Received December 11, 2018, accepted January 8, 2019, date of publication January 11, 2019, date of current version February 6, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2892425

Fast Pedestrian Detection in Surveillance Video Based on Soft Target Training of Shallow Random Forest

SANGJUN KIM[1], SOOYEONG KWAK[2], AND BYOUNG CHUL KO[1], (Member, IEEE)
[1]Department of Computer Engineering, Keimyung University, Daegu 42601, South Korea
[2]Department of Electronics and Control, Hanbat National University, Daejeon 34158, South Korea

Corresponding author: Byoung Chul Ko (niceko@kmu.ac.kr)

This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF), Ministry of Education, under Grant 2016R1D1A1A09916581, and in part by NRF through the Ministry of Science, ICT and Future Planning under Grant 2015R1C1A1A02037299.

ABSTRACT In recent years, deep learning algorithms have achieved top performances in object detection tasks. However, in real-time, systems having memory or computing limitations very wide and deep networks with numerous parameters constitute a major obstacle. In this paper, we propose a fast method for detecting pedestrians in surveillance systems having limited memory and processing units. Our proposed method applies a model compression technique based on a teacher–student framework to a random forest (RF) classifier instead of a wide and deep network because a compressed deep network still demands a large memory for a large amount of parameters and processing resources for multiplication. The first objective of the proposed compression method is to train a student shallow RF (S-RF), which can mimic the teacher RF's performance, by using a softened version of the teacher RF's output. Second, a deep network cannot easily detect small and closely located pedestrians in a surveillance video captured from a high perspective because of frequent convolutions and pooling processes. In this paper, adaptive image scaling and region of interest with S-RF were therefore combined to allow fast and accurate pedestrian detection in a low-specification surveillance system. In experiments, our proposed method achieved up to a 2.2 times faster speed and a 2.68 times higher compression rate than teacher RF and a better detection performance than several state-of-the-art methods on the Performance Evaluation of Tracking and Surveillance 2006, Town Centre, and Caltech benchmark datasets.

INDEX TERMS Pedestrian detection, model compression, teacher-student framework, random forest, shallower RF, surveillance video.

I. INTRODUCTION

Pedestrian detection is a fundamental task in computer vision applications, such as surveillance, advanced driver assistant systems (ADASs), robotics, entertainment and human-computer interfaces. Although it has been studied for many decades, accurate pedestrian detection remains an ongoing problem and presents potential challenges caused by different pedestrian postures, occlusion of pedestrians by objects or other pedestrians, non-rigid motion and variance in pedestrians' appearance caused by illumination changes. Among the various problems related to pedestrian detection in surveillance videos, the critical problems of occlusion and frequent pedestrian interactions in crowded scenes are the most challenging [1] and we focus on these problems in this paper.

In conventional pedestrian detection, the input images are densely up- and down-sampled according to predefined ratios to allow varying pedestrian sizes to be considered. Then, hand-crafted features are extracted from the candidate pedestrian regions in each size image using the scanning window method. Trained pedestrian detectors using a support vector machine (SVM) or AdaBoost classifier verify that candidate regions belong to the pedestrian or background class. Non-maximum suppression (NMS) is a post-processing algorithm that is responsible for merging all the detections that belong to the same object. Although conventional approaches require less computing power and memory than deep learning-based approaches, the feature extraction algorithms and the classifiers should be designed by a programmer and they cannot be jointly optimized to improve performance [2].

In contrast, deep learning-based pedestrian detection has recently exhibited state-of-the-art performances in pedestrian detection tasks. This approach performs end-to-end learning by significantly reducing the dependence of the detection on hand-crafted features and other preprocessing techniques. In particular, the convolutional neural network (CNN) has showed impressive accuracy as compared to conventional approaches because of its capability to learn discriminative features from raw pixels [3]. In CNN-based pedestrian detection, a kernel of size $n \times n$ is convolved with the input image in the convolution layers to produce a feature map. After the subsequent max-pooling layer, each feature map is also convolved with other kernels and the final feature maps are combined into a fixed-length feature vector that is then fed into the fully connected networks. The final softmax layer outputs classification scores over two classes, pedestrian and background. Although the detection accuracy of the deep learning-based approach is known to be better than that of conventional approaches, a few issues remain to be resolved to allow efficient pedestrian detection. Top performing systems usually involve very wide and deep networks with numerous parameters. However, a large-scale dataset and massive computing power are required for training, since these systems need to perform a very large number of multiplications. In addition, the very considerable number of parameters requires a large memory, and considerable skill and experience is also required for selecting suitable hyper parameters. These are the main reasons why wide and deep top performing networks are not well suited for applications with memory or time limitations [4]. The convolution and pooling layers of the CNN structure generate high-level sematic activation maps, which is one cause of the blurred boundaries between closely positioned pedestrians. As a result, CNN-based detectors are more likely to fail to locate each individual than conventional approaches as a result of inaccurate localization.

Surveillance videos tend to include a variety of perspective views, because the cameras are usually installed at an elevated location. Therefore, CNN-based detectors are not appropriate for detecting pedestrians in low resolution video when the altitude of the surveillance camera is high and the video thus includes various small-sized pedestrians. Another problem related to using a CNN-based detector is that it requires a large number of datasets for training and testing, but it is not easy to collect a large amount of training data for the surveillance camera under sufficiently different conditions to train the CNN. Moreover, to process multiple channel videos simultaneously, a CNN-based detector requires a high-level and massive computing device as compared to conventional detectors.

Therefore, in this study, we focused on developing a new fast pedestrian detection algorithm for surveillance cameras that can be run in a low-level computing device by applying the teacher-student framework to the conventional random forest (RF).

The remainder of this paper is organized as follows. In Section II, we describe pedestrian detection in videos captured by an elevated surveillance camera and the major contributions of this paper. In Section III, we introduce pedestrian detection using shallow RF (S-RF) based on a teacher-student learning framework. In Section IV, we present experiments demonstrating the accuracy and applicability of our proposed pedestrian detection method. Finally, our conclusions and scope for future work are presented in Section V.

II. RELATED WORKS

Because this paper presents a study on pedestrian detection in videos captured by an elevated surveillance camera, we introduce related research on various approaches for detecting pedestrians in surveillance camera videos.

Histograms of oriented gradient (HOG) [5] is the most widely used feature descriptor for pedestrian detection. Although a dense overlapping HOG grid provides good pedestrian detection results with a lower false positive rate than traditional Haar-like descriptors, it is also produces false positives when the pedestrian is similar in color and/or pattern to the background or misses pedestrians positioned in a crowd, as well as having a heavy computation demand [6].

To solve the missing pedestrian and false positive problems related to global feature descriptors such as HOG and local binary patterns (LBP) [7], the deformable part model (DPM) [8] was proposed for pedestrian detection based on mixtures of multiscale deformable parts and a latent SVM. The DPM is characterized by a coarse root filter that approximately covers an entire object and higher resolution part filters that cover smaller parts of the object. However, the DPM still cannot easily detect partially occluded pedestrians in surveillance videos, because it considers the score of the occluded parts in the final decision score. To solve this problem, Dehghan *et al.* [9] inferred occlusion information from the score of the parts and utilized only those parts having high confidence in their emergence by finding the most reliable set of parts that maximizes the probability of detection.

The performance of conventional approaches is in general limited by the representation power of the low-level hand-crafted features [1]. Therefore, CNN-based pedestrian detectors for surveillance systems have been attracting attention. Ouyang and Wang [10] proposed a deep model that jointly learns four components for pedestrian detection in a surveillance camera video: feature extraction, deformation handling, occlusion handling and classification. In this unified deep model, three components interact with each other in the learning process and each component is allowed to maximize its strength when cooperating with others. Chen *et al.* [1] converted the task of pedestrian detection into head-shoulder part detection to detect severely occluded pedestrians in surveillance videos. In their paper, they proposed a three-stage CNN cascade to capture the most discriminative information of the head-shoulder parts of pedestrians. Zhao *et al.* [3] used the

Edge Boxes algorithm [11] to obtain low-redundancy and a high quality of candidate windows with Fast R-CNN [12] architecture, which can extract thousands of region proposals and classify pedestrians at those locations based on a CNN. To reduce the run time of the region proposal of R-CNN, Faster R-CNN [12] was proposed, in which a region proposal network (RPN) that shares full-image convolutional features with the detection network was introduced. However, Faster R-CNN [12], as well as other CNN-based approaches, are still not appropriate for real-time pedestrian detection in surveillance systems. To reduce the processing time and improve the detection performance, you only look once (YOLO) [13] and YOLO 9000 [14] were proposed. These methods use a single neural network to predict the bounding boxes and class probabilities directly from full images in a single evaluation.

In recent years, research on small deep neural network architectures has been actively conducted to detect objects in embedded devices. For example, SqueezeNet [15], MobileNets [16], ShuffleNet [17], and TinySSD [18] are designed specifically to minimize model retaining object detection performance. Although tiny CNN architectures for object detection have shown a good performance, several problems related to them still remain to be resolved as mentioned in the Introduction. For example, in the case of TinySSD [18], the size of the network is greatly reduced through optimization, which is 26 times smaller than the Tiny YOLO [14] (60.5MB). However, the size of the model still exceeds 2.3MB and requires 571.09 million operations. Therefore, these limitations make it difficult to implement applications in real-time systems and constitute an obstacle to operating multiple channel videos simultaneously.

In addition to the feature extraction and classification algorithms related to pedestrian detection, variation in the camera's perspective affects the accuracy of pedestrian detection because the range of the image scaling level and multi-scale scanning can vary according to the camera's altitude and these two factors are closely related to the detection performance in terms of accuracy and run-time speed. To handle pedestrian detection in videos of surveillance cameras having different altitudes, Bae *et al.* [19] proposed scale of interest (SOI) and region of interest (ROI) estimation to minimize unnecessary computations in practical multiscale pedestrian detection. The role of the SOI is to determine the image-scaling level by estimating the perspective of the image and that of the ROI is to search the area of a scaled image. Ko *et al.* [6] proposed Hough windows maps (HWMs) for determining the levels of image scaling with a divide-and-conquer algorithm to reduce the computational complexity involved in processing surveillance video sequences. Moreover, an adaptive ROI for image scaling helps improve the detection accuracy and reduce the detection time.

Hattori *et al.* [20] proposed a spatially varying pedestrian appearance CNN model that takes into account the perspective geometry of the scene, because when a new surveillance system is installed in a new location, a scene-specific pedestrian detector must first be trained. To compensate for insufficient data resulting from frequently changing camera positions, this method used geometric scene data and a customizable database of virtual simulations of pedestrian motion instead of changing the ROI or image scaling level. Cai *et al.* [21] proposed a multi-scale CNN for a fast multi-scale pedestrian detection algorithm consisting of receptive fields of different scales and a scale-specific detector to produce a strong multi-scale pedestrian detector. Jiang *et al.* [22] proposed pedestrian detection based on sharing features across a group of CNNs that correspond to pedestrian models of different sizes. This method detects pedestrians of several different scales in a single layer of an image pyramid by sharing features in order to reduce the computational burden incurred by extracting features from an image pyramid.

A. CONTRIBUTIONS OF THIS WORK

To design a fast pedestrian detection scheme that is well suited for surveillance systems having a limited memory and processing unit, we introduce an algorithm for compressing deep and wide classification architectures into shallower ones. In this study, the proposed compression algorithm was applied to an RF classifier, which is an ensemble of decision trees, instead of to a CNN, because a CNN still demands a large amount of memory and processing resources, even when the depth of the layers is decreased by the proposed algorithm.

The major contributions of this paper are as follows.

- We describe the adoption of HWMs for determining the levels of image scaling and an adaptive ROI algorithm to reduce the amount of image scaling and sliding windows in surveillance camera videos.
- We explore new types of model compression algorithm that are realized by transferring the teacher-student framework to an RF model instead of using computationally heavy deep learning.
- We propose a model compression in which a teacher-student compression framework is applied to RF to allow training of a student shallow RF (S-RF), which is shallower than the teacher RF, using a softened version of the teacher's output.
- We prove that S-RF trained by soft target training is a reasonable method for mimicking the classification ability of a teacher classifier. In addition, it also an efficient method for detecting small-sized and closely positioned pedestrians in a high perspective surveillance video.
- We prove that the proposed S-RF efficiently and considerably decreases the processing time without sacrificing accuracy.

We describe the successful application of the proposed method to a benchmark dataset, and we confirm that its detection accuracy is similar to or higher than that of other CNN-based related methods with a shorter processing time.

III. TEACHER–STUDENT MODEL COMPRESSION

A. ESTIMATION OF IMAGE-SCALING LEVEL AND ADAPTIVE REGION OF INTEREST

The amount of image scaling and the number of search regions is a significant burden in pedestrian detection, because a multi-scale image pyramid requires frequent image scaling and the sliding windows should be applied at each scale for feature matching.

To reduce the amount of image scaling and sliding windows required to process surveillance camera videos, we adopted HWMs for determining the levels of image scaling and an adaptive ROI algorithm [6] for providing a different search ROI for each image scale. The image scaling level and corresponding ROI are changeable according to the perspective angle of the surveillance camera. As the feature, we use an oriented center-symmetric local binary pattern (OCS-LBP) [23], because it supports the gradient magnitude and pixel orientation simultaneously. To create a robust feature model for pedestrian occlusion, we compute the OCS-LBP from 4 × 4 adjacent sub-blocks and produce a single OCS-LBP descriptor with 128 dimensions by concatenating 8 types of local OCS-LBP descriptors from 16 sub-blocks.

As the pedestrian classification algorithm, we introduce an S-RF classifier trained using the proposed teacher-student training framework to separate candidate windows into pedestrian and non-pedestrian classes. The S-RF training procedure is described in detail in Section 3-C.

B. REVIEW OF TEACHER-STUDENT FRAMEWORKS

Although the performance of deep neural networks improves as the layers become deeper, they suffer from the disadvantage of increasing memory requirements for millions of parameters and computational complexity for millions of multiplications of filters. For these reasons, as mentioned above, a high-performance wide and deep network is not suitable for memory and time constrained applications [4], [24]. To reduce the memory required for numerous parameters and the computational burden at the inference time, several model compression frameworks have been proposed, such as parameter pruning and sharing [25], low-rank factorization [26], transferred/compact convolutional filters [27] and the teacher-student framework [4], [28]–[30]. It is known that among these four categories, the performance of the teacher-student framework matches or is superior to that of the teacher's framework and requires considerably fewer parameters and multiplications [24].

The teacher-student framework constructs a deep and wide teacher network having a high performance based on a large amount of training data and deep layers, and constructs a shallower student network with an equal performance based on the teacher network [4], [28]–[30]. As shown in Fig. 1, the student network is generated by using the probability values extracted by the softmax of the teacher network in the learning process instead of the class labels of the training data.

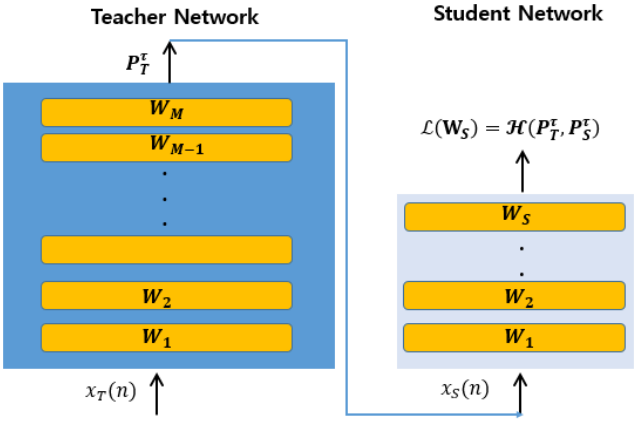


FIGURE 1. Teacher-student learning framework for compressing a teacher network into a student network. The softened outputs of the teacher network are used for training the target student network using other unlabeled training data by comparing the loss function ($\mathcal{L}$) and cross-entropy ($\mathcal{H}$) of the output of the teacher (P_T^τ) and the student (P_S^τ). (a) Teacher network having deep layers, (b) student network generated by using the probability values extracted from the teacher network.

The correlation between classes can be considered by using probability values (soft targets) instead of class labels (hard targets) for training data. Student networks that use non-hard target (soft targets) train the student network by using cross-entropy to reduce the difference in the output of the teacher and the student network.

However, as mentioned in the Introduction, a compressed CNN model still demands a large memory for very considerable amounts of parameters and processing resources for multiplication. For example, the representative teacher-student framework, FitNet [4], still requires 2.5 million parameters and 382 million multiplications, although the teacher network is reduced at a compression ratio of 3:6. Therefore, CNN-based deep top performing networks are not well suited for applications with memory or time limitations, even if model compression algorithms are applied.

In this study, we explored new types of model compression algorithms achieved by transferring the CNN-based teacher-student framework to the RF model, that is, an ensemble of decision trees. An RF is a decision tree ensemble classifier, where each tree is grown using a certain type of randomization. RFs have the capacity to process very large amounts of data with high training speeds, based on a decision tree. Moreover, this classifier has been shown to be effective in a large variety of high-dimensional problems, with a high computational performance and accuracy as compared to conventional SVM or AdaBoost classifiers. The structure of each tree in the RF is binary and is created in a top-down manner [31], [32]. An RF has a structure to which a CNN-based teacher-student framework can easily be applied because it can reduce the size of the forest by pruning the number of decision trees.

C. TRAINING OF SHALLOW RANDOM FOREST

In this study, we applied the teacher-student compression framework to RF to allow training of a student S-RF that is shallower than the teacher RF, using the softened version of the teacher's output.

Hinton *et al.* [28] trained a student network using real class labels and the softened output of an ensemble of a teacher network. The student network was trained to optimize the loss function ($\mathcal{L}$) based on two cross-entropies ($\mathcal{H}$).

$$\mathcal{L}\left(\mathbf{W}_S\right) = \mathcal{H}\left(y_{true}, P_S\right) + \lambda\mathcal{H}\left(P_T^{\tau}, P_S^{\tau}\right) \tag{1}$$

where the first term $\mathcal{H}$ means a cross-entropy between the real class labels $\mathbf{y}_{true}$ and the output of student networks $\boldsymbol{P_S}$. The second term means a cross-entropy between the softened output of the student network $\boldsymbol{P_S^{\tau}}$ and the teacher $\boldsymbol{P_T^{\tau}}$. However, it needs a tunable parameter $\boldsymbol{\lambda}$ to balance both cross entropies. Moreover, significant effort is required to label $\boldsymbol{y_{true}}$ for training data. However, unlabeled data help the student networks learn to approximate better the outputs of the fully trained teacher networks [29].

According to the experimental results presented in [29], soft target data (a class probability vector) are able to capture more information than the original hard target (0/1 labels) data by retaining the class relationship between the different classes and the input that has been internalized by the teacher. Moreover, unlabeled data help the student networks to learn to approximate better the outputs of the fully trained teacher networks. Therefore, in this study a new dataset B* consisting of soft target data for training a student RF was constructed instead of using the same dataset as that used for training the teacher RF. The soft target dataset B* of each sample is obtained from the pre-trained teacher RF.

First, the training pedestrian dataset is divided into dataset A for learning the teacher and a larger dataset B for learning the student. For training the teacher RF model, a training set A is given as a base for training component

$$\mathrm{A} = \{(\mathbf{x_i}, y_i)\,|i = 1, 2, \ldots N\}$$

where $\mathbf{x_i} = (x_{i1}, x_{i2}, \ldots, x_{iM})$ is an input vector with M (128) dimensions and $y_i = \{g_1, g_2, \ldots, g_C\}$ is a scalar (C is the number of classes and it has two classes), representing the class marked by the expert for $\mathbf{x_i}$. Dataset A labeled with a scalar 1 (pedestrian)/0 (non-pedestrian) is called a 'hard target'. The teacher RF is then trained to construct an ensemble of decision trees that minimizes the classification error using a labeled training set A.

In the training procedure of the teacher RF, the RF first chooses a random subset A′ from the training dataset, A. At node O, the sample A'_O is iteratively split into left and right subsets, A'_l and A'_r, by using the threshold, t, and split function, $f(v_i)$, for the feature vector, v. Then, several candidates are randomly created by the split function and threshold at the split node. From among these, the candidate that maximizes the information gain about the corresponding node is selected. The information gain, ΔE, is easily calculated by entropy estimation, according to

$$\Delta E = E\left(\mathrm{A}'_O\right) - \frac{\left|\mathrm{A}'_l\right|}{\left|\mathrm{A}'_O\right|}E\left(\mathrm{A}'_l\right) - \frac{\left|\mathrm{A}'_r\right|}{\left|\mathrm{A}'_O\right|}E(\mathrm{A}'_r) \tag{2}$$

where $E(\cdot)$ is the Shannon entropy of the classes in the set of training samples A′.

After the decision tree has been trained, the leaf nodes store statistical information containing the probability of each class that reached node $S, p_t(c_i|o), i \in 1, \ldots, C$. A random forest $\mathcal{T}$ then consists of a set of T trees and each tree $\mathcal{T}_t, t \in \{1, \ldots, T$ is trained on a randomly sampled subset of the training data A. The final class distribution of a sample $\mathbf{x}$ is generated by the ensemble (arithmetic averaging) of each distribution of all trees T:

$$p\left(c_i\,|\,\mathbf{x}\right) = \frac{1}{T}\sum\nolimits_{t=1}^{T} p_t(c_i|\mathbf{x}) \tag{3}$$

Training dataset B is then input to the teacher RF, which is trained using the corresponding hard targets. Unlike those of training dataset A, the input samples of dataset B have the same labels, consisting of a vector of class posterior probability summing to 1, expressed as

$$\mathbf{B} = \{(\mathbf{x_i}, \mathbf{p_i})\,|i = 1, 2, \ldots N\}$$

where $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iN})$ is the feature vector and $\mathbf{p}_i = (p_{i1}, p_{i2}, \ldots, p_{iC})$ is the class probability vector. p_{ij} is the probability for class j for sample i and the initial class probability vector $\mathbf{p}_i$ of the i-th sample must have the same class distribution as 1/C.

To obtain $\mathbf{p}_i$ in the original dataset B, each sample is applied to the teacher RF to calculate the class probability vector according to the results of Eq. (3) and relabel the original dataset B. After all the samples included in dataset B have been trained, a new dataset $\mathbf{B}^*$ is constructed:

$$\mathbf{B}^* = \left\{\left(\mathbf{x_i^*}, \mathbf{p_i^*}\right)|i = 1, 2, \ldots N^*\right\}$$

The new dataset $\mathbf{B}^*$ is transcribed with a class probability $\mathbf{p_i^*}$ that is called as 'soft target' as opposed to a hard target [29].

For training the decision tree of the student RF, the entropy estimation for evaluating the split function is calculated using output class distribution $\mathbf{p_i^*}$ estimated from the teacher RF, instead of Shannon entropy. Let us assume $\mathbf{B}'$ is the randomly selected subset of $\mathbf{B}^*$ and B'_O represents the samples at node O. The samples B'_O are iteratively split into left and right subsets B'_l and B'_r by using the entropy estimation:

$$\mathrm{E}\left(\mathrm{B}'_O\right) = -\frac{1}{N}\sum\nolimits_{i=1}^{N}\sum\nolimits_{j=1}^{C} p_{ij}^*\log(p_{ij}^*) \tag{4}$$

where p_{ij}^* is the j-th class distribution of sample i at node O and N is the total number of data classified into node O. Then, the information gain ΔE of node O is calculated by Eq. (2).

After decision tree Tr_t has been expanded, the final cross entropy is estimated to decide whether or not the candidate decision tree is selected to be a member of the student S-RF. The general form of the final cross entropy is

$$\mathrm{Tr}\left(\mathrm{T}, \mathrm{S}\right)_t = -\sum\nolimits_{i=1}^{N}\sum\nolimits_{j=1}^{C} p_{ij}^*(T)\log(p_{ij}^*(S_t)) \tag{5}$$

where $p_{ij}^*(T)$ represents the j-th class distribution of sample i of the B^* dataset transcribed by the teacher RF, and $p_{ij}^*(S_t)$

Algorithm 1 Procedure of Training Student Shallow Random Forest

```
Input:
B* : soft target dataset transcribed with a class probability
by output of teacher RF
D : maximum depth of a tree[1]
T': desired number of target trees of S-RF
N, C: number of samples of B* and class
S-RF: set of student S-RF
n(S-RF): number of trees in S-RF

While n(S-RF) ≤ T do
  Select subset B' from training set B*
  Grow an unpruned tree using the B' samples
  Step 1: Tree growing with D depth
  For d=1 to D do
    Each internal node randomly selects p variables and
    determines the best split function using only these
    variables
  Loop: Using different p-th variables, the split function
  $f(v_p)$ iteratively splits the training data into left ($B'_l$) and
  right ($B'_r$) subsets at node $O$.

      $B'_l = \{x \in B' | f(v_p) < t,$
      $B'_r = \left\{x \in B^{'} | f(v_p) \geq t\right\}.$

  The threshold $t$ is randomly chosen by the split function
  $f(v_p)$ in the range $t \in (\min f(v_p), \max f(v_p))$.
  Compute entropy $E\left(B^{'}_{O}\right)$of a function $f(v_p)$ using
  Eq. (4)
  Calculate information gain $\Delta E$ of node $O$ using Eq. (2)
  If ($\Delta E = \max$), then determine the best split function
  $f(v_p)$ for node $d$
  Else go to loop.
  Store the tree structure $Tr_t$ and probability distribution
  to leaf node.
  End of For
Step 2: Tree evaluation
  Compute the class probability, $p^*_{ij}$, on all B* data using
  the trained $t$-th decision tree $Tr_t$.
  The cross-entropy $Tr(T, S)_t$ about $Tr_t$ is estimated
  using Eq. (5)
  Minimization criteria of cross-entropy.
   IF $Tr(T, S)_t < \tau$,
    Then $S - RF \ni Tr_t$;
    Else remove $Tr_t$.
End While // End of student random forest growing
Step 3: Output of student S-RF
 The S-RF consists of $T'(T' <= T)$ soft target trained
decision trees and class probability $p^*_{ij}$ for each leaf node.
```

represents the j-th class distribution of sample i by the constructed decision tree t.

Algorithm 1 details the student RF training process using a soft target dataset.

Threshold τ for the minimization criterion of cross-entropy is 0.39 and detail experiment is described in Section IV-F.

Figure 2 shows the overall S-RF training procedure based on soft-target training data B$*$. After the t-th tree Tr_t is grown using information gain criteria and a random subset of B$*$ (Fig. 2(a)), tree evaluation is performed using the cross-entropy estimation of tree Tr_t to decide whether or not the candidate decision tree is selected as a member of student RF (Fig. 2(b)). The final S-RF consists of T' optimal trees selected after T iterations.

To train the teacher RF, we collected 4,250 images from Caltech dataset images [33] and YouTube consisting of 5,502 positive and 7,566 negative pedestrian samples as dataset A. In this study, we set the maximum size (number) of the teacher RF at 300 trees, because the accuracy no longer improves as the tree number of trees increases over 300. Then, dataset B was also generated from 1,700 images consisting of 2,200[1] positive and 3,000 negative samples using the rest of dataset. After the teacher RF was constructed using dataset A, dataset B was applied to the teacher RF and produced soft target training data B$*$.

IV. EXPEIMENTAL RESULTS

From among many datasets for evaluating pedestrian detection in video sequences, we chose two, Town Centre [34] and Performance Evaluation of Tracking and Surveillance (PETS) 2006 [35], because these two datasets were originally designed for evaluating pedestrian detection in videos captured by elevated surveillance cameras, which was the focus of this study. The resolution of Town Centre is high, 1920×1080 pixels, and its image capture rate is 25 frames per second (fps). It supports a ground truth consisting of 71,500 hand labeled head locations, with an average of 16 people visible at one time. The PET 2006 dataset includes multi-view camera sequences containing left-luggage scenarios at a train station in which the scene complexity increases. To evaluate the pedestrian detection performance, we used only a single viewpoint so that the evaluation would be performed under the same conditions as that for the comparison algorithms [36].

In addition, we compared performance with state-of-the-art researches for Caltech dataset to measure pedestrian detection performance in low-angle moving cameras.

In this study, we fixed the model size of the pedestrian at 63×27 pixels for the Town Centre, PETS 2006, and Caltech datasets. The image-scaling level was set at five by considering the camera perspective angle of the dataset: up-scaling to detect a pedestrian smaller than the model at ratios of 1:1.26, 1:1.03 and down-scaling to detect a pedestrian larger than the model at ratios of 1:0.77, 1:0.66 and 1:0.52, according to the results of the HWMs.

To evaluate the performance of the pedestrian detection, we measured the precision and recall, which are in general used to evaluate the performance of pedestrian detection.

[1] We set maximum tree depth to 20 using the same method found in [6].

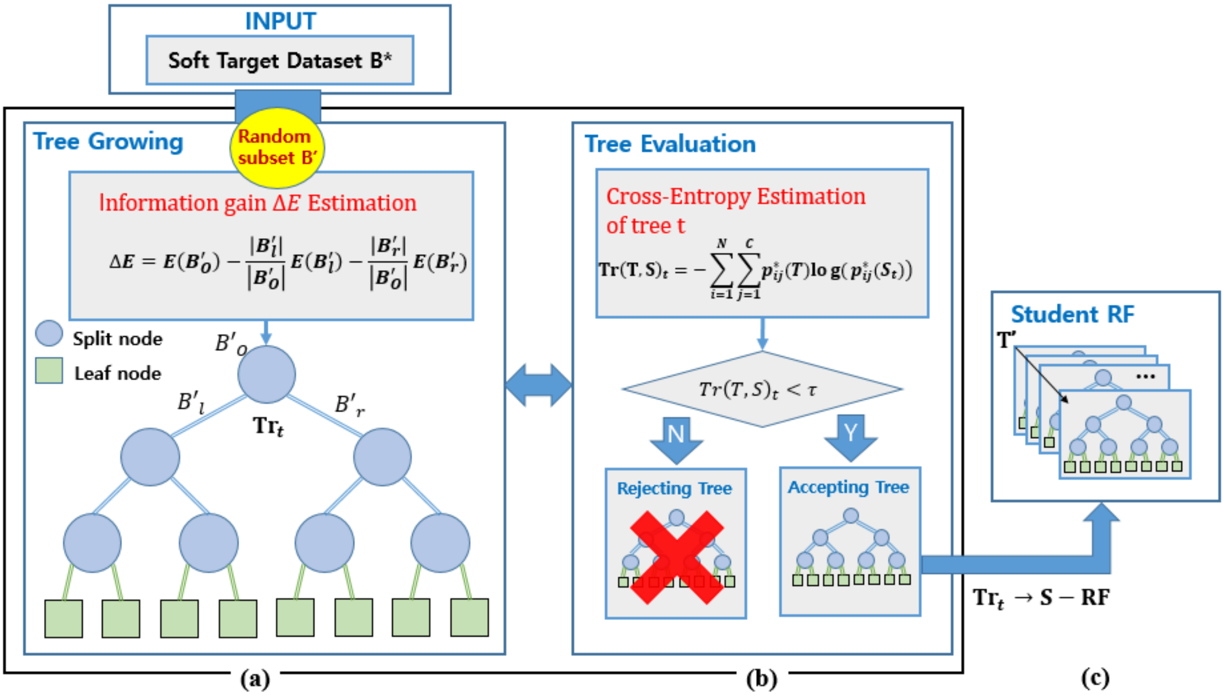

INPUT
Soft Target Dataset B*
Tree Growing
Random subset B′
Information gain ΔE Estimation
$\Delta E = E(B'_O) - \frac{|B'_l|}{|B'_O|}E(B'_l) - \frac{|B'_r|}{|B'_O|}E(B'_r)$
Split node
Leaf node
B'_O
B'_l
B'_r
Tr_t
Tree Evaluation
Cross-Entropy Estimation of tree t
$\mathrm{Tr}(\mathrm{T},\mathrm{S})_t = -\sum_{i=1}^{N}\sum_{j=1}^{C} p^*_{ij}(T)\log(p^*_{ij}(S_t))$
$Tr(T,S)_t < \tau$
N
Y
Rejecting Tree
Accepting Tree
Student RF
T'
…
$\mathrm{Tr}_t \rightarrow \mathrm{S-RF}$
(a)
(b)
(c)

FIGURE 2. Shallow random forest (S-RF) training procedure based on soft target training data B*. (a) Tree Tr_t grown using information gain criteria, (b) tree evaluation using the cross-entropy to decide whether or not candidate decision tree is selected as a member of student RF, and (c) the final S-RF consisting of T optimal trees.

A correct detection was counted if the overlap ratio between the detected bounding box and the ground truth bounding box exceeded 50%.

All the experiments were conducted using an Intel Core i7 processor with 8 GB of RAM running Microsoft Windows 10. In addition, all the RF approaches, including teacher RF and S-RF, were executed based on a CPU and the CNN-based state-of-the-art approaches were executed based on a single Titan-X GPU.

A. NUMBER OF OPTIMAL DECISION TREES FOR SHALLOW RANDOM FOREST

To determine the number of optimal trees of the S-RF, we compared the detection performance on the Town Centre dataset while changing the size (number) of trees as shown in Table 1. From the teacher RF consisting of 300 trees, we sequentially decreased the desired number of trees (T') to 250, 200, 150, 100, 50 and 30. In addition to precision and recall, we used additional standard criteria to measure the quality of the model compression and speed-up rate. The compression rate means the relative compression ratio of the S-RF to the teacher model, and similarly, the speed-up rate means the run time of the S-RF relative to the teacher model [24]. All the experiments were conducted based on a CPU, and a GPU was not used. As we can see from the Table 1, when the number of trees is 30, the number of parameters and the compression rate are excellent, but the precision and recall are relatively low. In contrast, as the number of trees is increased, the precision and recall rates are increase, but it can be seen that the number of parameters is relatively increased, and the speed and compression rate are lowered. Therefore, in this paper, we set the number of trees to 50 considering the compression rate and processing speed as well as the detection accuracy.

TABLE 1. Comparison of speed-up and compression rate using the teacher random forest and six compression methods that reduced the number of trees (M: million, ms: millisecond).

Compression	No. trees	No. params.	No. mult.	precision	recall	Speed-up	Compression Rate
Teacher RF	300	8.3 M	12,500	90.9 %	84.2 %	1 (106 ms)	1
S-RF 1	250	7.7 M	10,300	90.8 %	84.4 %	1.18 (89.4 ms)	1.07
S-RF 2	200	6.1 M	8,350	90.65%	84.71%	1.42 (74.2 ms)	1.34
S-RF 3	150	4.6 M	6,350	90.17%	84.7 %	1.76 (60 ms)	1.79
S-RF 4	100	3 M	4,200	90.0 %	84.9 %	2.2 (47.5 ms)	2.68
S-RF 5	50	1.5 M	2,050	89.4 %	85.7 %	2.78 (38.1 ms)	5.39
S-RF6	30	0.92 M	1,250	88.5 %	85.0 %	2.95 (35.8 ms)	8.99

B. DETECTION COMPARISON ON PETS2006 DATASET

To verify the effectiveness of the soft target training scheme, we compared its performance with that of six state-of-the-art methods: (1) DPM [8]; (2) the Faster R-CNN approach,

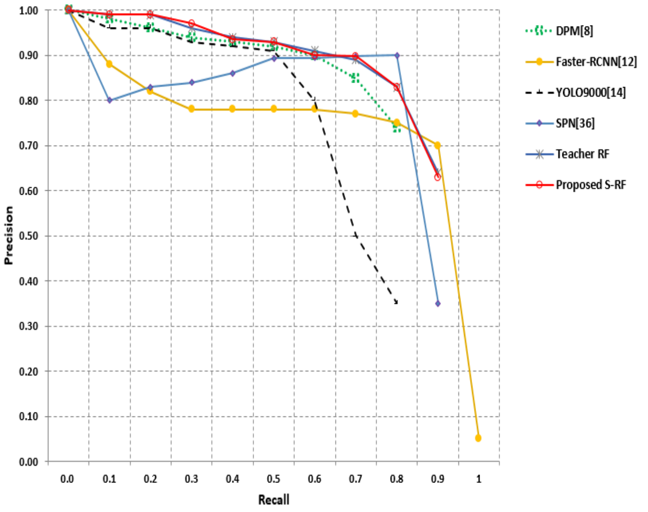


FIGURE 3. Precision-recall (PR) curve for six comparison approaches on PETS2006.

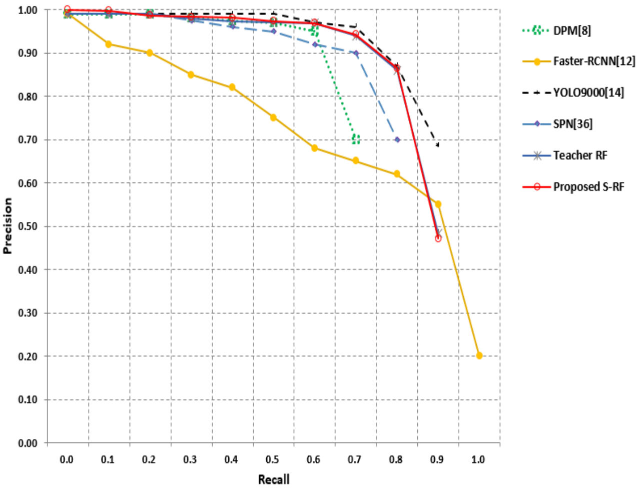


FIGURE 4. Precision-recall (PR) curve for six comparison approaches on Town Centre.

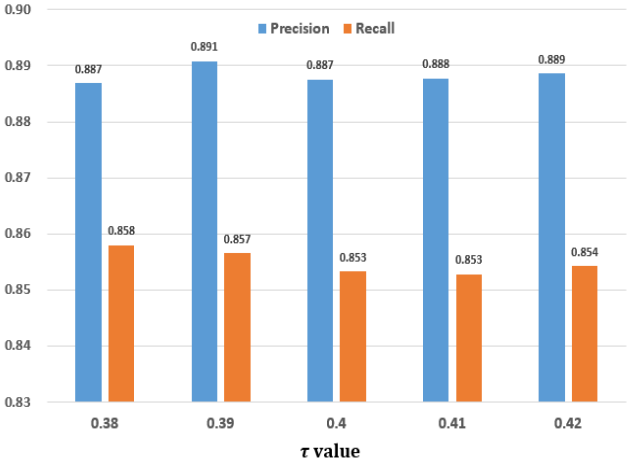


FIGURE 5. Five possible pairs of experimental results for determining the threshold τ of soft target training.

which shares full-image convolutional features with the detection network [12]; (3) the scene pose CNN network (SPN), which generates a scene-specific pedestrian detector and pose estimator [36]; (4) YOLO 9000, which is a real-time CNN-based object detection system over 9000 object categories [14]; (5) teacher RFs consisting of 300 trees (teacher RF); and (6) proposed S-RF consisting of 50 trees (proposed S-RF). Faster R-CNN and YOLO 9000 used pre-trained model parameters without performing fine-tuning. SPN used synthetic pedestrian dataset considering wider range of human poses for training.

In the first experiment using the PETS 2006 dataset, we predicted that two CNN-based methods, Faster R-CNN [12] and YOLO9000 [14], but not SPN [36], would produce a worse detection performance than the other methods as shown in Fig. 3. The main reason is that they are more likely than conventional approaches to fail to locate each individual in a perspective view, because of the small and blurred boundaries between closely positioned pedestrians.

The SPN approach showed a higher performance than Faster R-CNN, but the precision rate fluctuated according to the variation in the recall rate. In contrast, the DPM and RF-based approaches showed a better performance than the CNN-based approaches. The results confirmed that a methodology that uses ROI with scanning window is effective for pedestrian detection in surveillance videos. Moreover, the best performance of the proposed S-RF method is very similar to that of teacher RF and better than other state-of-the-art approaches.

C. DETECTION COMPARISON ON TOWN CENTRE DATASET

To achieve a fair performance comparison of the same six algorithms using an additional dataset, we also performed pedestrian detection using the Town Centre dataset. Figure 4 shows the precision and recall curves for the six methods. When the recall value was 0.5, the highest precision rates of the teacher RF, DPM [8], YOLO9000 [14] and the proposed S-RF algorithms were similar, about 0.97, and the patterns were similar to those shown in Fig. 4. Although the DPM algorithm [8] showed precision results similar to those of the proposed method up to a recall value of 0.5, the precision rate decreased rapidly when the recall value was 0.7. YOLO9000 [14] yielded a higher precision rate than other CNN-based approaches (Faster R-CNN [12] and SPN [36]), with recall values around 0.8. It also showed a somewhat higher precision rate than the proposed method when the recall value was larger than 0.8. Through our experiments using the additional Town Centre dataset, we verified that the proposed method gave a generalized performance as compared to CNN-based approaches, although the number of trees decreased considerably.

In summary, the proposed method showed a higher performance than the other state-of-the-arts methods and a similar performance pattern on the precision and recall curve. In particular, the proposed method showed better performance than other CNN-based methods for small-sized or occluded pedestrians located in the upper part of the image (see Fig. 6). The results indicate that S-RF trained by soft target training is a reasonable method to mimic the classification ability of a teacher classifier and also an efficient method for detecting small-sized and closely positioned pedestrians in a high perspective surveillance video.

D. DETECTION COMPARISON ON CALTECH DATASET

Additional experiment was performed on the Caltech dataset to test whether the proposed algorithm effectively detects pedestrians in videos with various sizes of pedestrians because approximately 70% of the pedestrian height of Caltech dataset is less than 100 pixels, including extremely small pedestrian less than 50 pixels. For this experiment, we divided dataset into two categories, following the typical protocol in literatures [39], [40]. First, *far* subset consists of non/partial occlusion pedestrians which are less than 45 pixels in height and *middle* subset has a height between 45 and 115 pixels. As the evaluation metric, we used the averaged log miss rated over the false positive per image, denoted as MR.

For evaluation, we used the standard test set of 4,024 Caltech images under two different performance protocols. To validate the detection performance of the proposed algorithm, we compared the quantitative results with that of six state-of-the-art methods focusing on detection of multi-scale pedestrians: (1) UDN+SS [10]; (2) Faster-RCNN [12]; (3) SA-Fast-RCNN, which use multi-scale CNN [21]; (4) F-DNN+SS, which uses a derivation of the Faster R-CNN; (5) TLL(MRF)+FGFA, where uses lightweight flow networks [38]; (6) TLL(MRF)+LSTM, where uses temporal feature aggregation [40]; and (7) proposed S-RF. For training, six comparison methods used the dense sampling of the training Caltech data (every 3th frame, resulted in 42782 images) and proposed method used the soft target training data B$*$.

As shown in Table 2, the propose S-RF method achieved lower MR to those of the state-of-the-arts for small-sized pedestrians. In detail, the proposed method had better MR performance for small-sized pedestrians as a result of 13.19% improvement over TTL(MRF)+LSTM [40] method. In contrast, the general CNN based methods [10], [12], [21] showed a good detection performance for middle-sized pedestrians. However, when the size of the pedestrian becomes smaller, the size of the image used for detection becomes smaller, which results in increasing the MR.

From the result, we know that the proposed student RF with adaptive image scaling is suitable algorithm for detecting small-sized pedestrians. However, when the pedestrian size is middle, MR performance is 4.68% lower than TTL(MRF)+LSTM [40]. This is because the camera perspective angle of the Caltech dataset is low, so pedestrians of various sizes overlap at the same ROI. Therefore, in the case of an image taken at a low camera perspective angle, our method is necessary to adjust the ROI and the image scaling level to reduce the MR.

TABLE 2. Comparison of log missing rate (MR, %) with recent state-of-the-art algorithms on standard dataset of Caltech (low is better)[2].

Methods	Average	*middle*	*far*
UDN+SS [10]	76.88	53.75	100
Faster-RCNN [12]	76.97	53.93	100
SA-FastRCNN [21]	75.92	51.83	100
F-DNN+SS [37]	55.26	33.15	77.37
TLL(MRF)+FGFA [38]	43.83	24.39	63.28
TLL(MRF)+LSTM [40]	41.86	**22.92**	60.79
Proposed S-RF	**37.60**	27.60	**47.60**

TABLE 3. Processing time comparison of five detection algorithms using PETS 2006 dataset.

Methods	YOLO9000 [14] (GPU)	SPN [36] (GPU)	DPM [8] (CPU)	Teacher RF (CPU)	S-RF (CPU)
FPS (ms)	19.9 ms	180 ms	210 ms	39 ms	20.8 ms

E. TIME COMPLEXITY

The purpose of soft target training based on a teacher student framework is to reduce the computational cost of the classifiers. The time complexity of the proposed S-RF was compared with that of four other methods, as shown in Table 2. DPM [8], teacher RF and S-RF were tested on an Intel Core i-7 CPU, and two CNN-based methods (YOLO 9000 [14] and SPN [36]) were tested on a Titan X GPU using PETS 2006. As shown in Table 3, the process time per frame (FPS) of YOLO9000 [14] is fastest, 19.9 ms, and is faster than the similar CNN-based SPN [36]. The second fastest method is the proposed S-RF at 20.8 ms, being faster than the similar CPU-based DPM [8] and teacher RF. Although the processing time of the proposed S-RF is 1.1 ms slower than that of YOLO90000 [14], we know that the proposed S-RF efficiently and considerably decreases the processing time without losing accuracy, because the processing time of YOLO9000 [14] is obtained on a GPU and that of the proposed S-RF is obtained on a CPU. The overall experimental results confirm that the proposed S-RF is more suitable than CNN-based or computationally heavy classifiers for low specification embedded surveillance systems.

F. DETERMINATION OF CROSS-ENTROPY THRESHOLD

Threshold τ, the minimization criterion of cross-entropy, is also an important parameter that reduces the size of the

[2] In the evaluation, we referred comparison results of [40] with recent state-of-the-art methods on standard test set of Caltech.

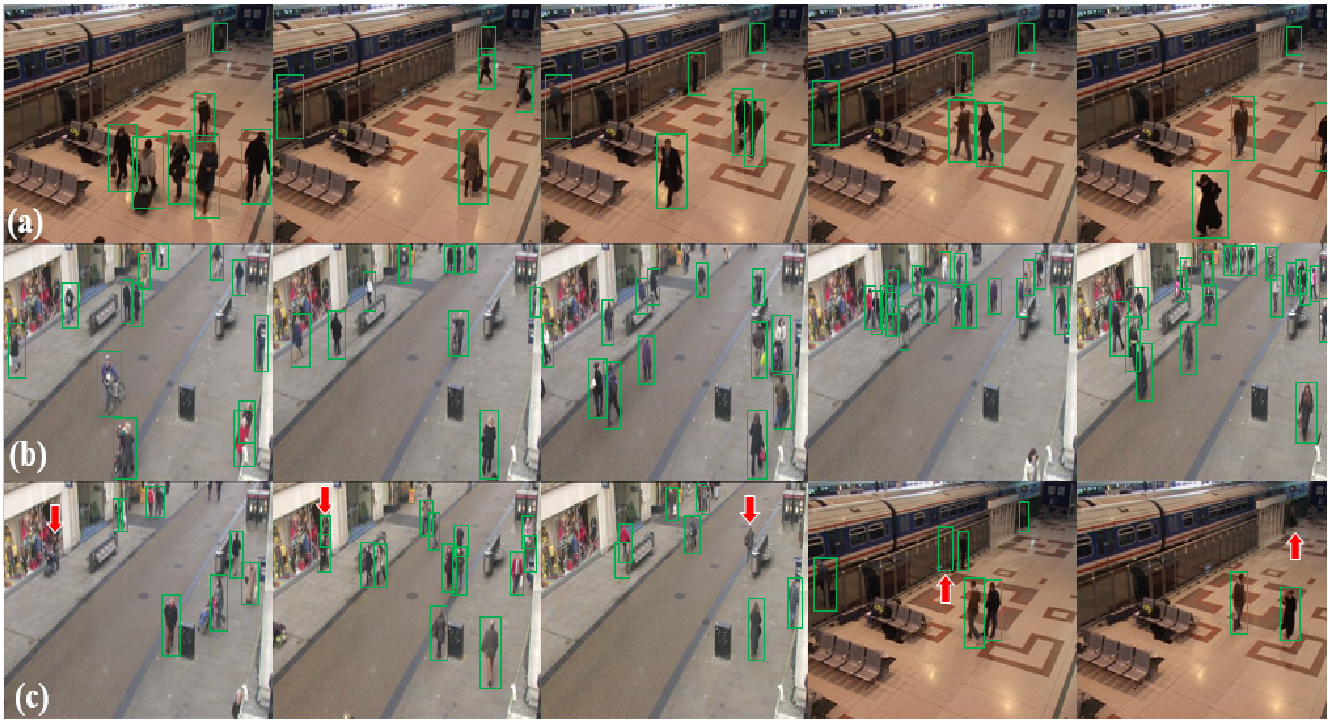


FIGURE 6. Pedestrian detection results in a surveillance camera using the proposed method, (a) the pedestrian detection results for PETS 2006, (b) the pedestrian detection results for Town Centre, (c) sample false detection results caused by a complicated or similar background and occlusion. The green rectangles indicate correctly detected pedestrians and the red arrows indicate false or missing detections.

teacher RF and the convergence speed of the S-RF training. To determine the appropriate threshold τ, we estimated the average detection precision and recall on the Town Centre dataset using five threshold values. As shown in Fig. 5, when the threshold τ is set to 0.39, the precision is slightly higher than for other values. A threshold of 0.38 yields a 0.001 higher recall value than one of 0.39. In contrast, the performance may be degraded when the threshold value is higher than 0.39. On the basis of these results, we set the cross-entropy threshold to 0.39, because the difference in precision between two values is higher than the difference in recall.

Figure 6 shows the pedestrian detection results of PETS 2006 (Fig. 6(a)) and Town Centre (Fig. 6(b) and (c)) using the proposed S-RF, in the case of occlusion. As shown in Fig. 6(a), the proposed approach detects pedestrians correctly when the pedestrians' sizes are significantly small, occluded by other pedestrians or the poses of the pedestrians differ. In Fig. 6(b), although in the Town Centre dataset more pedestrians appear on the street and the perspective angle is larger than that in PETS2006, the proposed S-RF also correctly detected pedestrians when they were small-sized and closely positioned in the high perspective surveillance video. However, the proposed S-RF still yields a few incorrect detections: it missed a pedestrian pushing a stroller (Fig. 6(c), first column) and falsely detected a pedestrian on a background that was similar in color and complicated (Fig. 6(c), second and fourth columns); it also missed a pedestrian when occluded by other objects and when the pattern of the background was similar to that of the pedestrian (Fig. 6(c), third and fifth columns).

Demo videos have been uploaded to our webpage, http://cvpr.kmu.ac.kr/SoftTarget.htm.

V. CONCLUSION

In this paper, we proposed a pedestrian detection algorithm that can detect small-sized and closely positioned pedestrians in a surveillance video when the camera is installed at a high location. Although deep learning, especially CNN, based approaches are known to achieve top performances in pedestrian detection tasks, they also require very wide and deep networks with numerous parameters, a large-scale dataset and massive computing power for kernel multiplication. Moreover, CNN-based detectors have difficulty not only detecting small-sized pedestrians in a low resolution video but also operating in a low specification embedded surveillance system. To detect small-sized pedestrians and determine the levels of image scaling, we adopted HWMs and an adaptive ROI algorithm instead of predicting the ROI using Faster R-CNN and YOLO9000. This study focused on exploring new types of model compression algorithms realized by transferring the teacher-student framework to an RF model instead of using heavy deep learning, because RF has a structure to which a teacher-student framework can be easily applied by reducing the size of the forest through pruning the number of decision trees. The proposed teacher-student compression model S-RF is a shallower version of the original teacher RF, using a softened version of the teacher's output. The experimental results proved that the proposed S-RF trained by soft target training is a reasonable method to mimic the classification ability of a teacher classifier. In addition, in high perspective surveillance datasets it also efficiently detected small-sized and closely positioned pedestrians and decreased the processing time considerably without losing accuracy.

In future work, we plan to improve our algorithm in order to reduce the false and missing detection rate when a pedestrian is similar to the background or is occluded by other objects by considering another feature. Moreover, although the proposed method exhibited a reasonable computational speed without degrading accuracy when run on a PC, a more compact S-RF version is needed so that it can be applied it to a low specification embedded board. In our opinion, it is reasonable to combine two or three of model compression algorithms to maximize the compression/speed up rates, for example, compressing the RF with teacher-student framework and performing a pruning method in each decision tree.

REFERENCES

[1] Q. Chen, W. Jiang, Y. Zhao, and Z. Zhao, "Part-based deep network for pedestrian detection in surveillance videos," in *Proc. Vis. Commun. Image Proces.*, Dec. 2015, pp. 1–4.

[2] Y. Tian, P. Luo, X. Luo, X. Wang, and X. Tang, "Pedestrian detection aided by deep learning semantic tasks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5079–5093.

[3] Z. Zhao, H. D. Bian Hu, and W. Cheng, "Pedestrian detection based on fast R-CNN and batch normalization," in *Proc. Int. Conf. Intell. Comput.*, Aug. 2017, pp. 735–746.

[4] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "FitNets: Hints for thin deep nets," in *Proc. Int. Conf. Learn. Represent.*, May 2015, pp. 1–10.

[5] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, USA, Jun. 2005, pp. 886–893.

[6] B. C. Ko, M. Jeong, and J. Nam, "Fast human detection for intelligent monitoring using surveillance visible sensors," *Sensors*, vol. 14, no. 11, pp. 21247–21257, 2014.

[7] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *Proc. 8th Eur. Conf. Comput. Vis.*, May 2004, pp. 469–481.

[8] 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.

[9] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah, "Part-based multiple-person tracking with partial occlusion handling," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1815–1821.

[10] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2056–2063.

[11] J. Hosang, R. Beneson, and B. Schiele, "How good are detection proposals, really?" in *Proc. Brit. Mach. Vis. Conf.*, Sep. 2014, pp. 1–12.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 779–788.

[14] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 6517–6525.

[15] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer. (2016). "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size." [Online]. Available: https://arxiv.org/abs/1602.07360

[16] A. G. Howard *et al.* (2017). "MobileNets: Efficient convolutional neural networks for mobile vision applications." [Online]. Available: https://arxiv.org/abs/1704.04861

[17] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.

[18] A. Wong, M. J. Shafiee, F. Li, and B. Chwyl. (2018). "Tiny SSD: A tiny single-shot detection deep convolutional neural network for real-time embedded object detection." [Online]. Available: https://arxiv.org/abs/1802.06488

[19] G. Bae, S. Kwak, H. Byun, and D. Park, "Method to improve efficiency of human detection using scalemap," *Electron. Lett.*, vol. 50, no. 4, pp. 265–267, Feb. 2014.

[20] H. Hattori, V. N. Boddeti, K. Kitani, and T. Kanade, "Learning scene-specific pedestrian detectors without real data," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3819–3827.

[21] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. 14th Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 354–370.

[22] X. Jiang, Y. Pang, X. Li, and J. Pan, "Speed up deep neural network based pedestrian detection by sharing features across multi-scale models," *Neurocomputing*, vol. 185, pp. 163–170, Apr. 2016.

[23] B.C. Ko, J. Y. Kwak, and J. Y. Nam, "Human tracking in thermal images using adaptive particle filters with online random forest learning," *Opt. Eng.*, vol. 52, no. 11, pp. 1–14, Nov. 2013.

[24] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 126–136, Jan. 2018.

[25] S. Han, J. Pool, J. Tran, and W. J. Dally, "Learning both weights and connections for efficient neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2015, pp. 1135–1143.

[26] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Speeding up convolutional neural networks with low rank expansions?" in *Proc. Brit. Mach. Vis. Conf.*, Sep. 2014, pp. 1–13.

[27] S. Dieleman, J. De Fauw, and K. Kavukcuoglu, "Exploiting cyclic symmetry in convolutional neural networks," in *Proc. IEEE Int. Conf. Mach. Learn.*, Jun. 2016, pp. 1889–1898.

[28] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proc. Adv. Neural Inf. Process. Syst. Workshop*, Dec. 2014, pp. 1–9.

[29] R. Price, K.-I. Iso, and K. Shinoda, "Wise teachers train better DNN acoustic models," *EURASIP J. Audio, Speech, Music Process.*, vol. 10, pp. 1–19, Apr. 2016.

[30] G. Urban *et al.*, "Do deep convolutional nets really need to be deep and convolutional?" in *Proc. Int. Conf. Learn. Represent.*, Aug. 2017, pp. 1–13.

[31] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[32] B. C. Ko, S. H. Kim, and J. Y. Nam, "X-ray Image classification using random forests with local wavelet-based CS-local binary patterns," *J. Digit. Imag.*, vol. 24, no. 6, pp. 1141–1151, 2011.

[33] 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.

[34] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *Proc. IEEE Conf. CVPR*, Jun. 2011, pp. 3457–3464.

[35] D. Thirde, L. Li, and F. Ferryman, "Overview of the PETS2006 challenge," in *Proc. 9th IEEE Int. Workshop Perform. Eval. Tracking Surveill. (PETS)*, Jun. 2006, pp. 47–50.

[36] H. Hattori, N. Lee, V. N. Boddeti, F. Beainy, K. Kitani, and T. Kanade, "Synthesizing a scene-specific pedestrian detector and pose estimator for static video surveillance," *Int. J. Comput. Vis.*, vol. 126, no. 9, pp. 1027–1044, 2018.

[37] 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. IEEE Winter Conf. Appl. Comput. Vis.*, Mar. 2017, pp. 953–961.

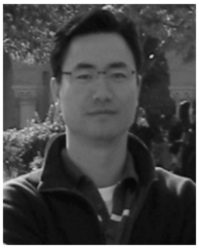
[38] X. Zhu, Y. Wang, J. Dai, L. Yuan, and Y. Wei, "Flow-guided feature aggregation for video object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 408–417.

[39] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, "Multispectral pedestrian detection: Benchmark dataset and baseline," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1037–1045.

[40] T. Song, L. Sun, D. Xie, H. Sun, and S. Pu, "Small-scale pedestrian detection based on topological line localization and temporal feature aggregation," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 1–16.

SANGJUN KIM received the B.S. degree in computer engineering from Keimyung University, Daegu, South Korea, in 2017, where he is currently pursuing the master's degree with the Computer Vision and Pattern Recognition Laboratory. His current research interest includes advance driver assistance systems using computer vision. He received the Best Paper Awards, in 2016 and 2017, from the Korean Institute of Information Scientist and Engineers.

SOOYEONG KWAK received the Ph.D. degree in computer science from Yonsei University, Seoul, South Korea, in 2010. From 2010 to 2011, she was a Senior Researcher with the Visual Display Division, Samsung Electronics. She is currently an Associate Professor with the Department of Electronics and Control Engineering, Hanbat National University. Her research interests include intelligent surveillance, sports motion analysis, and intelligent vehicle applications.



BYOUNG CHUL KO received the B.S. degree from Kyonggi University, Suwon, South Korea, in 1998, and the M.S. and Ph.D. degrees in computer science from Yonsei University, Seoul, South Korea, in 2000 and 2004, respectively. From 2004 to 2005, he was a Senior Researcher with Samsung Electronics, Suwon, where he was involved in the Ubiquitous Robot Companion Project on the subject of robot event detection and face recognition using charge-coupled device cameras. He is currently a Professor with the Department of Computer Engineering and the Vice Dean of the College of Engineering, Keimyung University, Daegu, South Korea. His current research interests include content-based image retrieval, vision-based fire detection, advance driver assistance systems, and biomedical image processing. He received several excellent paper awards from the Korean Institute of Information Scientists and Engineers. He was selected as the Best Researcher and Lecturer, in 2013, 2014, 2015, and 2018, at Keimyung University.