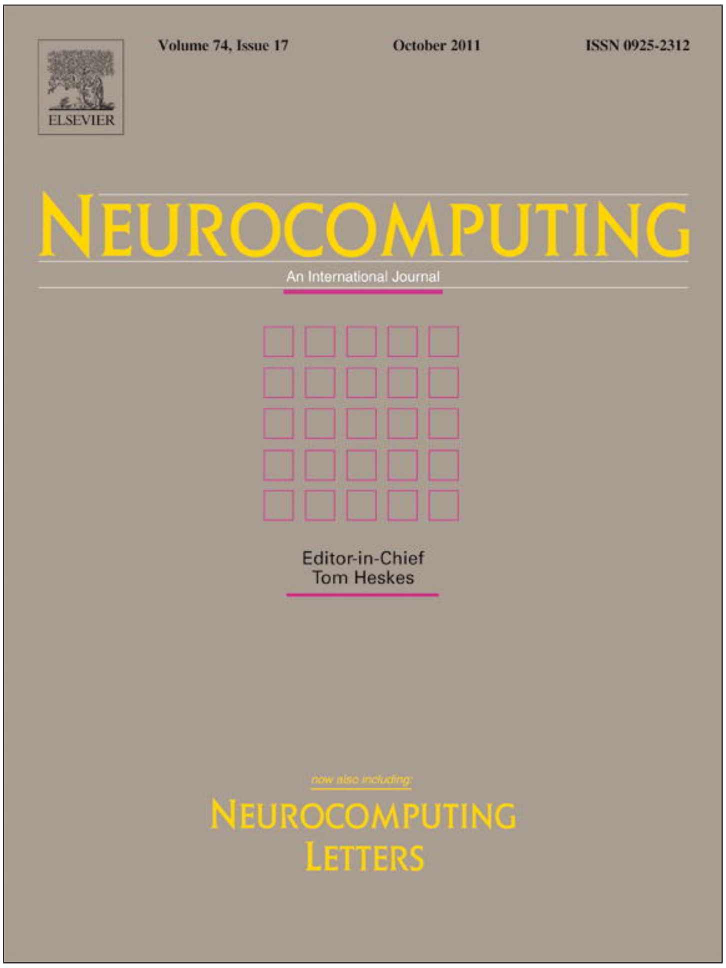

Volume 74, Issue 17
October 2011
ISSN 0925-2312
ELSEVIER
NEUROCOMPUTING
An International Journal
Editor-in-Chief
Tom Heskes
now also including:
NEUROCOMPUTING
LETTERS

Neurocomputing 74 (2011) 3343–3350

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Rapid pedestrian detection in unseen scenes

Xianbin Cao [a,1], Zhong Wang [a], Pingkun Yan [b,*], Xuelong Li [b]

[a] University of Science and Technology of China, Hefei, 230026, PR China
[b] Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, Shaanxi, PR China.

ARTICLE INFO

Article history:
Received 10 March 2011
Received in revised form
3 May 2011
Accepted 16 May 2011
Communicated by Liang Wang
Available online 2 July 2011

Keywords:
Pedestrian detection system
Adaptive detector
cascade classifier
Cross entropy
Ternary detection pattern

ABSTRACT

In this paper, a rapid adaptive pedestrian detection method based on cascade classifier with ternary pattern is proposed. The proposed method achieves its goal by employing the following three new strategies: (1) A method for adjusting the key parameters of the trained cascade classifier dynamically for detecting pedestrians in unseen scenes using only a small amount of labeled data from the new scenes. (2) An efficient optimization method is proposed, based on the cross entropy method and a priori knowledge of the scenes, to solve the classifier parameter optimization problem. (3) In order to further speed up pedestrian detection in unseen scenes, each strong classifier in the cascade employs a ternary detection pattern. In our experiments, two significantly different datasets, AHHF and NICTA, were used as the training set and testing set, respectively. The experimental results showed that the proposed method can quickly adapt a previously trained detector for pedestrian detection in various scenes compared with other existing methods.

1. Introduction

One of the applications of pedestrian detection systems is to detect pedestrians on moving vehicles in real-time under various scenes. The detector trained using training scenes may not reflect the characteristics of the new unseen scenes, which often leads to unsatisfactory detection results in unseen scenes. Therefore, a key problem is how to quickly and effectively adapt the detector to dynamic changing scenes for pedestrian detection.

Pedestrian detection methods based on classification are currently the mainstream in this field and the existing methods can be divided into two categories according to their ways of dealing with pedestrian detection in unseen scenes. One category is to use the trained classifier to detect pedestrians in unseen scenes directly [1–10]. The effectiveness of the methods is based on the premise that the difference between the training scenes and the unseen scenes is limited in general, so the originally trained classifier is still able to get acceptable detection performance in unseen scenes. However, the detection performance may decrease rapidly while the differences between the training scenes and the unseen scenes increase. The other kind of methods is to retrain a classifier in unseen scenes to detect pedestrians [11–14]. This kind of methods can be further divided into two groups. The first group fully utilizes the samples in unseen scenes and constructs a new classifier. However, it is difficult to obtain large number of labeled samples in unseen scenes. In addition, if only limited numbers of training samples are used, it is difficult to train a good classifier. Thus, these methods are difficult to use directly in practical applications. The second group of methods uses the existing training samples in old scenes and a small amount of samples in unseen scenes to retrain a new classifier [11–14]. The shortcoming of these methods is that the increased size of the training set leads to much longer training time.

This paper proposes a novel rapid method for pedestrian detection in unseen scenes based on cascade classifier structure. The motivations of our work are as follows. The detection ability of a cascade classifier is determined by the structure of l strong classifiers $\{C_i|i=1,\ldots,l\}$ in each layer and the corresponding thresholds $\{\theta_i|i=1,\ldots,l\}$, where each classifier C_i classifies an object into either pedestrian or non-pedestrian. Considering that the cascade structure of the strong classifier set $\{C_i\}$ has been proven effective under the training scene, we can keep the cascade structure and improve the ability of cascade classifier to adapt to various scenes by dynamically adjusting the threshold vector $\{\theta_i\}$. Moreover, we search for an efficient classifier, which can adapt to unseen scenes through quick optimization of threshold vector$\{\theta_i\}$. The goal is achieved by employing the following three new strategies:

(1) Only a small amount of labeled data from the unseen scenes needs to be used to adjust the key parameters of the trained cascade classifier dynamically for detecting pedestrians in unseen scenes. Compared to other previous methods, which

* Corresponding author.
E-mail addresses: xbcao@buaa.edu.cn (X. Cao), zhongw@mail.ustc.edu.cn (Z. Wang), pingkun@ieee.org (P. Yan), xuelong_li@ieee.org (X. Li).
[1] Also with the BeiHang University, Beijing 100083, PR China

0925-2312/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2011.05.019

completely retrain the classifier using a large amount of labeled data, the parameters in our method can be quickly optimized and thus the classifier can be adapted for pedestrian detection in the unseen scenes.

(2) An efficient optimization method is proposed, based on the cross entropy method [15] and a priori knowledge of the scenes, to solve the classifier parameters optimization problem. The method is able to exploit the information of samples in the unseen scenes, initialize the optimization algorithm from a good starting point, and obtain a satisfactory solution quickly.

(3) To satisfy the requirement on detection speed in unseen scenes, we propose a general ternary detection pattern based cascade classifier structure. Pedestrian and non-pedestrian objects are differentiated by comparing the stage threshold at each stage. Only those undetermined objects are moved to the next stage. Comparing with the traditional binary detection pattern, the ternary detection pattern can classify pedestrians and non-pedestrians in each layer at the same time, and therefore the detection can be performed faster.

The performance of our proposed method has been demonstrated on AHHF [1] pedestrians dataset and NICTA [16] pedestrians dataset. The experimental results showed that the proposed method can achieve performance of high detection speed, high detection rate and low false positive rate in unseen scenes.

The rest of the paper is organized as follows. In Section 2, we discuss the related works. In Section 3, the proposed rapid pedestrian detection method in unseen scenes is presented. The experimental results in various pedestrian detection datasets are demonstrated in Section 4. We finally give the concluding remarks in Section 5.

2. Related work

Pedestrian detection system needs to detect pedestrians quickly and accurately in continuously changing scenes in order to ensure the safe driving. Pedestrian detection has drawn a lot of interests from researchers in image processing, machine learning, etc.

Pedestrian detection based on image processing is a commonly used method in the pedestrian detection systems. Over the last decades, several image processing based approaches relating to the pedestrian detection problem have been developed. For example, there are optical flow based methods [17], stereo vision based methods [18,7], and model based methods [19–22]. Lipton. [17] presented a new and reliable flow technique called dynamic region matching to detect pedestrians, which can be combined with a motion detection algorithm to carry out real-time flow-based analysis of moving entities. The main shortcoming of this method is its slow detection speed. Zhao and Thorpe [18] presented a pedestrian detection method in a cluttered scene from a pair of moving cameras through stereo-based segmentation and neural network-based recognition. Grubb et al. [7] presented a pedestrian protection system based on a passive stereo vision configuration, which segments a scene into 3D objects. Their method then classifies each object as pedestrian or non-pedestrian and finally tracks the pedestrians in 3D. The shortage of the stereo vision based pedestrian detection methods is that the methods are only suitable for detecting pedestrians, which are close to the cameras. In addition, the computational complexity of the method is also very expensive.

Model based methods are also widely used for pedestrian detection. Those methods build a model using the motion gesture of pedestrians, and then classify an object as a pedestrian when the object meets the requirement of the model in detection process. For example, Cutler et al. [19] computes the degree of correlation among different interval images and uses Fourier transform to classify the pedestrians in the image. Zuriarrain et al. [20] presented a Markov chain Monte Carlo model based particle filter to detect multiple persons, which is dedicated to video surveillance applications. Beleznai and Bischof [21] presented a Bayesian detection framework to detect pedestrians in crowded scenes by contour integration and local shape estimation. Schwartz et al. [22] described a human detection method using partial least squares analysis to project the data onto a much lower dimensional subspace. The models in these methods are built for some particular scenes. If the scene changes, the original model may not be suitable for the scene any more. Therefore, the model based methods are difficult to adapt to the unseen scenes.

Classification based methods are the mainstream technologies in pedestrian detection system. There are two types of methods to detect pedestrians in unseen scenes. One type of the methods is to use the trained classifier to detect pedestrians in the unseen scenes directly. The other type is to retrain a new classifier in the unseen scenes.

So far, most study on pedestrian detection system belongs to the first type of methods. Early pedestrian detection methods mainly adopt single classifier such as neural networks based classifiers, Radial Basis Functions, and Support Vector Machines. For example, Oren et al. [23] first introduced the Support Vector Machines to pedestrian detection field and obtains a certain application. But the detection time is also too long. Gavrila [24] designs a pedestrian detection system based on Radial Basis Functions classifier. Considering that the detection rate of the method is too low, Gavrila and Geibel [2] used simulated annealing to optimize the RBF classifier and improved the detection performance. Franke and Joos [25] designed a neural network classifier successfully applied to pedestrian detection system. Szarvas et al. [26] presented a novel pedestrian detection method based on the use of a convolutional neural network classifier by automatically optimizing the feature representation to the detection task. Although single classifier based methods are simple and obtained certain success, the single classifier has imbalanced performance (e.g. low detection rate and high false positive rate). In order to increase the detection rate and decrease false positive rate, classifier based on cascade structure was proposed. Viola et al. [27,6] first proposed the cascade classifier and successfully applied it in the pedestrian detection system. Each strong classifier was trained using AdaBoost algorithm [27] and the system is very fast to detect pedestrians. Xu et al. [4] designed a cascade classifier, which consists of three SVM single classifiers for pedestrian detection. The detection rate and false positive rate of the cascade classifier are close to those of a single SVM classifier, but the detection speed is much slower. Xu et al. [28] proposed a new cascade of L1-norm minimization learning (LML) classifier to detect humans in images. The method trains a series of weak classifiers by LML and uses the integer optimization model to construct a strong classifier. Although these classification based methods have achieved successful applications, their performance degrades if the new scenes are significantly different from the training scenes.

The other category of methods is to retain a new classifier in unseen scenes to detect pedestrians. This kind of methods has drawn a lot of interests from researchers in object classification filed based on machine learning. The main idea is to use the original training dataset and a small amount of samples taken from the unseen scenes to form a large training set, and then train a new classifier to adapt to the unseen scenes. The recent arisen transfer learning method adopts a similar idea [11–14]. For example, Jiang and Chai [13] proposed a heuristic method to remove misleading training examples from the source domain and then constructed a new classifier for domain adaptation in natural language processing tasks. Wu and Dietterich [11] proposed an SVM framework using both inadequate training data and plenty of low quality auxiliary data for improving the image classification performance. Dai et al. [14] proposed a boosting algorithm TrAdaBoost, which was successfully applied to text classification using both training data and a small amount of samples. The key idea is to use boosting to filter out the different training data by automatically adjusting the weights of training instances. Liao et al. [12] proposed a new active learning method to select the unlabeled data in a target domain to be labeled with the help of the source domain data.

In this paper, we propose a rapid pedestrian detection method in unseen scenes. Different from the above mentioned methods, the proposed method does not need to retrain a new classifier, only uses a small amount of samples to adjust the trained classifier adapt to the unseen scenes. To the best of our knowledge, this is the first method that is able to detect pedestrians in unseen scenes without large scale retraining.

3. Rapid pedestrian detection in unseen scenes

This section presents the proposed rapid pedestrian detection method based on cascade structure, which consists of three parts. The first part describes the basic idea and the overall structure of our method. The second part introduces a ternary detection framework and its related analysis. Finally, the third part describes the cascade classifier key parameters (threshold vector $\{\theta_i\}$) optimization algorithm in details.

3.1. Overview

The proposed method is based on cascade classifier. As we know, the detection ability of a cascade classifier is determined by the structure of l strong classifiers $\{C_i|i=1,\ldots,l\}$ in each layer and the thresholds $\{\theta_i|i=1,\ldots,l\}$ where each classifier C_i classifies an object into either pedestrian or non-pedestrian. Assuming that the cascade structure of the strong classifier set $\{C_i\}$ has been well trained for the training scenes, the basic idea of our method is to keep the cascade structure and use the trained cascade structure to build a classifier that can be adapted to the unseen scenes through adjusting the classification thresholdθ_i. Different from those traditional training methods [2,4,6,11,14,27], the proposed method searches for an efficient classifier, which can be adapted to the unseen scenes quickly through optimizing the threshold vector $\{\theta_i\}$.

Let $D_s=\{x_i^s,y_i^s\}|_{i=1}^n$ be the training dataset in the old scenes with n samples, $D_a=\{x_i^a,y_i^a\}|_{i=1}^k$ be the auxiliary dataset that contains a small amount of k labeled samples randomly selected from the unseen scenes, and $D_t=\{x_i^t\}|_{i=1}^m$ denote test dataset from the unseen scenes with m samples. For the cascade classifier h_s, which was trained in the training scenes D_s, pedestrian detection is performed by using the strong classifier set $\{C_i\}$ with corresponding thresholds $\{\theta_i|i=1,\ldots,l\}$. Each classifier C_i classifies an object into either pedestrian or non-pedestrian. The classification function of the cascade classifier is denoted by $h_s(x)=C(\theta_1,\theta_2,\ldots,\theta_l;x)$, where x is an unlabeled sample for testing. Taking vector $\Theta=(\theta_1,\theta_2,\ldots,\theta_l)$ as the threshold vector of cascade classifier, the classification function can also be expressed as $h_s\ (x)=C(\Theta;x)$. Let $L(h_s(x))$ denote the classification error function of the classifier, the classification error function in auxiliary dataset can be computed as $\sum_{x\in D_a}L(C(\Theta',x))$, where $\Theta'=(\theta_1',\theta_2',\ldots,\theta_l')$ is a threshold vector in the process of threshold optimization.

Our goal is to minimize the above error function for adapting the previously trained classifier to the new scenes. The objective can be expressed as

$$Minimize:\sum_{x\in D_a}L(C(\Theta',x)) \tag{1}$$

Let $h_t(x)=C(\Theta^t;x)$ be the optimized classifier, where $\Theta^t=(\theta_1^t,\theta_1^t,\ldots,\theta_l{}^t)$ is the optimized threshold vector. It is difficult to compute the final threshold vector $\Theta^t=(\theta_1^t,\theta_2^t,\ldots,\theta_l{}^t)$. In this paper, we propose an optimization algorithm to solve this problem. The details are presented in Section 3.3.

In a traditional cascade classifier, there are several layers and each layer is trained by AdaBoost algorithm [27]. Each strong classifier adopts a binary detection pattern, which can only exclude non-pedestrians in each layer. To satisfy the requirement on detection speed in unseen scenes, we propose a general ternary detection pattern based cascade classifier structure. In each layer, the ternary detection pattern can classify pedestrians and non-pedestrians at the same time, and therefore the detection can be performed much faster. The details of the ternary detection pattern are presented in Section 3.2.

Fig. 1 shows the overall structure of our proposed method, which mainly consists of three steps: (1) Train a cascade classifier in the training dataset. (2) Use a small amount of samples taken from the unseen scene as auxiliary dataset to dynamically optimize the threshold vector of the above classifier to adapt to the unseen scenes. (3) Use the optimized classifier to detect pedestrians in unseen scenes using the ternary detection framework.

As we can see from Fig. 1, it has two classifiers. The first classifier is the one trained in the training scenes. The second classifier is the optimized classifier adapted for pedestrian detection in the unseen scenes. Our method optimizes the training scene classifier using auxiliary dataset to build the optimized classifier. The optimized classifier detects pedestrians in the unseen scene with help of ternary detection pattern. Algorithm 1 gives the overall flow of our method.

Algorithm 1. Overall flow of our method

Input: training dataset D_s, auxiliary dataset D_a and test dataset D_t;

Step 1: Train a cascade classifier in D_s and generate the initial threshold vector $\Theta=(\theta_1,\theta_2,\ldots,\theta_l)$;

Step 2: Build the final threshold vector $\Theta^t=(\theta_1^t,\theta_2^t,\ldots,\theta_l{}^t)$, which can be adapted to D_t using threshold vector optimization algorithm on D_a;

Step 3: Use the final threshold vector to construct the corresponding classifier, then detect pedestrians in D_t by using the ternary detection framework;

Output the detection results.

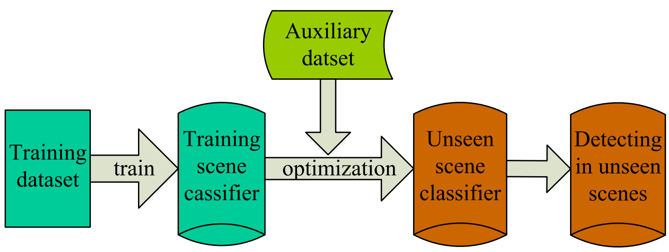


Fig. 1. Overall structure of the proposed algorithm.

3.2. Ternary detection framework

Cascaded classifier is now a commonly used classifier structure. Viola and Jones [27] and the follow-up work [7–9] proposed different cascade classifier and successfully applied to many applications such as image processing, pattern recognition, etc. Traditional cascade classifier contains several layers. Each layer can be considered as a binary detection step, which excludes non-pedestrian objects in each layer and move other undetermined objects into the next layer. The problem of binary detection is that the algorithm is not able to classify an object as pedestrian before the whole process is completed, not allowing the process to exit earlier.

In order to detect the pedestrians more efficiently, a ternary detection framework is proposed. In our work, a cascade classifier contains l layers, and each stage is a ternary detection pattern. At each stage, we introduce two parameters $\alpha_i,\beta_i(i=1,\ldots,l)$ as stage thresholds to classify the current objects into either pedestrians or non-pedestrians, where α_I is the threshold to classify pedestrians and β_i is the threshold to classify non-pedestrians. Layer thresholds are compared with the output of each strong classifier to classify the current object. Each single classifier inside the cascade structure is obtained by using the AdaBoost algorithm [27]. The proposed ternary detection framework is given in Algorithm 2.

Algorithm 2. Ternary detection algorithm

Input a test dataset, and the previously trained cascade classifier;

for $i=1,\ldots,l$;

Step 1: Compute the output of AdaBoost at the i th layerY_i, where Y_i denotes the output of layer$i$;

Step 2: if $Y_i<\beta_i$, Output non-pedestrian, break;

Step 3: if $Y_i>\alpha_i$, Output pedestrian, break;

Step 4: if $\beta_i<Y_i<\alpha_i$, continue;

Output pedestrian or non-pedestrian.

The proposed detection framework is more computationally efficient than binary detection in that it can exclude more non-pedestrian objects at each stage. In binary detection, the cascade classifier classifies an object as pedestrian only when the object is positive in all layers, while ternary detection algorithm can classify an object as pedestrian at any stage.

The theoretical interpretation of the algorithm is provided here. Consider that the binary detection and ternary detection patterns are used separately to classify an object, which is indeed a pedestrian. In binary detection, the object needs to be compared l times before it can be determined as a pedestrian. In the ternary detection, suppose that the current object is classified as a pedestrian at stage $i(i=1,\ldots l)$. And the probability of the current object classified as a pedestrian denoted asp_i. It needs to be compared i times to determine the object as a pedestrian. So the current object was classified as a pedestrian needs to be compared $E=\sum_{i=1}^{l}p_i i$ times in average. The probability of the object classified as a pedestrian at each stage is$p_i=1/l$. So the total number of comparison is $E=\sum_{i=1}^{n}p_i i=\frac{1}{l}\sum_{i=1}^{l}i=l+1/2$. In other words, the current object which is classified as a pedestrian needs to be compared $l+1/2$ times using a ternary pattern, while the object in binary detection needs to be compared l times. If we only consider pedestrian detection, ternary detection is $2\times l/l+1$ times faster than the binary detection. For non-pedestrian objects, the ternary detection algorithm and binary detection algorithm have the same detection speed in theory. Experiments in Section 4.2 verified the theoretical analysis.

In summary, the ternary detection framework proposed here is a simple cascaded classifier and it is a chain structure consisting of many single weak ternary classifiers. The common property of the two detection framework is that they can exclude non-pedestrians at each layer and the uncertain objects will be moved to the next layer. The main difference of these two detection methods is that the ternary detection can exclude pedestrians at each stage, but the binary detection method classifies an object as a pedestrian only when all the stages think it as pedestrian.

3.3. Threshold vector optimization

As we described in Section 3.1, our goal is to quickly search for the optimal threshold vector and construct the corresponding cascade classifier so that the error function of the classifier minimum in the auxiliary dataset can meet the requirement of continuously changing scenes. The problem is that Eq. (1) is difficult to be numerically expressed and the optimization may be too slow for a real-time application. Thus, it presents a great challenge for designing the algorithm to optimize the threshold vector $\Theta=(\theta_1,\theta_2,\ldots,\theta_l)$.

Numerous optimization algorithms have been proposed and proven to be successful in many applications, such as planning numerical optimization method [29], genetic algorithm [30], immune optimization [31], Co-Evolution [32], and global stochastic optimization method. However, since Eq. (1) cannot be numerical expressed, the traditional numeric optimization algorithms are not suitable for the problem. Meanwhile, global stochastic optimization method can find the optimal solution, but the optimization is very computationally expensive. Therefore, it is difficult to apply it to the optimization algorithm based on threshold vector $\Theta=(\theta_1,\theta_2,\ldots,\theta_l)$.

In this paper, we design an optimization algorithm using the cross entropy (CE) method combines heuristic knowledge optimization strategy. The CE method is an iterative method, which involves the following two phases: generation of a random sample from the data (trajectories, vectors, etc.) according to a specified distribution and updating the parameters of the random vector generator for producing better samples in the next iteration. The CE method has been successfully applied to a number of difficult combinatorial optimization problems. It can achieve good performance with fast convergence and high search rate. More importantly, the CE method is simpler than other optimization algorithms, which makes it more suitable for quickly searching the threshold vector optimized for unseen scenes.

To solve the specific problem in our work using the CE method, the following questions have to be properly answered: (1) How to generate random vector as the initial vector for optimization? For example, in our work, the problem is to generate threshold vector $\Theta=(\theta_1,\theta_2,\ldots,\theta_l)=((\alpha_1,\beta_1),(\alpha_2,\beta_2),\ldots,(\alpha_l,\beta_l))$. (2) How to update the threshold vector $\{(\alpha_i,\beta_i)\}(i=1,\ldots,l)$ based on the data to produce a better sample in the next iteration?

In our work, the above two problems are solved as follows.

3.3.1. Design random sampling operation

The CE based method itself can ensure the optimization algorithm to obtain good performance under any random distribution, but the optimization speed may be slow if the initialization is far away from the final target. To solve this problem, this paper presents a heuristic optimization strategy. We use the parameters of the classifier trained under the training scenes to set the initial value of $\{(\alpha_i,\beta_i)\}$, so that the optimization can start at a good point and search for the optimal solution in a much shorter time.

Let P_{ji} be the output of the jth positive sample in the ith layer of the old classifier and N_{ji} be the output of the jth negative sample in the ith layer of the old classifier. We set α_i',β_i' as the initial value of $\{(\alpha_i,\beta_i)\}$. $\alpha_i{}',\beta_i{}'$ can be obtained by

$$\alpha_i{}' = \frac{1}{k}\sum_{j=1}^{k} P_{ji}, \beta_i' = \frac{1}{k}\sum_{j=1}^{k} N_{ji}$$

which are the average output of all the positive samples and the negative samples in the ith layer, respectively. Similarly, we compute the corresponding variance values

$$\sigma_{\alpha_i{}'} = \frac{1}{k}\sum_{i=1}^{k}(P_{ji}-\alpha_i{}')^2, \quad \sigma_{\beta_i{}'} = \frac{1}{k}\sum_{i=1}^{k}(P_{ji}-\beta'{}_i)^2$$

Our random operation then uses the normal distribution $N(\mu_i,\sigma_i)$ to randomly generate some threshold vectors, where $\mu_i = (\alpha_i{}',\beta_i{}')$, $\sigma_i = (\sigma_{\alpha_i{}'},\sigma_{\beta_i{}'})$.

3.3.2. Adjusting strategy

To solve the second problem, according to the threshold vectors, which are generated randomly from the normal distribution $N(\mu_i,\sigma_i)$, we use the random threshold vector with the corresponding cascade classifier to detect pedestrians in the auxiliary dataset and compute its detection performance. Then the several best threshold vectors are selected and the mean is computed to update the next iteration.

The algorithm for optimizing $\{(\alpha_i,\beta_i)\}$ is given in Algorithm 2 in detail.

Algorithm 3. Optimization algorithm for threshold vector $\{(\alpha_i,\beta_i)\}$

Input a trained cascade classifier, initial threshold vector$\mu_0 = \{(\alpha_i,\beta_i)|_{i=1}^{l}\}$, iterate n times, auxiliary dataset D_a, $t=0$.

Step 1: generate some random threshold vectors using $N(\mu_i,\sigma_i)$, construct their corresponding cascade classifier to detect pedestrians in D_a and compute the detection performance of their corresponding classifier in D_a, then select several best threshold vectors;

Step 2: calculate the mean vector of these better vectors $\overline{\mu_i}$ and compute its variance vector$\overline{\sigma_i}$.

Step 3: update the next mean and variance vector:

$\mu_{i+1} = \lambda \times \mu_i + (1-\lambda) \times \overline{\mu_i}$

$\sigma_{i+1} = \lambda \times \sigma_i + (1-\lambda) \times \overline{\sigma_i}$;

Step 4: if $(t<n)$ t++, continue;

else output the vector $u_n = \{(\alpha_i{}^t,\beta_i{}^t)_{i=1}^{l}\}$, exit.

The output of the optimization algorithm is the final threshold vector $u_n = \{(\alpha_i{}^t,\beta_i{}^t)_{i=1}^{l}\}$, with which the corresponding cascade classifier can be constructed to detect pedestrians with the ternary detection pattern. The advantage of the optimization algorithm is in searching for the optimal threshold in a certain neighborhood through normal distribution, and using prior knowledge of the unseen scenes as the stating point. Thus, the optimization algorithm can find the optimal threshold vector efficiently.

4. Experiments

4.1. Dataset

In our experiments, the proposed method was tested on two datasets, AHHF [1] and NICTA [16]. The former was considered as the training scenes, and the latter was used as the unseen scenarios for performance testing. All experiments were carried out on an Intel 1.6 GHz (Dual-core) computer with 1GB DDR667 RAM.

The AHHF pedestrian dataset was collected from urban traffic videos captured in Hefei City, Anhui Province, China. It contains 1600 pedestrian samples. About 3000 high-quality negative samples were manually selected from objects similar to pedestrians. The pedestrian samples were randomly split into 2 groups, with 800 samples in each set. One of the sets was used for training and the other was used for testing. The pedestrian samples are 16×32 pixel pictures and the negative samples are 320×240 pixel pictures. We selected 828,685 negative samples for testing. The ratio of positive samples and negative samples was about 1:100. The NICTA pedestrian dataset contains a large number of pedestrian images with different sizes and dimensions. We selected 16×40 pixel pictures as pedestrian samples. There are 37,347 pedestrian images of this specification. The negative samples are also pictures in the size of 320×240. Pedestrian gestures in NICTA dataset are complex and diverse. Therefore, we clustered the dataset into five parts using the k-means algorithm and each part contains various forms of pedestrian images. We took the first part which includes 7249 pedestrian samples for training and the other parts which contain 29,918 samples for testing.

Fig. 2 shows some examples of the pedestrian images from the two datasets. It can be seen that the pedestrian images in the two datasets are quite different. Thus, the two datasets are suitable of being considered as different scenes and they are suitable for use in our experiments.

4.2. Performance comparison

4.2.1. Performance evaluation

4.2.1.1. Performance measures. For quantitatively evaluating the detection performance, *TPR*, *FPR* and F-measure were used to validate the proposed algorithm. F-measure is defined as the weighted harmonic mean of precision and recall. TPR is defined as true positive rate and also represents detection rate. FPR is defined as false positive rate. The formulas are as follows:

$$F = \frac{2 \times TP}{2 \times TP + FP + FN} \begin{cases} TPR = \frac{TP}{TP+FN} \\ FPR = \frac{FP}{TN+FP} \end{cases}$$

where *TP* (true positive) denotes the number of the positive samples, which are classified correctly; *FN* (false negative) is the number of the positive samples, which are wrongly classified as negative; *FP* (false positive) denotes the number of the negative samples, which are wrongly classified as positive; and *TN* (false negative) denotes the number of the negative samples, which are classified correctly.

4.2.2. Detection speed

Detection time is another standard to measure the detection performance between ternary detection pattern and binary detection pattern. The goal is to verify the effective of ternary detection pattern.

4.2.2.1. Comparison with conventional methods. We have performed four experiments to verify the performance of the proposed method. Experiment 1 is to verify the trained classifier performs not well in the unseen scene. Experiment 2 compared with detection performance of traditional methods and our methods in unseen scenes. Experiment 3 is detection speed comparison between ternary detection pattern and binary detection pattern based on cascade structure. Experiments 4 verify the robustness of optimized cascade classifier.

As we described in Section 4.1, we take AHHF dataset as training scenes and NICTA dataset as unseen scenes.

(A) Experiment 1: Verify the classifier which was trained in AHHF dataset and does not perform well in NICTA dataset

We trained a cascade classifier named Old-Cascade in the AHHF dataset. Table 1 gives detection rate and false positive rate of Old-Cascade in AHHF dataset and NICTA dataset.

Fig. 2. Example of pedestrian samples: (a) pedestrian samples in AHHF dataset (16 × 32) and (b) pedestrian samples in NICTA dataset (16 × 40).

Table 1
Comparison Of algorithm performance.

Detection results	Old-Cascade	
	TPR (%)	*FPR*
AHHF dataset	87.875	0.0078
NICTA dataset	68.83	0.0778

From Table 1, we can see that the classifier, which was trained in training scenes, does not perform well in unseen scenes, the detection rate is very low, and the false positive rate is very high. Fig. 3 gives the ROC curves of Old-Cascade under AHHF dataset and NICTA dataset. We can see that the Old-Cascade can obtain good performance in the same scene. Since AHHF dataset is different with NICTA dataset, the performance of Old-Cascade decreased clearly.

(B) Experiment 2: Performance comparison between traditional methods and our methods

We named Retrained-Cascade as a classifier, which is trained by AHHF dataset and auxiliary dataset. Similarly, Retrained-Cascade was also based on cascade structure. We also named the cascade classifier constructed by optimization algorithm with threshold vector as Optimized-Cascade. Experiment 2 compared with the detection rate of Old-Cascade, Retained-Cascade and Optimized-Cascade in NICTA dataset.

In our work, Old-Cascade is trained in the AHHF dataset which contained 6 layers. The optimization of the threshold vector was performed as follows. First, we selected 100 positive samples and 100 negative ones randomly from the first part of NICTA dataset as the auxiliary dataset. The Old-Cascade detected pedestrians in all the samples of the auxiliary dataset. The output for every single sample in each layer was calculated. The mean and variance of all the samples were calculated as the initial threshold vector. Then, 100 threshold vectors were randomly generated by the normal distribution using the initial threshold vector. Next, we selected 10 maximal F-measure scores with its corresponding threshold vectors and computed the mean vector and variance vector of these 10 vectors. The final threshold vector was obtained after 25 iterations. The final parameter vector was suitable for the unseen scenes.

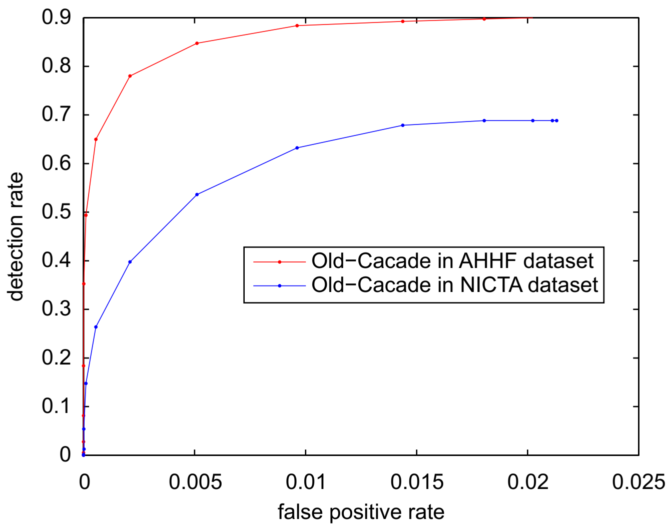


Fig. 3. ROC curve of Old-Cascade under AHHF and NICTA dataset.

Table 2 gives the detection rate of Old-Cascade, Retrained-Cascade, and Optimized-Cascade in NICTA dataset. As we can see from Table 2, the detection performance of Old-Cascade is very poor due to the large difference between AHHF dataset and NICTA dataset. The detection performance of Retrained-Cascade has a certain improvement. The reason is that the training set of Retrained-Cascade contains some auxiliary set. So Retained-Cascade has a certain impact when detecting pedestrians in NICTA dataset. Optimized-Cascade performed better than Retained-Cascade, because Optimized-Cascade generates some random threshold vectors through normal distribution and searches the optimal threshold vector in a certain neighborhood in each round. Retrained-Cascade has a certain effect due to the auxiliary dataset, but the scale of the auxiliary dataset is so small that it can not train a good

Table 2
Comparison of detection rate.

Classifier	Detection rate (*TPR*) (%)
Old-Cascade	68.83
Retrained-Cascade	77.07
Optimized-Cascade	82.73

Table 3
Time comparison.

Detection framework	Time (ms)	
	Pedestrian samples	Non-pedestrians samples
Binary detection	1981	19,172
Ternary detection	1232	14,503
Time ratio	1.6	1.32

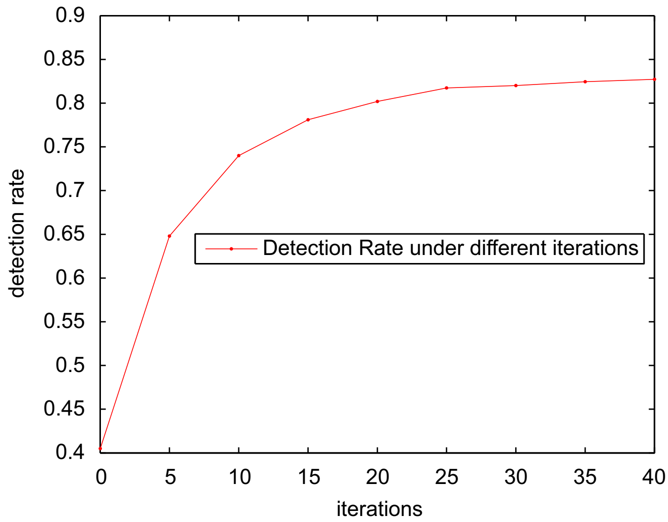


Fig. 4. Detection rate under different iterations.

classifier adapted for NICTA. Moreover, Retrained-Cascade cost expensive due to the scale of training set is increase.

(C) Experiment 3: Detection speed between ternary detection and binary detection
We compared with the detection speed between ternary detection pattern and binary detection pattern based on the same cascade structure. The detection algorithm uses a ternary detection, which can exclude pedestrians or non-pedestrians at each layer. Considering pedestrian detection only, we can see from Section 3.2, ternary detection is $2l/(l+1)=1.7$times faster than the binary detection in theory when the length of cascade $l=6$.
The test dataset contains 29,918 pedestrian samples and 828,685 negative samples. Table 3 shows the detection time of ternary detection and binary detection on the test dataset. Optimization algorithm based on threshold vector outputs the final threshold vector after 25 rounds, then construct the corresponding cascade classifier to detect pedestrians in test set. We can see from Table 3 that the ratio of the run time of two detection framework is close to theoretical value. Moreover, ternary detection framework is faster than binary detection algorithm for negative samples.
Fig. 4 gives the detection rate under different iterations. As we can see from Fig. 4, the detection rate of the Optimized-Cascade in NICTA dataset reaches 82% after 25 iterations and tends to become steady with larger iterations. Therefore, in this paper, we set the number of iterations to 25.

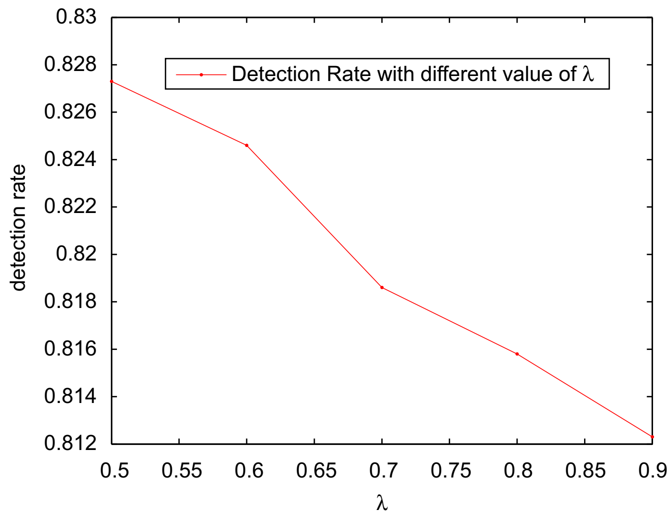


Fig. 5. Detection rate with different values of parameter λ in NICTA dataset.

(D) Experiment 4: Robustness of Optimized-Cascade
In the optimization algorithm, it has an adjustable parameter λ. Kroese et al. [15] indicates that the smoothing procedure with the smoothing parameter $\lambda(0.5 \le \lambda \le 0.9)$ works well in many cases, when $\lambda=1$, the algorithm will converge to a wrong solution.

In the experiment, parameter λ was set to 0.5. In order to validate the robustness of the optimization algorithm, a set of experiments with different values of λ were performed. It can be seen that different value of λ does not greatly influence the detection rate. When λ was set to 0.5, the detection rate could get the best result.

Fig. 5 gives the detection rate of the Optimized-Cascade under different parameter value of λ in NICTA dataset.

Compared to the existing methods, the advantages of the proposed method are as follows: (1) The proposed method can adapt to the unseen scenes for pedestrian detection. (2) The proposed method has excellent performance with high detection rate, low false positive rate and high detection speed. (3) The proposed method is robust and efficient.

5. Conclusion

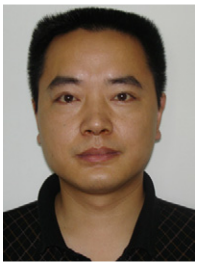
This paper proposes a method for rapid pedestrian detection in unseen scenes using cascade classifier structure. A new optimization algorithm is employed to generate an optimal threshold vector so that the cascade classifier can be adapted for unseen scenes using only a small amount of samples from the unseen scenes. At the same time, we present a general ternary detection pattern based cascade classifier structure. Comparing with the traditional binary detection pattern, the ternary detection pattern can classify pedestrians and non-pedestrians in each layer at the same time. Therefore, the detection can be done faster. Taking two different pedestrian datasets as the training set and the test set respectively, the experimental results showed that the proposed method is effective in that it can achieve higher detection rate, faster detect speed and lower false positive rate. It was also noticed in our experiments that the scale of the auxiliary dataset has certain influence on the performance of our proposed method. One of our further research directions is to quantify the relationship between the detection performance and the size of the auxiliary dataset.

Acknowledgments

The presented research work is supported by the National Basic Research Program of China (973 Program) (Grant No. 2011CB707000), the National Natural Science Foundation of China (Grant No. 61072093, 60972103), and the Foundation for Innovative Research Groups of the National Natural Science Foundation of China (No. 60921001).

References

[1] X.B. Cao, H. Qiao, J. Keane., A low-cost pedestrian detection system with a single optical camera, IEEE Trans. Intell. Transp. Syst. 9 (no. 1) (2008) 58–67 Mar.
[2] D.M. Gavrila, J. Geibel, Shape-based pedestrian detection and tracking, in: Proceedings of the IEEE Intelligent Vehicle Symposium, vol. 1, June 2002, pp. 8–14.
[3] G.M.A. Sessler, T. Martoyo, F.K. Jondral, RBF based multiuser detectors for UTRA-TDD, in: Proceedings of the IEEE Vehicular Technology Conference, 2001, vol. 1, pp. 484–486.
[4] F.L. Xu, X. Liu, K. Fujimura, Pedestrian detection and tracking with night vision,, IEEE Trans. Intell. Transp. Syst. 6 (no.1) (2005) 63–71 Mar.
[5] S. Maji, A.C..Berg, J. Malik. Classification using intersection kernel support vector machines is efficient, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, June 2008, pp. 1–8.
[6] P. Viola, M. Jones, D. Snow., Detecting pedestrians using patterns of motion and appearance, in: Proceedings of the IEEE Conference on Computer Vision, vol. 2, October 2003, pp. 734–741.
[7] G. Grubb, A. Zelinsky, L. Nilsson, M. Rilbe, 3D vision sensing for improved pedestrian safety, in: Proceedings of the IEEE Intelligent Vehicle Symposium, June 2004, pp. 19–24.
[8] A. Shashua, Y. Gdalyahu, G. Hayun, Pedestrian detection for driving assistance systems: single-frame classification and system level performance, in: Proceedings of the IEEE International Conference on Intelligent Vehicle Symposium, June 2004, pp. 1–6.
[9] P. Sabzmeydani, G. Mori, Detecting pedestrians by learning shapelet features, in: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, June 2007, pp. 1–8.
[10] O. Tuzel, F.M. Porikli, P. Meer., Pedestrian detection via classification on riemannian manifolds, IEEE Trans. Pattern Anal. Mach. Intell. 30 (no. 10) (2008) 1713–1727 October.
[11] P. Wu, T.G. Dietterich, Improving SVM accuracy by training on auxiliary data sources, in: Proceedings of the 21th International Conference on Machine Learning, vol. 69, July 2004, pp. 871–878.
[12] X. Liao, Y. Xue, L. Carin, Logistic regression with an auxiliary data source, in: Proceedings of the 22th International Conference on Machine Learning, vol. 119, August 2005, pp. 505–512.
[13] J. Jiang, C. Zhai, Instance weighting for domain adaptation in NLP, in: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, June 2007, pp. 264–271.
[14] W. Dai, Q. Yang, G. Xue, Y. Yu, Boosting for transfer learning, in: Proceedings of the 24th International Conference on Machine Learning, vol. 227, June 2007, pp. 193–200.
[15] D.P. Kroese, S. Porotsky, R.Y. Rubinstein., The cross-entropy method for continuous multiextremal optimization, Methodol. Comput. Appl. Probab. 8 (no. 3) (2006) 383–407 October.
[16] G. Overett, L. Petersson, N. Brewer, L. Andersson, N. Pettersson. A new pedestrian dataset for supervised learning, in: Proceedings of the IEEE Intelligent Vehicle Symposium, June 2008, pp. 373–378.
[17] Alan Lipton, Local application of optical flow to analyse rigid versus non-rigid motion, in: Proceedings of the International Conference on Compute Vision. Workshop Frame-Rate, Sepember 1999.
[18] L. Zhao, C.E. Thorpe, Stereo and neural network-based pedestrian detection, IEEE Trans. Intell. Transp. Syst. 1 (no.3) (2000) 148–154 Sep.
[19] R. Cutler, L.S. Davis, Robust real-time periodic motion detection, analysis, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 22 (no. 8) (Aug. 2000) 781–796.
[20] I. Zuriarrain, F. Learasle, N. Arana, M. Devy. An MCMC-based particle filter for multiple person tracking, in: Proceedings of the 19th International Conference on Pattern Recognition, December 2008, pp. 1–4.
[21] C. Beleznai, H. Bischof, Fast human detection in crowded scenes by contour integration and local shape estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 2009, pp. 2246–2253.
[22] W.R. Schwartz, A. Kembhavi, D. Harwood, L.S. Davis., Human detection using partial least squares analysis, in: Proceedings of the International Conference on Computer Vision, September 29–October 2, 2009, pp. 24–31.
[23] Oren M., Papageorgiou C., Sinha P., Osuna E., Poggio T., Pedestrian detection using wavelet templates, in: Proceedings of the International Conference on Computer Vision and Pattern Recognition, June 1997, pp. 193–199.
[24] D.M. Gavrila, Pedestrian detection from a moving vehicle, in: Proceedings of the European Conference on Computer Vision, 2000, pp. 37–49.
[25] U. Franke and A. Joos. Real-time stereo vision for urban traffic scene understanding, in: Proceedings of the IEEE Intelligent Vehicle Symposium, 2000, pp. 273–278.
[26] M. Szarvas, A. Yoshizawa, M. Yamamoto, J. Ogata. "Pedestrian detection with convolutional neural networks, in: Proceedings of the IEEE Intelligent Vehicle Symposium , June 2005, pp. 224–229.
[27] P. Viola, M. Jones., Rapid object detection using a boosted cascade of simple features, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, 2001, pp. 511–518.
[28] R. Xu, B. Zhang, Q. Ye, J. Jiao, Cascaded L1-norm minimization learning (CLML) classifier for human detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 3566–3569.
[29] S.S. Sritharan, Dynamic programming of the Navier–Stokes equations, Syst. Control Lett. 16 (no. 4) (1991) 299–307 April.
[30] K. Yen, L. Hanzo, Genetic-algorithm-assisted multiuser detection in asynchronous CDMA communications, IEEE Trans. Veh. Tech. 53 (no. 5) (Sep. 2004) 1413–1422.
[31] X.B. Cao, H. Qiao, Y.W. Xu., Negative selection based immune optimization, Adv. Eng. Software 38 (no. 10) (2007) 649–656 October.
[32] Y.P. Guo, X.B. Cao, H.Z. Yin, Co-evolutionary optimization algorithm with dynamic sub-population size, Int. J. Innov. Comput. Inform. Control. 3 (no. 2) (2007) 435–448 April.



Xianbin Cao received the B.S. degree in computer science and the M.S. degree in information and system from Anhui University, Hefei, China, in 1990 and 1993, respectively, and the Ph.D. degree in intelligent information processing from the University of Science and Technology of China (USTC), Hefei, in 1996. He has been with the USTC since 1996 and became an Associate Professor with the Department of Computer Science and Technology from 1999 to 2005, where he is currently a Professor and the Administrative Director of the Anhui Province Key Laboratory of Software in Computing and Communication, Hefei since 2005. Since 2009 he has also been a professor in the school of electronic and information engineering, Beihang University PR China and is the director of the lab of intelligent transportation system. His current research interests include intelligent transportation systems, airspace transportation management and intelligent computation. He has been publishing more than 100 books, book chapters, and papers in these areas since 1993.

Zhong Wang is a Ph.D. candidate in the Department of Computer Science and Technology, the University of Science and Technology. He received the B.S. degrees in computer science from AnQing Teachers College, China, in 2006. His research interests include machine learning, computer vision, and pedestrian detection. He is a student member of IEEE.

Pingkun Yan received the B.Eng. degree in electronics engineering and information science from the University of Science and Technology of China, Hefei, China and the Ph.D. degree in electrical and computer engineering from the National University of Singapore, Singapore. He is a full professor with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi′an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi′an 710119, Shaanxi, PR China. He was a Senior Member of the Research Staff of Philips Research North America, Briarcliff Manor, NY. Before that, he worked as a Research Associate with the Computer Vision Laboratory, University of Central Florida, Orlando, FL. His research interests include computer vision, pattern recognition, machine learning, and their applications. He is a senior member of the IEEE.

Xuelong Li is a full professor with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi′an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi′an 710119, Shaanxi, PR China.