Pattern Recognition Letters 90 (2017) 43–49

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Deep network aided by guiding network for pedestrian detection

Sang-Il Jung*, Ki-Sang Hong

Image Information Processing Laboratory, POSTECH, 77 Cheongam-Ro Nam-Gu Pohang Gyeongbuk 37673, Republic of Korea

ARTICLE INFO

Article history:
Received 17 May 2016
Available online 16 March 2017

MSC:
41A05
41A10
65D05
65D17

Keywords:
Pedestrian detection
Deep convolutional neural network

ABSTRACT

We propose a guiding network to assist with training a deep convolutional neural network (DCNN) to improve the accuracy of pedestrian detection. The guiding network is adaptively appended to the pedestrian region of the last convolutional layer; the guiding network helps the DCNN to learn the convolutional layers for pedestrian features by focusing on the pedestrian region. The guiding network is used only for training, and therefore does not affect the inference speed. We also explore other factors such as proposal methods and imbalance of training samples. By adopting a guiding network and tackling these factors, our method yields a new state-of-the-art detection accuracy on the Caltech Pedestrian dataset and presents competitive results with the state-of-the-art methods on the INRIA and KITTI datasets.

1. Introduction

The goal of pedestrian detection is to draw bounding boxes that tightly enclose pedestrians in a given image. A single bounding box should be found for each pedestrian; i.e., the detection task includes localization and classification. Localization specifies the positions of bounding boxes, and classification gives confidence regarding whether or not the box includes a pedestrian. Scanning an image in a sliding window fashion is a well-known approach to resolve this detection task. To detect various sizes of pedestrian, the scanning process must be conducted at every position and at multiple scales in an image. The AdaBoost-based algorithms [9,17,18,23,35] conducted in this fashion have achieved considerable success due to their fast processing speed. However, designing good hand-crafted features which might improve the detection accuracy is a difficult task.

Deep convolutional neural networks (DCNNs) have achieved accurate pedestrian detection [20,26]. DCNNs are too computationally expensive to be practical to scan an image in a sliding window fashion; therefore, the proposal-and-classification strategy is applied: first candidate (proposal) windows (bounding boxes) are extracted by a fast pedestrian detection method (e.g., AdaBoost-based method), then the windows are classified by DCNNs. This strategy originated from the R-CNN [13] in object detection, and achieves a good trade-off between the strong discriminative power and large computational burden of DCNNs. To improve detection accuracy, most work focused on designing a unique DCNN or on combining multiple DCNNs. DeepParts [26] used an ensemble of DCNNs (one for each body part), and SA-FastRCNN [20] combined two subnetworks to learn two different features depending on the instance scale (one for large and one for small). In this paper, we argue that a single typical DCNN can be acceptably discriminative if trained appropriately.

We propose a novel guiding network to assist with training the baseline network. The guiding network is adaptively appended to the region on the last convolutional layer of the baseline network; the region corresponds to the actual pedestrian of an input sample image. The underlying idea of the guiding network is to focus on the actual pedestrian region. We gather the training samples from the bounding boxes extracted by a proposal method; the samples are extracted from images by cropping the candidate region for a pedestrian with some contextual extension (Fig. 1). For contextual extension, all pixels surrounding the proposed bounding box with a specific scale ratio are added. Because the proposal methods do not consider the localization problem, a pedestrian is situated in an arbitrary position and scale in the proposal sample image that includes some biased background (first six images in Fig. 1). The biased background is caused by miscellaneous objects such as cars and trees, which often appear concurrently with pedestrians. This bias reduces the detection accuracy. For example, we have observed in experiments that car parts are often classified as pedestrians (Section 4.4). Focusing on the actual pedestrian region helps to reduce this bias by forcing the convolutional layers to learn the features of pedestrians. Our guiding network also focuses adaptively on the pedestrian-like pattern in the negative proposal image during the training process; the pedestrian-like pattern is a

* Corresponding author.
E-mail address: sijung@postech.ac.kr (S.-I. Jung).

http://dx.doi.org/10.1016/j.patrec.2017.02.018
0167-8655/

Fig. 1. Some proposal sample images which include some context around the proposed bounding boxes (dotted green rectangles). The red rectangles indicate the ground-truth bounding boxes. The sizes of the proposed bounding box and the extended box are 100 × 41 and 128 × 64, respectively. The first six images include pedestrians and the last two images do not. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

background pattern that might be misinterpreted as a pedestrian (last two images in Fig. 1). In this way, the guiding network helps to learn discriminative features for pedestrians. The guiding network is trained together with the baseline network by optimizing the combined loss function of both networks. The guiding network is used only during training; it is not used during testing, so it does not affect the inference speed.

Other factors that affect the detection accuracy include imbalance between the number of positive and negative samples in training data, and the proposal methods for extracting candidate bounding boxes. We explore these factors to improve the detection accuracy (Section 3).

Our main contributions are:

- We propose a novel guiding network to assist with learning the baseline network that performs classification and localization for pedestrian detection.
- We explore other factors such as proposal methods and the problem of imbalance between the number of positive and negative samples.
- We evaluate our method on three benchmarks: Caltech, INRIA and KITTI. We achieve a new state-of-the-art detection accuracy on the Caltech Pedestrian dataset and competitive results on the INRIA and KITTI datasets. The inference speed is fairly fast because our method does not increase the complexity of the network architecture.

The rest of this paper is organized as follows. Some work related to ours is reviewed in Section 2. The proposed methods are described in detail in Section 3. Experimental results that show the superiority of our method are given in Section 4.

2. Related work

AdaBoost has been widely used in pedestrian detection due to good classification accuracy and fast processing. Various hand-crafted features have been designed for AdaBoost; e.g., aggregated channel feature (ACF) [9], locally decorrelated channel feature (LDCF) [23], aggregated channel comparison feature (ACCF) [17], filtered channel feature (FCF) [35] and effective comparison feature (ECF) [18]. ACF aggregates 10 channels of the candidate pedestrian sample: three of LUV, six of gradient orientations, and one of gradient magnitude. The 10 channels can be transformed by local decorrelation (LDCF), or by various filter banks (FCF) to increase its discriminative power. ACCF is generated by comparing two values of random positions in the same channel and ECF is generated by selecting effective features among the comparison features. We used ACF and ECF as proposal methods in this paper. Convolutional channel features (CCF) [30] takes features learned by DCNN and uses them for AdaBoost learning.

DCNN-based pedestrian detection algorithms have achieved good detection accuracy. Hosang et al. [15] demonstrated an R-CNN pipeline with various proposal methods and variants of AlexNet [19]. TA-CNN [27] trained a DCNN for pedestrian detection jointly with auxiliary semantic tasks that include pedestrian attributes and scene attributes. Our guiding network is similar to TA-CNN in that an additional network assists in the original task, but we do not use external information such as semantic labels. DeepParts [26] addressed the occlusion problem by using an ensemble of DCNNs, but the computational cost is very high. CompACT-Deep [4] is a complexity-aware cascaded detector that combines various hand-crafted features with DCNN features seamlessly. A scale-aware pedestrian detection method based on Fast R-CNN (SA-FastRCNN) [20] combines two subnetworks (one for large scales and one for small scales) in a soft way to solve the scale problem. Hu et al. [16] further improved the detection accuracy by combining various information such as hand-crafted features, DCNN mid-layer features, semantic segmentation, and optical flows, then merging the information in the final scoring stage.

3. The proposed method

In this section, we describe our pedestrian detection method. We use a proposal-and-classification approach to reduce the computational burden of detecting pedestrians on multiple scales. To extract proposal samples, we use fast pedestrian detectors based on AdaBoost algorithm such as ACF [9] and ECF [18]. The candidate regions with some context are cropped from the image and resized to fixed size (128 × 64 in our system) while preserving the aspect ratio of bounding boxes. Then the regions are used as input to our DCNN for accurate classification and localization.

3.1. Baseline DCNN architecture

Our baseline network takes a proposal sample $\mathbf{x} \in \mathbb{R}^{128\times64\times3}$ as an input, and gives the classification score of pedestrian and the bounding box regression offset. The network consists of five convolutional blocks, two fully-connected layers, and two output layers for classification and localization. The five convolutional blocks are the same as those of the VGG-16 network [25], in which each convolutional block consists of two or three 3 × 3 convolutional layers, rectified linear unit layers, and a max-pooling layer. These convolutional layers produce a feature map of size 4 × 2 × 512. The feature map goes through a chain of two fully-connected layers of dimension 2048 followed by two split output layers. The first output layer is the binary classification layer (sample does/does not contain a pedestrian); the second output layer is the bounding box regression layer. This output layer architecture is taken from Fast R-CNN [12]. The loss L_B of the baseline network can be defined as

$$L_B = L_B^{cls} + L_B^{loc}, \tag{1}$$

where L_B^{cls} is classification loss (2-way softmax-log loss) and L_B^{loc} is localization loss (smooth$_{L1}$ loss).

We used three values $\mathbf{t} = (t^x, t^y, t^s)$ for bounding box regression offset where (t^x, t^y) denotes scale-invariant translation in each direction and t^s is a scale transformation; in this context, the regression is the transformation of a proposed bounding box to the regressed (localized) bounding box or to the target ground-truth bounding box. The bounding box aspect ratio of a pedestrian does not vary considerably, unlike other objects (e.g., aspect ratios of cars vary greatly with viewpoint); the mean aspect ratio of a pedestrian is 0.41 [10]. Therefore, to compute bounding box regression offset we used a single scale transformation factor t^s instead of two (t^w, t^h) width and height transformation. To adjust the ranges of L_B^{cls} and L_B^{loc}, we normalized the ground-truth localization targets to have zero mean and unit variance, as in Fast R-CNN [12].