A fast fused part-based model with new deep feature for pedestrian detection and security monitoring

Eric Juwei Cheng a, Mukesh Prasad b,⇑, Jie Yang b, Pritee Khanna c, Bing-Hong Chen a, Xian Tao d, Ku-Young Young a, Chin-Teng Lin b

1. Department of Electrical Engineering, National Chaio Tung University, Hsinchu, Taiwan
2. Centre for Artificial Intelligence, School of Computer Science, FEIT, University of Technology Sydney, Australia
3. Computer Science & Engineering Discipline, Indian Institute of Information Technology, Design and Manufacturing Jabalpur, India
4. Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, China

a r t i c l e i n f o

Article history:

Received 14 May 2019

Received in revised form 22 August 2019 Accepted 19 September 2019 Available online 9 October 2019

Keywords:

Pedestrian detection

Haar-like feature

Deep fused feature

Deformable partmodel

Security monitoring

a b s t r a c t

In recent years, pedestrian detection based on computer vision has been widely used in intelligent trans-portation, security monitoring, assistance driving and other related applications. However, one of the remaining open challenges is that pedestrians are partially obscured and their posture changes. To address this problem, deformable part model (DPM) uses a mixture of part filters to capture variation in view point and appearance and achieves success for challenging datasets. Nevertheless, the expensive computation cost of DPM limits its ability in the real-time application. This study propose a fast fused part-based model (FFPM) for pedestrian detection to detect the pedestrians efficiently and accurately in the crowded environment. The first step of the proposed method trains six Adaboost classifiers with Haar-like feature for different body parts (e.g., head, shoulders, and knees) to build the response feature maps. These six response feature maps are combined with full-body model to produce spatial deep fea-tures. The second step of the proposed method uses the deep features as an input to support vector machine (SVM) to detect pedestrian. A variety of strategies is introduced in the proposed model, includ-ing part-based to full-body method, spatial filtering, and multi-ratios combination. Experiment results show that the proposed FFPM method improves the computation speed of DPM and maintains the per-formance in detection.

1. Introduction

Pedestrian detection is a popular research field in computer vision. In recent decades, it has attracted much attention for its applications, ranging from intelligent transportation and robotics to security surveillance. Although the question of how to detect pedestrians has been thoroughly investigated [[1]](#page11), primarily due to the intra-class variation of pedestrians in clothing and articula-tion, the problem still remains challenging and continues to attract research. Viola and Jones [[2]](#page11) developed a cascaded method for fast detection, computing a calculus map with Haar-like features and automatically selecting features with Adaboost [[3]](#page11). Dalal and Triggs [[4]](#page11) proposed histogram of oriented gradients (HoG) features, which are fed to a linear support vector machine (LSVM) to improve the performance. Their report indicates that HoG features contain more object information, thus contributing to higher



detection rate. Those local features and classifiers will lay a solid foundation to further improve pedestrian detection.

Regarding pedestrian detection, the HoG feature collection is one of the most popular local feature collections and exhibits a low miss rate with various classifiers. Instead of using common general descriptors (e.g., Haar-like features, local binary patterns, HoG), some specific features are designed according to the attri-butes of pedestrian. Zhang et al. [[5]](#page11) proposes informed Haar-like features to describe the appearance of the up-right human body. Combination of different features also achieves impressed perfor-mance. Dollar et al. [[6]](#page11) introduces ChnFtrs detector which adopts HoG, gradient and color information as ‘‘Integral Channel Fea-tures”. ChnFtrs can achieve state-of-the-art performance for single part template detection. Dollar et al. [[22]](#page11) proposes a multiscale pedestrian detector operating using the construction of an image pyramid. Walk et al. [[23]](#page11) proposes a new feature, self-similarity on color channels, which consistently improves detection perfor-mance both for static images and for video sequences, across dif-ferent datasets. However, using mixed features may lead to high

2

computational cost. Therefore, determination of an effective solu-tion without limiting to local features is the hitch for researchers. The key problem involves the detection strategy rather than the combination of classifiers and features used.

In recent years, part based models [[7–13]](#page11) for pedestrian detec-tion have received wide consideration. In contrast with full-body methods [[3,14–19]](#page11), part-based models jointly learn distinct parts and their interrelationships to make a judgement. In addition, this approach is more effective and robust in detecting jointed objects such as animals and robots that appear with various postures and collusion. On the other hand, part-based models are successful in well-known challenges, e.g., PASCALVOC (the Pattern Analysis, Sta-tistical Modeling and Computational Learning Visual Object Classes). Felzenszwalb et al. [[11,12]](#page11) proposed deformable part model (DPM) that utilizes HoG features and a latent variable sup-port vector machine (LatSVM) to train the pedestrian part model, which suppresses the disturbance caused by varying pedestrian postures and multi-views. Ouyang et al. [[20]](#page11) proposes a deep model which considers the visible parts as hidden variables, through a probabilistic framework to learn the relationships among various parts. A parts selective DPM (PS-DPM) is proposed [[30]](#page11), which selectively chooses the original part filters and addi-tional part filters to detect pedestrians possessing an umbrella. Cai [[31]](#page11) proposes a pedestrian detection method with improved deformable part model by training the two-pedestrian deformable part model. Xiao [[32]](#page11) presents a method to improve pedestrian detection using the part-based saliency maps.

Although, DPM achieves impressive success in pedestrian detection, the drawback is that it require a long computation time due to the expensive computation cost of HoG. The generation of HoG feature produces too many feature vectors that makes its cal-culation time-consuming. This study proposes a fast fused part-based model (FFPM) that use a hierarchical adaptive boosting detector with Haar-like features instead of HoG features. The pro-posed approach is evaluated with DPM, and the results show that the proposed FFPM architecture successfully improves the compu-tation speed of DPM and retains the performance in pedestrian detection.

This paper is structured as follows. [Section 2](#page11) reviews the related work. The proposed FFPM is described in detail in [Section 3](#page11). The experimental results and discussion of the proposed FFPM are shown in [Section 4](#page11). Finally, [Section 5](#page11) is devoted to the conclusions and future work.

2. Adaptive boostingand deformable part model

2.1. Adaptive boosting

A matrix feature j can separate positive and negative samples by threshold h. It viewed as weak classifier h, as shown in Eq. [(1)](#page11):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| h ¼ | 1 | if pjf j < pjhj | ð1Þ |  |
| 0 | otherwise |  |
| where | f j | is Haar-like wavelet features, hj is threshold, pj is | 1 |  |

adjuster.

Comparing with other kinds of feature selection method, like Principal Component Analysis (PCA) [[26–28]](#page11), Independent Compo-nent Analysis (ICA) [[29]](#page11), AdaBoost has good select ability and increase detection rate. AdaBoost is used in converge weak classi-fiers from Haar-like wavelet set as a new strong classifier. The mathematical formula can be defined as follow in Eq [(2)](#page11):

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| C x | 8 | 1 | T | at ht ðxÞ P 21 | | | T | at | 2 |  |
|  | > |  | P |  |  |  | P |  |  |  |
| ð Þ¼< | | 0 | t¼1 |  |  |  | t¼1 |  | ð Þ |  |
| otherwise | | |  |  |
|  | > |  |  |  |
|  | : |  |  |  |  |  |  |  |  |  |

where h and C denote weak and strong classifier, respectively, a represents weight for each weak classifier, @t and ht ð xÞ are the weight and predict result for weak classifier, respectively.

2.2. Deformable part model

Felzenszwalb [[15,16]](#page11) proposes DPM that consist of one root fil-ter, several part filters and the position between them. DPM use a matching score calculated according to the detection result of root filter and part filters. The matching score for root and part filters is defined by dot product for parameter of filter and a move deform-able vector in HoG. Felzenszwalb change deformable problem as binary classification problem. For each sample x, the score for x can get by:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| f | x | Þ ¼ | max | | b | x; y | 3 |  |
|  | bð | z Z x | Þ | uð Þ | ð Þ |  |
|  |  |  | 2 ð |  |  |  |  |

where Β is parameter vector for model, z is hidden variables. Felzenszwalb defines a expanded SVM, called as Latent variable

support vector machine (LatSVM). One important feature is that when hidden informatinon has been provided for positive sample, training problem change to convex optimization problem, then by using gradient descent can get solution. Deformable part model is consist of three part: (a) One root filter (b) Several part filter (c) The position between part filter and root filter.

The model can split into three parts to initialize and train mixed model:

1. Initialize root filter: Based on the ratio of height and width to separate positive sample set P into m group.
2. Combine component: Combine the m initialized root filters as a non-component mixed model. Utilize LatSVM to retrain mixed models’ paramters for all positive and negative samples.
3. Initialize part filters: We used a heuristic algorithm to ini-tialize each part model. Each part model is consist of six models, their final positions can be found in the highest energy area.

In the DPM detection process, the score for each root position is the sum of response value for root filter and part filters. The math-ematical formula can be presented as follows Eq. [(4)](#page11):

|  |  |  |  |
| --- | --- | --- | --- |
|  | n |  |  |
| Scoreðx0; y0; lnÞ ¼ R0;l0ð x0; y0Þ þ Xi0 | | Di;l0 kð2ð x0; y0Þ þ ViÞ þ b | ð4Þ |
|  | ¼ |  |  |
| where | R0;l0 ðx0; y0Þ defines as response value for root | | filter, |
| Pin¼1Di;l0 | kð 2ð x0; y0Þ þ viÞ is the response value for part filters and | | |

k is the feature layer.

3. FFPM method

3.1. Architecture

The flow chart of the FFPM as shown in [Fig. 1](#page11) consists of three parts: part-boosting model, full-body boosting model and n classi-fier. In the part-boosting model, six adaptive boosting model corre-sponding to six body parts of pedestrian (i.e., head, left shoulder, right shoulder, hands, knees and feet) are used to extract the part features from the input image. The acquired part features are fed into full-body boosting model to learn the representation of pedes-trian. In the full-body boosting model, three adaptive boosting models are used to extract the deep features of the input images with different aspect ratio. The final detection result is determined by the classification result of support vector machine (SVM) and

|  |  |
| --- | --- |
| E.J. Cheng et al. / Measurement 151 (2020) 107081 | 3 |

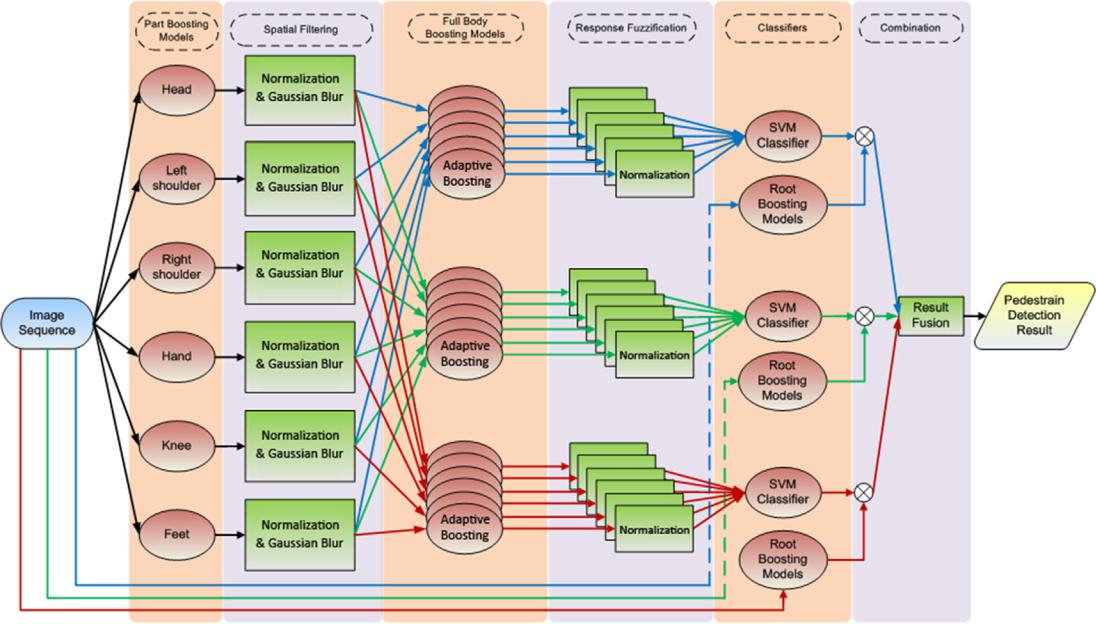


Fig. 1. Flow chart of the FFPM.

3.2.1. Part-boosting model



INRIA Person Dataset is used as traninging set in this paper as shown in [Fig. 2](#page11). [Fig. 3](#page11) depicts the cropped pedestrian images used for the training of part-boosting model. This paper uses six body parts for pedestrian detection: head, left shoulder, right shoulder, hands, knees, and feet. [Fig. 4](#page11) illustrates the body part images



Fig. 2. INRIA person dataset.

the root boosting model. The description and detail information of each process in [Fig. 1](#page11) is presented in the following sections.

3.2. Part-boosting model

The part-based feature extraction consists of two parts: part-

boosting models and spatial filtering. Part-boosting models use

six part filters to detect the corresponding parts of the pedestrians

from the input image. The output feature maps are normalized to

[0, 1] by spatial filtering. Then, feature response maps are gener-

ated for six distinct parts with Gaussian blurring that shares local

features to filter noise and disturbances.

Fig. 3.

Full body sample.

4



(a) Head (b) Left Shoulder

(c) Right Sholder (d) Hand

(e) Knees (f) Feet

Fig. 4. Six part features positive set. (a) Head. (b) Left shoulder. (c) Right sholder. (d) Hand. (e) Knees. (f) Feet.

cropped from pedestrian images described in [Fig. 3](#page11). For the train-ing of part-boosting model, the head images are normalized into 24\*28, left and right shoulders images to 24\*24, hands image to 24\*36, knees images to 24\*28 and feet images to 24\*36. By utilizing the adaptive boosting algorithm to train part-feature models, more powerful classifiers can be generated from numerous weak classi-fiers. Eq. [(5)](#page11) is used to extract six part features ([Fig. 5](#page11)).

|  |  |  |  |
| --- | --- | --- | --- |
| M |  |  |  |
| X1 | f mð xÞ | ð5Þ |  |
| F ð xÞ ¼ |  |
| m |  |  |  |
| ¼ |  |  |  |

where fm(x) denotes m weak classifiers, F(x) denotes the output of the final strong classifier. [Fig. 7](#page11) shows the part features with brighter regions representing higher response values. Notice that [Fig. 7](#page11)(a)–(f) vary from region to region; thus, it is essential to build part-based response score maps to normalize every sample.

3.2.2. Spatial filtering

[Fig. 8](#page11) indicates the flow chart of spatial filtering. This part dis-cusses how to use Gaussian Blur, which is shown in [Fig. 8](#page11) to modify part features from different distribution intervals into same inter-

|  |  |
| --- | --- |
|  | 5 |

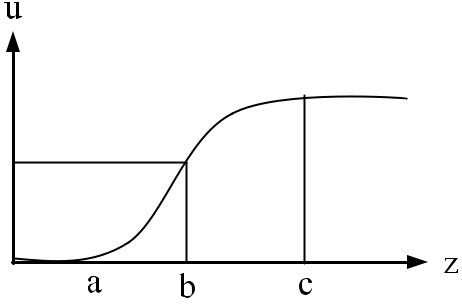
Gaussian MF, S-shaped MF and Z-shaped MF, etc. This study uti-lizes an ‘S’-type recursive function rather than other classical recursive functions or traditional maximum-minimum quantized methods mainly because the S-type recursive function is smoother and more flexible when addressing the z-axis and especially robust to rigid points. [Fig. 6](#page11) shows a S-type recursive function, and the transform formulation of the S-type recursive is as follows in Eq.



[(6)](#page11). [Fig. 7](#page11) shows the output part feature maps from the part-boosting model. It can be observed that these part feature maps have strong response at the position of the corresponding parts. For instance, the head feature map in [Fig. 7](#page11)(a) shows strong response to the head of pedestrian. To enhance the contrast of part feature maps, we use S-type recursive function described in Eq. [(6)](#page11) to normalize the part feature maps. The normalized feature maps shown in [Fig. 9](#page11) is close to binary images.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | 8 | 0 |  | z a 2 | | | |  | for z < a | 9 |  |  |
| S z ; a; b; c | | > | | 2 |  | c a |  | |  |  | for a 6 z 6 b | > | 6 |  |
| > | |  |  |  |  | |  | 2 |  | > |  |
|  |  | | > |  |  |  | |  | z c |  | > |  |  |
|  |  | | > |  |  |  | |  |  |  | > |  |  |
| ð |  | | Þ ¼ < | 1 | 2 | | | | c a |  | for b < z 6 c | = | ð Þ |  |
|  |  | | > | 1 |  |  | |  |  | for z > c | > |  |  |
|  |  | | > |  |  |  | |  |  |  |  | > |  |  |
|  |  | | > |  |  |  | |  |  |  |  | > |  |  |
|  |  | | > |  |  |  | |  |  |  |  | > |  |  |
|  |  | | : |  |  |  | |  |  |  |  | ; |  |  |
| Fig. 5. Original image. | (2) Gaussian Blurring | | | | | | | | |  |  |  |  |  |

Fig. 6. S-type recursive function.



val. The paper applies pixel wise methods to normalize different part features and filter the whole image to suit the second layer of the full-body feature extraction.

(1) Normalization

Membership Function (MF) is a basic idea in Fuzzy Theory, it can be used to describe property of fuzzy sets. Value for member-ship function is between 0 and 1. For DPM, the higher of value means it is more likely the predicted part and vice sersa. Member-ship function can divided as continuous and discrete MF. The for-mer type can separate as Triangular MF, Trapezoidal MF, Bell MF,

Gaussian blurring [[21]](#page11) is an image processing method on the spatial domain. Through Gaussian blurring, pixel value are shared to its neighbor if a region contains several higher value pixels; that is, the edge in the part feature maps becomes more smooth via per-forming Gaussian blurring. This operation is also helpful to reduce the noise within the feature maps. In the two-dimensional domain, the Gaussian blurring function can be defined as follows:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Gð x; yÞ ¼ | 1 | e | x2 þy2 | ð7Þ |  |
|  | 2r2 |  |
| 2pr2 |  |

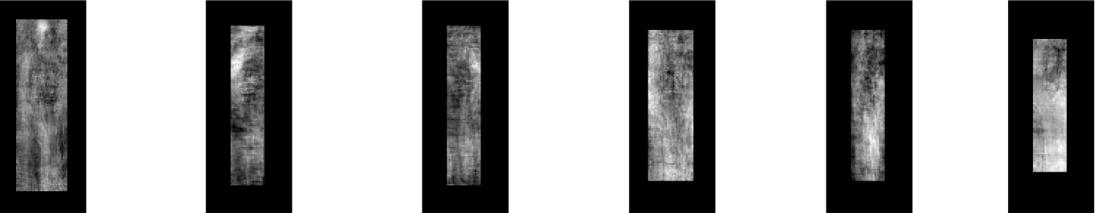
where r denotes the standard deviation of the normal distribution, which can be calculated as:

r ¼ 0:3 ðð SizeofMask 1Þ 0:5 1Þ þ 0:8 ð8Þ

Using ‘S’-type recursive function as the transform formula, pro-cessing the transformed part features using Gaussian blurring on the spatial domain, thus sharing them with neighboring pixels, achieves better performance, especially in terms of robustness. [Fig. 10](#page11)(a)–(f) show the results after Gaussian blurring.

3.3. Full-body feature extraction

Although, the body parts of pedestrian can be detected by the part boosting model, it is not sufficient to correctly capture the position of pedestrian because the relative position of body parts are changed due to the view point of camera and the pose of pedes-trian. To resolve this problem, this paper proposes full-body boost-



(a) Head (b) Left shoulder (c) Right shoulder (d) Hand (e) Knee (f) Foot

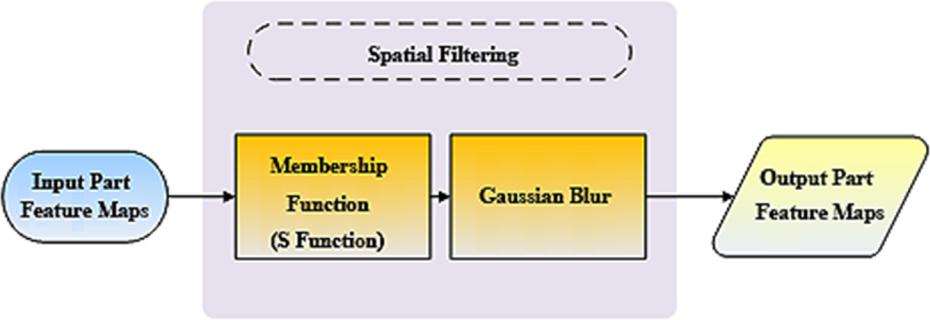
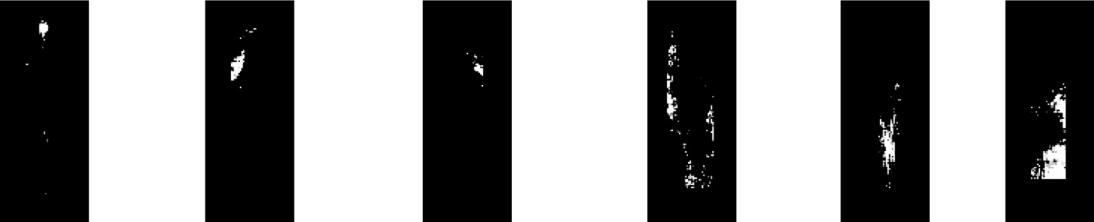
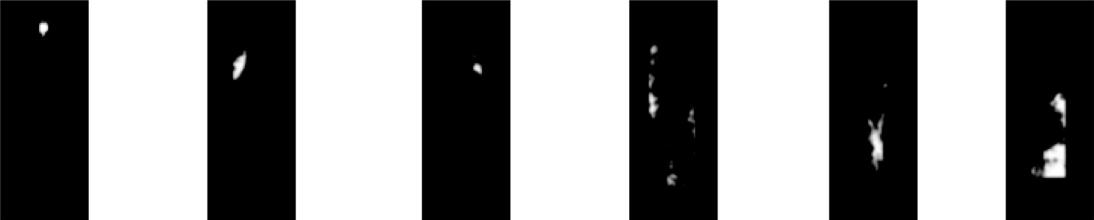
Fig. 7. Part features for six body parts. (a) Head. (b) Left shoulder. (c) Right sholder. (d) Hand. (e) Knee. (f) Foot.

Fig. 8. Flowchart of the spatial filtering.



(a) Head (b) Left shoulder (c) Right shoulder (d) Hand (e) Knee (f) Foot

Fig. 9. Binary features of six body parts. (a) Head. (b) Left shoulder. (c) Right sholder. (d) Hand. (e) Knee. (f) Foot.



(a) Head (b) Left shoulder (c) Right shoulder (d) Hand (e) Knee (f) Foot

Fig. 10. Gaussian map of six body parts. (a) Head. (b) Left shoulder. (c) Right sholder. (d) Hand. (e) Knee. (f) Foot.

ing model to hierarchically learn deep features from the part fea-ture maps. The full-body boosting model contains two parts: (1) deep feature extraction and (2) classification using SVM.

For pedestrian detection, the shape and size of pedestrians vary with respect to their pose and position. For this reason, the full-body boosting model employs three detection windows with dif-ferent aspect ratios. The full-body boosting model consist of three adaptive boosting models that are used to extract the features from three aspect ratios of input images. The aspect ratios of input images are illustrated in [Fig. 12](#page11). The adaptive boosting models are trained by parts feature maps described in [Fig. 10](#page11) for learning the deep features that combines the knowledge of all parts fea-tures. The visualization of the deep features are shown in [Fig. 11](#page11). Each deep feature has high response to the body part of interest, which means that the full-body boosting model can detect all body parts by deep features.

In the full-body boosting model, six full-body deep feature maps are acquired from each adaptive boosting model, and then passed into 3-way SVM for classification, as described in [Fig. 1](#page11). However, the magnitude of full-body deep feature maps are real number; therefore, it’s required to normalize the feature maps to

[0, 1] before the classification. [Fig. 13](#page11) illustrates the output feature maps normalized by S-type recursive function.

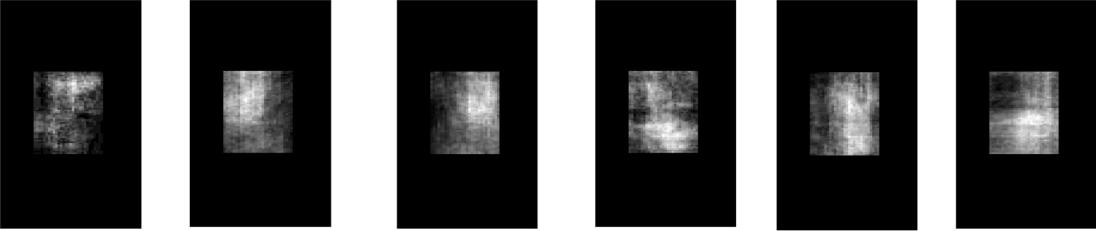
3.4. Classifier and combination strategy

In this study, the overall classification task is performed by SVM and the root-boosting model that is described in [Section 3.4.2](#page11). If the overlapping region between the positive detection of SVM and root-boosting model is large enough, this approach can help to reduce the computational cost and the false positive rate of the proposed model. The detail information is described in the fol-lowing sections.

3.4.1. Support vector machine

Support vector machine (SVM) is a supervised machine learning method which comes from statistical theory. It is used to solve bin-ary classification problem, calclulate optimal separate hyperplane for positive and negative sample space, get the minimal classifica-tion error. The principle is similar to neural network that use approximation for get expect accuracy for any polynomial func-tion. The formula for SVM is as below:

|  |  |
| --- | --- |
|  | 7 |

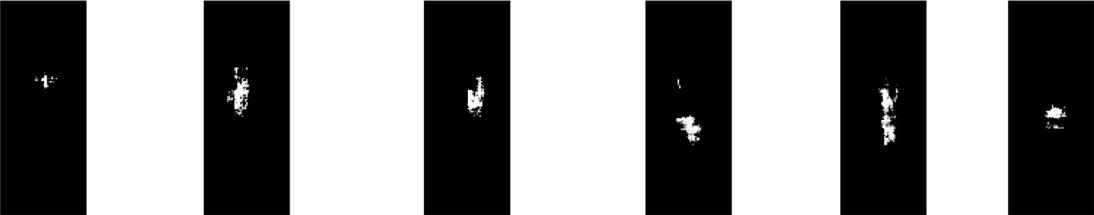


(a) Head (b) Left Sholder (c) Right Sholder (d) Hand (e) Knee (f) Feet

Fig. 11. The deep features learnt by the full-body boosting model. (a) Head. (b) Left shoulder. (c) Right sholder. (d) Hand. (e) Knee. (f) Feet.



Fig. 12. The training data of FFPM with different aspect ratios. The aspect ratio of images in (a) 30\*78, (b) 30\*92 and (c) 24\*94.



(a) Head (b) Left shoulder (c) Right shoulder (d) Hand (e) Knee (f) Foot

Fig. 13. Normalized deep feature maps. (a) Head. (b) Left shoulder. (c) Right sholder. (d) Hand. (e) Knee. (f) Foot.

fðxÞ ¼ WT X þ b ð9Þ

After transforming six full body features with complementary relationships, we combine six full body features into a joint six dimensional vector of pedestrian features and train an SVM with those vectors. The pedestrian features vector is defined as follows:

Vector ¼ ½Head; LeftShoulder; RightShoulder; Hand; Knee; Foot

ð10Þ

In this paper, the LIBSVM [[24]](#page11) toolbox is used to realize the SVM algorithm. Different classifiers are trained on a pedestrian database of three aspect ratios by setting the core function of SVM RBF, it has

two critical parameters (gamma (-g) and cost (-c)). Cross validation was implemented to avoid overfitting and to determine the opti-mal combination of parameters.

3.4.2. Root-boosting model

For the purpose of filtering out non-candidate regions quickly, we adopt adaptive boosting algorithm to train several full body detectors with different aspect ratios. [Fig. 12](#page11)(a)–(c) are different ratios full body pedestrian positive samples, the amount for first ratio of type is 520, the second type is 522, the last type include 195 positive samples. There are 1218 background pictures as the negative samples for the three type of ratios. Changing the scale size of each full body detector and slide across the image to find candidate regions contain a pedestrian possibility.

3.4.3. Multi-ratios combination strategy

This section introduces the combination strategy for the poten-tial candidate area and three aspect ratios of the full body models. The goal of achieving a combined result is to reduce the total com-

putational cost and reduce the number of false positives. Finally, by combining three multi-view prediction frames, one can obtain the real pedestrian region. The combination strategy must be executed under the same aspect conditions. Thus, combining the result of



(a) Detection result of SVM

(b) Detection result of the root-boosting model

(c) Detection result from the SVM and root-boosting model

(d) Detection result with different aspectratio (e) Final detection result

Fig. 14. Candidate regions of pedestrian. Bounding boxes with different colors stand for different aspectratio. (a) Detection result of SVM. (b) Detection result of the root-boosting model. (c) Detection result from the SVM and root-boosting model. (d) Detection result with different aspectratio. (e) Final detection result.

|  |  |
| --- | --- |
|  | 9 |

the SVM model as shown in [Fig. 14](#page11)(a) with the result of the full body model as shown in [Fig. 14](#page11)(b) provides the potential candidate area as shown in [Fig. 14](#page11)(c). The result has to be combined under the exact condition of combining prediction frames.

The prediction frames combining condition can be calculated as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| area Bp \ Bp | 1 | P 0:5 | ð11Þ |  |
| area Bp [ Bp | 1 |  |
|  |  |  |

When two prediction frames satisfy the condition, the algo-rithm will calculate new prediction frames:

|  |  |
| --- | --- |
| NewBox ¼ MeanðallBoxÞ | ð12Þ |

There are two situations for combining prediction frames: (1) with the same aspect ratio and (2) with the three different aspect ratios as shown in [Fig. 14](#page11)(d). The result based on the former result and combines it with the final result as shown in [Fig. 14](#page11)(e).

4. Experiment

The dataset used in this research is one of the most popular per-son datasets: the INRIA Person dataset, in which images have com-plex backgrounds and various pedestrian postures and lighting conditions, thus making detection extremely troublesome. The res-olution of images varies from 320 240 to 1280 960. The data-set was divided into training and testing datasets. Next sections will introduce the major baselines and two experimental results with respect to efficiency, average accuracy and miss rate.

4.1. PASCAL-VOC estimate metrics

PASCAL-VOC is a well known challenge for object detection, recognition, and classification in the field of computer vision. The estimate metrics may change based on the duplicate area for pre-dict and actual frame. [Fig. 15](#page11) shows duplicate area for a pedestrain: green full line is actual pedestrain frame, blue full line is predict pedestrain frame, there are some duplicate area between them. The condition of correct pedestrian is defined as in Eq. [(13)](#page11).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| area ðB det \ BÞ | P | 0:5 | 13 |  |
|  |
| area ðB det [ BÞ |  |
|  | ð Þ |  |

where B is actual pedestrian frame, Bdet defines as predict pedes-trian frame.

Average precision (AP) is the evaluation metric, which can be calculated as follows:

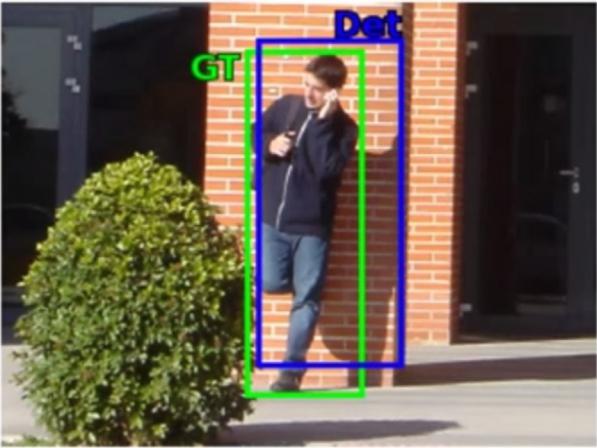


Fig. 15. Duplicate area.

|  |  |  |
| --- | --- | --- |
| 1 |  |  |
| AveP ¼ Z | pð r Þdr | ð14Þ |
| 0 |  |  |

The recall rate equals correct pedestrian number divided by the total pedestrian number:

|  |  |  |  |
| --- | --- | --- | --- |
| Recall ¼ | TruePositives | ð15Þ |  |
| TotalPositives |  |

The precision rate equals correct pedestrian number divided by the total prediction box number:

|  |  |  |  |
| --- | --- | --- | --- |
| Precision ¼ | TruePositives | ð16Þ |  |
| ðTruePositives þ FalsePositivesÞ |  |

4.2. Experimental results

[Fig. 16](#page11) indicates the relationship for Recall and Precision. It shows that the recall for VJ method is around 0.06, precision drop

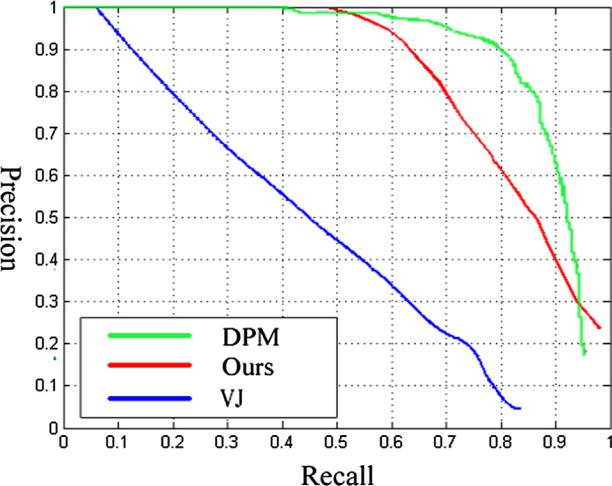


Fig. 16. The curve of precision and recall.

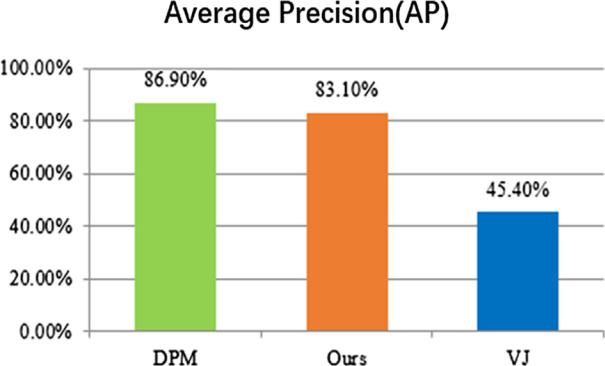


Fig. 17. Average precision.

Table 1

Average precision comparison.

|  |  |  |
| --- | --- | --- |
| Detector | AP (%) | Time (s) |
|  |  |  |
| DPM [[11]](#page11) | 86.90% | 12.85 s |
| VJ [[18]](#page11) | 45.40% | 0.77 s |
| Proposed method | 83.10% | 7.1 s |
|  |  |  |

Table 2

Performance comparison of proposed method with other different methods with different features.

|  |  |  |  |
| --- | --- | --- | --- |
| Detector | Feature | Classifier | Average Miss Rate |
|  |  |  |  |
| MT-LDCF [[19]](#page11) | HOG + gradients + color | Adaboost | 11% |
| Franken33 [[17]](#page11) | HOG + gradients + color | Adaboost | 13% |
| Informed Haar [[16]](#page11) | Informed Haar | Adaboost | 14% |
| VeryFast [[14]](#page11) | HOG + gradients + color | Adaboost | 16% |
| CrossTalk [[15]](#page11) | HOG + gradients + color | Adaboost | 19% |
| LatSVM-V2 (DPM) [[11]](#page11) | HOG | latent SVM | 20% |
| FPDW [[22]](#page11) | HOG + gradients + color | Adaboost | 21% |
| ChnFtrs [[6]](#page11) | HOG + gradients + color | Adaboost | 21% |
| MultiFtr + CSS [[23]](#page11) | HOG + self-similarity | latent SVM | 25% |
| Proposed Method | Deep Haar | Adaboost + RBF kernel SVM | 31% |
| FeatSynth [[24]](#page11) | HOG + texture | linear SVM | 31% |
| MultiFtr [[25]](#page11) | HOG + Haar | Adaboost | 36% |
| HogLbp [[26]](#page11) | HOG + LBP | linear SVM | 39% |
| Pls [[27]](#page11) | HOG + color + texture | PLS + QDA | 40% |
| HikSvm [[28]](#page11) | HOG | HIK SVM | 43% |
| LatSvm-V1(DPM) [[12]](#page11) | HOG | latent SVM | 44% |
| HOG [[4]](#page11) | HOG | linear SVM | 46% |
| FtrMine [[25]](#page11) | HOG + gradients + Haar + color | Adaboost | 58% |
| VJ [[18]](#page11) | Haar | Adaboost | 72% |
| Poselnv [[29]](#page11) | HOG | RBF kernel SVM | 80% |
|  |  |  |  |



(a) Crowded environments I (b) False positive sample

(c) Crowded environments II

Fig. 18. Detection results. (a) Crowded environments I. (b) False positive sample. (c) Crowded environments II.

rapidly from 1; recall for DPM method is in 0.4, precision drop slowly; recall for the proposed method is around 0.5, the precision is similar to DPM. In all, it can be observed that the proposed method has higher stability than the VJ method but less stability than the DPM. Also, the recall for the proposed method is intersect with DPM around 0.95, that means the same accuracy. In the aspect

of recall, the proposed method performs better than DPM with 0.98 and 0.96, respectively, which means the proposed method finds more true positive samples. [Fig. 17](#page11) shows the average precision that the score for DPM, the proposed and JV method are 86.9%, 83.1% and 45.4% respectively. The difference between the proposed method with DPM and VJ are 3.8%, 37.7% respectively. [Table 1](#page11) pro-

|  |  |
| --- | --- |
|  | 11 |

vides the average accuracy as shown in [Table 1](#page11). The average preci-sion of VJ is 45.4%, whereas for the proposed method is 83.1%.

4.2.1. Evaluation metric of speed

The executing speed can be calculated as below:

|  |  |  |  |
| --- | --- | --- | --- |
| Speed ¼ | TimeofTestingimage | ð17Þ |  |
| NumbersofTestingimage |  |

As [Table 1](#page11) shows, the execution speed of each image is 12.85 s when the DPM is used, whereas the speed of VJ is 0.77 s. The pro-posed method processes within 7.1 s. In comparison with the DPM, the proposed algorithm trades a slight reduction in accuracy for a significant improvement in speed, greater than 44.75%.

4.2.2. Efficiency estimate metrics

In this paper, we also evaluate the system performance com-pared with VJ, whereas the whole performance consists of average accuracy and computational cost, which can be calculated as:

|  |  |  |  |
| --- | --- | --- | --- |
| AP | APVJ | ð18Þ |  |
| Time | TimeVJ |  |

Transforming from Eq. [(18)](#page11), [Table 1](#page11) shows the DPM can achieve 3.43% average accuracy, whereas the proposed method can achieve

5.96% average accuracy. With respect to the whole performance, this study improves on the DPM by 1.74 times, thus manifesting higher efficiency than the DPM.

4.3. Comparison with state-of-the-art methods

[Table 2](#page11) shows the performance of othermachine learning based methods in terms of the average miss rate. The formulation of the miss rate is

|  |  |
| --- | --- |
| Missrate ¼ 1 Recall | ð19Þ |

The average miss rate is sampled uniformly from 10–2 to 100. Benefit from well defined features, informed Haar method has lower miss rate. The proposed method uses several Adaboost clas-sifiers with Haar-like feature for distinct body parts to produce response maps, feeding these response maps to construct spatial full body model. To our best knowledge, it is the first concept to bring up as a deep Haar feature.

4.4. The result for a real time pedestrian database

[Fig. 18](#page11) shows the testing results of our pedestrian detection sys-tem. The testing database contains various images with various backgrounds, lighting, different angles, and pedestrian postures. In [Fig. 18](#page11)(a), orange arrow shows the pedestrian occluded by other pedestrians, whereas in [Fig. 18](#page11)(b), green arrow shows the false negative detection resulting from the pedestrian reflection on the mirror. This pedestrian detection model can runs well based on dif-ferent background, illumination, angles and gestures, but still can-not solve inverted image and similar color for clothes and backgound problems well.

5. Conclusion and future work

Pedestrian detection are often used to judge the efficiency of object detection system. Traditional pedestrain detection only extract full-body features, resulting in a limited precision. The extensive experiments used in this paper demonstrates that FFPM represents a good tradeoff between speed and performance com-pared with DPM. Nevertheless, there still have room for improving the computation speed of the proposed system. In addition, the

performance of FFPM is limited by the non-rigid deformation and scale change of pedestrian since Haar-like features doesn’t have scale-invariant property. How to address these problems is the crucial issue in our future work.

Declaration of Competing Interest

The authors declare that they have no known competing finan-cial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported in part by the Australian Re-search Council (ARC) under discovery grant DP180100670 and DP180100656. Research was also sponsored in part by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0022 and W911NF-10-D- 0 0 02/TO 0 023 and in part by the Taiwan Ministry of Science and Technology MOST 106-2218-E-009-027-MY3.

References

1. [P. Dollár, C. Wojek, B. Schiele, P. Perona, Pedestrian detection: an evaluation of](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0005) [the state of the art, IEEE Trans. Pattern Anal. Mach. Intell. 34 (4) (2012) 743–](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0005) [761](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0005).
2. [P. Viola, M. Jones, Robust real-time face detection, Int. J. Comput. Vision 57 (2)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0010) [(2004) 137–154](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0010).
3. [R. Lienhart, J. Maydt, An extended set of haar-like features for rapid object](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0015) [detection, Int. Conf. Image Process. 1 (2002)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0015).
4. [N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, IEEE](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0020) [Comput. Soc. Conf. Comput. Vision Pattern Recognit. 1 (2005) 886–893](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0020).
5. [S. Zhang, C. Bauckhage, A.B. Cremers, Informed haar-like features improve](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0025) [pedestrian detection, IEEE Int. Conf. Comput. Vision Pattern Recogni. (2014)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0025) [947–954](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0025).
6. P. Dollar, Z. Tu, P. Perona, S. Belongie, Integral channel features, BMVC, 2009.
7. [D. Forsyth, Object detection with discriminatively trained part-based models,](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0035) [IEEE Trans. Pattern Anal. Mach. Intell. 47 (2) (2014) 6–7](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0035).
8. [M. Bergtholdt, J. Kappes, S. Schmidt, C. Schnörr, A study of parts-based object](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0040) [class detection using complete graphs, Int. J. Comput. Vision 87 (1–2) (2009)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0040) [93–117](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0040).
9. [M. Sun, S. Savarese, Articulated part-based model for joint object detection and](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0045) [pose estimation, IEEE Int. Conf. Comput. Vision (2011) 723–730](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0045).
10. [M. Andriluka, S. Roth, B. Schiele, Discriminative appearance models for](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0050) [pictorial structures, Int. J. Comput. Vision 99 (3) (2012) 259–280](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0050).
11. [P.F. Felzenszwalb, R.B. Girshick, D. Mcallester, D. Ramanan, Object detection](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0055) [with discriminatively trained part based models, IEEE Trans. Pattern Anal.](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0055) [Mach. Intell. 32 (9) (2010) 1627–1645](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0055).
12. [P. Felzenszwalb, D. McAllester, D. Ramanan, A discriminatively trained,](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0060) [multiscaled, deformable part model, IEEE Conf. Comput. Vision Pattern](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0060) [Recognit. (2008)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0060).
13. [J. Xu, S. Ramos, D. Vazquez, A.M. Lopez, Domain adaptation of deformable part-based models, IEEE Trans. Pattern Anal. Mach. Intell. 36 (12) (2014) 2367–](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0065) [2380](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0065).
14. [R. Benenson, M. Mathias, R. Timofte, L.V. Gool, Pedestrian detection at 100](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0070) [frames per second, CVPR (2012)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0070).
15. [P. Dollar, R. Appel, W. Kienzle, Crosstalk cascades for frame-rate pedestrian](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0075) [detection, CVPR (2012)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0075).
16. [S. Zhang, C. Bauckhage, A.B. Cremers, Informed haar-like features improve](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0080) [pedestrian detection, CVPR (2014)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0080).
17. [M. Mathias, R. Benenson, R. Timofte, L. Van Gool, Handling occlusions with](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0085) [franken classifiers, ICCV (2013)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0085).
18. [P. Viola, M.J. Jones, D. Snow, Detecting pedestrians using patterns of motion](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0090) [and appearance, 9th IEEE Int. Conf. Comput. Vision 63 (2) (2005) 153–161](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0090).
19. [C. Zhu, Y. Peng, A boosted multi-task model for pedestrian detection with](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0095) [occlusion handling, IEEE Trans. Image Process. 24 (12) (2015) 5619–5629](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0095).
20. W. Ouyang, X. Zeng, X. Wang, Partial occlusion handling in pedestrian detection with a deep model, IEEE Trans. Circuits Syst. Video Technol. 26 (11) (2016) 2123–2137, <https://doi.org/10.1109/TCSVT.2015.2501940>.
21. [R.a. Haddad, A.N. Akansu, A class of fast gaussian binomial filters for speech](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0105) [and image processing, IEEE Trans. Signal Process. 39 (3) (1991) 723–727](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0105).
22. [P. Doll´ar, S. Belongie, P. Perona, The fastest pedestrian detector in the west,](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0110) [British Machine Vision Conf. (2010)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0110).
23. [S. Walk, N. Majer, K. Schindler, B. Schiele, New features and insights for](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0115) [pedestrian detection, CVPR (2010)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0115).
24. [A. Bar-Hillel, D. Levi, E. Krupka, C. Goldberg, Partbased feature synthesis for](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0120) [human detection, ECCV (2010)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0120).

12 E.J. Cheng et al. / Measurement 151 (2020) 107081

1. [P. Doll](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0125)´[ar, Z. Tu, H. Tao, S. Belongie, Feature mining for image classification, IEEE](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0125) [Conf. Comput. Vision Pattern Recognit. (2007)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0125).
2. [X. Wang, T.X. Han, S. Yan, An HOG-LBP human detector with partial occlusion](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0130) [handling, Comput. Vision, 12th IEEE Int. Conf. Comput. Vision (2009) 32–39](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0130).
3. [W. Schwartz, A. Kembhavi, D. Harwood, L. Davis, Human detection using](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0135) [partial least squares analysis, IEEE Intl. Conf. Comput. Vision (2009)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0135).
4. [S. Maji, A.C. Berg, J. Malik, Classification using intersection kernel support](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0140) [vector machines is efficient, CVPR (2008)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0140).
5. [Z. Lin, L.S. Davis, A pose-invariant descriptor for human detection and](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0145) [segmentation in European Conf, Comput. Vision (2008)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0145).
6. [Y. Shimbo, Y. Kawanishi, D. Deguchi, I. Ide, H. Murase, Parts selective DPM for](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0150) [detection of pedestrians possessing an umbrella, Intelligent Vehicles](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0150) [Symposium (IV), 2016 IEEE, 2016](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0150).
7. [Y. Cai, Z. Liu, X. Sun, L. Chen, H. Wang, Research on pedestrian detection](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0155) [technology based on improved DPM model, IEEE 7th Annual International](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0155) [Conference on CYBER Technology in Automation, Control, and Intelligent](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0155) [Systems (CYBER), 2017](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0155).
8. [F. Xiao, B. Liu, R. Li, Pedestrian object detection with fusion of visual attention](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0160) [mechanism and semantic computation, Multimedia Tools Appl. 1 (1) (2019)](http://refhub.elsevier.com/S0263-2241(19)30947-9/h0160).