (VCV) [16]. In VCG, all image areas which have any probability of being vehicles will be selected. In this step, prior knowledge of vehicles may often be used, such as horizontal/vertical edges, symmetry, color, shadow, and texture. In VCV, the image areas selected in VCG will be further verified to eliminate those which are not vehicles. In this step, a two-class classification framework is often used and a classifier that can distinguish vehicles from nonvehicles will be trained from a set of training images. In our work, we will also follow this two-step framework.

The FIR images reflect information on the temperature of objects. Therefore, FIR image cannot show detailed information available in the visible domain such as texture, color, and shadows. Obviously, the existing vehicle detection algorithms used with visible spectrum cameras are not suitable for far infrared cameras because of the inherent differences between images generated by cameras in the IR and visible spectra. Due to this, vehicle candidate generation mainly focuses on hot spots in the image.

In the VCG step of our method, a low threshold is firstly applied to the pixel values. The pixels with values that are lower than the threshold are considered to be low temperature areas and are removed, while hot spots in the images are preserved. Since the relationship between the temperature of the object and the brightness in FIR image is constant for a specific FIR camera, the threshold can be chosen manually. In our application, the threshold is set at 150 which corresponds to a temperature of 30°C. Figure 2 shows the original FIR image and the processed image separately.

In the processed image, there obviously exist many hot spots that are not belonging to vehicles such as the hot road surface, hot road lamps, or other sources. To eliminate these sources of interference, a connected region searching algorithm is applied on the processed image and all regions that do not satisfy the rules below are considered to be nonvehicle hot spots:

(1) Rule 1: connected regions with a length/width ratio below 0.3 or above 1.5 are considered to be nonvehicle hot spots.

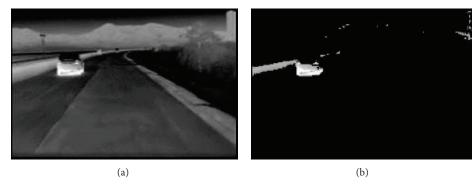


FIGURE 2: Hot area detection in an FIR image (processed image on right).



FIGURE 3: Vehicle candidate generation.

(2) Rule 2: connected regions with less than 120 pixels are considered to be nonvehicle hot spots.

This further processing eliminates many nonvehicle hot spots. The remaining hot spots will all be considered to be vehicle candidates. As seen in Figure 3, two vehicle candidates are identified in this particular FIR image.

## 3. Vehicle Candidate Contour Segmentation

In a traditional visible spectrum image-based vehicle detection framework, a two-class classifier will be trained and used to classify all vehicle candidates that are identified. However, traditional image feature descriptors such as Haar and HOG are more suitable for local pixel information representation and in an FIR image; the detailed information is relatively small. Therefore, features such as the vehicle candidate shape or contours will be used, which are based on global information rather than pixel values. In this section, contour segmentation is firstly required to segment the full vehicle candidate objects.

One way to find continuous contours in a gray scale image is using edge detection to find continuous contours in the edge image. However, since edges are generally discrete and independent, the contours that are generated may appear broken and branching. Another method transforms the scale image to a binary image first and then uses the chain code method to generate contours in the binary image. For this method, it is critical that binary image is of good quality.

Traditional binary segmentation methods are usually based on a global threshold obtained from a pixel histogram. This type of method is suitable when the inner foreground image has small variations, such as pedestrians in an FIR image. However, for our application, the lower part of the vehicle such as the wheels is usually brighter than the upper part of the vehicle such as the windows. Therefore, a global threshold based method could easily eliminate the upper part of vehicles in an FIR image.

Based on the analysis above, a maximum distance based local adaptive threshold determination method is proposed to generate the binary segmentation. This method firstly sets a global threshold based on a histogram to get a binary image. Then, a maximum distance based local threshold will be further decided for each subregion around the edge of the binary image that has been generated. The main steps of the local threshold determination method are given below.

- (1) Set the processing area of each vehicle candidate. Specifically, the original vehicle candidate areas are expanded twice vertically and 1.5 times horizontally.
- (2) Apply a median filter on the vehicle candidate area to eliminate obvious noise.
- (3) Set I(x, y) as the vehicle candidate area image. Set a global threshold using the OTSU method. Fill in the blank area of the binary image and get the edge image of I(x, y) which is set as E(x, y).
- (4) Choose the edge points with a five-point gap in E(x, y). Then, set the selected edge points as the center and find the selected edge point  $5 \times 5$  neighborhood region to obtain a new local threshold for this small region, which is decided using the maximum distance based method. This method is based on the concept that a threshold should divide a gray scale histogram into two parts. The best threshold produces the largest gap between the mean value of the two parts and the mean value of the whole image. The distance measurement function is

$$R(T) = \frac{\left[\sum_{i=1}^{T} i P_i - \sum_{i=1}^{M} i P_i P_i(T)\right]^2}{P_i(T) \left[1 - P_i(T)\right]},$$
 (1)

where M is the total number of gray scale levels,  $P_i$  is the proportion of gray scale level i, and T is the threshold,  $P_i(T) = \sum_{i=1}^{T} P_i$ .

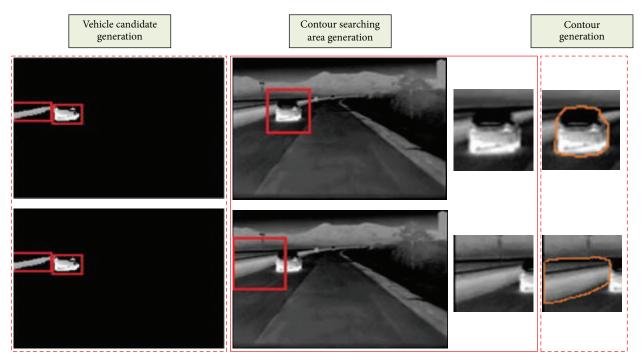


FIGURE 4: Vehicle candidate contour generation.

Then, transform the small region to a binary image with this new threshold and set the new binary image to be bi(x, y).

(5) Repeat step (4) until the complete areas around each of the selected edge points are processed and a full contour of the vehicle candidates is obtained.

The processed result of the contour generation method is shown in Figure 4.

#### 4. Vehicle Candidate Verification

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

The machine learning based method is very popular and effective for vehicle candidate verification tasks in the visible spectrum. Within the many existing machine learning methods, SVM (support vector machines) and Adaboost are the two most common classifiers [17–20]. However, most classifiers including these two classifiers are based on a shallow learning model and can be modeled as a structure consisting of one input layer, one output layer, and a single hidden layer. Recently, a new machine learning structure called deep learning has been proposed, which has a hierarchical architecture that can be exploited for representation learning and pattern classification. The deep model is superior to the existing shallow models as it can learn multiple levels of representation and abstraction of image data.

There are many types of deep architectures such as deep belief networks (DBN) and deep convolution neural networks (DCNN) which use DBN as a typical deep learning structure, as first proposed by Hinton et al. [21] and used in many tasks such as MNIST classification, 3D object recognition, and voice recognition. In our work, DBN is applied

and a classifier is trained for vehicle candidate verification tasks.

4.1. Deep Belief Network (DBN) for Vehicle Candidate Verification. In this subsection, the overall architecture of the DBN classifier for vehicle candidate verification will be firstly introduced.

Let *X* be the set of training samples which contain vehicle contour images and nonvehicle contour images which are generated manually by our group. *X* consists of *K* samples which are shown below:

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

In X,  $X_k$  is a training sample and all samples are resized to  $I \times J$ . Y represents the labels corresponding to X, which can be written as

$$Y = [y_1, y_2, \dots, y_k, \dots, y_K].$$
 (3)

In *Y*,  $y_k$  is the label vector of  $\mathbf{X}_k$ . If  $\mathbf{X}_k$  belongs to a vehicle,  $y_k = (1, 0)$ . Otherwise,  $y_k = (0, 1)$ .

The purpose of the vehicle candidate verification task is to learn the mapping function from the training data *X* to the label data *Y* based on a given training set. With this trained mapping function, unknown contour images can be classified as either vehicle or nonvehicle.

In this task, a DBN architecture is applied to address this problem. Figure 5 shows the overall architecture of the DBN. It is a fully interconnected deep 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 a  $I \times J$  neural number which is equal to

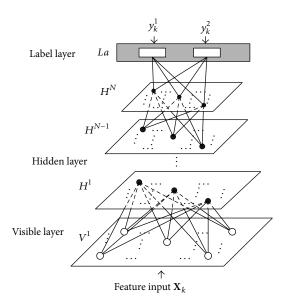


FIGURE 5: Proposed DBN for vehicle candidate verification.

the dimensions of the training feature, that is, the pixel values of the training samples. On the top, the La layer just has two states which can be either (1,0) or (0,1).

The learning process of the DBN has two main steps. In the first step, the parameters of the two adjacent layers will be refined with the greedy-wise reconstruction method. This step will be repeated until the parameters of all the hidden layers are fixed. This first step is also called the pretraining process. In the second step, the whole pretrained DBN will be fine-tuned with the La layer information based on back propagation. This second step can be considered to be the supervised training step.

4.2. Pretraining Method. In this subsection, the training method of the whole DBN for vehicle candidate verification will be presented.

Assume that the size of the upper layer is  $P \times Q$  whose size is decided in bilinear projection as mentioned in Zhong et al.'s work [22]. In this step, the parameters of the two adjacent layers will be refined using the greedy-wise reconstruction method proposed by Hinton et al. [21]. The visible input layer  $V^1$  and the first hidden layer  $H^1$  are taken here as an example; other adjacent layers use the same pretraining method.

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 state  $(\mathbf{v}^1, \mathbf{h}^1)$  in this RBM is

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

where  $\theta^1 = (\mathbf{A}^1, \mathbf{b}^1, \mathbf{c}^1)$  are the parameters between the visible input layer  $V^1$  and the first hidden layer  $H^1$ ,  $A^1_{ij,pq}$  are the symmetric weights from input neural (i,j) in  $V^1$  to the hidden neural (p,q) in  $H^1$ , and  $b^1_{ij}$  and  $c^1_{pq}$  are the (i,j)th and (p,q)th bias of  $V^1$  and  $H^1$ . So, the RBM has the following joint distribution:

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

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

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

Then, the conditional distribution over the visible input state  $\mathbf{v}^1$  in layer  $V^1$  and the hidden state  $\mathbf{h}^1$  in  $H^1$  can be given by the logistic function:

$$p\left(\mathbf{h}^{1} \mid \mathbf{v}^{1}\right) = \prod_{p,q} p\left(h_{pq}^{1} \mid \mathbf{v}^{1}\right), p\left(h_{pq}^{1} \mathbf{v}^{1}\right)$$

$$= \sigma\left(\sum_{i=1,j=1}^{i \leq I,j \leq J} v_{ij}^{1} A_{ij,pq}^{1} + c_{pq}^{1}\right),$$

$$p\left(\mathbf{v}^{1} \mid \mathbf{h}^{1}\right) = \prod_{i,j} p\left(v_{ij}^{1} \mid \mathbf{h}^{1}\right), p\left(v_{ij}^{1} \mid \mathbf{h}^{1}\right)$$

$$= \sigma\left(\sum_{p=1,q=1}^{p \leq P,q \leq Q} h_{pq}^{1} A_{ij,pq}^{1} + b_{ij}^{1}\right),$$

$$(7)$$

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

Finally, the weights and biases can be updated step by step using random Gaussian distribution values  $A^1_{ij,pq}(0)$ ,  $b^1_{ij}(0)$ , and  $c^1_{pq}(0)$  with the Contrastive Divergence algorithm [23]. The updating formulas are

$$\begin{split} A_{ij,pq}^{1} &= \vartheta A_{ij,pq}^{1} \\ &= \vartheta A_{ij,pq}^{1} \\ &+ \varepsilon_{A} \left( \left\langle v_{ij}^{1} \left( 0 \right) h_{ij}^{1} \left( 0 \right) \right\rangle_{\text{data}} - \left\langle v_{ij}^{1} \left( t \right) h_{ij}^{1} \left( t \right) \right\rangle_{\text{recon}} \right), \quad (8) \\ b_{ij}^{1} &= \vartheta b_{ij}^{1} + \varepsilon_{b} \left( v_{ij}^{1} \left( 0 \right) - v_{ij}^{1} \left( t \right) \right), \\ c_{pq}^{1} &= \vartheta c_{pq}^{1} + \varepsilon_{c} \left( h_{pq}^{1} \left( 0 \right) - h_{pq}^{1} \left( t \right) \right). \end{split}$$

Here,  $\langle \cdot \rangle_{\rm data}$  represents the expectation with respect to the data distribution and  $\langle \cdot \rangle_{\rm recon}$  represents the reconstruction distribution after one step. The step size t is set to 1.

As mentioned previously, the whole pretraining process will be performed on lower layer groups  $(V^1, H^1)$  to upper layer groups  $(H^{n-1}, H^n)$  one at a time.

4.3. Global Fine-Tuning. In the above unsupervised pretraining process, the greedy layer-wise algorithm is used to



FIGURE 6: SAT NV628 FIR camera.

learn the DBN parameters. In this subsection, a traditional back propagation algorithm will be used to fine-tune the parameters  $\theta = [\mathbf{A}, \mathbf{b}, \mathbf{c}]$  using the information of the label layer La.

Since the pretraining process has already identified strong initial parameters, the back propagation step is just used to finely adjust the parameters so that local optimum parameters  $\theta^* = [\mathbf{A}^*, \mathbf{b}^*, \mathbf{c}^*]$  can be found. At this stage, the learning objective is to minimize the classification error  $[-\sum_t \mathbf{y}_t \log \widehat{\mathbf{y}}_t]$ , where  $\mathbf{y}_t$  and  $\widehat{\mathbf{y}}_t$  are the real label and the output label of data  $\mathbf{X}_t$  in layer N.

#### 5. Experiments and Analysis

5.1. DBN Based Vehicle Verification Effect. The proposed DBN based vehicle verification method is trained on our image dataset captured by a SAT NV628 FIR camera as shown in Figure 6. The total number of samples for training and testing are 2700 and 500, respectively.

By using the proposed method, four different architectures of 2D-DBN are applied. They all contain one visible layer and one label layer, but with one, two, three, and four hidden layers, respectively. In training, the critical parameters of the proposed 2D-DBN in experiments are set to  $\alpha=0.4$  and  $\theta=0.75$  and the image samples for training are all resized to  $32\times32$  pixels.

The detection results of the four 2D-DBN architectures and two common shallow models (SVM and Adaboost) are shown in Table 1. It is observed that the 2D-DBN with three hidden layers maintains the highest detection rate in the test set. It is also seen that deep architecture performs much better than the existing shallow models.

5.2. System Overall Effect. All the methods described below are tested using the same image dataset containing 5000 images captured by our group. In the dataset, there are 6382 vehicles and around 37% in near range (less than 25 m), 61% in medium range (25 m to 75 m), and 12% in far range (more than 75 m). Some of the vehicle candidate generation effects are shown in left column of Figure 7. Based on the

Table 1: Vehicle detection rates with different DBN structures and two shallow models.

Classifier types	Correct detection	Correct rate
2D-DBN (one hidden layer)	455/500	91.0%
2D-DBN (two hidden layers)	469/500	93.8%
2D-DBN (three hidden layers)	474/500	94.8%
2D-DBN (four hidden layers)	468/500	93.60%
HOG + SVM	438/500	87.6%
Haar + Adaboost	442/500	88.4%

Table 2: Overall vehicle detection effects comparison.

Authors	TP rate	FP rate	Notes
Andreone et al. [11]	85.8%	8.70%	30 Hz frame rate
Besbes et al. [12]	91.4%	3.48%	24 Hz frame rate
Our framework	93.9%	1.13%	21 Hz frame rate

vehicle candidate generation results, the DBN based vehicle candidate verification method is further applied. Some of the vehicle candidate verification results are shown in Figure 7. The left column shows identified vehicle candidates marked in red and the right column shows the verified vehicles marked with a blue rectangle.

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 is seen that the proposed vehicle detection framework exhibits the lowest false positive (FP) rate while achieving the highest true positive (TP) rate, which is 2.5% higher than that of Bassem's method.

Figure 8 shows a group of vehicle sensing results in a continuous video. The blue rectangles represent correct vehicle detection and the red rectangles represent missed detection or false detection. From the results, it can be seen that some light spots with similar shapes to vehicles are recognized to be vehicles, such as in the fourth image. Vehicles which are not in the rear view are easily missed. Besides, the strong occlusion between different vehicles easily causes missed detection because they may be decided as one object in vehicle candidate generation step. Generally, most vehicles in the rear view are detected correctly and pedestrians and bicycles are not falsely detected.

# 6. Conclusion

In this work, a new method is proposed for night-time vehicle detection in far infrared images. Compared with existing methods, a maximum distance based local adaptive threshold determination method is proposed to generate the vehicle candidate and a deep learning framework is introduced to perform vehicle candidate verification. Overall, this two-step vehicle detection method achieves the highest vehicle detection rate compared with existed state-of-the-art methods. Additionally, the processing time is below 50 ms per frame which satisfies requirements for real-time applications.

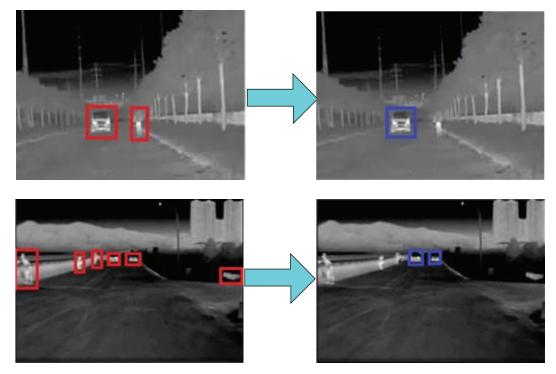


FIGURE 7: Vehicle candidate verification results.

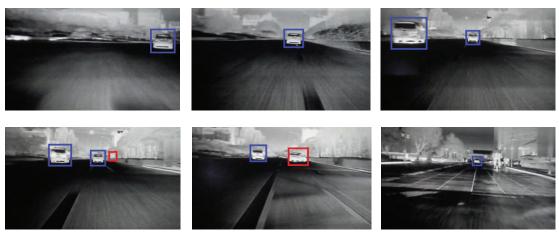


FIGURE 8: Vehicle sensing results in a continuous video.

#### **Conflict of Interests**

The authors declare that they have no conflict of interests.

# Acknowledgments

This work has been supported by the National Natural Science Foundation of China under the Grants 61403172, 61203244, and 51305167, China Postdoctoral Science Foundation (2014M561592), China Postdoctoral Science Foundation Special Funding (2015T80511), Information Technology Research Program of Transport Ministry of China under the Grant 2013364836900, Natural Science Foundation of Jiangsu Province (BK20140555), and Jiangsu University Scientific

Research Foundation for Senior Professionals (12JDG010, 14JDG028).

### References

- [1] http://asirt.org/Initiatives/Informing-Road-Users/Road-Safety-Facts/Road-Crash-Statistics.
- [2] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: a survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1773–1795, 2013.
- [3] H. Wang, C. Yuan, and Y. Cai, "Smart road vehicle sensing system based on monocular vision," *Optik*, vol. 126, no. 4, pp. 386–390, 2015.

[4] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: a survey," *Machine Vision and Applications*, vol. 25, no. 3, pp. 727–745, 2014.

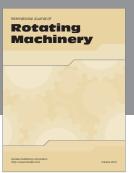
- [5] R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Ten years of pedestrian detection, what have we learned?" in *Computer Vision—ECCV 2014 Workshops*, vol. 8926 of *Lecture Notes in Computer Science*, pp. 613–627, Springer, 2015.
- [6] F. Zaklouta and B. Stanciulescu, "Real-time traffic sign recognition in three stages," *Robotics and Autonomous Systems*, vol. 62, no. 1, pp. 16–24, 2014.
- [7] S. Sivaraman and M. M. Trivedi, "A general active-learning framework for on-road vehicle recognition and tracking," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 2, pp. 267–276, 2010.
- [8] C. Hermes, J. Einhaus, M. Hahn, C. Wöhler, and F. Kummert, "Vehicle tracking and motion prediction in complex urban scenarios," in *Proceedings of the IEEE Intelligent Vehicles Symposium* (*IV '10*), pp. 26–33, IEEE, June 2010.
- [9] A. Ramirez, E. Ohn-Bar, and M. Trivedi, "Integrating motion and appearance for overtaking vehicle detection," in *Proceedings* of the 25th IEEE Intelligent Vehicles Symposium, pp. 96–101, IEEE, June 2014.
- [10] J. Arróspide, L. Salgado, M. Nieto, and F. Jaureguizar, "On-board robust vehicle detection and tracking using adaptive quality evaluation," in *Proceedings of the 15th IEEE International Conference on Image Processing (ICIP '08)*, pp. 2008–2011, IEEE, San Diego, Calif, USA, October 2008.
- [11] L. Andreone, P. Antonello, M. Bertozzi, A. Broggi, A. Fascioli, and D. Ranzato, "Vehicle detection and localization in infra-red images," in *Proceedings of the IEEE 5th International Conference* on Intelligent Transportation Systems, pp. 141–146, Singapore, 2002.
- [12] B. Besbes, A. Apatean, A. Rogozan, and A. Bensrhair, "Combining SURF-based local and global features for road obstacle recognition in far infrared images," in *Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems (ITSC '10)*, pp. 1869–1874, September 2010.
- [13] P. Hurney, P. Waldron, F. Morgan, E. Jones, and M. Glavin, "Night-time pedestrian classification with histograms of oriented gradients-local binary patterns vectors," *IET Intelligent Transport Systems*, vol. 9, no. 1, pp. 75–85, 2015.
- [14] Q. Liu, J. Zhuang, and S. Kong, "Detection of pedestrians at night time using learning-based method and head validation," in *Proceedings of the IEEE International Conference on Imaging* Systems and Techniques (IST '12), pp. 398–402, IEEE, July 2012.
- [15] O. Pauplin and J. Jiang, "DBN-based structural learning and optimisation for automated handwritten character recognition," *Pattern Recognition Letters*, vol. 33, no. 6, pp. 685–692, 2012.
- [16] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: a review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 5, pp. 694–711, 2006.
- [17] 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.
- [18] S. S. Teoh and T. Bräunl, "Symmetry-based monocular vehicle detection system," *Machine Vision and Applications*, vol. 23, no. 5, pp. 831–842, 2012.
- [19] R. Sindoori, K. S. Ravichandran, and B. Santhi, "Adaboost technique for vehicle detection in aerial surveillance," *International Journal of Engineering & Technology*, vol. 5, no. 2, pp. 765–769, 2013.

[20] 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.

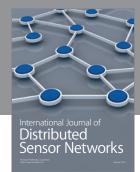
- [21] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.
- [22] S. Zhong, Y. Liu, and Y. Liu, "Bilinear deep learning for image classification," in *Proceedings of the 19th ACM International Conference on Multimedia*, pp. 343–352, ACM, 2011.
- [23] F. Wood and G. E. Hinton, "Training products of experts by minimizing contrastive divergence," Tech. Rep., Brown University, Providence, RI, USA, 2012.











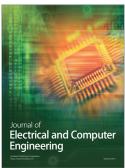


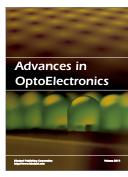




Submit your manuscripts at http://www.hindawi.com











International Journal of Antennas and

Propagation





