Associated Metric Coding Network for Pedestrian Detection

Shuai Chen and Bo Ma(✉)

Beijing Laboratory of Intelligent Information Technology,
Beijing Institute of Technology, Beijing, China
{chenshuai,bma000}@bit.edu.cn

Abstract. Convolutional neural networks (CNNs) have played a significant role in pedestrian detection, owing to their capacity of learning deep features from original image. It is noteworthy that most of the existing generalized objection detection networks must crop or warp the inputs to fixed-size which leads to the low performance on multifarious input sizes. Moreover, the lacking of hard negatives mining constrains the ability of recognition. To alleviate the problems, an associated work network which contains a metric coding net (MC-net) and a weighted association CNN (WA-CNN), is introduced. With region proposal net in low layer, MC-net is introduced to strengthen the difference of intra-class. WA-CNN can be regarded as a network to reinforce the distance of inter-class and it associates the MC-net to accomplish the detection task by a weighted strategy. Extensive evaluations show that our approach outperforms the state-of-the-art methods on the Caltech and INRIA datasets.

Keywords: Pedestrian detection · Region proposal net
Weight association CNN · Metric coding net

1 Introduction

Detecting the pedestrians from original images which contain a wealth of object information, such as car, tree and the sky, is a very challenging work. The significant advances in traditional model [4,26,28] and deep model [11,15–17,22] in this area have been witnessed in recent years.

The traditional model which extracts the low-level features (*e.g.* HoG [4], Haar [26], HoG-LBP [28]) from images and then selects rich representations to train the classifier(*e.g.* SVM [4], boosting classifiers [6]), is a widely used strategy, but it is hard to be optimized unitedly for decreasing the error rate.

In the deep model, CNNs have played a significant role in the pedestrian detection, owing to their capacity of learning representative and discriminative features from the original images. For example, as the number of negatives is significantly larger than that of positive ones in one database, Tian et al. [24]

B. Ma—This work was supported in part by the National Natural Science Foundation of China (No. 61472036).

J. Yang et al. (Eds.): CCCV 2017, Part III, CCIS 773, pp. 120–131, 2017.
https://doi.org/10.1007/978-981-10-7305-2_11

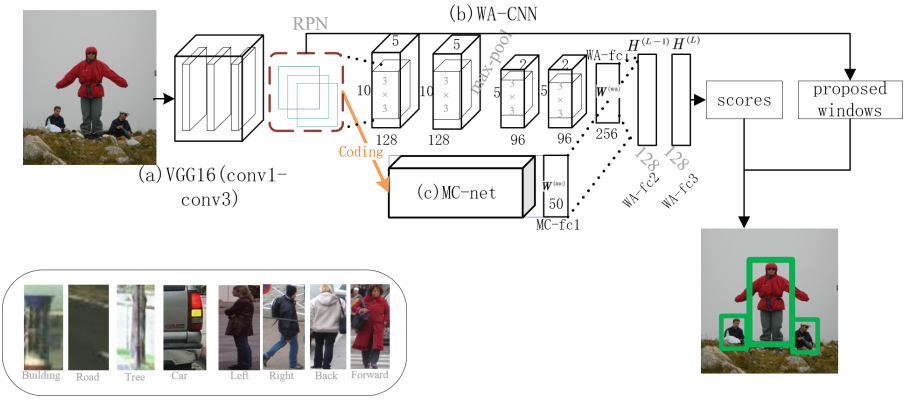

Fig. 1. The associated work network contain three parts: (a) the RPN in low layer. (b) weighted association CNN. (c) metric coding net

transfered scene attribute information from existing background scene segmentation databases to the pedestrian dataset for learning representative features.

However, most of the previous deep models must crop or warp the images to fixed-size (*e.g.* 224×224 in VGG16 [23]) which leads to the low performance on multifarious input sizes [9]. To solve this problem, spatial pyramid pooling in deep convolutional networks (SPP-net) [9] has been proposed to pool the feature maps with arbitrary size before the full-connected layers using spatial pyramid pooling. Further, [21] proposed a Region Proposal Network (RPN) that it shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. But these strategies are not suitable for little size inputs (the size of pedestrian) with a very deep CNN (*e.g.* VGG16, VGG19 [23]), meanwhile lacking of any strategy for hard negatives mining constrains the ability of recognition in these methods. The problems have been attracting increasing attention for accurate, yet efficient, pedestrian detection. [32] redesign RPN for pedestrian size and unite Boosted Forests (BF) to mine the hard negatives. But using RPN united the BF in low layer simply will lost many rich features for large sizes and they are hard to be optimized unitedly.

Driven by these observations and considered the excellent performance of VGG16 and RPN. We propose an effective baseline for pedestrian detection, apply RPN in low layer for generating the proposal windows with arbitrary size. Furthermore, as the mining of negative samples is significant, an associated work network feeding with the labeled multi-class negative and positive samples, is introduced in our work. It contains two networks: metric coding net (MC-net) and weighted association CNN (WA-CNN). MC-net which is based on metric learning theory, is devised to reinforce the intra-class distance. With the strategy, the network will encode the feature by the template parameter, and the generated codes can be seen as the comparability determination between the feature and

the template parameter. Finally, WA-CNN which is designed to strengthen the inter-class difference by a deep model, associates the metric codes to accomplish the detection task using a weighted loss function.

This work has the following main **contributions**. (1) MC-net is devised to reinforce the intra-class distance. Feeding with the feature maps extracted from labeled viewpoint pedestrian and no-pedestrian images, the template parameter will be trained. After the training, the metric codes are encoded by the net, and the codes can be seen as a comparability measurement between the inputs and the template parameter. (2) WA-CNN is proposed to reinforce the distance of inter-class in our network with a deep CNN network, and it will associate the metric codes to accomplish the detection task with a weighted loss function.

2 From Supervised Generalized Max Pooling to the Template of Multi-class

Our goal is to propose a template learning mechanism that we attempt to represent multi-class by vectors. Motivated by a property proposed in Generalized Max Pooling (GMP) [12] that the dot-product similarity between the max-pooling representation (a vector) and a feature matrix is a constant value:

$$\boldsymbol{\psi'}\boldsymbol{\phi} = \boldsymbol{\alpha}, \tag{1}$$

where $\boldsymbol{\psi'}$ is the feature matrix, $\boldsymbol{\phi}$ is max-pooling representation, $\boldsymbol{\alpha}$ is a vector with all elements being a constant value and the value of the constant has no influence [12].

We generalize the max-pooling representation to be a template vector (representing one class). Because of the randomicity of $\boldsymbol{\alpha}$, we enforce the dot-product similarity between the feature and the template vector to be a generalized max pooling vector which is the mean value of all max-pooling vectors belonging to one class. The primal formulation is

$$\sum_{i=1}^{\tau} \boldsymbol{\psi'}_{\boldsymbol{i}}\boldsymbol{\phi} = \boldsymbol{\alpha}, \tag{2}$$

where $\boldsymbol{\psi}'_{\boldsymbol{i}}$ is the i-th labeled feature matrix of τ images belonging to one class. The vector $\boldsymbol{\alpha}$ can be seen as a supervised method to calculate the max pooling, and it can be seen as a comparability determination between the template vector and the feature matrix as well.

In order to learn the template vector, we can turn Eq. (2) into a least square regression problem

$$\Gamma = \frac{1}{2}||\sum_{i=1}^{\tau} \boldsymbol{\psi}'_{\boldsymbol{i}}\boldsymbol{\phi} - \boldsymbol{\alpha}||^2. \tag{3}$$

To the problem, we must calculate several template vectors to represent multi-class in this system. Thus, we introduce metric coding net (MC-net) to generalize vectors $\boldsymbol{\phi}$ as a template parameter which is learned by a neural network, to represent multi-class template.

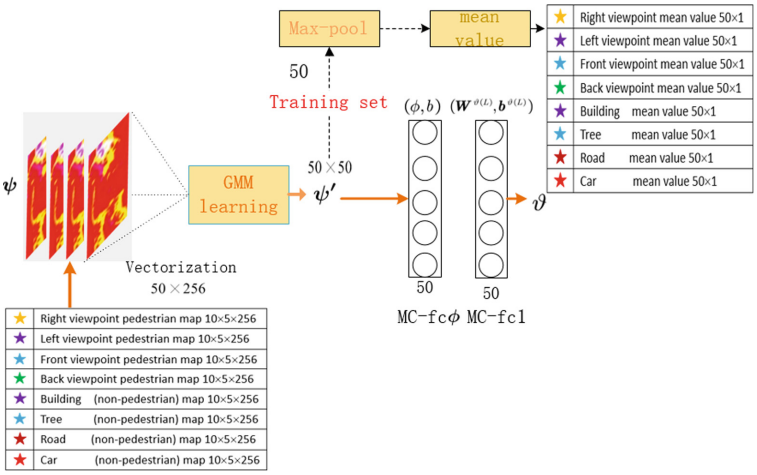

Fig. 2. Metric coding net with GMM learning and two full-connected layers.

2.1 Formulation of MC-Net

Because the low layers focus on the local features and they encode more discriminative features to capture intra-class variations [27]. Inspired by this property, MC-net is introduced to reinforce the intra-class distance by the low layer feature maps.

The net employs the feature maps $\boldsymbol{\psi} \in \mathbb{R}^{10\times5\times256}$ as input and all feature maps are reshaped into a matrix ($\boldsymbol{\psi''} \in \mathbb{R}^{50\times256}$) with 50-dimensional vector and 256 feature maps by vectorization.

To derive a more compact and discriminative representation, we utilize Gaussian Mixture Model (GMM) to model the generation process of feature maps. Assume that the feature maps are subject to parametric distribution $\mathcal{P}_\lambda(\boldsymbol{\psi''})$. Then, $\mathcal{P}_\lambda(\boldsymbol{\psi''})$ can be written as

$$\mathcal{P}_\lambda(\boldsymbol{\psi''}) = \sum_{t=1}^{T} \omega_t p_t(\boldsymbol{\psi''}), \tag{4}$$

where p_t is the t-th component of GMM with

$$p_t(\boldsymbol{\psi''}) = \frac{1}{(2\pi)^{d/2}|\boldsymbol{\Sigma}_t|^{1/2}} e^{\left(-\frac{1}{2}(\boldsymbol{\psi''}-\boldsymbol{\mu}_t)^T \boldsymbol{\Sigma}_t{}^{-1}(\boldsymbol{\psi''}-\boldsymbol{\mu}_t)\right)}, \tag{5}$$

and $\lambda = \{\omega_t, \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t\}_{t=1,\cdots,T}$ (T is 25 in our work) denotes the parameters of GMM training by $\boldsymbol{\psi''}$. Because the weight parameters bring little additional information, we use $\boldsymbol{\psi'} = \{\boldsymbol{\mu}_t, \Sigma_t\}_1^T \in \mathbb{R}^{50\times50}$ to represent the feature maps.

The coding framework contains one full-connected layer (MC-fcϕ, input map 1) at the beginning. The weight ϕ of MC-fcϕ can be seen as the template vector in Eq. (2), the output

$$\boldsymbol{\vartheta} = \boldsymbol{\psi'}\boldsymbol{\phi} + \boldsymbol{b} \tag{6}$$

will be calculated using Eq. (3) as the loss function and $\boldsymbol{\alpha} = \boldsymbol{\vartheta} - \boldsymbol{b}$.

To strengthen the capacity of representing the multi-class, we increase one full-connected layer (MC-fc1) in the framework and initialize the weights of MC-fc1 and MC-fc$\boldsymbol{\phi}$ randomly. The detailed setting is in Fig. 1(c). The forward propagation is passed from MC-fc$\boldsymbol{\phi}$ without activation function to MC-fc1 by

$$\boldsymbol{\vartheta} = \boldsymbol{W}^{\vartheta(L)}\left(\boldsymbol{\psi'}\boldsymbol{\phi} + \boldsymbol{b}\right) + \boldsymbol{b}^{\vartheta(L)}, \tag{7}$$

where $\boldsymbol{\vartheta}$, $\boldsymbol{W}^{\vartheta(\boldsymbol{L})}$, $\boldsymbol{b}^{\vartheta(L)}$ indicate the top-layer feature vector, weights and bias respectively, $\boldsymbol{\phi}$, $\boldsymbol{b}$ are the weight and the bias parameter of MC-fc$\boldsymbol{\phi}$ respectively. Without the activation function, the two layers can be combined by linear combination, Eq. (7) can be written as

$$\boldsymbol{\vartheta} = \boldsymbol{\psi'}\left(\boldsymbol{W}^{\vartheta(L)} \oplus \boldsymbol{\phi}\right) + \left(\boldsymbol{b} \oplus \boldsymbol{b}^{\vartheta(L)}\right) \tag{8}$$

where $\oplus$ is the operation of linear combination. This method is equivalent to increase the dimension of $\boldsymbol{\phi}$ simply. We use the non-linear activation function to reconstitute the network

$$\boldsymbol{\vartheta} = \boldsymbol{W}^{\vartheta(L)}\left(ReLu\left(\boldsymbol{\psi'}\boldsymbol{\phi} + \boldsymbol{b}\right)\right) + \boldsymbol{b}^{\vartheta(L)}, \tag{9}$$

where $ReLu$ is the rectified linear function [13]. The output

$$\boldsymbol{\vartheta} = \boldsymbol{\psi'}\left(\boldsymbol{W}^{\vartheta(L)} \bowtie \boldsymbol{\phi}\right) + \left(\boldsymbol{b} \bowtie \boldsymbol{b}^{\vartheta(L)}\right) \tag{10}$$

can be seen as the comparability determination between the input feature $\boldsymbol{\psi'}$ and the multi-class template parameter, $\bowtie$ is a generalized symbol of non-linear combination by $ReLu$.

2.2 The Training of the Net

Let $\boldsymbol{B} = \{(\boldsymbol{\psi}, \boldsymbol{\alpha}_i)\}_{i=1}^{K}$ be the training set, K is the number of training images. Specifically, corresponding to the max pooling vector in Eq. (2), $\boldsymbol{\alpha}_i$ denotes eight labeled mean values, where we pool $\boldsymbol{\psi'}$ to $\boldsymbol{\psi}^{\max} \in \mathbb{R}^{50\times 1}$ by max-pooling and calculate the mean value $\boldsymbol{\alpha}_i$ in each labeled class.

Corresponding to Eq. (3),

$$E^{(MC)} = \frac{1}{2}||\boldsymbol{\vartheta} - \boldsymbol{\alpha}||^2, \tag{11}$$

is used as the loss function, where $\boldsymbol{\vartheta}$ is the output of the net, $\boldsymbol{\alpha}$ is the labeled mean value, as show in Table 1.

During the training, we set a maximum epoch number (2000) and the training process will be terminated when the objective converges on a relative little value.

Table 1. The labeled mean value.

	Labeled class ($\boldsymbol{\alpha}$)	Symbol
Pedestrian viewpoint	Front viewpoint	$\boldsymbol{\alpha}_{front}$
	Left viewpoint	$\boldsymbol{\alpha}_{left}$
	Right viewpoint	$\boldsymbol{\alpha}_{right}$
	Back viewpoint	$\boldsymbol{\alpha}_{back}$
Non-pedestrian	Building	$\boldsymbol{\alpha}_{building}$
	Tree	$\boldsymbol{\alpha}_{front}$
	Road	$\boldsymbol{\alpha}_{road}$
	Car	$\boldsymbol{\alpha}_{car}$

Therefore, after the training, the parameter $\left[\left(\boldsymbol{W}^{\vartheta(L)} \bowtie \boldsymbol{\phi}\right), \left(\boldsymbol{b}^{\vartheta(L)} \bowtie \boldsymbol{b}\right)\right]$ can be seen as a generalized template. After inputting an arbitrary image, the output will be the metric code $\boldsymbol{\vartheta}$, it can be used directly as the comparability metric between the image and template.

3 Weighted Association CNN

As analyzed in [27], different layers encode different types of features and higher layers semantic concepts on object categories. Motivated by the property, WA-CNN is proposed to reinforce the distance of inter-class by a deep model.

3.1 The Formulation of WA-CNN

Let $\boldsymbol{D} = \{(\boldsymbol{\psi}, y'_i)\}_{i=1}^{K}$ be the training feature maps set, where $y'_i = (y_i, \varrho_i^p, \varrho_i^n)$ is a three-tuple. y_i indicates whether a feature map is pedestrian or not. Binary labels $\varrho_i^p = \{\varrho_i^{pj}\}_{j=1}^{4}$, $\varrho_i^p = \{\varrho_i^{nj}\}_{j=1}^{4}$ represent the viewpoint pedestrian and non-pedestrian, and the labels are shown in Fig. 1.

As shown in Fig. 1(b), WA-CNN employs feature maps $\boldsymbol{\psi} \in \mathbb{R}^{10\times5\times256}$ as input by stacking four convolutional layers (WA-conv1 to WA-conv4), one max-pool and three full-connected layers (WA-fc1 to WA-fc3), and the detailed setting is shown in Fig. 1(b). For all these layers, we utilize the rectified linear function [13] as the activation function.

As shown in Fig. 1(b), to strengthen the discriminant validity of intra-class, WA-CNN associates the metric codes which are generated by MC-net,

$$\begin{aligned}\boldsymbol{H}^{(L-1)} = ReLu(&\boldsymbol{W}^{(wa)}\boldsymbol{H}^{(L-2)} + \boldsymbol{b}^{(wa)} \\ &+ \boldsymbol{W}^{(mc)}\boldsymbol{\vartheta} + \boldsymbol{b}^{(mc)}),\end{aligned} \tag{12}$$

where $\boldsymbol{H}^{(L)}$ is the top-layer feature vector of WA-CNN, $\boldsymbol{W}^{(wa)}$, $\boldsymbol{b}^{(wa)}$ and $\boldsymbol{W}^{(mc)}$, $\boldsymbol{b}^{(mc)}$ are the parameter matrices corresponding to the two networks respectively.

We use

$$\begin{aligned} E^{(WA)} &= -\sum_{i=1}^{K} \log p\left(y_i, \boldsymbol{\varrho}^p, \boldsymbol{\varrho}^n | \boldsymbol{\psi}, \boldsymbol{\vartheta}\right) \\ &= -y \log p\left(y | \boldsymbol{\psi}, \boldsymbol{\vartheta}\right) - \sum_{i=1}^{4} \boldsymbol{\varrho}^{pi} \log p\left(\boldsymbol{\varrho}^p | \boldsymbol{\psi}, \boldsymbol{\vartheta}\right) \\ &\quad - \sum_{j=1}^{4} \boldsymbol{\varrho}^{nj} \log p\left(\boldsymbol{\varrho}^n | \boldsymbol{\psi}, \boldsymbol{\vartheta}\right), \end{aligned} \tag{13}$$

as the loss function and the loss function is expand to three parts, the main pedestrian, the viewpoint pedestrian and the non-pedestrian. The main task is to predict the pedestrian label y. $\boldsymbol{\varrho}^{pi}$, $\boldsymbol{\varrho}^{nj}$ are the i-th pedestrian estimations and the j-th non-pedestrian estimations. $p\left(y|\boldsymbol{\psi}, \boldsymbol{\vartheta}\right)$, $p\left(\boldsymbol{\varrho}^p|\boldsymbol{\psi}, \boldsymbol{\vartheta}\right)$, $p\left(\boldsymbol{\varrho}^n|\boldsymbol{\psi}, \boldsymbol{\vartheta}\right)$ are modeled by softmax functions.

3.2 The Training of WA-CNN

Because the main task is to predict the pedestrian label, the others are the auxiliary tasks. Thus, in the phase of training, we reformulate Eq. (11) using ω and ε to associate multiple tasks by a weighted strategy as the following

$$\begin{aligned} E^{(WA)} &= -y \log p\left(y | \boldsymbol{\psi}, \boldsymbol{\vartheta}\right) - \sum_{i=1}^{4} \omega_i \boldsymbol{\varrho}^{pi} \log p\left(\boldsymbol{\varrho}^p | \boldsymbol{\psi}, \boldsymbol{\vartheta}\right) \\ &\quad - \sum_{j=1}^{4} \varepsilon_j \boldsymbol{\varrho}^{nj} \log p\left(\boldsymbol{\varrho}^n | \boldsymbol{\psi}, \boldsymbol{\vartheta}\right). \end{aligned} \tag{14}$$

In our work, ω and ε can be values between zero and one. we set $\forall \omega_i = 0.1,\ i = 1, 2, ...4$, $\forall \varepsilon_j = 0.1,\ j = 1, 2, ...4$ simply.

With the training set $\boldsymbol{D} = \{(\boldsymbol{\psi}, y'_i)\}_{i=1}^{K}$, WA-CNN is trained to reinforce the distance of inter-class, further.

4 Overview on Our Method

Figure 1 shows our pipeline of pedestrian detection, where VGG16 (conv1–conv3) with RPN extract the candidate regions from the images with arbitrary image size. The generated feature maps of candidate regions $\boldsymbol{\psi} \in \mathbb{R}^{10\times5\times256}$ will be reconstituted to the set $\boldsymbol{B} = \{(\boldsymbol{\psi}, \boldsymbol{\alpha}_i)\}_{i=1}^{K}$ and $\boldsymbol{D} = \{(\boldsymbol{\psi}, y'_i)\}_{i=1}^{K}$ with the labels, and they will be the training sets for our associated work network.

Our associated work network contains two network, MC-net and WA-CNN. WA-CNN can be seen a network to reinforce the distance of inter-class, on the contrary, the Mc-net plays the role in enhancing the instance difference of intra-class.

As show in Fig. 2, with the training set $\boldsymbol{B} = \{(\boldsymbol{\psi}, \boldsymbol{\alpha}_i)\}_{i=1}^{K}$ which contain the labeled viewpoint pedestrian and non-pedestrian images, MC-net learn the template parameter $\left[\left(\boldsymbol{W}^{\vartheta(L)} \ltimes \boldsymbol{\phi}\right), \left(\boldsymbol{b}^{\vartheta(L)} \ltimes \boldsymbol{b}\right)\right]$. After the training, it codes the feature maps $\boldsymbol{\psi}$ with template parameter, and the output $\boldsymbol{\vartheta}$ is a comparability determination between the input map and the template parameter.

Finally, the outputs of MC-net and WA-CNN are jointly learned by the two full-connected layers of our network. And the weighted loss function is designed to accomplish the detection task with the joint features.

5 Experiments

To evaluate the performance of our detector on Caltech-Test [7] and INRIA [8] datasets, the evaluation protocol is following with [7] strictly.

The training data generated by transferring scene attribute information from existing background scene segmentation databases to seventeen attributes in pedestrian dataset by TA-CNN [24]. We only use eight attributes (showing in Table 1) as the training data, and the data are reconstituted into two parts: the viewpoint pedestrian (left, right, front, back) and the non-pedestrian (tree, car, road and building). Note that our network does not employ any motion and context information.

For Caltech-test reasonable subset, all results of our network are obtained by training on the reconstituted training data and evaluating on Caltech-Test (set06–set10). And, to evaluate the generalization capacity of the our network, we report overall results on INRIA-test in this section. All results of our network are obtained by training on reconstituted training data and evaluating on INRIA-test.

Hereinafter, RPN will be fine-turning in our network and WA-CNN and MC-net which are the brand-new network, will be evaluated on the performance and effectiveness (Table 2).

Table 2. The runtime and performance on Caltech.

Method	Window process	Hardware	Time/img(s)	MR (%)
LDCF	-	CPU	0.6	24.8
CCF	-	Titan Z GPU	13	17.3
Ours	RPN	GeForce GTX 1080 × 2	**0.9**	**13.74**

5.1 Effectiveness of Different Components in WA-CNN

Under the framework of deep neural networks, we compare the result of our network (WA-CNN+non-linear MC-net) with WA-CNN+linear MC-net and WA-CNN without MC-net to verify the capacity of MC-net in representing multi-class.

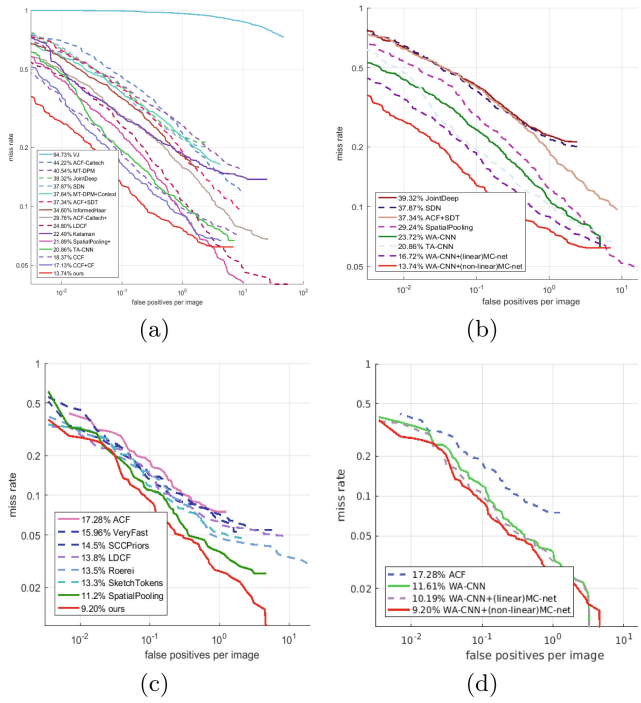

Fig. 3. Results on Caltech-test reasonable subset and INRIA dataset: (a), (c): Overall performance (b), (d): Log-average miss rate reduction procedure.

Caltech-Test reasonable subset: we systematically study the effectiveness of different components on our network. After the training, WA-CNN without MC-net gets **23.71**% miss rate. With this baseline, we implement WA-CNN+linear MC-net. As shown in Fig. 3(b), WA-CNN+linear MC-net gets **16.72**% miss rate, and it gets **7** % improvement. To verify the capacity of the non-linear MC-net in our network, we re-train MC-net with non-linear activation function. The result shows in Fig. 3(b), MC-net (non-linear) achieves **13.74**% performance, and improves **3.02**% than linear MC-net. The result is also compared with the other deep models: JointDeep, SDN, ACF+SDT, SpatialPooling, TA-CNN, and our method gets the lowest miss rate.

INRIA: WA-CNN without MC-net gets **11.61**% miss rate and WA-CNN+linear MC-net gets **10.19**% miss rate. MC-net (non-linear) improves **9.20**% performance, as shown in Fig. 3(d). The results show that our network has a good capacity on the generalization.

5.2 Comparisons with State-of-the-Art Methods

Finally, the results of our network with existing best-performing methods which contain handcrafted features and deep neural networks are evaluated.

Caltech-Test reasonable subset: we compare the result of our network with existing best-performing methods, including VJ [25], HOG [4], ACF-Caltech [5], MT-DPM [29], MTDPM+Context [29], JointDeep [16], SDN [11], ACF+SDT [20], InformedHaar [33], ACF-Caltech+ [14], SpatialPooling [19], LDCF [14], Katamari [3], SpatialPooling+ [18],TA-CNN [24],CCF [30], CCF+CF [30]. Figure 3(a) reports the results, and our method achieves the smallest miss rate (**13.74**%) compared to all existing methods.

INRIA: We compare the result of our network with existing best-performing methods, including ACF, VeryFast [1], SCCpriors [31], LDCF, Roerei Z [2], SketchTokens [10], SpatialPooling, and parts method mentioned in Sect. 4.1. As shown in Fig. 3(c), our method achieves the lowest miss rate.

Fig. 4. Compares with ACF, our network show a good performance in hard negatives mining, the scores in hard negatives is discriminative than ACF.

5.3 Evaluation on Hard Negatives Mining

In order to evaluate our method on hard negatives mining intuitively, we chose twenty images (hard negatives) which are difficult to other methods in Caltech dataset. The scores of ACF are as our baseline in the evaluation, and the scores of our network are calculated by softmax functions. To be fair, the scores of ACF are normalized to [−1,1]. Figure 4 shows that our method have a good performance to mine the hard negatives.

6 Conclusions

In this paper, with the plenty negative and positive samples, MC-net and WA-CNN are introduced to associated work for mining the hard negatives in pedestrian detection. They enforce the intra-class and inter-class differences using the properties of low-level and high-level features in a CNN model. Under the network, the problem of input size was alleviated by a flexible using on RPN. Extensive experiments demonstrate the effectiveness of the proposed method.

References

1. Benenson, R., Mathias, M., Timofte, R., Van Gool, L.: Pedestrian detection at 100 frames per second. In: CVPR, pp. 2903–2910. IEEE (2012)
2. Benenson, R., Mathias, M., Tuytelaars, T., Van Gool, L.: Seeking the strongest rigid detector. In: CVPR, pp. 3666–3673 (2013)
3. Benenson, R., Omran, M., Hosang, J., Schiele, B.: Ten years of pedestrian detection, what have we learned? In: Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8926, pp. 613–627. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16181-5_47
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. CVPR **1**, 886–893 (2005)
5. Dollár, P., Appel, R., Belongie, S., Perona, P.: Fast feature pyramids for object detection. TPAMI **36**(8), 1532–1545 (2014)
6. Dollar, P., Tu, Z., Perona, P., Belongie, S.: Integral channel features. In: BMVC (2009)
7. Dollar, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: an evaluation of the state of the art. TPAMI **34**(4), 743–761 (2012)
8. Ess, A., Leibe, B., Van Gool, L.: Depth and appearance for mobile scene analysis. In: ICCV, pp. 1–8 (2007)
9. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8691, pp. 346–361. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10578-9_23
10. Lim, J.J., Zitnick, C.L., Dollár, P.: Sketch tokens: a learned mid-level representation for contour and object detection. In: CVPR, pp. 3158–3165 (2013)
11. Luo, P., Tian, Y., Wang, X., Tang, X.: Switchable deep network for pedestrian detection. In: CVPR, pp. 899–906 (2014)
12. Murray, N., Perronnin, F.: Generalized max pooling. In: CVPR, pp. 2473–2480 (2014)
13. Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: ICML, pp. 807–814 (2010)
14. Nam, W., Dollár, P., Han, J.H.: Local decorrelation for improved pedestrian detection. In: NIPS, pp. 424–432 (2014)
15. Ouyang, W., Wang, X.: A discriminative deep model for pedestrian detection with occlusion handling. In: CVPR, pp. 3258–3265 (2012)
16. Ouyang, W., Wang, X.: Joint deep learning for pedestrian detection. In: ICCV, pp. 2056–2063 (2013)
17. Ouyang, W., Zeng, X., Wang, X.: Modeling mutual visibility relationship in pedestrian detection. In: CVPR, pp. 3222–3229 (2013)

18. Paisitkriangkrai, S., Shen, C.: Pedestrian detection with spatially pooled features and structured ensemble learning. In: TPAMI, pp. 1243–1257 (2016)
19. Paisitkriangkrai, S., Shen, C., van den Hengel, A.: Strengthening the effectiveness of pedestrian detection with spatially pooled features. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8692, pp. 546–561. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10593-2_36
20. Park, D., Zitnick, C.L., Ramanan, D., Dollár, P.: Exploring weak stabilization for motion feature extraction. In: CVPR, pp. 2882–2889 (2013)
21. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: TPAMI, p. 1 (2016)
22. Sermanet, P., Kavukcuoglu, K., Chintala, S., LeCun, Y.: Pedestrian detection with unsupervised multi-stage feature learning. In: CVPR, pp. 3626–3633 (2013)
23. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
24. Tian, Y., Luo, P., Wang, X., Tang, X.: Pedestrian detection aided by deep learning semantic tasks. In: CVPR, pp. 5079–5087 (2015)
25. Viola, P., Jones, M.J.: Robust real-time face detection. IJCV **57**(2), 137–154 (2004)
26. Viola, P., Jones, M.J., Snow, D.: Detecting pedestrians using patterns of motion and appearance. IJCV **63**(2), 153–161 (2005)
27. Wang, L., Ouyang, W., Wang, X., Lu, H.: Visual tracking with fully convolutional networks. In: ICCV, pp. 3119–3127 (2015)
28. Wang, X., Han, T.X., Yan, S.: An HOG-LBP human detector with partial occlusion handling. In: ICCV, pp. 32–39. IEEE (2009)
29. Yan, J., Zhang, X., Lei, Z., Liao, S., Li, S.Z.: Robust multi-resolution pedestrian detection in traffic scenes. In: CVPR, pp. 3033–3040 (2013)
30. Yang, B., Yan, J., Lei, Z., Li, S.Z.: Convolutional channel features. In: ICCV, pp. 82–90 (2015)
31. Yang, Y., Wang, Z., Wu, F.: Exploring prior knowledge for pedestrian detection. In: BMVC, pp. 1–12 (2015)
32. Zhang, L., Lin, L., Liang, X., He, K.: Is faster R-CNN doing well for pedestrian detection? In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 443–457. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_28
33. Zhang, S., Bauckhage, C., Cremers, A.B.: Informed Haar-like features improve pedestrian detection. In: CVPR, pp. 947–954 (2014)