

# An Improvement of Pedestrian Detection Method with Multiple Resolutions

Guodong Zhang<sup>1</sup>, Peilin Jiang<sup>2</sup>, Kazuyuki Matsumoto<sup>1</sup>, Minoru Yoshida<sup>1</sup>, Kenji Kita<sup>1</sup>

<sup>1</sup>Faculty and School of Engineering, Tokushima University, Tokushima, Japan

<sup>2</sup>Xi'an Jiaotong University, Xian, China

Email: zhang-g@hotmail.co.jp, pljiang@xjtu.edu.cn, matumoto@is.tokushima-u.ac.jp, mino@is.tokushima-u.ac.jp, kita@is.tokushima-u.ac.jp

**How to cite this paper:** Zhang, G.D., Jiang, P.L., Matsumoto, K., Yoshida, M. and Kita, K. (2017) An Improvement of Pedestrian Detection Method with Multiple Resolutions. *Journal of Computer and Communications*, 5, 102-116.

<https://doi.org/10.4236/jcc.2017.59007>

**Received:** October 10, 2016

**Accepted:** July 14, 2017

**Published:** July 17, 2017

Copyright © 2017 by authors and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## Abstract

In object detection, detecting an object with 100 pixels is substantially different from detecting an object with 10 pixels. Many object detection algorithms assume that the pedestrian scale is fixed during detection, such as the DPM detector. However, detectors often give rise to different detection effects under the circumstance of different scales. If a detector is used to perform pedestrian detection in different scales, the accuracy of pedestrian detection could be improved. A multi-resolution DPM pedestrian detection algorithm is proposed in this paper. During the stage of model training, a resolution factor is added to a set of hidden variables of a latent SVM model. Then, in the stage of detection, a standard DPM model is used for the high resolution objects and a rigid template is adopted in case of the low resolution objects. In our experiments, we find that in case of low resolution objects the detection accuracy of a standard DPM model is lower than that of a rigid template. In Caltech, the omission ratio of a multi-resolution DPM detector is 52% with 1 false positive per image (1FPPI); and the omission ratio rises to 59% (1FPPI) as far as a standard DPM detector is concerned. In the large-scale sample set of Caltech, the omission ratios given by the multi-resolution and the standard DPM detectors are 18% (1FPPI) and 26% (1FPPI), respectively.

## Keywords

Deformable Part Model, Pedestrian Detection, Multi-Resolution; Latent SVM

## 1. Introduction

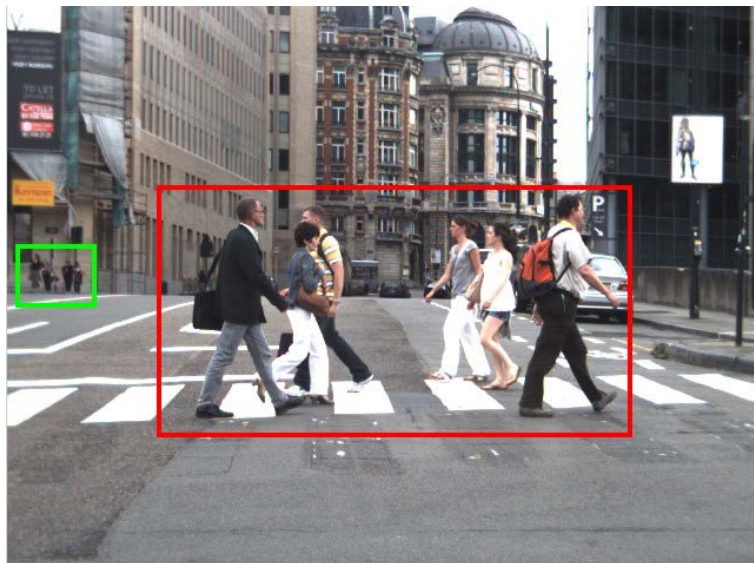
Pedestrian detection has been a hotspot in computer vision research [1]. The corresponding detection algorithm has been developed towards high precision and instantaneity [2] [3]. For a driverless automobile, the usage of which has

become popular nowadays, its intelligent system should be able to detect the locations and quantities of pedestrians ahead, to analyze the road conditions, and to guarantee the safety of these pedestrians [4]. For such cases, the pedestrian detection is an inevitable procedure. The pedestrian detection problem is difficult because that the target people often have various characteristics and the surrounding environments also change frequently [5].

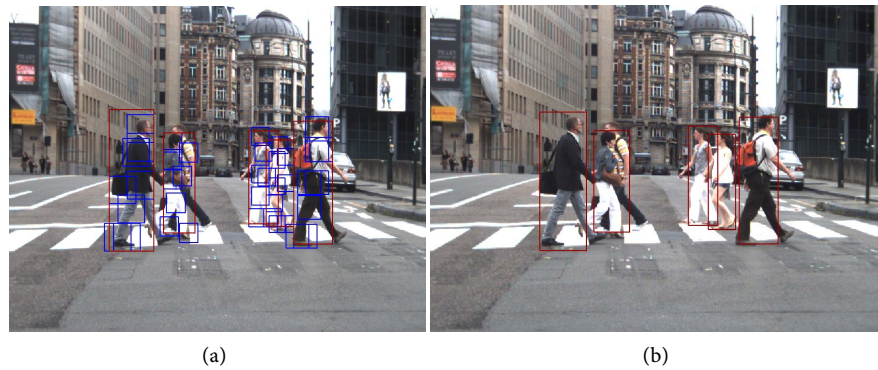
The pedestrian sizes in real world are different from each other. Besides the height diversity of different people, many imaging differences are incurred by the different distances between people and the camera. **Figure 1** shows a high resolution corresponds to the large pedestrian scale and a low resolution corresponds to the small pedestrian scale in the process of pedestrian detection.

Pedestrians contain rich information in the case of high resolution [6], and it is more likely for them to be detected. Even if they are locally overlapped, many algorithms have the capability to detect these targets [7]. However, in the case low resolution, the pedestrians which contain a small amount of information cannot be detected easily. Meanwhile, low resolution pedestrians are very vulnerable to the interferences of the surrounding environments. In most cases, a detection algorithm has a much better detection result for the high resolution pedestrians than that for the low resolution pedestrians. Dalal and Triggs [8] proposed a HOG detector. If the detection window is fixed to  $128 \times 64$  pixels during training and detection, this detector can generate good effects at the time of detecting pedestrians with pixels greater than  $128 \times 64$ . However, when the target pedestrians are smaller than  $128 \times 64$ , the detector almost fails to detect any pedestrian. Although the target can be increased to larger than  $128 \times 64$  pixels by means of interpolation, the detection accuracy is still brought down. The DPM pedestrian detector makes use of a root filter and several part filters to describe the pedestrians. Information in the pedestrians of high resolution is sufficient.

**Figure 2(a)** and **Figure 2(b)** are results obtained by utilizing a standard DPM



**Figure 1.** The pedestrians with Multiple resolution in a sample picture.



**Figure 2.** The detection result of standard DPM. (a) The part filter and root filter in DPM (b) The detection result of standard DPM.

detector to detect pedestrians in **Figure 1**. It is obvious that the small-scale pedestrians cannot be detected successfully. Therefore, the overall detection effect can be improved if we can improve the detection effect for low resolution pedestrians and prevent affecting the detection effect for high resolution pedestrians.

In this paper, we propose a multi-resolution DPM pedestrian detection algorithm, which takes advantage of the standard DPM framework in training the pedestrian with the resolution factor as a hidden variable. For the high resolution pedestrians, the response can be figured out in the first place. And its location can be estimated with the combination of this high resolution response and the response under a corresponding low resolution. However, for the low resolution pedestrians, the judgment over possible locations of these targets is carried out by only calculating the responses under the low resolution. High resolution and low resolution are only intuitive concepts in the common sense. In addition, resolution is closely associated with the heights of pedestrian samples.

Structure of this paper is as follows. In section 2, we thoroughly illustrate the DPM model for pedestrian detection, depicts the DPM learning algorithm, and describe the parameter initialization and the training procedures. In section 3, we illustrate the improved DPM algorithm in case of multi-resolution targets, by analyzing the features of pedestrian detection under multi-resolution, and describing the improved multi-resolution DPM pedestrian detection algorithm in detail. In Section 4, we apply this improved algorithm to a general dataset to comparatively analyze the experimental results.

## 2. Overview of Related Theory

### 2.1. Deformable Part Model

The deformable part model (DPM) consists of a root filter and several part filters to describe the pedestrians. Specifically, the root filter describes each pedestrian as a whole, while each part filters describe a part of the pedestrian, such as the head and hand [9]. In this way, the constructed model can effectively capture the pedestrian information, and adapt well to the changes of body posture and dressing of the pedestrian [10].

The DPM pedestrian detection is a x-resolution detection method. However, to some extent the algorithm is able to adapt to different resolutions, because of the following three reasons.

Firstly, the DPM features are based on the image pyramid HOG features [11], which are adaptable to the scale variation within a certain range. Secondly, because the available data sets often consist of a large number of pedestrian samples, we have enough information for training a DPM model. For example, over 84% positive samples in the Caltech pedestrian database are over 30 pixels in height, over 16% positive samples are more than 80 pixels in height, and around 69% positive samples are between 30 pixels and 80 pixels in height.

## 2.2. Hard Example Mining (SVM)

In the training procedure, there are usually more negative samples than the available positive samples. Taking pedestrian detection for example, the images of pedestrians are positive samples, and the images without pedestrian are negative samples. In this case,  $10^5$  samples can be generated from an image, most of which are negative samples. It is almost impossible to take all negative samples into consideration. Therefore, we select the positive samples and the hard examples for constructing a training set. The hard examples are referred to those which are incorrectly classified at the first time. The Bootstrapping classification algorithm is employed for training an initial negative sample set. The algorithm collects the incorrectly classified samples at the first time, add these samples to the negative sample set to form the hard samples. The process is repeated for several times until a good classification result is achieved. We define the hard example and easy sample as follows:

$$H(\beta, D) = \{(x, y) \in D \mid yf_{\beta}(x) < 1\} \quad (1)$$

$$E(\beta, D) = \{(x, y) \in D \mid yf_{\beta}(x) > 1\} \quad (2)$$

where,  $H(\beta, D)$  denotes the incorrectly classified samples at the first time or the samples located within the classification boundary  $E(\beta, D)$  denote the correctly classified samples. The samples on the classification boundary do not belong to  $H(\beta, D)$  or  $E(\beta, D)$ .  $\beta^*(D) = \arg \min_{\beta} L_D(\beta)$ .

Because  $L_D$  is strictly convex,  $\beta^*(D)$  is the single result of the optimized problem. Given a sample library  $D$ , we want to find a small sample set  $C$  with  $C \subseteq D$  and  $\beta^*(C) = \beta^*(D)$ . To solve this problem, we firstly define an initial set which contains all the training samples. We train an LSVM model, and renew the previous set by removing the simple samples and by adding the new hard examples.

## 2.3. Hard Example Mining with LSVM

For the LSVM, mining hard examples is equivalent to optimize  $D_{(z_p)(\beta)}$  rather than  $L_D(\beta)$ . This constraint turns the whole optimization problem to a convex optimization problem.

As for the hard example mining with SVM, we define an set with samples in the form of  $(x, z)$ , in which  $z \in Z(x)$ . In the real application, the set consists of  $\Phi(x, z)$  rather than  $(x, z)$ . We define a vector set  $F = (i \cdot v)$ , in which  $i$  denotes sample index, and  $v = \Phi(x, z)$  with  $z \in Z(x_i)$ . Because the hidden variable  $z$  is not fixed, for each sample  $x_i$ , there may be multiple corresponding  $(i, v) \in F$ . Then we define  $I(F)$  as the index of vectors in the vector set  $F$ , and define the target function for  $\beta$  with the feature vectors in  $F$ :

$$L_F(\beta) = \frac{1}{2} \|\beta\|^2 + C \sum_{i \in I(F)} \max\left(0, 1 - y_i \left(\max_{(i,v) \in F} \beta \cdot v\right)\right) \quad (3)$$

$L_F$  can be optimized with the gradient-descent algorithm. We use  $V(i)$  as the set of feature factors  $v$ .

The gradient-descent algorithm is described as follows.

1.  $a_t$  is the learning rate for iteration  $t$ .
2.  $i \in I(F)$  is the index for samples in  $F$ .
3.  $v_i = \arg \max_{v \in V(i)} \beta \cdot v$ .
4. If  $y_i(\beta v_i) \geq 1$ , then  $\beta = \beta - a_t(\beta)$ .
5. Otherwise,  $\beta = \beta - a_t(\beta - C y_i v_i)$ .

We set  $\beta^*(F) = \arg \min_{\beta} L_F(\beta)$  and try to find  $\beta^*(F) = \beta^*(D(Z_p))$  in a sample set  $D(Z_p)$  of small size.

As for the hard example mining with standard SVM, we define the feature vectors for hard and simple example in training set  $D$  as follows.

$$H(\beta, D) = \left\{ (i, \Phi(x, z_i)) \mid z_i = \arg \max_{z \in Z} \beta \cdot \Phi(x_i, z) \text{ and } y_i(\beta \cdot \Phi(x_i, z_i)) < 1 \right\}$$

$$E(\beta, F) = \{ (i, v) \in F \mid y_i(\beta \cdot v) > 1 \} \quad (4)$$

We find the hard examples by calculating  $\beta^*(D(Z_p))$ .  $F_1$  is defined as the initial feature vector set.

The LSVM hard example mining algorithm is given below:

1. Train model with  $\beta_t := \beta^*(F_t)$ .
2. If  $H(\beta_t, D(Z_p)) \subseteq F_t$ , stop the iteration and return  $\beta_t$ .
3. Remove the simple samples by  $F'_t := F_t \setminus X$ , where  $X \subseteq E(\beta_t, F_t)$ .
4. Add new hard examples by  $F_{t+1} := F'_t \cup X$ , where  $X \cap H(\beta_t, D(Z_p)) \setminus F_t \neq \emptyset$ .

In the 3rd step simple samples are removed from the training set, while in the 4th step new hard examples are added to the training set. The entire iteration procedure terminates when there are no hard examples to add.

### 3. Pedestrian Detection with Multi-Resolution DPM

#### 3.1. Fixed Resolution Model

Let  $x$  represent an image window, and  $\mathcal{O}(x)$  represent the image feature. As many slide window detection algorithms, we have

$$f(x) > 0 \quad \text{where } f(x) = w \cdot \mathcal{O}(x) \quad (5)$$

in which  $x$  is marked as pedestrian. We train the above model with the positive

and negative samples of the training set  $(x_i, y_i)$ , in which  $y_i \in \{-1, 1\}$ . The commonly available training algorithms include SVM and boosting, and we employ the linear SVM for training the parameter  $w$ :

$$w^j = \arg \min_w \frac{1}{2} w \cdot w + C \sum_i \max(0, 1 - y_i w \cdot \varnothing(x_i)) \quad (6)$$

in which  $x_i$  is assumed to be of a fixed size during training and testing. We define a feature vector  $\varnothing(x)$  to deal with the windows of different sizes.

### 3.2. Models with Fixed Resolutions

If an image contains objects of different resolutions at the same time, the detector of a fixed resolution usually cannot detect all different objects simultaneously. Because we can describe the different distances of pedestrians in an image, for each window  $x$  we can define a binary variable  $s$  to represent the distance of a pedestrian. We use  $s=0$  to represent the distant target pedestrians, and use  $s=1$  to denote the close target pedestrian. Our classifier is the same as the previous one.  $f(x, s) = w \cdot \varnothing(x, s)$ ,

$$\varnothing(x, s) = \begin{cases} \begin{bmatrix} \varnothing_0(x) \\ 1 \\ 0 \\ 0 \end{bmatrix} & \text{if } s=0 \end{cases} \text{ and } \varnothing(x, s) = \begin{cases} \begin{bmatrix} 0 \\ 0 \\ \varnothing_1(x) \\ 1 \end{bmatrix} & \text{if } s=1 \end{cases} \quad (7)$$

where  $\varnothing_0(x)$  and  $\varnothing_1(x)$  denote the features at different scales, such as a pedestrian of 50 pixels and a pedestrian of 100 pixels.

### 3.3. Multi-Scale Multi-Resolution Model

For the close target pedestrians with  $s=1$ , we can transform a model for high-resolution targets to two models for different resolution targets. For instance, we can transform a 100 pixels window into two windows of 50 pixels, and calculate the features at the small window scale. In this way, we can transform a model for high-resolution target to two models for different resolution targets:

$$\varnothing(x, s, z) = \begin{cases} \begin{bmatrix} \varnothing_1(x) \\ 1 \\ 0 \\ 0 \end{bmatrix} & \text{if } s=0 \end{cases} \text{ and } \varnothing(x, s, z) = \begin{cases} \begin{bmatrix} \varnothing_0(x) \\ 0 \\ \varnothing_1(x) \\ 1 \end{bmatrix} & \text{if } s=1 \end{cases} \quad (8)$$

With the above formula we can transform the object features at a fixed resolution into features at different resolutions. However, because  $\varnothing_0(x)$  is different at different resolutions, the linear SVM is not suitable for training models.

### 3.4. Multi-Resolution DPM Algorithm for Pedestrian Detection

A significant feature of the above method is that a rigid template is used for object detection at both large and small scales. The description operators at low levels (e.g., HOG feature) are adaptable to small image deformation [12]. However, such method is not applicable in case of large scales. For example, HOG feature detector is invariant to different postures of a 50 pixels height pedestrian,

but not invariant to the 100 pixels height pedestrians. If we are to detect a large-scale target, we can choose a low-resolution template. And if we hope to gain more information, we can select a high-resolution template. For a good adaptability to the deformation at a large scale, we adopt a DPM model. As a hidden parameter  $z$  is defined in the DPM model, we use  $\varnothing_1(x, z)$  as the combination of HOG feature and the deviation.

$$\varnothing(x, s, z) = \begin{cases} \varnothing_0(x) \\ 1 \\ 0 \\ 0 \end{cases} \text{ if } s=0 \text{ and } \varnothing(x, s, z) = \begin{cases} \varnothing_0(x) \\ 1 \\ \varnothing_0(x, z) \\ 0 \end{cases} \text{ if } s=1 \quad (9)$$

The classifier passes through all the hidden variables at last, and calculates  $f(x, s) = \max_z w \cdot \varnothing(x, s, z)$ :

$$f(x, s) = \begin{cases} w_0 \cdot \varnothing_0(x) + b_0 & \text{if } s=0 \\ w_0 \cdot \varnothing_0(x) + \max_z \varnothing_1(x, z) + b_1 & \text{if } s=1 \end{cases} \quad (10)$$

$f(x, s)$  is transformed into a standard linear template for calculating the response at a low resolution. For calculating the response at a high resolution,  $f(x)$  would need to search all part models to find the model which makes the maximum response. Suppose the distances between different parts and the root filter are independent from each other, the following formula can be calculated with the  $QP$  algorithm:

$$\max_z w_1 \cdot \varnothing_1(x, z) = \max_z \sum_j w_j \cdot \varnothing(x, z_j) + \sum_{j,k \in E} w_{jk} \cdot \varnothing(z_j, z_k) \quad (11)$$

in which  $z_j$  denotes the location of part  $j$ ,  $w_j$  denotes the template of part  $j$ ,  $w_{kj}$  denotes the deformable model of part  $j$  and  $k$ , and  $E$  denotes the boundary.  $\varnothing(x, z_j)$  represents the HOG feature at location  $z_j$ , and  $\varnothing(z_j, z_k)$  represents the deformation difference between part  $j$  and part  $k$ . For a given training set  $(x_i, s_i, y_i)$ , we can employ the LSVM algorithm for training the model parameter  $w$ .

At the stage of model training, we set  $s_i = 1$  if the training sample has a large scale ( $h > 0$ ), in which  $s_i$  is no longer a hidden variable. And we set  $s_i = 0$  if the training sample has a small scale ( $h < 50$ ), in which  $s_i$  is not a hidden variable, either. If the training sample has a medium scale ( $50 < h < 100$ ), the training sample can be considered as both a high resolution object and a low resolution object. In this case,  $s_i$  becomes a hidden variable, which can be added to the set of hidden variables of the LSVM model for training. The rough procedure consists of the random initialization of variables  $s_i$  and  $z_i$ , the calculation of model parameter  $\beta$  in model training, and the acquisition of value for the hidden variable in accordance to the maximum response which would be taken into the next iteration.

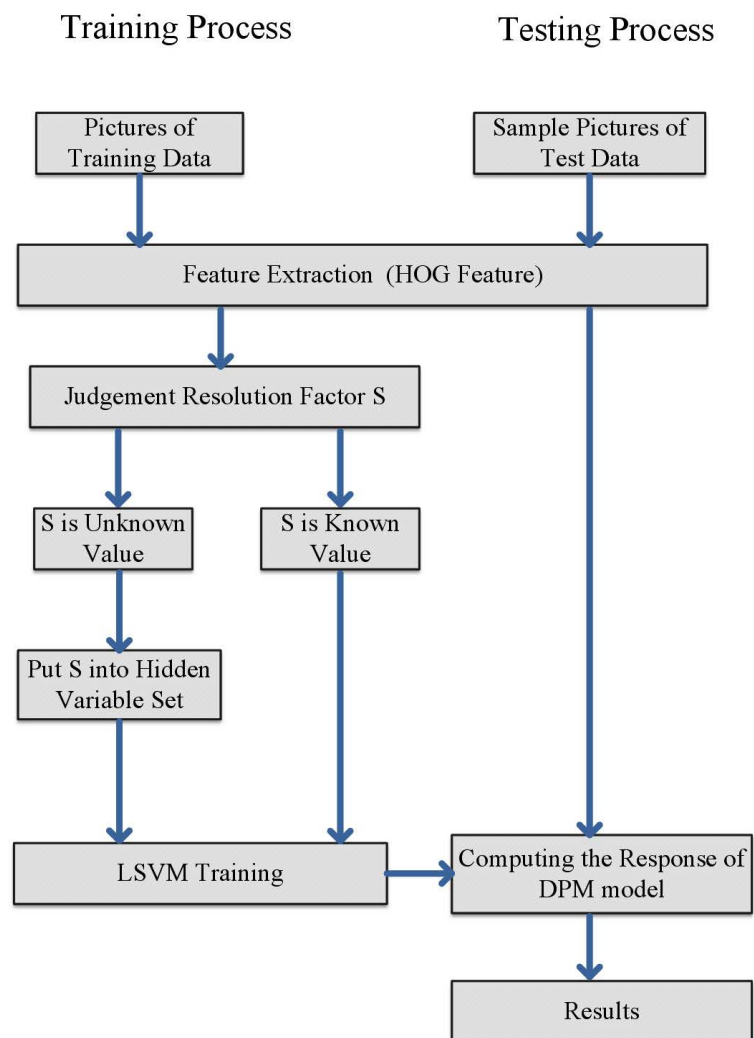
## 4. Experiment Process and Dataset

### 4.1. Overview of the Algorithm

Our proposed multi-resolution DPM is similar to a hybrid deformable model

with two target models. But there are also big differences between these methods. Firstly, many parameters are shared in our deformable model, while all parameters in the hybrid deformable model are independent from each other. Secondly, our multi-resolution deformable model consist a different procedure for the variable  $s_i$ . At the training state,  $s_i$  is a hidden variable, while at the test state  $s_i$  would become a visible variable. The procedure of pedestrian detection by multi-resolution DPM is shown in **Figure 3**.

First, DPM parameters  $w_0$  and  $w_1$  are initialized by the initialization method as illustrated before.  $w_0$  is the model parameter at a low resolution, while  $w_1$  is the model parameter at a high resolution. At the training state, we set the value of  $s$  by considering the height  $h$  of a trained sample, for which  $s = 1$  if  $h > 100$  and  $s = 0$  if  $h < 50$ . For the samples with  $s = 1$ , LSVM can be used to train a standard DPM model, which renders the model parameter  $w_1$ . For the samples with  $s = 0$ , a linear SVM can be used to train a DPM model (no hidden variable is involved), which renders the model parameter  $w_0$ . For the samples with height  $50 < h < 100$ , we add  $s$  to the set of hidden variables, and



**Figure 3.** The processing steps of pedestrian detection on multiple resolutions.



train the model with the LSVM algorithm.

## 4.2. Evaluation Method

### Whole Image Evaluation

The detection result for an image consist of a set of bounding boxes ( $BB$ ) and a corresponding confidence score [13]. If the detection result ( $BB_{dt}$ ) and the standard result ( $BB_{gt}$ ) has a great extent of overlapping, we consider the detection result matches the standard result. We define that a detection result matches the standard result if they have over 50% parts overlapped:

$$a_0 = \frac{area(BB_{dt} \cap BB_{gt})}{area(BB_{dt} \cup BB_{gt})} > 0.5 \quad (12)$$

Each ( $BB_{dt}$ ) can match to at most one ( $BB_{gt}$ ), which means that every detected ( $BB_{dt}$ ) can only match one ( $BB_{gt}$ ) but not multiple ( $BB_{gts}$ ). Therefore, if a detection result ( $BB_{dt}$ ) could match multiple ( $BB_{gts}$ ) we only select the ( $BB_{gt}$ ) with the highest confidence as the final detection result. If a ( $BB_{dt}$ ) is not matched with any ( $BB_{gt}$ ), it is labeled as false positive. And if a ( $BB_{gt}$ ) is not matched with any ( $BB_{dt}$ ), it is labeled as false negatives as well.

## 4.3. Experiment Dataset

### 1. INRIA

The INRIA data set is a static pedestrian detection database which has been widely employed in recent researches. The training set consists of 614 positive samples (containing 2416 pedestrians) and 1218 negative samples. The test set consists of 288 positive samples (containing 1126 pedestrians) and 453 negative samples. Images in the INRIA data set are mainly collected from google, GRAZ-01 and personal photographs.

### 2. Caltech Pedestrian Database

The Caltech Pedestrian Database is a large scale database. It consists of videos of  $640 \times 480$  pixel with 30 frames per second, captured by in-vehicle cameras for about 10 hours. Within these videos, 250,000 frames (around 137 minutes), 350,000 bounding boxes, and 2300 pedestrians are manually annotated by human experts. The data set is divided into 10 sets, among which sets 00 - 05 are used for training, and sets 06 - 10 are used for testing. In our experiment, we also employ sets 00 - 05 for training and sets 06-10 for testing.

Pillor [14] divide samples in the Caltech Pedestrian Library into distant ( $h < 30$ ), medium ( $30 < h < 80$ ) and near  $h > 80$  types, according to the heights ( $h$ ) of pedestrians.

The model feature of DPM is based on the HOG feature of the image pyramid, and DPM has good adaptability to a certain range of scale changes. Usually the pedestrian samples in our data set are not too small, so there is enough information for building the DPM model.

In the Caltech pedestrian database, the average height of the samples is 48 pixels. More than 84% of the positive samples have height greater than 30 pixels,

more than 16% of the positive samples have height greater than 80 pixels, and about 69% of the positive samples have height from 30 to 80 pixels. In the case of high-resolution ( $h > 100$ ), the DPM model show very good results in traditional DPM model tests. In this work we defined targets with height  $h > 100$  as the high resolution target, targets with height  $h < 50$  as the low resolution targets, and targets with height  $50 < h < 100$  as the unknown resolution targets in which case the resolution factor is treated as a hidden variable.

## 5. Results and Discuss

### 5.1. The Result in INRIA

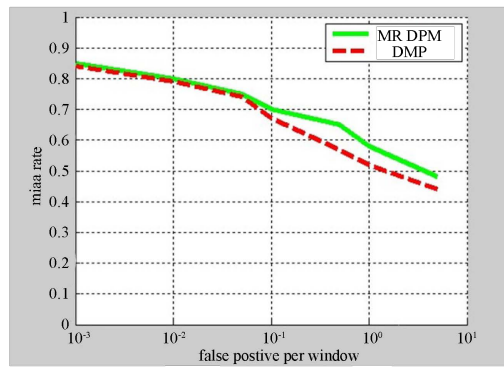
There are nearly one thousand pictures in INRIA pedestrian database. Our model is trained on the INRIA training set, and evaluated on the INRIA test set. The multi-resolution DPM-based pedestrian detection algorithm acquires a precision of 87.2% on INRIA pedestrian database, which is slightly better than 86.9% as the precision of the standard DPM. This is mainly because that there are too few picture samples in this data set, and the pedestrian scale is too big in the pictures. Only a few pictures contain the small-scale pedestrians, in which case the multi-resolution DPM detector cannot change the detection method at the high resolution. Therefore, the detection result is almost the same as the standard DPM result.

### 5.2. The Result in Caltech Database

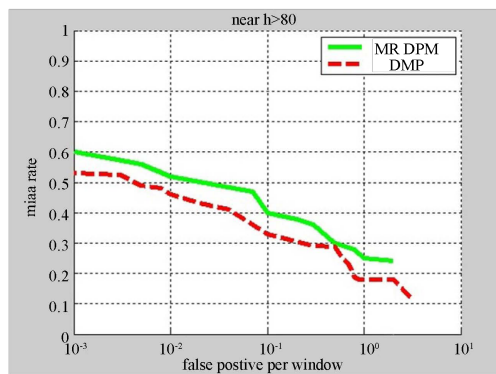
Training our model on the Caltech pedestrian database is more challenging than on the INRIA dataset. The number of samples in the Caltech pedestrian dataset is large, which is much larger than the number of samples in the INRIA dataset. The Caltech pedestrian dataset consists samples of  $640 \times 480$  pixel resolution, in which many pedestrian targets of small scales are included.

In this section, set0 to set5 in Caltech database are employed for training, and set6 to set11 are chosen for testing. **Figures 4(a)-(d)** show the detection results for all-distance, near-distance, middle-distance and far-distance samples, respectively. Specifically, the near-distance corresponds to pedestrians with height  $h > 80$  pixel, the middle-distance corresponds to pedestrians with height  $h > 30$  but  $h < 80$ , and the far-distance corresponds to pedestrian with height  $h < 30$ .

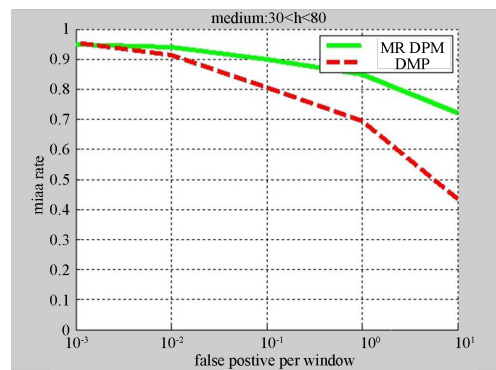
The experiment result suggests that the multi-resolution DPM algorithm renders better detection results than that the standard DPM in terms of all testing sets, **Figure 4(a)** shows the result based on all test samples. The multi-resolution DPM-based pedestrian detection algorithm achieves a missing rate of 52% (1FPPI), which is much better than the missing rate 59% (1FPPI) achieved by the DPM pedestrian detection algorithm. This result suggesting that multi-resolution DPM has better detection effect than the standard DPM. In terms of the large-scale samples ( $h > 80$ ) in Caltech database, the multi-resolution DPM-based detection algorithm renders a missing rate of 18% (1FPPI), while the standard DPM-based detection algorithm renders a missing rate of 26% (1FPPI),



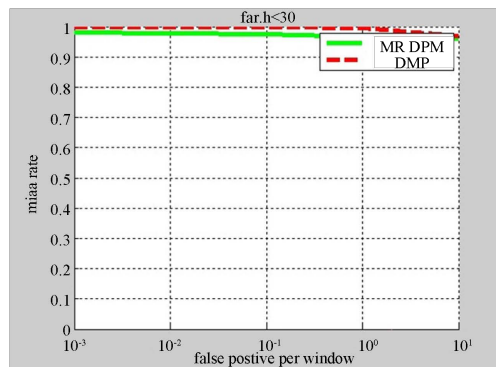
(a)



(b)



(c)



(d)

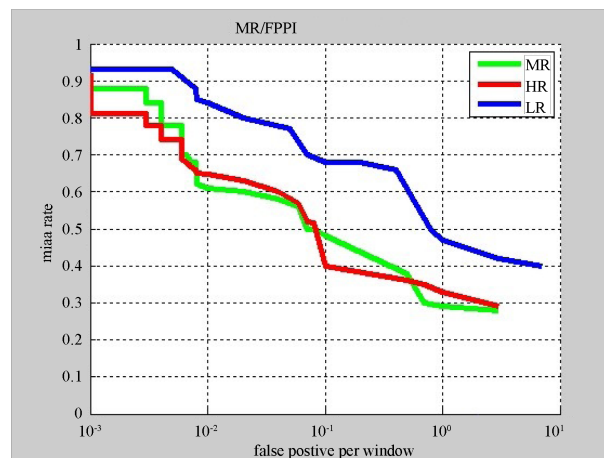
**Figure 4.** The result on Caltech pedestrian data set by different distance. (a) The result on all-distance samples; (b) The result on near-distance; (c) The result on middle-distance; (d) The result on far-distance samples.

which suggests that the multi-resolution DPM also outperforms the standard DPM for large-scale targets. For small-scale samples, the difference in both models is not as obvious as shown in **Figure 4(b)**, which is mainly because, that the useful information at small scales is very limited (the objects at less than 30 pixels are known as small-scale objects). At this point, the detection algorithm cannot acquire enough information for detection.

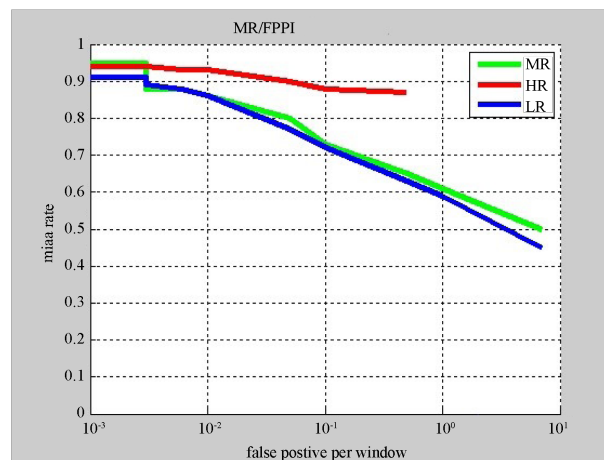
### 5.3. The Result in Part of Caltech Database

In order to better explain the experiment result, we further select 3000 pictures from the standard Caltech database for evaluation. This experiment is conducted to compare the detection effects of the detection algorithm at a high resolution (corresponding to the standard DPM algorithm), the detection algorithm at a low resolution (corresponding to merely the root filter-based DPM algorithm), and the proposed multi-resolution DPM algorithm.

The experiment result is shown in **Figure 5**, in which LR represents the low-resolution detection algorithm, HR represents the high-resolution detection al-



(a)



(b)

**Figure 5.** The results of different-scale targets on the subset of the Caltech pedestrian data set. (a) Large-scale target; (b) Small-scale target.

gorithm, and MR represents the multi-resolution detection algorithm.

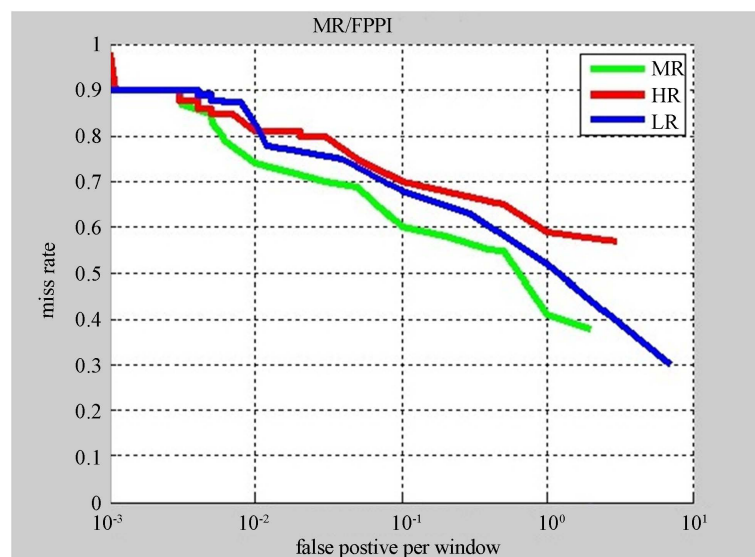
The results of large-scale target ( $h > 90$ ) detection as shown in **Figure 5(a)**, the multi-resolution DPM algorithm achieves similar performance as the standard DPM algorithm. Both algorithms are better than the low-resolution detector.

**Figure 5(b)** shows the terms of the small-scale target ( $h < 90$ ) detection, we find that the detection effect of the standard DPM algorithm drops quickly as the target scale grows small. The detection effect of the multi-resolution DPM algorithm is slightly lower than that of the rigid template.

According to the overall comparison is shown in **Figure 6**, the multi-resolution DPM algorithm is better than the high-resolution detection algorithm on the testing set. We find that the missing rate on test set is 52% with a rigid template. This is lower than 59%, which is the missing rate with the standard DPM algorithm. This is because most of the pedestrian samples are with height  $30 < h < 80$ , and even the high-resolution algorithm cannot detect the small-scale targets. The standard DPM algorithm achieves a good detection effect for the large-scale targets.

## 6. Conclusion

In this paper we proposed a Pedestrian Detection Method at Multiple Resolution. Especially during pedestrian detection under the high resolution, such an algorithm can generate very significant effects. However, targets in images acquired in the real world are under diverse resolutions in most cases. Considering this, the standard DPM is subjected to great limitations. Here, a multi-resolution DPM algorithm based on the standard DPM algorithm is presented. In this way, pedestrian detection is fixed to different resolutions. For example, pedestrians under the high resolution can be detected through a deformable part model, while those under the low resolution are detected based on the rigid template. In



**Figure 6.** The result on the subset of the Caltech pedestrian data set.

Caltech, omission ratio of a multi-resolution DPM detector was 52% (1FPPI); comparatively, it became 59% (1FPPI) as far as a standard DPM detector was concerned. In the large-scale sample set of Caltech, omission ratio of the multi-resolution and the standard DPM detectors were 18% (1FPPI) and 26% (1FPPI) respectively. The general results of proposed method are better than the standard DPM.

## Acknowledgements

This research was partially supported by JSPS KAKENHI Grant Numbers 15K00425, 15K00309.

## References

- [1] Markus, E. and Gavrilu, D.M. (2009) Monocular Pedestrian Detection: Survey and Experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **31**, 2179-2195. <https://doi.org/10.1109/TPAMI.2008.260>
- [2] Bagheri, M., Siekkinen, M. and Nurminen, J.K. (2016) Cloud-Based Pedestrian Road-Safety with Situation-Adaptive Energy-Efficient Communication. *IEEE Intelligent Transportation Systems Magazine*, **8**, 45-62. <https://doi.org/10.1109/MITS.2016.2573338>
- [3] Guo, A., Xu, M., Ran, F., *et al.* (2016) A Real-Time Pedestrian Detection System in Street Scene. *International Journal on Smart Sensing and Intelligent Systems*, **9**, 1592-1613.
- [4] Zhe, L. and Davis, L.S. (2010) Shape-Based Human Detection and Segmentation via Hierarchical Part-Template Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**, 604-618. <https://doi.org/10.1109/TPAMI.2009.204>
- [5] Zhe, L., Hua, G. and Davis, L.S. (2009) Multiple Instance Feature for Robust Part-Based Object Detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, Miami, 20-25 June 2009.
- [6] Liu, Y., Lasang, P., Siegel, M., *et al.* (2016) Multi-Sparse Descriptor: A Scale Invariant Feature for Pedestrian Detection. *Neurocomputing*, **184**, 55-65. <https://doi.org/10.1016/j.neucom.2015.07.143>
- [7] An, M.S. and Kang, D.S. (2015) A Method of Robust Pedestrian Tracking in Video Sequences Based on Interest Point Description. *International Journal of Multimedia and Ubiquitous Engineering*, **10**, 35-46. <https://doi.org/10.14257/ijmue.2015.10.10.04>
- [8] Navneet, D. and Triggs, B. (2005) Histograms of Oriented Gradients for Human Detection. 2005 *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, 20-25 June 2005.
- [9] Afsar, P., Cortez, P. and Santos, H. (2015) Automatic Visual Detection of Human Behavior: A Review from 2000 to 2014. *Expert Systems with Applications*, **42**, 6935-6956. <https://doi.org/10.1016/j.eswa.2015.05.023>
- [10] Kim, S. and Cho, K. (2015) Design of High-Performance HOG Feature Calculation Circuit for Real-Time Pedestrian Detection. *Journal of Information Science and Engineering*, **31**, 2055-2073.
- [11] Lin, C.F., Chen, C.S., Hwang, W.J., *et al.* (2015) Novel Outline Features for Pedestrian Detection System with Thermal Images. *Pattern Recognition*, **48**, 3440-3450. <https://doi.org/10.1016/j.patcog.2015.04.024>
- [12] Felzenszwalb, P.F., *et al.* (2010) Object Detection with Discriminatively Trained

Part-Based Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**, 1627-1645. <https://doi.org/10.1109/TPAMI.2009.167>

- [13] Piotr, D., *et al.* (2012) Pedestrian Detection: An Evaluation of the State of the Art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **34**, 743-761. <https://doi.org/10.1109/TPAMI.2011.155>
- [14] Piotr, D., Belongie, S. and Perona, P. (2010) The Fastest Pedestrian Detector in the West. *BMVC*, **2**.



Scientific Research Publishing

**Submit or recommend next manuscript to SCIRP and we will provide best service for you:**

Accepting pre-submission inquiries through Email, Facebook, LinkedIn, Twitter, etc.

A wide selection of journals (inclusive of 9 subjects, more than 200 journals)

Providing 24-hour high-quality service

User-friendly online submission system

Fair and swift peer-review system

Efficient typesetting and proofreading procedure

Display of the result of downloads and visits, as well as the number of cited articles

Maximum dissemination of your research work

Submit your manuscript at: <http://papersubmission.scirp.org/>

Or contact [jcc@scirp.org](mailto:jcc@scirp.org)

