Proceedings of the 2nd World Congress on Electrical Engineering and Computer Systems and Science (EECSS'16)
Budapest, Hungary – August 16 – 17, 2016
Paper No. MVML 109
DOI: 10.11159/mvml16.109

Extended Joint Deep Learning for Pedestrian Detection

Dae Jin Jo, Hyeon Seok Yang, Young Shik Moon[†]
Department of Computer Science and Engineering, Hanyang University
55 Hanyangdaehak-ro, Sangnok-gu, Ansan, Gyeonggi-do, 15588, Republic of Korea
djjo@visionlab.or.kr; hsyang@visionlab.or.kr; ysmoon@hanyang.ac.kr

Abstract **-** In this paper, we propose an extended version of Unified Deep Network (UDN). The Extended UDN (EUDN) uses multiple deformation models that operate independently of each other and mixture of the responses of the models to estimate the detection label. The deformation models of the EUDN jointly learned in order to complement each other through penalized in-diversity loss measured from the average correlation between the models. In our experiments, we show that combining independently the deformation models (which are even if worse than existing one) reduces the error in the manner similar to the ensemble learning, and considering diversity of the individual models is more effective without considering diversity. Our approach is evaluated on the Caltech datasets and achieves better performance than the UDN.

Keywords**:** pedestrian detection, deep learning, unified deep network, ensemble learning

1. Introduction

Detecting pedestrians from images is an important topic in computer vision with many fundamental applications in automotive safety, robotics, and video surveillance. The wide variety of appearances of pedestrians due to body pose, occlusions, clothing, cluttered backgrounds, and articulation makes this task challenging.

Several approaches have been proposed in past decades to handle these challenges. The approaches can be separated into three groups. The first group of approaches is based on using discriminative feature and suitable classifier (HOG [2], ACF [3]). The second is based on using deformable part model [4], and the third is based on handling occlusion [5].

Recently, with the progress of deep learning, convolutional neural networks have been shown to achieve good performance in computer vision problems. Unified Deep Network (UDN) [1], which is also a type of CNN, unifies the approaches we described above into a deep model and shows successful results. However, the network uses only one deformation model.

In this paper, inspired by the work of Felzenszwalb [4], we extend the UDN to have more deformation models and propose a method to fuse the results for models in the manner similar to ensemble method. Moreover, our network is trained by considering the diversity among the models.

This paper is organized as follow: in Section 2 we briefly introduce the UDN. We describe our method in Section 3. The experimental results are described in Section 4, then final conclusions are presented in Section 5.

2. Unified Deep Network

UDN, which is specified in Table 1, is proposed to jointly learn feature extraction, parts deformation, occlusions, and person-to-person relations. The second convolutional layer of UDN convolves the feature maps, obtained from the first convolutional layer, with 20 kernels of different shapes and different sizes. Then the feature maps are used to calculate the part scores in deformation layer. These part scores are divided into three hierarchical levels, designed to consider mutual visibility relationship. The scores are fed to each level of visibility reasoning layer and propagated to estimate detection label. In order to prevent disturbing of the imperfect part scores in the previous level, the visibility reasoning layer includes extra nodes at level 2 and 3 (We denote the number of the nodes of the visibility reasoning layer as the number of parts plus the number of the extra nodes in the table). The output of the layer at level 3 is fed to a two-way softmax which produces a distribution over the two class labels.

[†] Corresponding Author

Table 1: Structure of the UDN.

Layer	1	2	3	4	5				6
Type	Convolutional	Average Pooling	Convolutional	Deformation	Visibility Reasoning				Output
Specification	64 × 9 × 9	4 × 4	20 part kernels	20 part scores	level	1	2	3	softmax
					nodes	6	7+7	7+7	

3. Proposed Method

3.1. Extended Unified Deep Network (EUDN)

The proposed network is presented in Fig 1. The network can have N deformation models that operate independently of each other. A deformation model is separated into two parts: deformation part model and visibility reasoning model. The deformation part models can have different number and shape of convolutional kernels and the visibility reasoning models can also be composed independently of each other in the different visibility relationship. Each deformation model provides two responses, positive and negative, for a given image. The detection label estimation y is obtained through combination of the responses for each class.

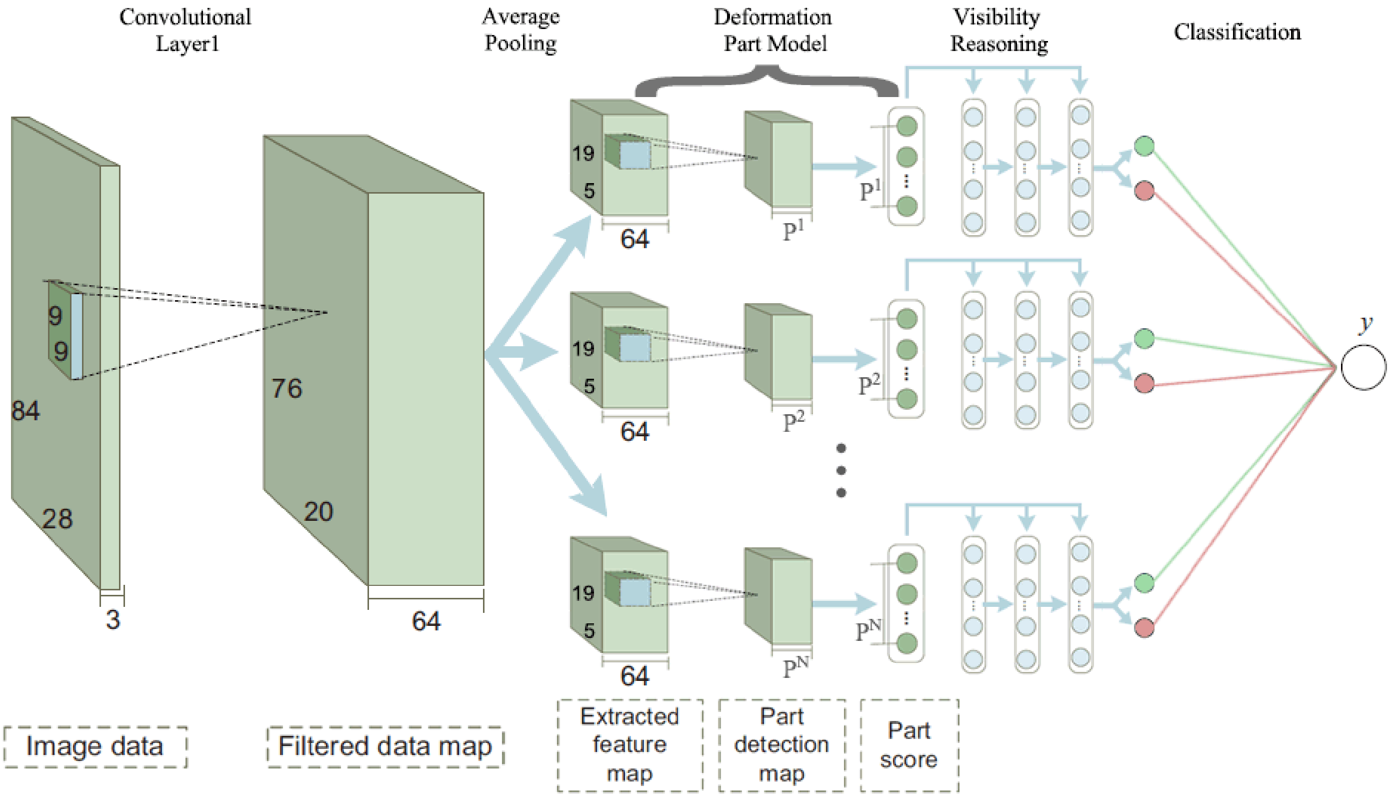


Fig. 1: Overview of our deep network model.

3.2. Visibility Reasoning and Classification

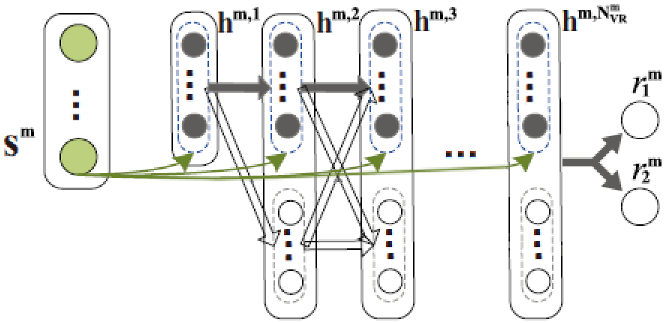
The m-th deformation part model provides P^m part scores $\mathbf{s}^m = \{s_1^m, \dots, s_{P^m}^m\}$ for $m = 1, \dots, N$. $\mathbf{s}^m$ is then used for visibility reasoning and for obtaining the responses for each class. Fig. 2 shows the visibility reasoning model. The score and visibility of j-th part at level l of m-th model are denoted as $s_j^{m,l}$ and $h_j^{m,l}$, respectively. The visibility and the number of parts at level l of m-th model are denoted as $\mathbf{h}^{m,l} = [h_1^{m,l}, \dots, h_{p_l^m}^{m,l}]$ and P_l^{m} respectively. Given $\mathbf{s}^m$, the model for inference is as follows:

$$
\begin{gathered}
\tilde{h}_j^{\mathrm{m},1} = \sigma(b_j^{m,1} + g_j^{m,1} s_j^{m,1}) \\
\tilde{h}_j^{m,l+1} = \sigma\big(\tilde{\mathbf{h}}^{m,l}\mathbf{w}_{*,j}^{m,l} + b_j^{m,l+1} + g_j^{m,l+1} s_j^{m,l+1}\big), \qquad l = 1, \dots, (N_{VR}^m - 1) \\
r_i^m = \sigma(\,\tilde{\mathbf{h}}^{m,N_{VR}^m}\mathbf{w}_i^{m,res} + b_i^{m,res})
\end{gathered} \tag{1}
$$

where σ is the sigmoid function, and N_{VR}^m is the number of visibility reasoning layers. $g_j^{m,l}, s_j^{m,l}, \mathbf{W}^{m,l}, \mathbf{w}^{m,res}$, and b^m are parameters to be learned. The extra hidden nodes, represented by white circles in Fig. 2, can be used in the same reason of the UDN. They do not use detection scores and have the term $g_j^{m,l+1} s_j^{m,l+1} = 0$ in Eq. (1), while the hidden nodes with the term $g_j^{m,l+1} s_j^{m,l+1} \neq 0$ are represented by gray circles.

The deformation models provide N responses $\mathbf{r}_i = \{r_i^1, \dots, r_i^N\}$ for class i. A softmax function is used to find the predicted probability that a given input image includes pedestrians. In a detection problem, the function can be expressed as in Eq. (2) where $\mathbf{w}_i^{cls}$ and b_i^{cls} are the parameters to be learned. The probability that a given input image does not include any pedestrian is given by $1 - \tilde{y}$.

$$
\tilde{y} = \frac{\exp(\mathbf{r}_1\mathbf{w}_1^{cls} + b_1^{cls})}{\exp(\mathbf{r}_1\mathbf{w}_1^{cls} + b_1^{cls}) + \exp(\mathbf{r}_2\mathbf{w}_2^{cls} + b_2^{cls})} \tag{2}
$$



$\mathbf{s}^{\mathbf{m}}$
$\mathbf{h}^{\mathbf{m},1}$
$\mathbf{h}^{\mathbf{m},2}$
$\mathbf{h}^{\mathbf{m},3}$
...
$\mathbf{h}^{\mathbf{m},N_{VR}^m}$
$r_1^{\mathbf{m}}$
$r_2^{\mathbf{m}}$

Fig. 2: The visibility reasoning model.

3.3. Considering Diversity of the Deformation Models

The EUDN is an ensemble classifier that combines multiple deformation models. It is well known that the performance of the combination of an ensemble depends on the diversity among its individual models. One of the measures of the diversity is the correlation factor among the outputs of the models.

The total error of a classifier can be expressed as the sum of two error, $E_{\text{total}} = E_{add} + E_{bayes}$, where E_{bayes} is the optimal Bayes error and E_{add} is the added error that comes from the obtained boundary of the classifier. In [7], they show that positively correlated classifiers only slightly reduce the added error, uncorrelated classifiers reduce the added error by a factor of 1/*N*, and negatively correlated classifiers reduce the error even further. In this study, we adopt this scheme to train our network.

The EUDN is trained with the proposed loss function that is penalized by the in-diversity measured from the average correlation. The loss function L is defined as follows:

$$
L = L^{CE} + \sum_i \lambda_i L_i^{inDiv} \tag{3}
$$

$$
L^{CE} = -\frac{1}{K}\sum_k^K y_k^{gnd}\log(\tilde{y}_k) + (1 - y_k^{gnd})\log(1 - \tilde{y}_k) \tag{4}
$$

$$
L_i^{inDiv} = \frac{1}{N(N-1)}\sum_{m=1}^N \sum_{m \neq l} Corr(r_i^m, r_i^l) \tag{5}
$$

As shown in Eq. (3), our loss function is a combination of cross entropy L^{CE} and in-diversity loss L^{inDiv} for each response of the deformation models. These in-diversity losses are multiplied by the relative importance weights λ_i. The cross entropy is defined as Eq. (4) where K is the number of samples, y_k^{gnd} is the ground-truth label of k-th sample that $y_k^{gnd} \in \{0, 1\}$, and $\tilde{y}$ is the estimation of detection label. The in-diversity loss, defined as Eq. (5), is measured on the average correlation. A response for class *i* of *m*-th model is denoted as r_i^m.

In order to learn the parameters of each layer in Fig. 1, the prediction error is back propagated through $\mathbf{r}_i$. The gradient for $\mathbf{r}_i$ is as follows:

$$\frac{\partial L}{\partial r_i^{k,m}} = \frac{\partial L_{CE}}{\partial r_i^{k,m}} + \lambda_i \frac{\partial L_i^{inDiv}}{\partial r_i^{k,m}} \qquad (6)$$

$$\frac{\partial L_i^{inDiv}}{\partial r_i^{k,m}} = \frac{1}{N(N-1)} \sum_{P=1}^{N} \sum_{P \neq l} \frac{\partial Corr_{P,l}}{\partial r_i^{k,m}} \qquad (7)$$

where *m* and *k* denote the *m*-th model and the *k*-th sample, respectively.

4. Experiments

The proposed method is evaluated on the Caltech dataset [6] in the same way as [1]. In the deep learning model, the batch size is 60 with the relative importance weight $\lambda_1 = \lambda_2 = 0.5$. In this experiment, we use three different deformation models that are considered different views of the pedestrians. We investigate the influence of the proposed loss function by testing various designs of EUDN.

The compared approaches are HOG [2], ACF[3], DPM [4], HogLbp [5], ConvNet-U-MS [8] and UDN [1]. Existing approaches use various features, deformable part models and different learning approaches.

4.1. Deformation Models

The part models used in this experiment are shown Fig. 3. In the figure, (a) is the model used in UDN and (b) and (c) are the models that are designed to complement existing one and represent pedestrian's side pose and scaled view, respectively. The visibility reasoning models are specified in Table 2. (a) is the model used in UDN and (b) and (c) are additional models designed in the similar manner as the UDN. The numbers of nodes of the layers are denoted by the number of the parts plus the number of the extra nodes.

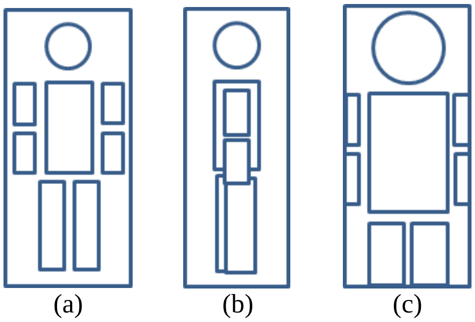


Fig. 3: The part models.

Table 2: Structure of the visibility reasoning models.

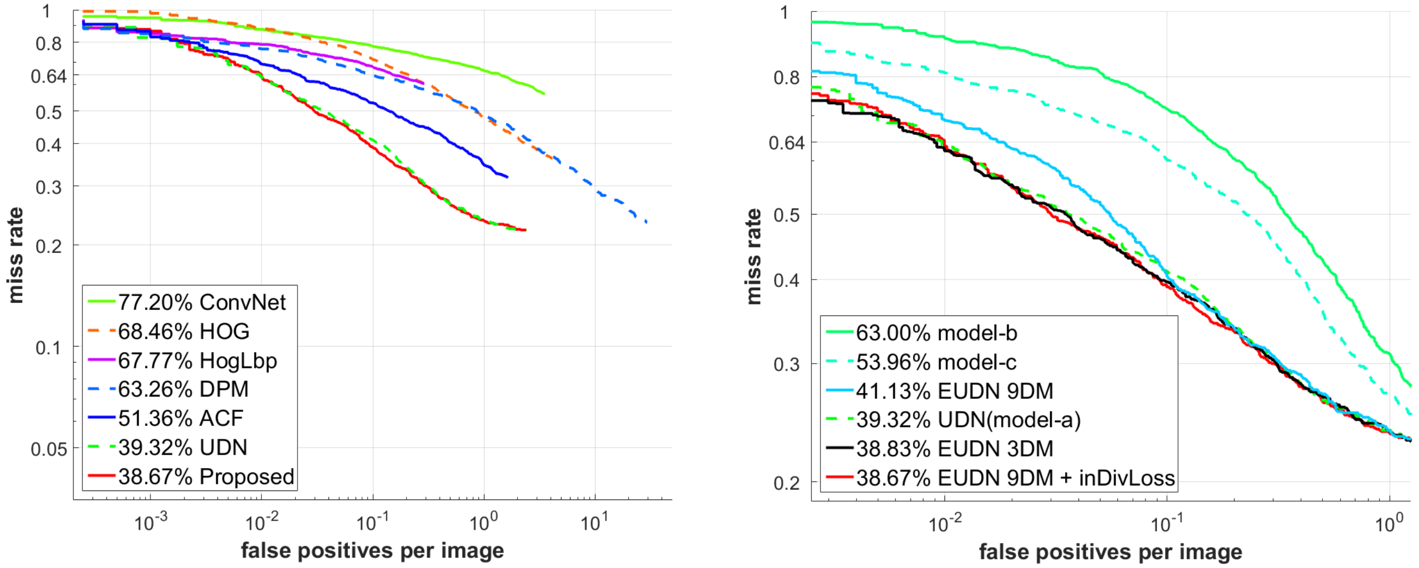
Model	(a)	(b)	(c)
number of parts	20	13	15
number of parts for level1	6	6	7
number of parts for level2	7 + 7	3 + 3	5 + 5
number of parts for level3	7 + 7	4 + 4	3 + 3

4.2. Results

To evaluate the performance, the Caltech-Train dataset is used to train our model. At the training stage, there are approximately 60,000 negative samples and 4,000 positive samples from the Caltech-Train dataset. Fig. 4 (a) shows the overall experimental results on the Caltech-Test. The proposed method reduces the average miss rate by about 0.7%, compared with UDN.

In this experiment, we investigate various designs of proposed deep models on this dataset. Comparisons are shown in Fig. 4 (b). We evaluate the performance of each single deformation model. The model-*a* is the UDN, and model-*b* and model-*c* are the additional models we described in Section 4.1. As the results show, the two models are underperformed relative to the existing model. However, the combined model (denoted as EUDN 3DM) shows better performance than the UDN.

To investigate the influence of proposed loss function in Section 3.3, we compare two models that are trained with and without the in-diversity loss. We design the models to have nine deformation models (denoted as EUDN 9DM), three sets of the three models. As shown in the result, the model that trained without the in-diversity loss is rather worse than the EUDN 3DM, while the other model is better. This means that if we do not consider the diversity of multiple models, there are more redundant parameters to be learned; therefore the model would lead to overfitting. On the other hand, if we consider diversity of the models, the model would be more generalized model.



(a) (b)

Fig. 4: Comparison of the performances on the Caltech-Test dataset.
(a) performance comparison with existing methods. (b) proposed methods compared to UDN.

5. Conclusion

This paper proposes an extended unified deep model that jointly learns its multiple deformation models. By considering diversity among the models, the EUDN achieves better performance than the UDN on Caltech dataset. The experimental results reveal that the additional deformation models can be complemented each other and the proposed loss function makes the network, which has multiple independent models, learn more generalized model. We believe that the proposed network can be improved by using suitable input channels and by designing more elaborate deformation model. However, a limitation of the EUDN is that its deformation models should be designed manually. Future research can address this issue by jointly learning of deformation models that generated by modified network architecture.

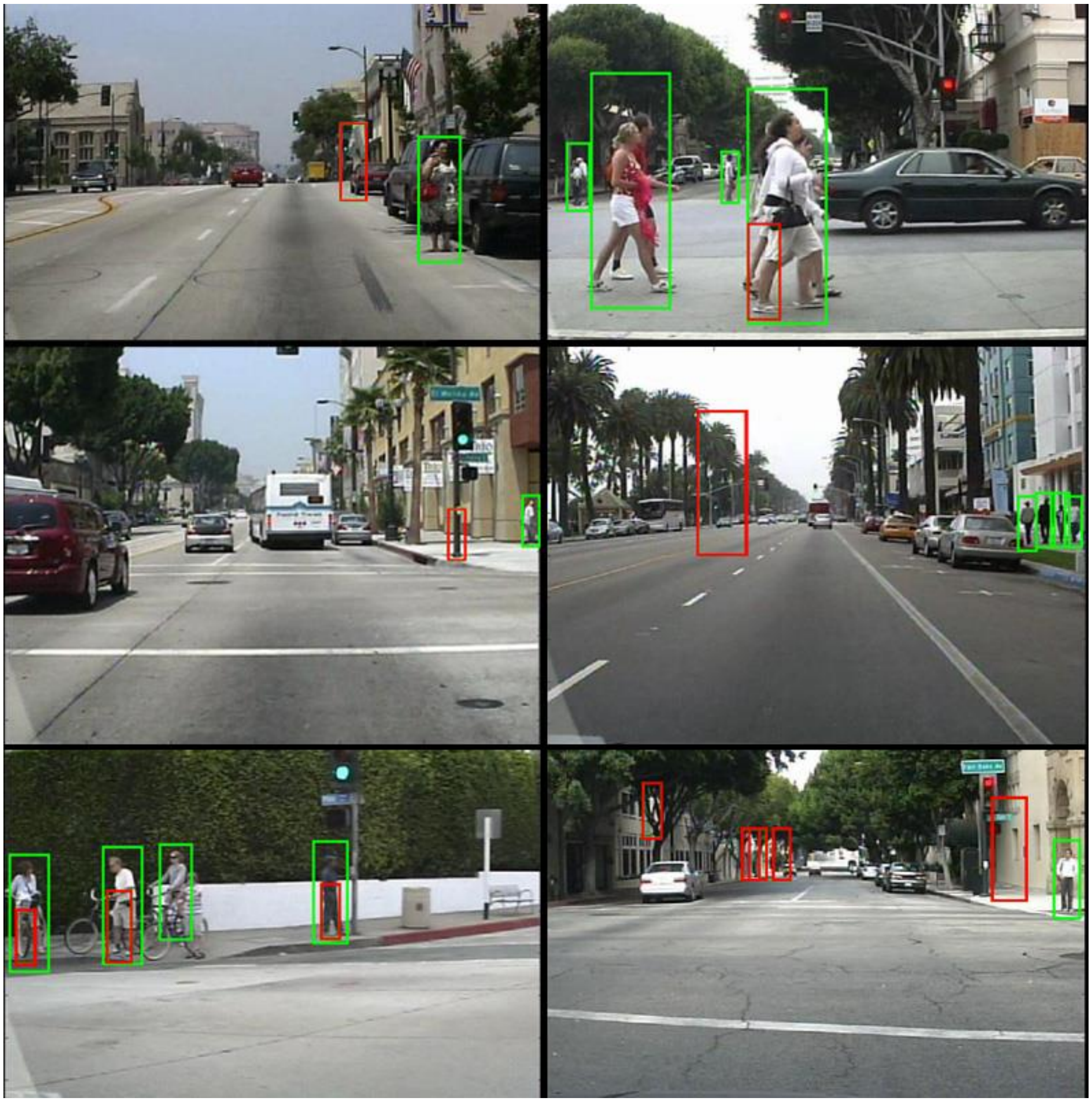


Fig. 5: Detection Examples. The green and red boxes are true and false positives, respectively.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (No. 2012002464).

References

[1] W. Ouyang and X. Wang, “Joint deep learning for pedestrian detection,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2056-2063, 2013.

[2] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 886-893, 2005.

[3] P. Dollár, R. Appel, S. Belongie and P. Perona, “Fast feature pyramids for object detection,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1532-1545, 2014.

[4] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part based models,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627-1645, 2010.

[5] X. Wang, X. Han and S. Yan, “An hog-lbp human detector with partial occlusion handling,” in *IEEE 12th International Conference on Computer Vision*, pp. 32-39, 2009.

[6] P. Dollár, C. Wojek, B. Schiele and P. Perona, “Pedestrian detection: an evaluation of the state of the art,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743-761, 2012.

[7] K. Tumer and J. Ghosh, “Linear and order statistics combiners for pattern classification,” in *Combining artificial neural nets*, A. Sharkey, Ed. London: Springer-Verlag, pp. 127-161, 1999.

[8] P. Sermanet, K. Kavukcuoglu, S. Chintala and Y. Lecun, “Pedestrian detection with unsupervised and multi-stage feature learning,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3626-3633, 2013.