Group Cost-Sensitive BoostLR With Vector Form Decorrelated Filters for Pedestrian Detection

Chengju Zhou, Meiqing Wu, and Siew-Kei Lam, *Member, IEEE*

***Abstract*— Pedestrian detection has achieved notable progress in the field of computer vision over the past decade. However, existing top-performing approaches suffer from high computational complexity which prohibits their realization on embedded platforms with low computational capabilities. In this paper, we propose a robust and fast pedestrian detection framework which is based on the Filtered Channel Feature (FCF) approach. The proposed framework exploits vector-form decorrelated filters to extract more discriminative channel features while benefiting from low computational complexity. A novel group cost-sensitive BoostLR (Boosting with Loss Regularization) algorithm is proposed to train the classifier. The proposed training strategy provides more emphasis to the harder samples by exploring the variations of negatives selected from different rounds in hard negative mining processing, and hence is able to boost the overall detection performance. In addition, the proposed method also benefits from the BoostLR framework to achieve better generalization. Experiments on the well-known Caltech, INRIA and CityPersons pedestrian detection datasets show that our proposed approach achieves the best detection performance among all of the state-of-the-art non-deep learning methods and can run one order of magnitude faster than classical FCF methods (e.g. Checkerboards).**

***Index Terms*— Pedestrian detection, decorrelated channel feature, cost-sensitive, boosting.**

I. Introduction

PEDESTRIAN detection plays an essential role in a wide range of applications including intelligent vehicles, surveillance and robotics. Previous works [1]–[13] have demonstrated that pedestrian detection is a challenging problem due to high intra-class variation, highly cluttered background, inconsistent illumination, etc. Apart from the need for high robustness, real-world applications often necessitate that pedestrian detection algorithms run in real-time with limited computational resources (e.g. embedded systems employed in autonomous vehicle and robotics) [14]–[16]. This imposes the requirement of low computational complexity on pedestrian detection algorithms in many real-world applications.

Recently, the Filtered Channel Feature (FCF) framework [10] has gained wide attentions and several works [10], [17]–[20] have demonstrated its effectiveness and efficiency for pedestrian detection. As indicated in [10], the FCF approaches vary based on the filters used in feature extraction. For example, Checkerboards exploited a set of checkerboard pattern as filters to extract filtered channel features [10]. RotatedFilters explored several decorrelated filters that are designed based on the orientation of the aggregated channels [20]. In addition to exploiting different filters for pedestrian detection, some works have explored priors or variants in the training data to boost the detection performance. For example, the height of pedestrian is used to separate the training set into different subsets [18], [21] based on the assumption that pedestrians with lower height are more difficult to detect. In [19], the posterior probability estimations from the previous stage are employed to partition the training samples into subsets with different detection difficulties. In addition, the learning algorithm also plays an important role in FCF framework. It aims to explore the filtered channel features and recognize the pedestrian from background. RealBoost [22] is widely used in existing FCF works [10], [17], [20]. The limitations of RealBoost are that it is too sensitive to outliers (e.g. due to annotation errors in the dataset), and it does not have the capacity to control the generalization of learned model which often induces severe overfitting.

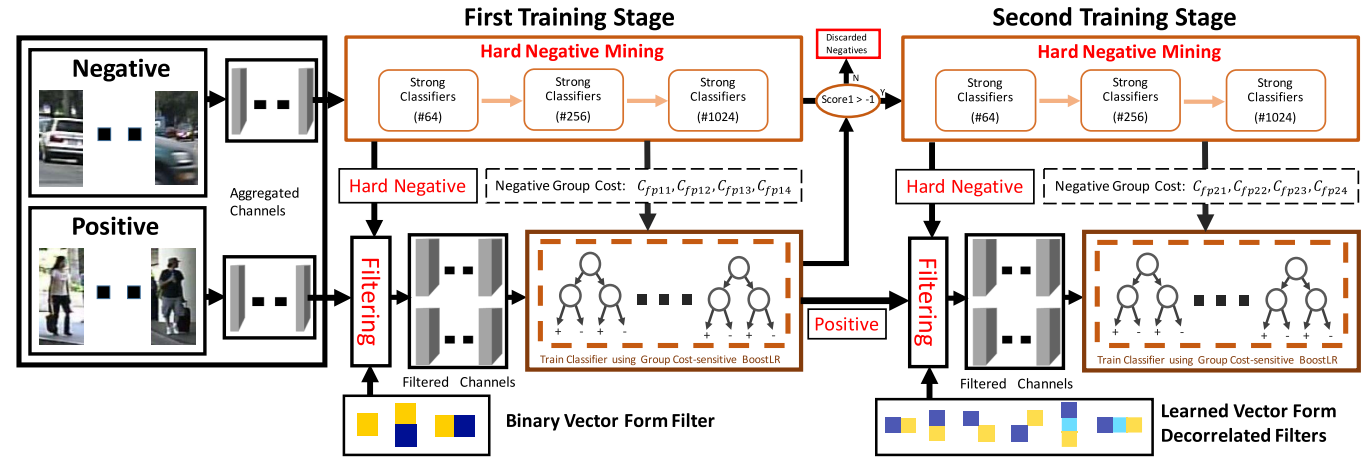
In this paper, we proposed a novel two-stage pedestrian detection framework that is based on the FCF approach. We explore vector-form[1] decorrelated filters which can combine the advantages of effective feature representation of decorrelated channel features and low computational complexity of vector-form filters. Experiments show that the proposed filters not only achieve better performance than the matrix-form filters used in existing FCF methods but also benefit from significant lower computational complexity. In order to further improve the detection performance, we propose a novel group cost-sensitive BoostLR algorithm which achieves better generalization for the learned model and explores the intrinsic variants of negatives selected from the commonly-used hard negative mining strategy. Specifically, BoostLR with α-tunable regularization loss [23] is exploited, in which the sensitivity to outliers varies between the asymptotically constant weights of LogitBoost and the exponential weights of AdaBoost as α changes. In addition, BoostLR formulates the shrinkage strategy that is widely used in boosting

Manuscript received July 26, 2018; revised February 22, 2019, May 23, 2019, and July 23, 2019; accepted October 2, 2019. This work was supported by the National Research Foundation Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) Programme with the Technical University of Munich at TUMCREATE. The Associate Editor for this article was N. Zheng. *(Corresponding author: Chengju Zhou.)*

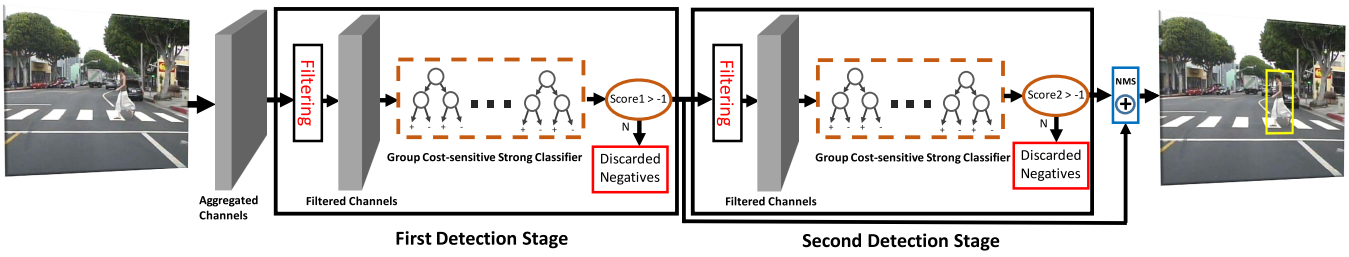
The authors are with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: zhou0271@e.ntu.edu.sg).

Digital Object Identifier 10.1109/TITS.2019.2948044

[1]We group the filters into two categories: matrix-form filters and vector-form filters. The matrix-form filters have dimension of $m \times n$ with $m > 1$ and $n > 1$ while vector-form filters have the length of m for different orientations, e.g. vertical ($m \times 1$), horizontal ($1 \times m$) and diagonal (non-zeros on the diagonals of $m \times m$ matrix).



(a)



(b)

Fig. 1. Illustration of the proposed detection framework for: a) training, b) testing. We use identical filters in both the training and testing procedures. The aggregated channels computed from the input image are used to obtain the filtered channels with vector-form filters. The binary vector-form filters are used in the first training and testing stage, while learned vector-form decorrelated filters are exploited in the second training and testing stage. During training, the negatives are grouped from different rounds of hard negative mining process and assigned corresponding costs. In the testing phase, the candidates that are recognized as positive in the first detection stage serve as inputs to the second detection stage. The final detection results are obtained after Non-Maximum Suppression (NMS) based on the averaged scores from the first and second detection stages.

family, which improves the generalization of the proposed algorithm. The negatives that are mined from different rounds of hard negative mining processing are assigned different costs which enforces more penalties on the mis-classified harder negatives.

The main contributions of this paper are summarized as follows:

1) We proposed a novel two-stage pedestrian detection framework as shown in Figure 1. Binary vector-form filters are used in the first stage, and in the second stage we employ 6 vector-form decorrelated filters (2 for vertical orientation, 2 for horizontal orientation and 2 for diagonals) for each of the 10 aggregated feature channels (3 LUV color channels, 1 gradient magnitude channel, and 6 channels for Histogram of Oriented Gradients) as described in [14]. The binary vector-form filters used in the first stage aims to solve the computation bottleneck of existing FCF framework and discard easy negatives. The vector-form decorrelated filters employed in the second stage focus on exploring local discriminative information to boost the detection performance while keeping the computational efficiency.
2) We performed extensive studies to show that decorrelated filters that have the same sign elements (i.e. weighted-sum filters where the coefficients are either all positive or all negative) show weaker discrimination than other filters. This important insight explains why previous FCF methods that adopt such weighted-sum filters (e.g. Checkerboards and RotatedFilters) achieve inferior detection performance even when large and complex filters are employed. Our work is the first to learn vector-form decorrelated filters from training samples using the FCF framework. In contrast to existing decorrelated filters methods [17], [18] that learn matrix-form decorrelated filters, our proposed method can extract more discriminative features and has the advantage of lower computational complexity.
3) We proposed a novel group cost-sensitive BoostLR learning algorithm for training the pedestrian detector. The inherent trait of commonly-used hard negative mining strategy is explored to assign different costs for negatives with different difficulties. These costs are then employed in the proposed algorithm to provide more emphasis to the negatives mined from latter rounds that exhibit higher detection difficulty, and hence is able to boost the overall detection performance. The proposed method also benefit from the advantage of BoostLR framework, where the margin loss can serve as the regularizer, in order to achieve better generalization.
4) We performed extensive evaluations using the well-known Caltech, INRIA and CityPersons datasets to show that our proposed method achieves the best

detection performance (i.e. MR[2] 11.98% on Caltech dataset, MR 10.71% on INRIA dataset and MR 23.25% on CityPersons dataset) among all of the state-of-the-art non-deep learning methods and can run one order of magnitude faster than classical FCF methods (e.g. Checkerboards).

The rest of paper is organized as follows. Section II conducts a review of existing top-performing pedestrian detection methods. The proposed vector-form decorrelated filters and group cost-sensitive BoostLR algorithm are introduced in Section III. Section IV presents extensive experimental results on well-known Caltech, INRIA and CityPersons datasets to demonstrate the effectiveness and efficiency of the proposed method over state-of-the-art methods. Finally, Section V concludes the paper.

II. Related Work

The general framework of pedestrian detection can be decomposed into feature representation and object classification. Based on how the features of pedestrian image are constructed, existing methods can be divided into two families: hand-crafted [10], [14], [24]–[29] and learned from data [11], [17], [30]–[32].

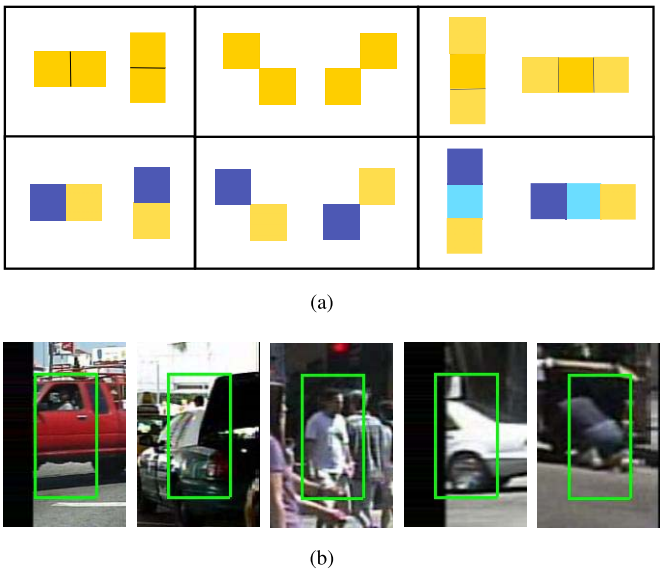
For hand-crafted features, HOG (Histogram of Oriented Gradients) [24] and ChnFtrs (Channel Features) [27] are well studied in the literature. The HOG feature and its variants have dominated the task of pedestrian detection for a long time after its introduction. However, research in recent years shows that the detection performance are inferior compared to other kinds of features. For features that are learned from data, the DCNN (Deep Convolution Neural Network) feature and decorrelated channel feature have attracted large amount of attention in the pedestrian detection community. In particular, all of the recent top-performing approaches [11], [30] on well-known Caltech pedestrian detection dataset rely on DCNN features.

However, the computational complexity and storage consumption of DCNN feature based methods limit their deployment in real-world scenarios. The DCNN features, which are used in Faster R-CNN [33] and SSD [34] detection framework, often requires very deep model (VGG [35]) which leads to high computational complexity. For example, VGG-16/19 incurs 15.3/19.6 billion FLOP for input image resolution 224x224 [36]. To accelerate the detection process, the DCNN feature based methods typically rely on high-end discrete GPUs [11], [30], [37] which require high power supply and active cooling. This limits their deployment on power constrained embedded systems (e.g. those used for autonomous driving and robotics [38]). Some works [39]–[42] have proposed network compression and acceleration for realizing deep convolution neural network on embedded platforms. Others have tried to design more compact networks, e.g. GoogLeNet [43], ResNet [36], MobileNet [44] and YOLO [45], [46]. However, the computational complexity of the detection framework that adopts these networks are still very high. For example, Faster R-CNN for MobileNet with input image resolution 600x600 still incurs 30.5 billion FLOP on the COCO dataset [44]. As such, the DCNN feature based methods are still not suitable for embedded platforms that have tight computational resources or employ battery as their main power source.

The Filtered Channel Feature (FCF) framework have gained popularity in pedestrian detection community after it was introduced in [10]. This framework decomposes the detection process into three correlated steps: aggregated channel computation, filtering over aggregated channel and object classification. In the FCF framework, the filters used in the second step are not restricted to hand-crafted or learned from data. The decorrelated channel feature was introduced in [17] where a set of decorrelated filters with size of 5×5 are learned from positive training samples for each aggregated channel [14]. These filters are then applied over corresponding aggregated channels to obtain decorrelated channel features. A method called Checkerboards [10] is proposed in which a naive set of checkerboard patterns are employed as filters to evaluate whether hand-crafted filters from statistical information of pedestrian [29] are superior to a naive set of checkerboard patterns. The same authors later proposed another method called RotatedFilters [20] in which a set of carefully tailored filters were designed based on the orientation of corresponding aggregated channels. The filters from Checkerboards and RotatedFilters demonstrated superiority in detection performance but suffer from high computational complexity due to the employment of large number of filters (i.e. 61 filters per channel) [10] or the need to filter over high resolution aggregated channels (i.e. one upsampled octave for input image and no downsampling for aggregated channels) [20]. To solve the high computational complexity in filtering step, [18] proposed to learn small multi-scale matrix-form decorrelated filters i.e., size of 2×2 and 3×3. The binary vector-form filters are explored in [19] and are shown to achieve notable improvement in the detection performance.

In addition to improve the detection performance from the perspective of feature extraction, some works have attempted to explore the variants in the training data. A group cost-sensitive Adaboost is proposed in [21] which takes advantage of the fact that training samples with low resolution are often difficult to classify. The positive samples are divided into low and high resolution set and more penalties are assigned to mis-classified low resolution samples. However, this partition is often influenced by the annotation errors, which are common in the large scale public datasets with some examples in Figure 2(b). In [18], a cost-sensitive learning algorithm with fused loss is proposed to explore the variants in training samples and alleviate the influence of annotation errors. This work has no discrimination between positives and negatives when assigning cost. However, the mis-classified positives should have larger cost than the negatives since missing detections (i.e. false negatives) are harder to recover. The work in [19] proposed to assign larger cost for mis-classified positives and divide the negatives into several sets. But the partition of negatives in [19] is based on the posterior probability estimation from previous detection stage which implies that it can only be used in a multi-stage detection framework. This requirement

[2]Log-average miss rate (MR) on False Positive Per Image (FPPI) ranges of $[10^{-2}, 10^{-0}]$. Lower MR represents better detection performance.



(a) (b)

Fig. 2. a) Visualization of learned vector-form decorrelated filters (Brighter pixels mean for positive numerical value while blue pixels mean negative numerical value. Best viewed in color), b) Examples of erroneous annotations from Caltech dataset.

restricts the first detection stage from taking advantage of variants in the training data. Furthermore, there is still a wide gap in the detection performance between top-performing FCF methods and deep learning methods which prevents their applicability in real-world scenarios.

The proposed detection framework in this paper differs from existing works in both feature representation and learning strategy. In our work, we exploit vector-form filters instead of matrix-form filters used in [10], [17], [18], and [20]. The vector-form filters are also employed in [19], but these are hand-crafted binary filters. Unlike [19], the vector-form filters used in the second stage of the proposed framework are decorrelated filters that are learned from training data. In contrast to the cost-sensitive strategies used in [18], [19], and [21], we partition the negatives into different subsets based on the commonly-used hard negative mining process. This strategy explores the intrinsic benefits for negatives with different detection difficulties. To the best of our knowledge, our work is the first to adopt this strategy for cost-sensitive learning. Compared with other boosting-like algorithms [18], [19], [21], the proposed method allows for the control of regularization strength and robustness to outlier with the link function and binding function respectively. This is important when training the detector since the training data often have some annotation errors which would affect the performance of trained detector. In such scenarios, the robustness to outliers would play an important role to reduce the effects of outliers on the learned model. Furthermore, in real-world scenarios, different kinds of mis-classification have different costs. For example, in autonomous driving, a missing detection of pedestrian may lead to fatal accident which means that this kind of missing detection should be associated with a higher cost when training the detector. Other boost-like algorithms often focus on only a single aspect (e.g. cost-sensitive in [21]) while the proposed method can handle multiple aspects at the same time using several hyperparameters (i.e. by using α to tune robustness to outliers, σ to control regularization strength and C_* for various mis-classifications).

III. Proposed Framework

In this section, we present the proposed framework from the perspective of feature representation and learning strategy. Figure 1(a) and Figure 1(b) illustrate the training and testing procedure of the proposed framework. It can be observed that the aggregated channels are first computed from input image. The aggregated channels consist of 10 channels of the same dimension, including 3 LUV color channels, 1 gradient magnitude channel, and 6 channels for Histogram of Oriented Gradients [14]. The filtered channels are then obtained from the aggregated channels using binary vector-form filters in the first stage and learned vector-form decorrelated filters in the second stage. It is worth noting that identical filters are used in the corresponding stages of training and testing procedure. During training, the negatives selected from different rounds of hard negative mining process are assigned different costs. The classifier is then trained using the proposed learning algorithm with filtered channel features and group costs at corresponding training stage. In the testing phase, the pedestrian candidates that passed both detection stages are classified as pedestrian and the confidence score is the combination of scores from the two detection stages. The final detection results are then obtained after Non-Maximum Suppression (NMS).

In Section III-A, we will introduce the proposed vector-form decorrelated filters that are able to extract more discriminative features than existing methods. Next, in Section III-B, we will present the proposed group cost-sensitive BoostLR (Boosting with Loss Regularization) learning algorithm which explores the variants in hard negatives and alleviates the influence of annotation errors in the training data.

A. Vector Form Decorrelated Filters

In order to take advantage of good feature representation of decorrelated filters and the computational efficiency of vector-form filters, we propose to learn vector-form decorrelated filters. The vector-form filters can be characterized by their dimensions and orientations. e.g. vertical filter with dimension of $m \times 1$, horizontal filter with dimension of $1 \times m$ and diagonal filter (non-zeros on the diagonals of $m \times m$ matrix). Different vector-form filters can extract information from specific orientation of local region. For example, the vertical discriminative information can be obtained using vertical vector-form filters. To fully explore the local information, we propose to learn vector-form decorrelated filters in vertical, horizontal and diagonal orientations. The covariance matrix $\mathbf{\Sigma}$ is first computed on the set of patches extracted from each aggregated channel [17]. Then the eigen-vector from the eigen-decomposition of $\mathbf{\Sigma}$ are employed as decorrelated filters. Consequently, the number of vector-form decorrelated filters for a specific orientation (e.g. vertical orientation) is equal to the length of filter. For instance, there are m learned vector-form decorrelated filters if the length of filter is m.

Some examples of learned vector-form decorrelated filters that we have investigated are shown in Figure 2(a). These include the vertical and horizontal filters with length of 2, diagonal filters with length of 2, and vertical and horizontal filters with length of 3. The corresponding eigen-values of the decorrelated filters decrease from upper to lower row.

The filters in the upper row of Figure 2(a) are weighted sum of pixels (e.g. [0.7141, 0.7001])[3] while filters in lower row are weighted discrepancy of pixels (e.g. [0.6934, -0.7205]). Similar phenomenon has also been observed in the decorrelated filters used in LDCF [17] and [18] where the decorrelated filter that corresponds to largest eigen-value has same sign coefficients. In order to better describe the filters, we partition the decorrelated filters into two subcategories: *weighted-sum filter* where the sign of filter coefficients are always the same, and *weighted-discrepancy filter* where the sign of filter coefficients are not the same. The decorrelated filters in upper row of Figure 2(a) are weighted-sum filter while filters in lower row are weighted-discrepancy filter. Our investigation on Caltech dataset in Section IV-B shows that the weighted-discrepancy filters have better discrimination capability than weighted-sum filters. When weighted-discrepancy and weighted-sum decorrelated filters are exploited together, the detection performance is slightly better than the case with weighted-sum filters, but is still much inferior compared with the case of weighted-discrepancy filters as illustrated in Figure 4(a). We will provide detail discussion of these results in Section IV-B. Therefore, only weighted-discrepancy vector-form decorrelated filters are employed in the second detection stage of the proposed framework.

B. Group Cost-Sensitive BoostLR

It has been well recognized that the regularization is important to assure good generalization of a classifier. Classical regularization often enforces a cost on classifier complexity through constraining parameters. Masnadi-Shirazi and Vasconcelos [23] finds that the margin losses can also serve as regularizers of posterior class probability estimations. In this section, we first introduce the boosting algorithm with tunable loss regularization (BoostLR). Then present our proposed group cost-sensitive BoostLR.

1) Boosting With Loss Regularization (BoostLR): The classifier $H(\boldsymbol{x})$ is defined to map a feature vector $\mathbf{x}$ to a class label y and can be expressed as,

$$H(\mathbf{x}) = sgn[p(\mathbf{x})] \tag{1}$$

where $p(\mathbf{x})$ is the classifier predictor and sgn refers to the operation that retrieves the sign. Given a non-negative loss $L(\mathbf{x}, y)$, the optimal predictor $p^*(\mathbf{x})$ can be obtained by minimizing the expectation of the loss function, i.e. risk,

$$R(p) = E_{\mathbf{X},Y}[L(p(\mathbf{x}), y)] \tag{2}$$

In practical, the optimal predictor is estimated by minimizing an empirical risk of Eq. 2. The empirical risk from a set of training samples $(\mathbf{x}_i, y_i)_{i=1}^N$, where $\mathbf{x} = (x_1, \ldots, x_d)^T \in \mathbb{R}^d$ is the feature vector for each training sample and $y \in \{-1, 1\}$ is the corresponding label, can be written as

$$R_{emp}(p) = \frac{1}{N}\sum_i L(p(\mathbf{x}_i), y_i) \tag{3}$$

Learning from finite samples can easily lead to over-fitting. One method to alleviate the problem of over-fitting is to enforce a cost on classifier complexity, by constraining parameters (e.g. adding the norm of parameters as penalty term). Another way is to use a margin loss $L_\phi(y, p(\mathbf{x})) = \phi(yp(\mathbf{x}))$ which assigns a penalty to correctly classified examples that are close to the decision boundary. In [23], it is demonstrated that the margin losses can themselves serve as regularizers of posterior class probability and the generalization of learned classifiers can be controlled with a combination of link function $f_\phi^*(\eta)$ defined in Eq. A.A2 (to determine the regularization strength) and binding function $\beta_\phi(v)$ defined in Eq. A.A10 (to determine the robustness to outliers due to annotations errors in the dataset). The details of proof are shown in Appendix A. Boosting with weights rule in Eq. A.A9 is denoted Boosting with Loss Regularization (BoostLR).

In boosting family, the exponential loss (i.e. $\phi(v) = e^{-v}$) and logistic loss (i.e. $\phi(v) = \log(1+e^{-v})$) are commonly-used loss functions. Based on the weights rule of exponential loss and logistic loss in Eq. A.B1 and Eq. A.B2, a tunable regularization loss is proposed in BoostLR [23] and its derivative can be written as

$$\phi'_\sigma(v) = -(1 - \frac{e^{\frac{v}{\sigma}}}{1+e^{\frac{v}{\sigma}}})\frac{1-\alpha}{2-3\alpha}(e^{-\alpha\frac{v}{\sigma}} + e^{\alpha\frac{v}{\sigma}}), \quad \alpha \in [0, \frac{1}{2}] \tag{4}$$

where $\sigma = \frac{\mu}{\xi+1}$, $\alpha = \frac{\xi}{1+\xi}$, $0 \le \xi \le 1$ and $0 < \frac{1}{\mu} \le 1$ is the shrinkage factor. As α varies, the $\phi'_\sigma(v)$ interpolates between the derivative of logistic loss and exponential loss. Hence, the loss $\phi_\sigma(v)$ is called α-tunable regularization loss in BoostLR [23]. Appendix B details the derivation of Eq. 4 from exponential loss and logistic loss. The weight update functions of α-tunable regularization loss with different α values are shown in Figure 3. σ is set to $\frac{1}{\xi+1}$ in the plot. It can be observed that the weight function becomes an update rule of exponential loss when α is 0.5 and of logistic loss when α is 0.

In GradientBoost framework, one can design a strong classifier by combining a set of weak classifiers and optimizes this problem using the gradient descent in weak classifier space $\mathcal{W}$. Let $p(\mathbf{x}) = G(\mathbf{x})$ and define $G(\mathbf{x})$ as

$$G(\mathbf{x}) = \sum_{t=1}^{T} g^{(t)}(\mathbf{x}) \tag{5}$$

With $\phi(v)$ as loss, the empirical risk on training set can be written as,

$$R_{emp}(G) = \frac{1}{N}\sum_{i=1}^{N}\phi(y_i G(\mathbf{x}_i)) \tag{6}$$

Using GradientBoost framework [47], the predictor can be updated at each iteration t,

$$G^{(t)}(\mathbf{x}) = G^{(t-1)}(\mathbf{x}) + g^{(t)}(\mathbf{x}) \tag{7}$$

[3]The element values may be negative (e.g. [-0.7092, -0.7050]) in some learned vector-form decorrelated filters.

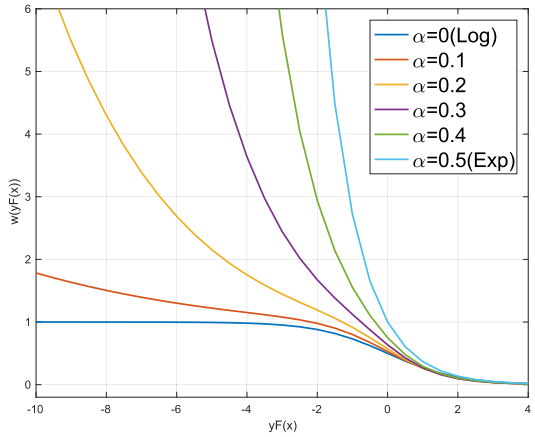


Fig. 3. Weight function of the $\alpha-$tunable regularization loss, for different values of α.

where $g^{(t)}(\mathbf{x})$ is the gradient of $R_{emp}(G)$ in weak classifier space $\mathcal{W}$ and can be expressed as

$$\begin{aligned} g^{(t)}(\mathbf{x}) &= \arg\max_{g} \sum_{i=1}^{N} -y_i \phi'(y_i G^{(t-1)}(\mathbf{x}_i)) g(\mathbf{x}_i) \\ &= \arg\max_{g} \sum_{i=1}^{N} y_i w^{(t)}(\mathbf{x}_i) g(\mathbf{x}_i) \end{aligned} \tag{8}$$

where

$$w^{(t)}(\mathbf{x}_i) = -\phi'(y_i G^{(t-1)}(\mathbf{x}_i)) \tag{9}$$

Substituting the derivative of loss function $\phi'(y_i G^{(t-1)}(\mathbf{x}_i))$ with Eq. A.A9, Eq. 9 can be rewritten as

$$w^{(t)}(\mathbf{x}_i) = -(1 - [f_\phi^*]^{-1}(y_i G^{(t-1)}(\mathbf{x}_i)))\beta'_\phi(y_i G^{(t-1)}(\mathbf{x}_i)) \tag{10}$$

Boosting with these weights updating rule is denoted Boosting with Loss Regularization (BoostLR). The weight updating rule of α-tunable regularization loss for boosting learning can be obtained by substituting $\phi'(y_i G^{(t-1)}(\mathbf{x}_i))$ using Eq. 4 as

$$w^{(t)}(\mathbf{x}_i) = (1 - \frac{e^{\frac{y_i G^{(t-1)}(\mathbf{x}_i)}{\sigma}}}{1 + e^{\frac{y_i G^{(t-1)}(\mathbf{x}_i)}{\sigma}}}) \frac{1-\alpha}{2-3\alpha} (e^{-\alpha \frac{y_i G^{(t-1)}(\mathbf{x}_i)}{\sigma}} + e^{\alpha \frac{y_i G^{(t-1)}(\mathbf{x}_i)}{\sigma}}) \tag{11}$$

2) Group Cost-Sensitive BoostLR: The weight update rule using Eq. 4 is cost-insensitive and does not incorporate prior knowledge of training samples as the weight is determined only by the margin $\upsilon = yG(\mathbf{x})$. Suitable prior knowledge can be exploited to train a better classifier. For example, the height of training sample is used as prior knowledge in [18] and [21] based on the assumption that samples with lower height are usually more difficult to recognize. However, the detection difficulty of samples can be influenced by many factors, e.g. the occlusion level. Taller training samples with large degree of occlusion may be more difficult to recognize than un-occluded samples with low height. As such, the cost-sensitive strategy in [18] and [21] may not be reliable. In the proposed approach, the non-pedestrian samples that are mined from different rounds of hard negative mining process show different detection difficulties. The detection difficulty of negatives can be explored to pay more attention to the harder negatives. Suppose there are K rounds in hard negative mining processing and S_k denote the set of negatives mined from k-th round. The group cost-sensitive loss can be expressed as,

$$L(\mathbf{x}, y) = \begin{cases} 0, & \text{if } H(\mathbf{x}) = y \\ C_{fn}, & \text{if } y = 1, H(\mathbf{x}) = -1 \\ C_{fp}^k, & \text{if } y = -1, H(\mathbf{x}_{S_k}) = 1, k = 1, \ldots, K \end{cases} \tag{12}$$

where C_{fn} is the cost for mis-classifying a pedestrian as non-pedestrian. C_{fp}^k is the cost for mis-classifying a non-pedestrian as pedestrian in the k-th round of hard negative mining. In object detection, it is essential to ensure that objects are not missed, as missing detections for objects are very difficult to recover. As a result, the cost for false negatives should be larger than false positives which means that $C_{fp}^k < C_{fn}$. We set $C_{fn} = 1$ and then search the optimal value for C_{fp}^k. The corresponding update rule for group cost-sensitive BoostLR can be written as,

$$\phi'(C_*\upsilon) = -C_*(1 - \frac{e^{\frac{C_*\upsilon}{\sigma}}}{1 + e^{\frac{C_*\upsilon}{\sigma}}}) \frac{1-\alpha}{2-3\alpha} (e^{-\alpha \frac{C_*\upsilon}{\sigma}} + e^{\alpha \frac{C_*\upsilon}{\sigma}}) \tag{13}$$

where C_* is the cost for different groups defined in Eq. 12. In contrast to cost-insensitive BoostLR weight update rule, Eq. 13 assigns larger weight on the sets with higher cost which imposes that the learning algorithm pays more attentions to harder training samples. The proposed two stage learning framework is presented in Algorithm 1.

IV. Results and Discussion

In this section, we evaluate the detection performance and execution time of the proposed method on well-known Caltech, INRIA and CityPersons pedestrian detection datasets. To ensure a fair comparison for the execution time, we implemented all the methods on the same platform, i.e., 3.5GHz Intel Xeon E5-1650 CPU with single thread execution. We have not relied on GPUs in our experiments.

A. Experiment Setup

1) Caltech Dataset: In Caltech dataset [48],[4] the training data is augmented by extracting one of every 3 frames instead of every 30 frames from the raw videos following the approach in [10]. We refer to this training set as Caltech10x training set. There are 42,782 images in the training set for hard negative mining. The Caltech test set consists of 4024 images which includes 1014 positive samples. The evaluation metric is log-average miss rate (MR) on False Positive Per Image (FPPI) in $[10^{-2}, 10^{-0}]$ under Reasonable setup (pedestrians that are at least 50 pixels tall and at least 65% visible [11]). In addition, we also tested our trained model on the new annotations of Caltech test set provided by [20], which has corrected some errors in the original annotations. We denote the results of the original and new annotations as MR and MR$_N$ respectively.

[4]www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/

Algorithm 1 Group Cost-Sensitive BoostLR for Pedestrian Detection

Input: Training set $(\mathbf{x}_i, y_i)_{i=1}^N$, a set of weak learner $\{g_i(\mathbf{x})\}_{i=1}^M$, number of training iteration T, and number of hard negative mining round K,

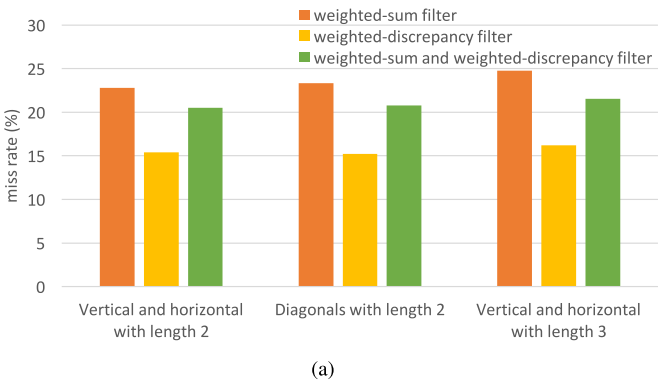
Output: Detector $H(\mathbf{x})$

Obtain group cost C_{fn} and $[C_{fp}^1, \ldots, C_{fp}^K]_{s_1}$ for the first stage
Set $G_1^{(0)}(\mathbf{x}) = 0$, $w_i^{(1)} = 1/N, i = 1, \ldots, N$;
for $t_1 = 1$ to T **do**
  Choose optimal weak learner $g_1^{(t_1)}(\mathbf{x})$ based on weight $w_i^{(t1)}$
  Update predictor $G_1^{(t_1)}(\mathbf{x}) = G_1^{(t_1-1)}(\mathbf{x}) + g_1^{(t_1)}(\mathbf{x})$;
  Update weights through Eq. 13;
end for
Obtain group cost C_{fn} and $[C_{fp}^1, \ldots, C_{fp}^K]_{s_2}$ for the second stage
Set $G_2^{(0)}(\mathbf{x}) = 0$, $w_i^{(1)} = 1/N, i = 1, \ldots, N$;
for $t_2 = 1$ to T **do**
  Choose optimal weak learner $g_2^{(t_2)}(\mathbf{x})$ based on weight $w_i^{(t2)}$
  Update predictor $G_2^{(t_2)}(\mathbf{x}) = G_2^{(t_2-1)}(\mathbf{x}) + g_2^{(t_2)}(\mathbf{x})$;
  Update weights through Eq. 13;
end for
return detector $H(\mathbf{x}) = sgn[G_1^{(t_1)}(\mathbf{x}) + G_2^{(t_2)}(\mathbf{x})]$

The model size used in the proposed method is 128×64. For each stage, three rounds of hard negative mining (64, 256, 1024, 5120 trees respectively) are used and 150,000 negatives are added to the training set in each round. During decision tree learning, we randomly selected 1/16 features from a large feature pool and the depth of the decision tree is limited to 6 in the first stage, and 8 in the second stage. The strides of both sliding window and aggregated channel shrinkage factor are 4, and each image is upsampled by one octave. The bin size is set to 1 when computing aggregated channels. The C_{fn} is set to 1 and an arithmetic sequence is employed ($[C_{fp}^1, \ldots, C_{fp}^K, 1]$) when searching optimal cost for negative subsets in each training stage from cross-validation experiments. For parameter σ and α, we set $\sigma = \frac{5}{\xi+1}$ and select optimal α from cross-validation experiments.

2) INRIA Dataset: In INRIA dataset [24],[5] there are 614 positive images and 1218 negative images in the training set. The trained model is evaluated on 288 testing images using MR on FPPI ranges of $[10^{-2}, 10^{-0}]$. For each stage, three rounds of hard negative mining (32, 128, 512, 4096 trees respectively) are used and the number of decision tree is limited to 512 in the second stage. 11,000 negatives are added in each hard negative mining round in both stages. Due to the small scale of training image in INRIA dataset, the max depth of decision tree is limited to 2 in the first stage and 3 in the second stage. The bin size is set to 4 when computing aggregated channels. Other settings are same with Caltech dataset.

[5]http://pascal.inrialpes.fr/data/human/



(a)

Fig. 4. The detection performance (MR) of different types of vector-form decorrelated filters with respect to weighted-sum filter, weighted-discrepancy filter and their combinations on Caltech dataset.

3) CityPersons Dataset: The CityPersons dataset [49] is built upon the Cityscapes dataset [50], in which the images are collected from multiple cities across Europe. In our experiments, we use the original training and validation split which comprises of 2975 and 500 images respectively. Following the evaluation setting in [49], the MR on FPPI ranging between $[10^{-2}, 10^{-0}]$ across different occlusion levels is used to evaluate the trained model, which includes four setups: Reasonable (pedestrian height ranges between $[50, +\infty]$ pixels and visibility locates in $[0.65, +\infty]$), Small (pedestrian height ranges between [50, 75] pixels and visibility locates in [0.65, $+\infty$]), Heavy (pedestrian height ranges between [50, $+\infty$] pixels and visibility locates in [0.2, 0.65]) and All (pedestrian height ranges between [20, $+\infty$] pixels and visibility locates in [0.2, $+\infty$]). The model size of the proposed method for CityPersons dataset is 128×64 and five rounds of hard negative mining (64, 256, 2014, 2048, 4096, 6144) are used. The maximum depth of decision tree are set to 8 in both stages. Other settings are the same with Caltech dataset.

B. Ablation Experiments

In this subsection, we conduct ablation experiments on Caltech dataset.

1) Effect of Weighted-Sum and Weighted-Discrepancy Vector-Form Decorrelated Filters: As indicated in Section III-A, the learned vector-form decorrelated filters can be partitioned into *weighted-sum* and *weighted-discrepancy* filters as depicted in Figure 2(a). In order to investigate the influence of these two kinds of filters on detection performance, we conduct experiments on Caltech dataset. The detection performance with different vector-form decorrelated filters is illustrated in Figure 4(a). It can be observed that the MR corresponding to weighted-sum filters is much higher than weighted-discrepancy filters. When the weighted-sum and weighted-discrepancy filters are used together, the MR is still much higher than the case of only using weighted-discrepancy filters. This implies that the weighted-sum filters have a negative impact on the detection performance. In order to further verify our assumption, we re-trained the LDCF detector based on the setting in [17]. We change the max depth of decision tree

TABLE I

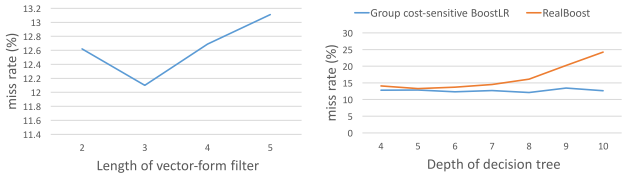
DETECTION PERFORMANCE (MR) WITH DIFFERENT STRATEGIES ON CALTECH DATASET. ✓INDICATES THAT CORRESPONDING STRATEGY IS ADOPTED

Group cost-sensitive BoostLR	Decorrelated vector-form filters	MR(%)
		29.80
✓		13.62
	✓	23.54
✓	✓	12.10

from 5 to 7 and conduct experiments with and without the weighted-sum decorrelated filters which correspond to the largest eigen-value. The MR without weighted-sum filter is 23.68% while the MR with weighted-sum filter is 25.92%. This further demonstrates that weighted-sum filters fail to capture discriminative information from local region and negatively impact the detection performance. Since the weighted-sum filters are widely used in existing FCF methods (e.g. Checkerboards [10] and RotatedFilters [20]), our finding also explains why existing FCF methods achieve inferior detection performance even though more complex filters are employed. Therefore, the weighted-sum filters are not employed in the proposed method.

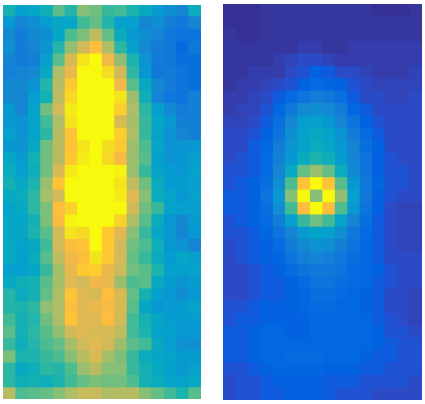
2) Effect of Proposed Group Cost-Sensitive BoostLR and Vector-Form Decorrelated Filters: In this work, we improved the detection performance by using novel decorrelated features and a new learning strategy. In order to evaluate the influence of each modification on the detection performance, we perform experiments on Caltech dataset. The experiment results are shown in Table I. The RealBoost and binary vector-form filters are employed as the baseline in these experiments. From the table, it can be observed that the MR is very high if the proposed learning strategy is not employed (i.e. when RealBoost is used instead of the proposed learning strategy). This is mostly due to the exponential loss of RealBoost which is too sensitive to outliers and has no parameter to control the regularization strength of learned classifier. One can also observe that much lower MR can be obtained if only the proposed group cost-sensitive BoostLR is employed. This demonstrates the advantage of the proposed learning strategy over RealBoost. Compared with binary filters used in [19], the proposed vector-form decorrelated filters achieve lower MR regardless of the adopted learning strategy. This implies that the proposed decorrelated filters can effectively explore the local region and extract better discriminative features. These observations show that both the proposed learning strategy and vector-form decorrelated filters contribute to significant improvement in the detection performance.

3) Effect of Length of Vector-Form Decorrelated Filters: The length of vector-form filter determines the scope of discriminative information extraction. In order to find the optimal length of vector-form filters for pedestrian detection, we conduct experiments to evaluate the detection performance when the length of vertical and horizontal vector-form filters are progressively increased. Please note that the diagonal vector-form filters with length of 2 are always employed when conducting experiments. For each type of vector-form decorrelated filter, only the decorrelated filter corresponding to second



(a) (b)

Fig. 5. a) The detection performance (MR) of varying length of vector-form decorrelated filters, and b) detection performance (MR) of varying max depth of decision tree with different learning strategies on Caltech dataset.



(a) (b)

Fig. 6. a) The spatial distribution of features selected by the second stage detector on Caltech dataset. Brighter pixels indicate that the features are selected in high frequency and blue pixels correspond to low frequency. b) Correlation map illustration in aggregated channel. The pixel of interest is surrounded by several brighter pixels. The brighter pixels mean higher correlation while the blue pixels represent lower correlation. Best viewed in color.

largest eigen-value is selected since it is weighted-discrepancy filter. The MR with varying length of vector-form decorrelated filters is shown in Figure 5(a). It can be observed that the lowest MR is achieved when the max length of vector-form filters is 3. In addition, the MR increases rapidly when the length of vector-form filters is larger than 3. This is mostly due to the fact that the decorrelated filters with length of 4 and 5 collect information from larger local region which has less correlation. In order to demonstrate this, we provide the following two maps: spatial distribution of feature selected by the second stage detector as shown in Figure 6(a), and correlation map illustration in aggregated channel from positive training samples as shown in Figure 6(b). The pixel of interest in Figure 6(b) is surrounded by several brighter pixels, especially in the vertical and horizontal orientations. One can observe that the features located in the middle region of pedestrian are selected in high frequency. The correlation map of feature from high frequency region shows that high correlation only exists for adjacent pixels. Specifically, vertical and horizontal adjacent pixels have higher correlation than diagonal ones. This is reasonable as the distance from diagonal adjacent pixels is larger than vertical and horizontal ones. These observations explain why higher MR is achieved when the length of vector-form decorrelated filters is larger than 3.

TABLE II

AVERAGE EXECUTION TIME PER FRAME (SECONDS) AND DETECTION PERFORMANCE (MR) OF FCF METHODS ON CALTECH DATASET. NOTE THAT THE EXECUTION TIME IS OBTAINED BASED ON MATLAB/C++ IMPLEMENTATION RUNNING ON A WORKSTATION WITH SINGLE THREAD

	Aggregated Channel	Filtering	Classification	Total Time (s)	MR(MR$_N$) (%)
ACF	0.045	-	0.061	0.106	29.76(26.42)
LDCF	0.046	0.220	0.035	0.301	24.80(22.02)
RotatedFilters	0.384	9.309	6.931	16.625	19.20(15.27)
Checkerboards	0.496	22.117	20.709	43.319	18.47(14.39)
MS-MFDF	0.193	0.135	0.163	0.491	14.63(11.15)
B-VFF	0.153	0.044	0.094	0.291	14.62(11.95)
Ours	0.238	0.081	0.385	0.704	12.10(9.25)
Ours_Acc	0.201	0.110	0.251	0.562	11.98(9.18)

4) Effect of Max Depth of Decision Tree With Group Cost-Sensitive BoostLR and RealBoost: The evaluation of the max depth of decision tree is often ignored in existing FCF based methods, where shallow decision trees are often employed. For example, max depth of decision tree is 5 and 4 in LDCF [17] and Checkerboards [10] respectively. In order to better understand the influence of the decision tree, we conduct experiments with varying max depth of decision tree using the proposed group cost-sensitive BoostLR and RealBoost respectively. Figure 5(b) shows the MR with different max depth of decision tree on Caltech dataset. It can be observed that MR with RealBoost increases rapidly when the depth is larger than 6, in which overfitting is observed. This implies that the RealBoost is not suitable with deeper decision tree due to the lack of regularization term that plays an important role in controlling the generalization of learned classifier. When the proposed learning strategy is employed, one can observe that the MR continues to decrease when the max depth is larger than 6. This is because the proposed learning strategy can control the generalization ability of the learned classifier by link function (the regularization strength) and binding function (the robustness to outliers) [23]. These experiment results demonstrate the superiority of our proposed group cost-sensitive BoostLR over RealBoost.

C. Comparisons With State-of-the-Art Methods on Caltech Dataset

In [19], two acceleration strategies: Selective Classification and Selective Scale Processing are proposed and have been demonstrated to notably improve the detection speed while maintaining robustness. We incorporated these two acceleration strategies into our proposed framework which are indicated as *Our_Acc* in Table II. With these acceleration strategies, the proposed method obtained further gain in detection performance (MR reduces from 12.10% to 11.98%) and speed (execution time reduces from 0.704 to 0.562 second per image). In the remaining paper, only results of the proposed method with acceleration strategies are reported.

Figure 7 provides the comparison of detection performance with state-of-the-art methods on Caltech benchmark. It can be observed that the proposed approach achieves the best detection performance among non-deep learning methods. Compared to existing FCF methods, the MR of our method is significantly lower, i.e., 11.98% whereas MR of Checkerboards [10], MS-MFDF [18] and B-VFF [19] are 18.47%, 14.63% and 14.62% respectively. Similar results can be observed when the evaluations are conducted on the new annotations of Caltech test set. Compared to the filters employed in Checkerboards and MS-MFDF, the proposed vector-form decorrelated filters are much simpler but achieve better detection performance. This demonstrates the effectiveness of the proposed vector-form decorrelated filters and the group cost-sensitive BoostLR learning strategy.

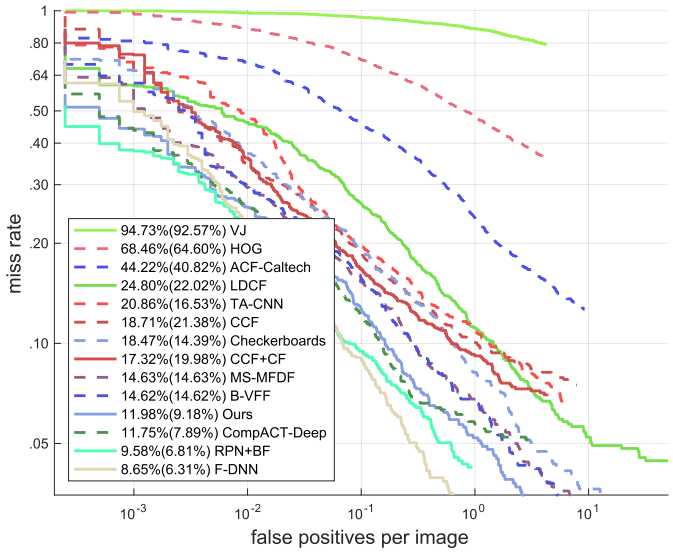


Fig. 7. Detection performance comparisons with state-of-the-art methods on Caltech dataset. The numbers indicate the value of MR (MR$_N$).

The run-time comparison of proposed method and state-of-the-art FCF methods are reported in Table II. It is noteworthy that we only conduct comparisons with the methods that have released their trained model so that we can run these methods on same platform. It can be observed that the ACF [14] achieves the lowest run-time since there is no filtering step in ACF. The run-time of proposed method is one order of magnitude lower than RotatedFilters [20] and Checkerboards [10], while achieving much better detection performance. Although MS-MFDF and B-VFF can run faster than the proposed method, the MR of these two methods are about 2.6% higher than ours. These results clearly demonstrate that the proposed methods can achieve better trade-off between detection performance and run-time than state-of-the-art pedestrian detection methods.

D. Comparisons With State-of-the-Art Methods on INRIA Dataset

The detection performance of proposed method and the state-of-the-art methods on INRIA dataset are shown

TABLE III

AVERAGE EXECUTION TIME PER IMAGE (SECONDS) AND DETECTION PERFORMANCE (MR) ON INRIA DATASET. NOTE THAT THE EXECUTION TIME IS OBTAINED BASED ON MATLAB/C++ IMPLEMENTATION RUNNING ON A WORKSTATION WITH SINGLE THREAD

	Aggregated Channel (s)	Filtering (s)	Classification (s)	Total Time (s)	MR (%)
ACF	0.024	-	0.012	0.036	17.28
LDCF	0.051	0.243	0.105	0.399	13.79
MS-MFDF	0.079	0.198	0.022	0.299	10.91
B-VFF	0.075	0.027	0.019	0.121	10.97
Ours	0.077	0.064	0.029	0.170	10.71

TABLE IV

AVERAGE EXECUTION TIME PER IMAGE (SECONDS) AND DETECTION PERFORMANCE (MR) ON CITYPERSONS DATASET. NOTE THAT THE EXECUTION TIME IS OBTAINED BASED ON MATLAB/C++ IMPLEMENTATION RUNNING ON A WORKSTATION WITH SINGLE THREAD

	Reasonable (%)	Small (%)	Heavy (%)	All (%)	Total Time (s)
ACF	46.05	54.98	75.03	68.02	9.04
LDCF	44.24	54.42	77.78	66.89	8.30
Checkerboards	47.56	61.86	74.94	67.85	270.54
Ours	23.25	47.93	63.56	49.94	39.33

in Figure 8. One can observe that our proposed method achieves the best detection performance (MR is 10.71%) among non-deep learning methods. Compared with ACF [14] and LDCF [17], the proposed method achieves 6.57% and 3.08% lower on MR respectively. The proposed method still achieves better performance than MS-MFDF [18] and B-VFF [19]. The average execution time of several FCF based methods on INRIA dataset are shown in Table III. It can be observed that the proposed method can run much faster than LDCF [17] and MS-MFDF [18]. The detection time of B-VFF is a little lower than proposed method due to the use of low complexity binary filters. These results further demonstrate the effectiveness and efficiency of the proposed detection framework.

E. Comparisons With FCF Based Methods on CityPersons Dataset

Since CityPersons is a new pedestrian detection dataset, there are no previously reported detection results on this dataset using FCF detection framework. In order to compare with existing FCF based methods, we conduct experiments using existing FCF based methods which includes ACF [14], LDCF [17] and Checkerboards [10]. We mostly used the default settings when training ACF, LDCF and Checkerboards detector When training Checkerboards on CityPersons dataset, we choose to set the maximum size of filter to 4×3, resulting in a total of 39 filters. The detection performance and running time of ACF, LDCF, Checkerboards and proposed method are shown in Table IV. It can be observed that the proposed method achieves much lower MR on all four evaluation setups. It can also be observed that Checkerboards obtains a slightly higher MR value than ACF and LDCF, which implies that utilizing a large amount of Checkerboard pattern filters does not contribute to achieving better detection performance. The detection time of Checkerboards is about 6.8 times more than our method as more filters are employed. These results demonstrate the general applicability of the proposed method.

F. Comparisons With Deep Learning Based Methods on Caltech and INRIA Datasets

For Caltech dataset, as shown in Figure 7, F-DNN (Fused-DNN) [30] which fuses several deep convolutional neural networks achieves the best detection performance. For INRIA dataset, the RPN+BF [11] that combines a region proposal

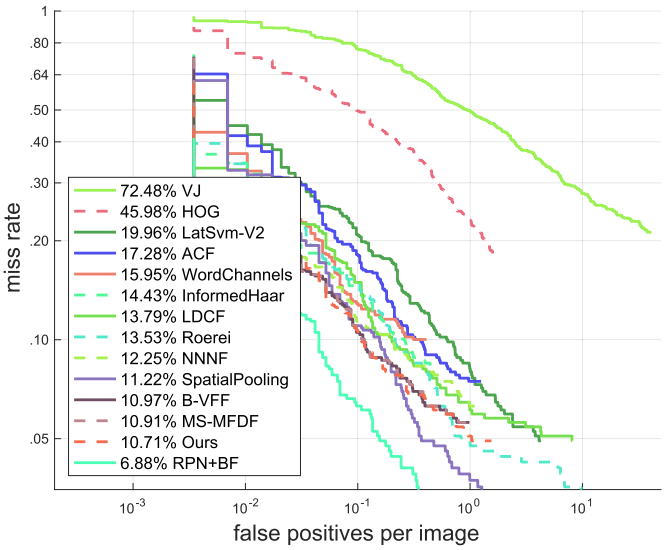


Fig. 8. Detection performance (MR) comparisons with state-of-the-art methods on INRIA dataset.

network and a boosted forest obtains the lowest MR as depicted in Figure 8. Although RPN+BF and F-DNN obtain better detection performance than the proposed approach, their performance heavily relies on very deep models (e.g. VGG [35]) and requires additional operations to refine the detection results from the detection framework of Faster R-CNN [33] (e.g. RPN+BF) and SSD [34] (e.g. F-DNN). In fact, the MR of using only RPN in RPN+BF and SSD in F-DNN are 14.90% and 13.06% respectively, which are inferior to our proposed method (i.e 11.98%). Due to the high computational complexity of deep convolution neural network, the deep learning based methods (i.e. VGG network) require high-end discrete GPUs (e.g. NVIDIA Titan X) to accelerate the detection process, which has been reported to be more than 60X faster than the dual-core Xeon CPU.[6] Even with the powerful GPU, the F-DNN is reported to run at only about 0.3 second per image on Caltech dataset which cannot meet the real-time requirements [30]. As highlighted in [37], the run-time of deep learning based object detection methods is more than 10X longer when executed on CPU.

In order to compare the computational complexity of the proposed method and deep learning based methods,

[6]https://github.com/jcjohnson/cnn-benchmarks

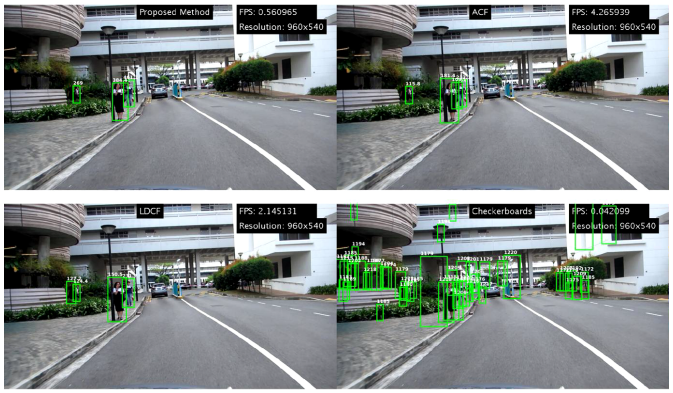


Fig. 9. Snapshot of pedestrian detection comparison video in real-world scenario. Full video is found here: https://youtu.be/otoK5hlOOFk.

we calculate the number of floating-point operations in both methods. It is worth noting that the floating-point comparisons in the decision tree are also calculated as they are the main operations in the classification step of the proposed method. The proposed method requires about 1.1 billion FLOP. The VGG-16 network is employed to calculate the FLOP of DCNN as it is the backbone network in many deep learning based methods (e.g. RPN+BF [11] and F-DNN [30]). The FLOP of VGG-16 network for input image resolution 224 $\times$ 224 is 15.3 billion [36]. For Caltech dataset, the image size is 480 $\times$ 640 and the estimated complexity of VGG-16 network on Caltech image is 93.7 billion FLOP. Hence, the complexity of deep learning based pedestrian detection method is at least 85X higher than the proposed method. It is worth mentioning that the computational complexity of deep learning based methods is only considered for the backbone network and does not take into account the complexity of other steps, e.g. proposal generation and classification in the detection framework of Faster R-CNN. Besides, this complexity also assumes that the input image is not upsampled although the deep learning based methods often increase the resolution of input image to obtain better detection performance [37]. For example, the input image is first upsampled by a factor of 1.5X prior to feeding it into region proposal network in RPN+BF [11].

YOLO [46] is a fast detection framework and can run at 40 FPS with powerful GPU (e.g. NVIDIA Titan X). However, the FLOP of YOLO v2 for input image resolution 608 $\times$ 608 is about 62.94 billion [7] which is about 57X higher than the proposed method. We run YOLO v2 in our workstation using single thread and it requires about 12 seconds to process a single image which will not meet the real-time requirements of most applications.

G. Detection results comparisons of FCF methods in real-world scenario

In order to compare the detection performance of existing FCF methods (e.g. ACF, LDCF and Checkerboards) in real-world scenario, we undertook a field trial in our university campus where the camera is mounted on a moving vehicle. Fig. 9 shows a snapshot of the video. The video resolution is 960 $\times$ 540 and the FPS (Frame Per Second) is obtained when executing the pedestrian detection algorithms on 3.5GHz Intel Xeon E51650 CPU with only a single thread. It can be observed from the video that the proposed method obtains tighter bounding boxes around pedestrians than ACF and LDCF, and can recognize the partially occluded pedestrians while ACF and LDCF fail to detect pedestrians in some cases. Although Checkerboards can obtain tight bounding boxes around pedestrians, it induces much more false alarms than the proposed method, ACF and LDCF, especially in cluttered background regions. The FPS is shown on the top right of each sub-screen. We can observe that the FPS of the proposed method is lower than ACF and LDCF, but is much higher than Checkerboards which highlights the high computational complexity of Checkerboards. In real-world scenarios, the occluded pedestrians are common. This implies that the ACF and LDCF are not suitable for real-world deployment. Checkerboards can recognize occluded pedestrian but its high computational complexity will limit its potential deployment on platforms with constrained computational resources (e.g. embedded systems employed in autonomous vehicle).

[7]https://pjreddie.com/darknet/yolo/

V. CONCLUSION

We proposed a robust and runtime-efficient two-stage pedestrian detection framework. The proposed method utilizes vector-form decorrelated filters to extract more discriminative features and a group cost-sensitive BoostLR learning algorithm to explore the variants of negatives mined from commonly-used hard negative mining strategy in order to improve the detection performance. The learned vector-form decorrelated filters are employed in the second detection stage to capture more details from local regions while at the same time benefiting from low computational complexity compared to the conventional matrix-form filters. The intrinsic variants in negatives are explored through employment of group cost-sensitive BoostLR learning algorithm. The experiment results show that the proposed method achieves best detection performance among non-deep learning methods on well-known Caltech, INRIA and CityPersons benchmarks and can run one order of magnitude faster than classical FCF methods (e.g. Checkerboards).

Even though the proposed method achieves better performance than existing FCF based methods, there are still opportunities to further improve its performance. In particular, the proposed method uses decorrelated filters to extract features which can be considered as local features due to the limitation of filter size. The local features cannot capture the relationship of body parts (e.g. head and foot) that are not adjacent, which may limit the potential of the proposed framework. However, the global features cannot be extracted by simply using larger decorrelated filters as information from larger distance has less correlation. In order to further improve the detection performance of the proposed method, we plan to explore methods for effectively extracting global features in our future work.

APPENDIX A
PROOF THAT MARGIN LOSSES ACT AS REGULARIZERS OF POSTERIOR CLASS PROBABILITIES

In [23], the predictor $p(\mathbf{x})$ can be expressed as a composition of two functions,

$$p(\mathbf{x}) = f(\eta(\mathbf{x})) \tag{A.A1}$$

where $\eta(\mathbf{x}) = P_{Y|\mathbf{X}}(1|\mathbf{x})$ is the posterior probability function and $f : [0, 1] \rightarrow \mathbb{R}$ is a link function that maps the posterior class probabilities $\eta(\mathbf{x})$ to classifier predictions $p(\mathbf{x})$ [22], [51], [52]. The problem of learning optimal predictor $p^*(\mathbf{x})$ can be decomposed into the sub-problem of learning optimal link function $f^*(\eta(\mathbf{x}))$ and estimating the posterior function $\eta(\mathbf{x})$. Since $f^*(\eta(\mathbf{x}))$ can usually be determined analytically, this can be estimated as $\eta(\mathbf{x})$, whenever $f^*(\eta(\mathbf{x}))$ is one-to-one mapping. $f^*_\phi(\eta)$ that respects the loss function $L_\phi(y, p(\mathbf{x})) = \phi(yp(\mathbf{x}))$ can be obtained by minimizing conditional risk $D_\phi(\eta, f)$ and can be written as,

$$\begin{aligned} f^*_\phi(\eta) &= \arg\min_f D_\phi(\eta, f) \\ D_\phi(\eta, f) &= \eta\phi(f_\phi(\eta)) + (1-\eta)\phi(-f_\phi(\eta)) \end{aligned} \tag{A.A2}$$

where we omit the dependence on $\mathbf{x}$ for notational simplicity. In the case of margin loss, the optimal link function $f^*(\eta)$ is usually unique and computable in closed-form by solving $\eta\phi'(f^*_\phi(\eta)) = (1-\eta)\phi'(-f^*_\phi(\eta))$ for f^*_ϕ [23]. When f^*_ϕ is invertible, the posterior probability can be recovered from

$$\eta(\mathbf{x}) = [f^*_\phi]^{-1}(p^*(\mathbf{x})) \tag{A.A3}$$

At this time, the loss $\phi(\upsilon)$ is said to be proper. And proper loss has the following structure as

$$\phi(\upsilon) = D^*_\phi([f^*_\phi]^{-1}(\upsilon)) + (1 - [f^*_\phi]^{-1}(\upsilon))[D^*_\phi]'([f^*_\phi]^{-1}(\upsilon)) \tag{A.A4}$$

where $\upsilon = yp(\mathbf{x})$ is the margin.

Suppose we have a predictor estimation as

$$\hat{p}^*(\mathbf{x}) = p^*(\mathbf{x}) + \epsilon_p(\mathbf{x}) \tag{A.A5}$$

where $\epsilon_p(\mathbf{x})$ is the prediction error. If $\epsilon_p(\mathbf{x})$ has small amplitude, then the estimation of probabilities $\hat{\eta}(\mathbf{x})$ can be approximated by its Taylor series expansion around p^* as

$$\hat{\eta}(\mathbf{x}) \approx [f^*_\phi]^{-1}(p^*(\mathbf{x})) + \epsilon_\eta(\mathbf{x}) \tag{A.A6}$$

with

$$\epsilon_\eta(\mathbf{x}) = \{[f^*_\phi]^{-1}\}'(p^*(\mathbf{x}))\epsilon_p(\mathbf{x}) \tag{A.A7}$$

If $|\{[f^*_\phi]^{-1}\}'(p^*(\mathbf{x}))| < 1$, the probability estimation error $\epsilon_\eta(\mathbf{x})$ has smaller magnitude than the prediction error $\epsilon_p(\mathbf{x})$. Hence, for equivalent prediction error $\epsilon_p(\mathbf{x})$, a loss function $\phi(\upsilon)$ with inverse link function $[f^*_\phi]^{-1}(p^*(\mathbf{x}))$ of smaller growth rate $|\{[f^*_\phi]^{-1}\}'(p^*(\mathbf{x}))| < 1$ produces more accurate posterior probability estimation. Since the optimal link f^*_ϕ can be directly computed in closed-form from the loss function $\phi(\upsilon)$ by solving the minimization problem in Eq. A.A2, the loss function $\phi(\upsilon)$ acts as a regularizer of posterior probability estimation when the growth rate of inverse link function $[f^*_\phi]^{-1}(p^*(\mathbf{x}))$ is smaller than one. Hence, the regularization strength of loss $\phi(\upsilon)$ is determined by the link function $f^*_\phi(\eta)$.

Let us define

$$\rho_\phi(\upsilon) = \frac{1}{|\{[f^*_\phi]^{-1}\}'(\upsilon)|} \tag{A.A8}$$

as the regularization strength of $\phi(\upsilon)$. If $\rho_\phi(\upsilon) > 1$, then $\phi(\upsilon)$ is denoted a regularization loss.

When taking derivatives respect to υ on both sides of Eq. A.A4, we can get the following equation

$$\phi'(\upsilon) = (1 - [f^*_\phi]^{-1}(\upsilon))\beta'_\phi(\upsilon) \tag{A.A9}$$

where

$$\beta_\phi(\upsilon) = [D^*_\phi]'([f^*_\phi]^{-1}(\upsilon)) \tag{A.A10}$$

is called binding function of $\phi(\upsilon)$ in [23]. It can be observed that the binding function $\beta_\phi(\upsilon)$ actually defines a one-to-one mapping between the link function f^*_ϕ and the derivative of the risk D^*_ϕ, which implies that the $\beta_\phi(\upsilon)$ "*binds*" link and risk. Under mild conditions, the binding function $\beta_\phi(\upsilon)$ is a monotonically decreasing odd function which determines the behavior of $\phi(\upsilon)$ away from the origin (e.g. large margin for outliers in training set). For example, in GradientBoost framework, the weight of training sample is determined by the derivative of loss function (as in Eq. A.A9). Therefore, the derivative of binding function $\beta'_\phi(\upsilon)$ is proportional to the weight of training sample. If $\beta'_\phi(\upsilon)$ produces extreme high magnitude for large margin (e.g. outliers), the boosting learning process will be dominated by the outliers and the learned model only works well for outliers. From the above analysis, it can be inferred that, the regularization strength is controlled by the link function $f^*_\phi(\eta)$ while the robustness to outlier is determined by the binding function $\beta_\phi(\upsilon)$.

APPENDIX B
DERIVATION OF α-TUNABLE REGULARIZATION LOSS IN BOOSTLR

The exponential loss (i.e. $\phi(\upsilon) = e^{-\upsilon}$) and logistic loss (i.e. $\phi(\upsilon) = \log(1 + e^{-\upsilon})$) are commonly-used loss functions in boosting family, and their link functions are invertible and have the form of $\frac{1}{2}\log\frac{\eta}{1-\eta}$ and $\log\frac{\eta}{1-\eta}$ respectively by solving Eq. A.A2. Accordingly, Eq. A.A9 which corresponds to exponential loss and logistic loss can be written as

$$\phi'(\upsilon) = -(1 - \frac{e^{2\upsilon}}{1 + e^{2\upsilon}})(e^{-\upsilon} + e^{\upsilon}) \tag{A.B1}$$

$$\phi'(\upsilon) = -(1 - \frac{e^{\upsilon}}{1 + e^{\upsilon}}) \tag{A.B2}$$

By introducing a parameter ξ, the Eq. A.B1 and Eq. A.B2 can be unified as

$$\phi'(\upsilon) = -(1 - \frac{e^{(\xi+1)\upsilon}}{1 + e^{(\xi+1)\upsilon}})\frac{1}{2-\xi}(e^{-\xi\upsilon} + e^{\xi\upsilon}), \quad \xi \in [0, 1] \tag{A.B3}$$

which interpolates between the derivative of exponential loss ($\xi = 1$) and logistic loss ($\xi = 0$). Hence, the derivative of

tunable regularization loss can be written as

$$\phi'_\sigma(v) = -(1 - \frac{e^{\frac{v}{\sigma}}}{1+e^{\frac{v}{\sigma}}})\frac{1-\alpha}{2-3\alpha}(e^{-\alpha\frac{v}{\sigma}} + e^{\alpha\frac{v}{\sigma}}), \quad \alpha \in [0, \frac{1}{2}] \tag{A.B4}$$

where $\sigma = \frac{\mu}{\xi+1}$ and $\alpha = \frac{\xi}{1+\xi}$, and $0 < \frac{1}{\mu} \leq 1$ is the shrinkage factor. As α varies, the $\phi'_\sigma(v)$ interpolates between the derivative of logistic loss and exponential loss. Hence, the loss $\phi_\sigma(v)$ is called α-tunable regularization loss in BoostLR [23].

According to Eq. A.A8 and Eq. A.A9, the regularization strength of loss $\phi_\sigma(v)$ in Eq. 4 can be written as:

$$\rho_{\phi_\sigma}(v) = \frac{1}{|\{[f^*_{\phi_\sigma}]^{-1}\}'(v)|} = \frac{1}{|\frac{\partial(\frac{e^{\frac{v}{\sigma}}}{1+e^{\frac{v}{\sigma}}})}{\partial v}|} = \frac{(1+e^{-\frac{v}{\sigma}})^2}{\frac{1}{\sigma}e^{-\frac{v}{\sigma}}} \tag{A.B5}$$

The following can be obtained from Eq. A.B5.

$$\begin{aligned}(1+e^{-\frac{v}{\sigma}})^2 - \frac{1}{\sigma}e^{-\frac{v}{\sigma}} &= 1 + 2e^{-\frac{v}{\sigma}} + e^{-2\frac{v}{\sigma}} - \frac{1}{\sigma}e^{-\frac{v}{\sigma}} \\ &= 1 + (2 - \frac{1}{\sigma})e^{-\frac{v}{\sigma}} + e^{-2\frac{v}{\sigma}}\end{aligned} \tag{A.B6}$$

Since $\xi \in [0, 1]$ and $0 < \frac{1}{\mu} \leq 1$, then $2 - \frac{1}{\sigma} = 2 - \frac{\xi+1}{\mu} \geq 0$, and Eq. A.B6 is larger than 0 for any v which implies that the regularization strength $\rho_{\phi_\sigma}(v)$ in Eq. A.B5 is larger than 1. Hence, the loss $\phi_\sigma(v)$ used in Eq. 4 is a regularization loss. The σ is called regularization gain in [23] as it controls the regularization strength $\rho_{\phi_\sigma}(v)$ by manipulation of the loss margin $v = yp(\mathbf{x})$. α determines the robustness to outliers (i.e. large margin) through controlling the magnitude of $\beta'_\phi(v)$ (e.g. $\beta'_{\phi_\sigma}(v) = -1$ when $\alpha = 0$ and $\beta'_{\phi_\sigma}(v) = -(e^{-\frac{v}{\sigma}} + e^{\frac{v}{\sigma}})$ when $\alpha = \frac{1}{2}$).

REFERENCES

[1] S. Zhang, C. Bauckhage, and A. B. Cremers, "Efficient pedestrian detection via rectangular features based on a statistical shape model," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 763–775, Apr. 2015.

[2] W. Liu, B. Yu, C. Duan, L. Chai, H. Yuan, and H. Zhao, "A pedestrian-detection method based on heterogeneous features and ensemble of multi-view–pose parts," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 813–824, Apr. 2015.

[3] X. Li *et al.*, "A unified framework for concurrent pedestrian and cyclist detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 2, pp. 269–281, Feb. 2017.

[4] J. Miseikis and P. V. K. Borges, "Joint human detection from static and mobile cameras," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 1018–1029, Apr. 2015.

[5] J. Baek, J. Kim, and E. Kim, "Fast and efficient pedestrian detection via the cascade implementation of an additive kernel support vector machine," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 902–916, Apr. 2017.

[6] M. You, Y. Zhang, C. Shen, and X. Zhang, "An extended filtered channel framework for pedestrian detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1640–1651, May 2018.

[7] Q. Li, H. Wang, Y. Yan, B. Li, and C. W. Chen, "Local co-occurrence selection via partial least squares for pedestrian detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 6, pp. 1549–1558, Jun. 2017.

[8] M. Pedersoli, J. Gonzàlez, X. Hu, and X. Roca, "Toward real-time pedestrian detection based on a deformable template model," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 355–364, Feb. 2014.

[9] R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Ten years of pedestrian detection, what have we learned?" in *Proc. ECCV*. Cham, Switzerland: Springer, 2014, pp. 613–627.

[10] S. Zhang, R. Benenson, and B. Schiele, "Filtered channel features for pedestrian detection," in *Proc. CVPR*, Jun. 2015, vol. 1, no. 2, p. 4.

[11] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?" in *Proc. ECCV*. Cham, Switzerland: Springer, 2016, pp. 443–457.

[12] A. D. Costea, R. V. Varga, and S. Nedevschi, "Fast boosting based detection using scale invariant multimodal multiresolution filtered features," in *Proc. CVPR*, Jun. 2017, pp. 993–1002.

[13] Q. Ye, J. Liang, and J. Jiao, "Pedestrian detection in video images via error correcting output code classification of manifold subclasses," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 193–202, Mar. 2012.

[14] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014.

[15] A. D. Costea, A. V. Vesa, and S. Nedevschi, "Fast pedestrian detection for mobile devices," in *Proc. ITSC*, Sep. 2015, pp. 2364–2369.

[16] A. D. Costea and S. Nedevschi, "Semantic channels for fast pedestrian detection," in *Proc. CVPR*, Jun. 2016, pp. 2360–2368.

[17] W. Nam, P. Dollár, and J. H. Han, "Local decorrelation for improved pedestrian detection," in *Proc. NIPS*, 2014, pp. 424–432.

[18] C. Zhou, M. Wu, and S.-K. Lam, "Group cost-sensitive boosting with multi-scale decorrelated filters for pedestrian detection," in *Proc. BMVC*, 2017, pp. 48.1–48.12.

[19] C. Zhou, M. Wu, and S.-K. Lam, "Fast and accurate pedestrian detection using dual-stage group cost-sensitive realboost with vector form filters," in *Proc. MM*, 2017, pp. 735–743.

[20] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "How far are we from solving pedestrian detection?" in *Proc. CVPR*, Jun. 2016, pp. 1259–1267.

[21] C. Zhu and Y. Peng, "Group cost-sensitive boosting for multi-resolution pedestrian detection," in *Proc. AAAI*, 2016, pp. 3676–3682.

[22] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors)," *Ann. Statist.*, vol. 28, no. 2, pp. 337–407, 2000.

[23] H. Masnadi-Shirazi and N. Vasconcelos, "A view of margin losses as regularizers of probability estimates," *J. Mach. Learn. Res.*, vol. 16, no. 1, pp. 2751–2795, 2015.

[24] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, vol. 1, no. 1, pp. 886–893.

[25] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *Proc. CVPR*, Jun. 2010, pp. 2241–2248.

[26] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[27] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *Proc. BMVC*, 2009, p. 91.

[28] J. Yan, Z. Lei, L. Wen, and S. Z. Li, "The fastest deformable part model for object detection," in *Proc. CVPR*, Jun. 2014, pp. 2497–2504.

[29] S. Zhang, C. Bauckhage, and A. B. Cremers, "Informed Haar-like features improve pedestrian detection," in *Proc. CVPR*, Jun. 2014, pp. 947–954.

[30] X. Du, M. El-Khamy, J. Lee, and L. Davis, "Fused DNN: A deep neural network fusion approach to fast and robust pedestrian detection," in *Proc. WACV*, Mar. 2017, pp. 953–961.

[31] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun, "Pedestrian detection with unsupervised multi-stage feature learning," in *Proc. CVPR*, Jun. 2013, pp. 3626–3633.

[32] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Convolutional channel features," in *Proc. CVPR*, Jun. 2015, pp. 82–90.

[33] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. NIPS*, 2015, pp. 91–99.

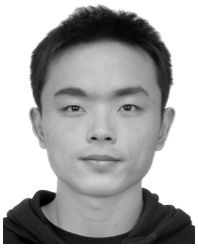
[34] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. ECCV*. New York, NY, USA: Springer, 2016, pp. 21–37.

[35] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, Jun. 2016, pp. 770–778.

[37] J. Huang *et al.*, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proc. CVPR*, Jun. 2017, pp. 7310–7311.

[38] S. Liu, J. Tang, Z. Zhang, and J.-L. Gaudiot, "CAAD: Computer architecture for autonomous driving," 2017, *arXiv:1702.01894*. [Online]. Available: https://arxiv.org/abs/1702.01894
[39] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," 2015, *arXiv:1510.00149*. [Online]. Available: https://arxiv.org/abs/1510.00149
[40] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Speeding up convolutional neural networks with low rank expansions," 2014, *arXiv:1405.3866*. [Online]. Available: https://arxiv.org/abs/1405.3866
[41] J. L. Holi and J.-N. Hwang, "Finite precision error analysis of neural network hardware implementations," *IEEE Trans. Comput.*, vol. 42, no. 3, pp. 281–290, Mar. 1993.
[42] M. Kim and P. Smaragdis, "Bitwise neural networks," 2016, *arXiv:1601.06071*. [Online]. Available: https://arxiv.org/abs/1601.06071
[43] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. CVPR*, Jun. 2015, pp. 1–9.
[44] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: https://arxiv.org/abs/1704.04861
[45] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. CVPR*, Jun. 2016, pp. 779–788.
[46] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. CVPR*, Jun. 2017, pp. 7263–7271.
[47] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, 2001.
[48] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, Apr. 2012.
[49] S. Zhang, R. Benenson, and B. Schiele, "CityPersons: A diverse dataset for pedestrian detection," in *Proc. CVPR*, Jun. 2017, pp. 4457–4465.
[50] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. CVPR*, Jun. 2016, pp. 3213–3223.
[51] H. Masnadi-Shirazi and N. Vasconcelos, "On the design of loss functions for classification: Theory, robustness to outliers, and savageboost," in *Proc. NIPS*, 2009, pp. 1049–1056.
[52] M. D. Reid and R. C. Williamson, "Composite binary losses," *J. Mach. Learn. Res.*, vol. 11, pp. 2387–2422, Sep. 2010.



Chengju Zhou received the M.S. degree from the School of Computer Science and Technology, Tianjin University, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His current research interest includes object detection for urban traffic scene understanding.

Meiqing Wu received the Ph.D. degree from the School of Computer Science and Engineering, Nanyang Technological University (NTU), Singapore, in 2016. She is currently a Research Fellow with the School of Computer Engineering, NTU. Her current research interests include stereo vision, motion analysis, object detection, and tracking for urban traffic scene understanding.

Siew-Kei Lam received the B.A.Sc., M.Eng., and Ph.D. degrees from Nanyang Technological University (NTU), Singapore. He is currently an Assistant Professor with the School of Computer Engineering (SCE), NTU. He has published more than 75 international refereed journals and conferences in design methodologies for heterogeneous and reconfigurable systems, embedded vision and autonomous systems, and high-speed computer arithmetic. His research investigates methods for realizing custom computing solutions in embedded systems.