Pedestrian Detection Based on Deep Convolutional Neural Network with Ensemble Inference Network

Hiroshi Fukui [1] Takayoshi Yamashita [1] Yuji Yamauchi [1] Hironobu Fujiyoshi [1] Hiroshi Murase[2]

***Abstract*— Pedestrian detection is an active research topic for driving assistance systems. To install pedestrian detection in a regular vehicle, however, there is a need to reduce its cost and ensure high accuracy. Although many approaches have been developed, vision-based methods of pedestrian detection are best suited to these requirements. In this paper, we propose the methods based on Convolutional Neural Networks (CNN) that achieves high accuracy in various fields. To achieve such generalization, our CNN-based method introduces Random Dropout and Ensemble Inference Network (EIN) to the training and classification processes, respectively. Random Dropout selects units that have a flexible rate, instead of the fixed rate in conventional Dropout. EIN constructs multiple networks that have different structures in fully connected layers. The proposed methods achieves comparable performance to state-of-the-art methods, even though the structure of the proposed methods are considerably simpler.**

I. INTRODUCTION

In driving assistance systems, object detection is generally conducted using Light Detection and Ranging (LIDAR) or vision-based approaches that use single or multiple frames recorded by an in-vehicle camera. LIDAR captures a three-dimensional point cloud from the light reflected by an object. Google automatic driving system employs LIDAR because it records high-quality depth information and is adequate in the research field. However, LIDAR is a very expensive technology for regular vehicles.

There are two approaches using in-vehicle cameras: methods based on a single frame and those based on multiple frames. Although single-frame methods enable real-time detection, it is required that improve the detection accuracy in various situations. On the other hand, multiple-frame techniques achieve high detection accuracy via the use of motion and context information. The motion information is extracted from the optical flow, and context information is estimated by superpixels. These techniques extract rich information, but are time consuming. In addition, the motion information requires corresponding points between frames to be detected. Single-frame methods can be combined with other approaches. For example, LIDAR-based approaches use single-frame pre-processing to reduce the search region. In this paper, we consider a single-frame method with the aim of popularizing driving assistance systems based on in-vehicle cameras.

The combination of Histograms of Oriented Gradients (HOG) features and Support Vector Machines (SVM) has commonly been used for pedestrian detection [1]. HOG features focuses on the gradient of the local region, and is robust to small variations in pose. Following Dalal work, several related approaches have been proposed [2] [3] [4] [5]. To handle large pose variations, Felzenszwalb proposed the Deformable Part Model (DPM) [2], which detects pedestrians with a whole pedestrian model and part regions. DPM attained the best performance in a pedestrian detection benchmark.

Pedestrian detection based on Deep Convolutional Neural Networks (CNN) achieved the high detection accuracy in the pedestrian detection benchmark [6]. The CNN proposed by LeCun has been attracting attention [7], and Krizhevsky applied CNN to object recognition [11]. The CNN employs a Rectified Linear Unit (ReLU) as an activation function, and uses Dropout to obtain generalization [8]. Dropout randomly removes a fixed ratio of units. This ratio is usually set to 50%, and the response value of selected units is zero. Different units are selected in each iteration of the training process. Although networks trained with Dropout exhibit improved generalization performance, this random selection is only applied to the training process.

In this paper, to achieve better generalization, we propose Random Dropout and Ensemble Inference Network (EIN) for the training and classification processes, respectively. Random Dropout selects units at random with a flexible rate, instead of the fixed rate used in conventional Dropout. EIN constructs multiple networks that have different structures in fully connected layers. In the remainder of this paper, we first describe some related work on conventional CNN. We then introduce the proposed Random Dropout and EIN method. Finally, we investigate the architecture of EIN through a series of experiments, and compare the performance of the proposed technique with that of state-of-the-art methods.

II. RELATED WORKS

Object detection is a fundamental topic in pattern recognition. Pedestrian detection based on boosted cascade classifiers was proposed by Snow [10], with Haar-like features used to extract differences between frames. Although this approach is as fast as boosted cascade-based face detection, it is not especially accurate.

Dalal et al. have proposed HOG features that capture gradient information of small regions [1]. These features are robust to variations in the appearance of pedestrians. The SVM classifier is trained using HOG vectors. The combination of HOG features and the SVM classifier has

[1]Chubu University, 1200 Matsumoto cho, Kasugai, Aichi, Japan. fhiro@vision.cs.chubu.ac.jp

[2]Nagoya University, Furo cho, Chikusa ku, Nagoya, Aichi, Japan.

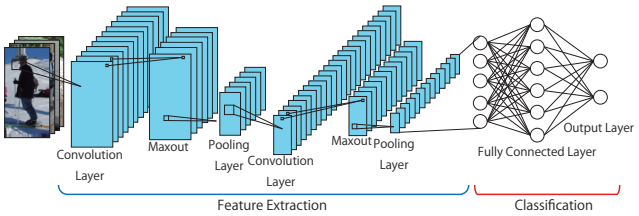


Fig. 1. Conventional CNN architecture

become popular in the field of pedestrian detection. There are many extensions to Dalal approach [3] [5] [4].

DPM archives high accuracy pedestrian detection that is robust to pose variations [2]. It estimates probabilities across the whole pedestrian region and some part regions at the same time, and uses these to detect pedestrian regions. The whole pedestrian model and each part region are based on HOG features extracted from several resolutions. DPM recorded the best performance in the pedestrian detection benchmark in 2010. In addition, DPM has been applied to point clouds captured from LIDAR [12]. Instead of HOG features, this point cloud-based DPM uses the number of points in each cell as a feature.

Conventional methods consist of feature extraction, which is manually designed, and trained classification, which is based on machine learning techniques. In particular, feature design requires human knowledge to ensure robustness. To overcome this problem, deep learning-based approaches have attracted attention. Deep learning, especially in CNN, has excellent feature representation and classification abilities. Krizhevsky has shown the potential of CNN in a 1000-class object recognition benchmark (ILSVRC). After this breakthrough, Deep Learning approaches have been applied to many problems, such as house number recognition, scene labeling, and object detection. Ouyang et al. proposed the concept of Joint Deep Learning, which is based on hierarchical neural networks. First, features are extracted from partial pedestrian regions, and these are then combined hierarchically. Joint Deep Learning is robust to pose variations, and recorded the top score in the Caltech pedestrian benchmark . R-CNN is one of the best detection approaches based on Deep Learning [13]. This method determines regions of interest in an image using the superpixel method, and extracts features using CNN. The extracted features are passed to an SVM that is trained for each object and, finally, the specific object position is detected. Although Joint Deep Learning and R-CNN are capable of highly accurate pedestrian detection, their networks have a complicated structure.

III. Convolutional Neural Network

As shown in Fig. 1, The CNN contains an alternate succession of convolutional layers and Pooling layers. This is based on the notion of local receptive fields discovered by Hubel [14]. There are several types of layers, including input layers, convolutional layers, pooling layers, and classification layers. Besides the raw data, each input layer also takes edges and normalized data. The convolutional layer has M kernels with size $Kx \times Ky$ that are filtered in order to input data. The filtered responses from all the input data are then subsampled in the pooling layer. Scherer [15] found that max pooling can lead to faster convergence and improved generalization, and Boureau [16] analyzed theoretical feature pooling. Max Pooling outputs the maximum value in certain regions, such as in a 2×2 pixel. Convolutional layers and pooling layers are laid alternately to create the deep network architecture. Finally, the classification layer outputs the probability of each class through a softmax connection of all weighted nodes in the previous layer.

CNN utilizes supervised learning in which filters are randomly initialized and updated through backpropagation [17]. The Backpropagation estimates the connected weights and minimize E by gradient descent as shown in Eq. (1) and Eq. (2).

$$E = \frac{1}{2}\sum_{n=1}^{N} E_n \tag{1}$$

$$\mathbf{W}^{(l)} \leftarrow \mathbf{W}^{(l)} + \Delta\mathbf{W}^{(l)} = \mathbf{W}^{(l)} - \eta\frac{\partial E_n}{\partial \mathbf{W}^{(l)}} \tag{2}$$

Note that $\{n|1, ..., N\}$ is the training sample. η is the training ratio, $\mathbf{W}^{(l)}$ is the weight that connects in layer l to the next layer $(l+1)$. The error for each training sample E_n is the sum of the differences between the output value and the label. $\Delta\mathbf{W}^{(l)}$ is represented as shown in Eq. (3).

$$\Delta\mathbf{W}^{(l)} = -\eta\delta^{(l)}\mathbf{y}^{(l-1)} \tag{3}$$

$$\delta^{(l)} = \mathbf{e}\phi(\mathbf{A}^{(l)}) \tag{4}$$

$$\mathbf{A}^{(l)} = \mathbf{W}^{(l)} \cdot \mathbf{y}^{(l-1)} \tag{5}$$

$\mathbf{y}^{(l-1)}$ is the output in the $(l-1)$th layer and $\mathbf{e}$ is the output nodes error. $\mathbf{A}^{(l-1)}$ is the accumulated value connected to node from all nodes in the $(l-1)$th layer. The local gradient descent is obtained by Eq. (4). The activation function ϕ may be a sigmoid, hyperbolic tangent, or ReLU [11]. The connected weights in the entire network are updated concurrently for a predetermined number of iterations, or until some convergence condition is satisfied.

Dropout is an efficient method to reduce over-fitting and improve generalization [8]. It randomly selects units with a probability of 50%. The features of the selected units are then used for optimization during each iteration of the training process. Dropout is also used for training in the fully connected layer.

IV. Proposed Method

We propose two techniques based on Dropout: Random Dropout for the training process, and Ensemble Inference Network (EIN) for the classification process. Details of these techniques are as follows.

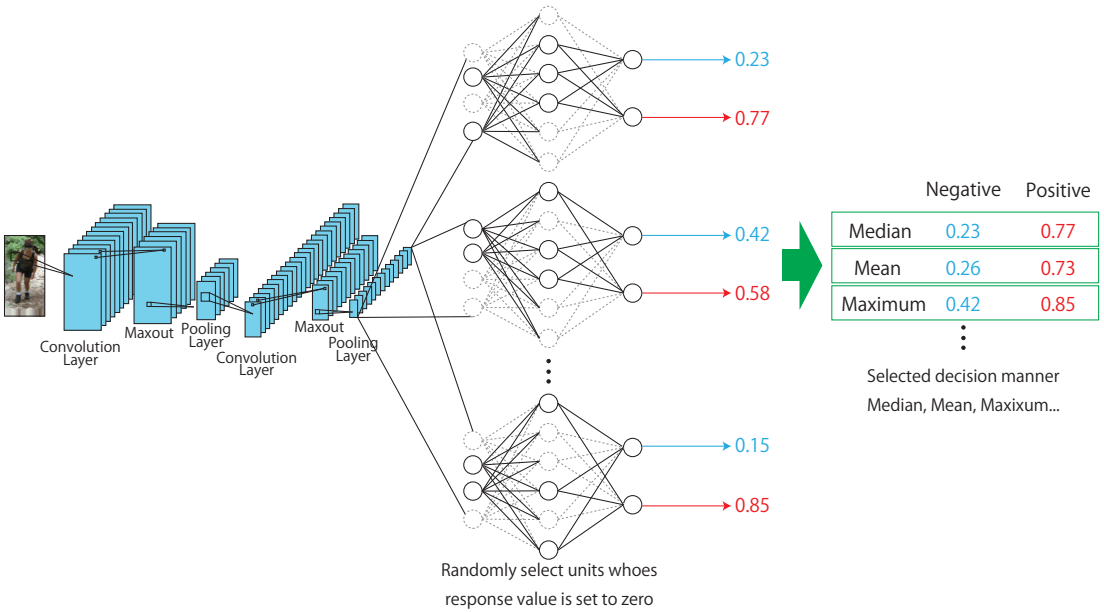


Fig. 3. Algorithm of EIN.

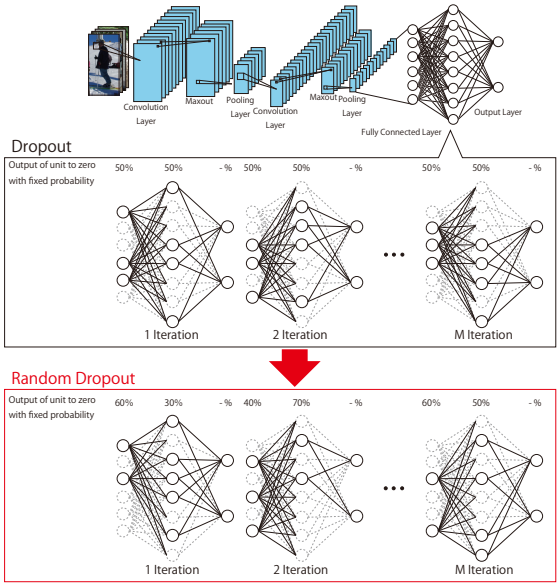


Fig. 2. Algorithm of conventional Dropout and Random Dropout.

A. Random Dropout

Conventional Dropout consists of setting the output of each hidden unit to zero with a fixed probability (usually 50%). The hidden units that are set to zero are randomly selected in each iteration. Dropout produces robustness by neglecting certain information from the layer below. We extend this Dropout technique by applying a random probability in each iteration to give Random Dropout. Figure 2 illustrates the update process in conventional Dropout and the proposed Random Dropout. Whereas conventional Dropout always sets 50% of the units to zero, Random Dropout applies a different ratio for each iteration. We first define the Random Dropout probability as between 30% and 70%. We set this ratio to 60% and 30% for each layer in the first iteration, then change this to 40% and 70% for each layer in the second iteration. For example, we set this ratio to 60% and 30% for each layer in the first iteration, then change this to 40% and 70% for each layer in the second iteration. We aim to obtain generalization using a flexible Random Dropout ratio.

B. Classifier using EIN

EIN is intended to remove connections from the previous layer, similar to Dropout, in the classification process. We randomly select units whose response value is set to zero with 50% probability. The classification process of conventional CNN feeds the input values to the trained network immediately. Unlike conventional CNN, EIN feeds the input value to the fully connected layers of the trained network at different times. At each time, some units of the fully connected layers are randomly set to zero. In other words, our method forms different networks from the original trained network by randomly selecting different units. We compare the decision manner based on median, mean or maximum for pedestrian detection independently in experiments. The steps of the EIN process are as follows.

1) Feature maps: First, the input image $\mathbf{I}$ is convoluted with filter $\mathbf{V}$, and feature maps are generated with activation function ϕ, as shown in Eq. (6).

$$h = \phi\left(\mathbf{V}^T\mathbf{I} + \mathbf{b}\right) \tag{6}$$

Here, $\mathbf{b}$ is a bias term. We employ the Maxout activation function [9], which selects the maximum value from K feature maps at each unit, as shown in Eq. (7).

$$h'_i = \max_{k\in[1,K]} h_{ik} \tag{7}$$

We utilize max pooling to select the maximum value in each certain region, and obtain robustness against small shape variations by resizing the feature map, as shown in Eq. (8).

$$h_i'' = \max_{p \in P_i} h_p' \tag{8}$$

Although units in the fully connected layer will change according to the random selection, the convolution layer and pooling layer remain the same. Thus, we can share the output values from the final pooling layer.

2) Structured by some networks: The response of randomly selected units is set to zero in the fully connected layer and classification layer, as shown in Eq.(9).

$$h_j^{(l)} = \phi\left(\mathbf{W}^{(l)}\mathbf{x} + \mathbf{b}^{(l)}\right) \cdot m_j^{(l)} \tag{9}$$

Note that the feature map obtained in Step 1 is defined as $\mathbf{x}$. $\mathbf{W^l}$ and b^l are the connection weights and bias of the lth layer, respectively. m controls the response value h_j^l, and is set to 0 or 1. When m is set to 0, the response value is 0, and when m is set to 1, the response value is 1. We construct N networks with randomly selected zero-response units. The probability of each class in network n is given by the soft-max function in Eq. (10).

$$O_{nc} = \frac{\exp\left(\mathbf{W}_j^{(L)}\mathbf{h}_j^{(L)} + b_j^{(L)}\right)}{\sum_{c=0}^{C}\exp\left(\mathbf{W}_c^{(L)}\mathbf{h}_c^{(L)} + b_c^{(L)}\right)} \tag{10}$$

3) Finaly output Classifier value: The final output class is obtained from the probability O_{nc} for each class in each network. All O_{nc} stored into probability set $\mathbf{s_c}$ and median is selected $\mathbf{S_c^{Median}}$ from them.

V. Pedestrian Detection using Proposed Method

For pedestrian detection, we employ sliding window approach to detect multiple pedestrians at different scale. In the detection process, it is necessary to distinguish whether pedestrians are present or not in a huge number of windows. We apply a HOG+SVM classifier to reduce the processing time to find pedestrian candidate regions and the proposed CNN is applied to them. The proposed CNN is applied to pedestrian candidate windows.

VI. Experiments

We evaluate the performance of Random Dropout and EIN using the Caltech Pedestrian Dataset and Daimler Mono Pedestrian Benchmark Dataset. In addition, we evaluate the efficiency of Random Dropout and several EIN architectures that have a different number of networks and different final output decisions (maximum or average). We also compare the proposed method with, CNN, HOG [1], HogLbp [3], LatSvm-V2 [2], VJ [10], DBN-Isol [25], ACF [26], ACF-Caltech [26], Pls [20], FPDW [22], ChuFtrs [18], CrossTalk [19], RandomForest [5], MultiResC [21], Roerei [23], MOCO [27], Joint Deep [6] and Switchable Deep Network [28] for Caltech Pedestrian Dataset. For the Daimler Mono Pedestrian Benchmark Dataset, we compare with CNN, HOG [1], Shapelet [29], HogLbp [3], LatSvm-V2 [2], VJ [10], RandomForest [5], MultiFtr [30], and MLS [4].

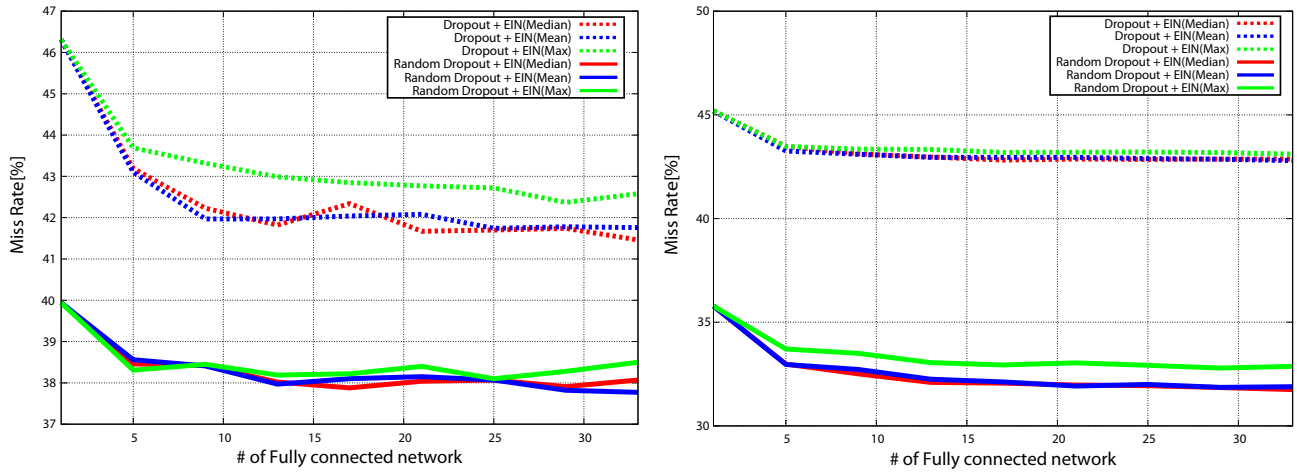
TABLE I
Using CNN structures

		Caltech Dataset	Daimler Dataset
Input		108 × 36 × 3	96 × 48 × 1
Layer 1	Weight Filter	40, 9 × 5	100, 5 × 3
	Max Pooling	2 × 2	2 × 2
	Maxout	2	2
Layer 2	Weight Filter	64, 5 × 3	80, 5 × 4
	Max Pooling	2 × 2	2 × 2
	Maxout	2	2
Layer 3	Weight Filter	32, 3 × 3	70, 5 × 4
	Max Pooling	2 × 2	2 × 2
	Maxout	2	2
Layer 4	# of Fully connected	1,000	1,000
Layer 5	# of Fully connected	500	500
Layer 6	# of Fully connected	100	100
Output	Softmax	2	2

The structures of CNN are shown in Table 1. The Caltech Pedestrian Dataset has 4,000 positive samples and 200,000 negative samples for training, and 8,273 test images. We augmented 101,808 positive samples from the originals by shifting, rotation, mirroring, and scaling. For the Daimler Mono Pedestrian Benchmark Dataset, we augmented 250,560 positive samples from 31,320 samples; the dataset also has 254,356 negative samples. The total iteration of updating parameters is 500,000 with 5 minibatchs and training ratio η is set to 0.01.

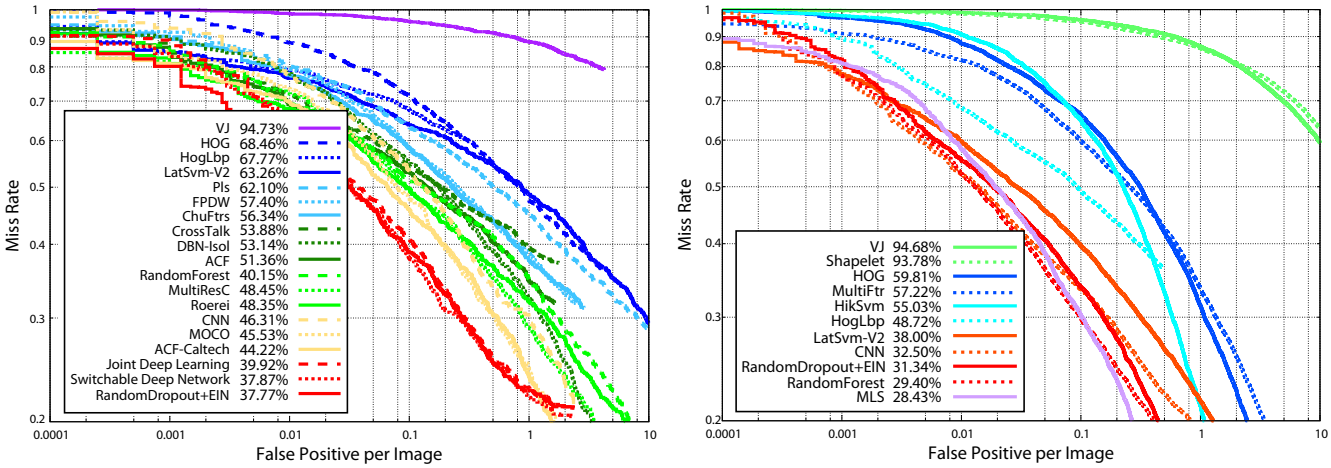
A. Performance comparison with the number of networks

First, we consider the best architecture of EIN by changing the number of networks and the determining way of final output. We evaluate the architectures of EIN that contain network from 1 to 33. Each network retained the same structure in the convolutional layer and pooling layer, but the fully connected layers varied according to the random selection of units. When the number of networks is 1, we have a conventional network. We select the mean probability from all networks. Figures 4(a) and (b) show the evaluation results using the Caltech Pedestrian Dataset and Daimler Mono Pedestrian Benchmark Dataset, respectively. First, the difference between the solid and broken lines indicates the performance of Dropout and Random Dropout. In Caltech Pedestrian Dataset case, we can see that Random Dropout improves the accuracy by about 6%. The architecture that has 33 networks with mean selection achieves the best accuracy, a miss rate of 37.77% with the Caltech Pedestrian Dataset. From this evaluation, EIN with multiple networks and different structures in the fully connected layers obtains better detection accuracy. In Daimler Mono Pedestrian Benchmark Dataset, whereas the miss rate of CNN are 35.78%, our methods significantly improves to 31.34% at 33 ensemble networks with mean decision manner.



(a) Caltech Pedestrian Dataset (b) Daimler Mono Pedestrian Bemchmark Dataset

Fig. 4. Performance comparison with the number of networks



(a) Caltech Pedestrian Dataset (b) Daimler Mono Pedestrian Benchmark Dataset

Fig. 5. Compare proposed methods and conventional methods

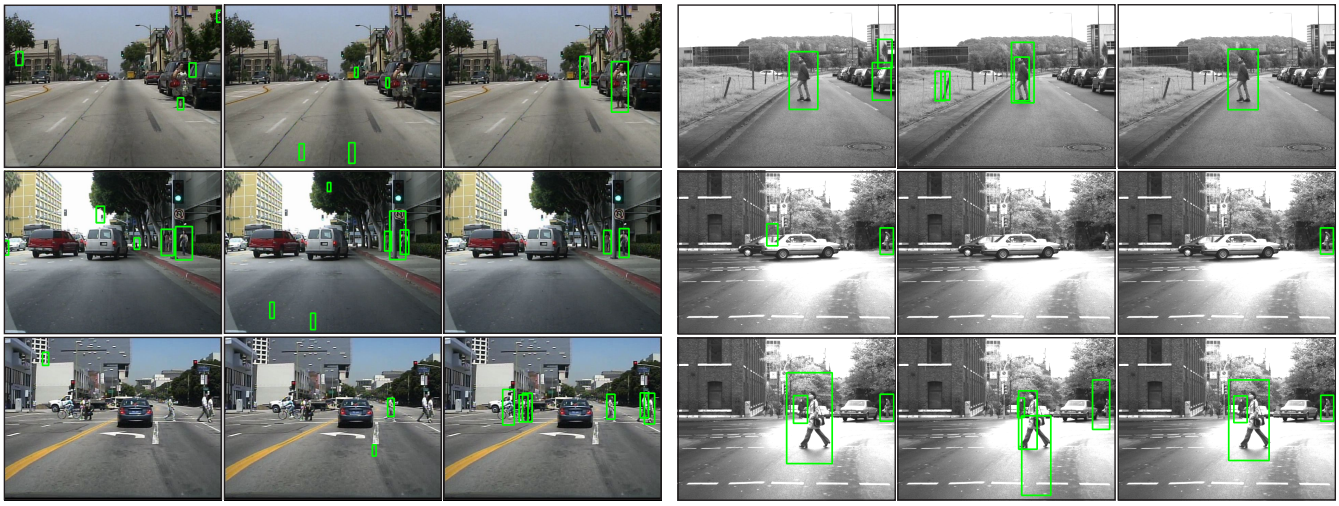
B. Performance comparison with conventional methods

We compare our method with state-of-the-art methods using the Caltech Pedestrian Dataset. As shown in Figure 5(a), with an FPPI of 0.1, our method gives a 10.5% increase in accuracy compared to conventional CNN. In addition, the proposed method achieves similar performance to Switchable Deep Network, which constructs a complex network structure. This demonstrates that a simple network architecture is sufficient to achieve state-of-the-art performance in Deep Learning methods. We show a comparison result on Daimler Mono Pedestrian Benchmark Dataset in Figure 5(b). FPPI is improved 2.0% accuracy than conventional CNN at 0.1, compared to CNN making the current Daimler Mono Pedestrian Benchmark Dataset top performance, it improves detection accuracy about 31.32% and archives comparable performance to state-of-the-art methods.

Figure 6 shows a pedestrian detection result in Caltech Pedestrian Dataset and Daimler Mono Pedestrian Benchmark Detection. The results in the first and fourth columns are examples of detection by HOG+SVM, second and fifth columns are examples of detection by DPM. The pedestrian detection result in the third and sixth columns are result of the proposed methods. As shown Figure 6, while conventional methods does not detect small pedestrians, proposed method detects them in same scene. The proposed methods are capable to detect hard situation such as occlusion, various poses. Moreover, it can reduce the false positive.

VII. Conclusion

We proposed two techniques for improvement of pedestrian detection by random selection of the units based on the Dropout. Random Dropout that is randomly setting corresponding value of the units to zero with flexible rate. EIN constructs multiple networks that have different structure in fully connected layer with decision manner of final output. We achieve the comparable performance to state-of-the-art methods in Deep Learning approach, even the structure of



(a) Caltech Pedestrian Dataset (b) Daimler Mono Pedestrian Benchmark Dataset

Fig. 6. Visualized pedestrian detection

proposed methods are significant simple. In future work, we will try to reduce the computational cost for real time processing.

REFERENCES

[1] N.Dalal, B. Triggs, "Histograms of oriented gradients for human detection", Computer Vision and Pattern Recognition, 2005.

[2] P. Felzenszwalb, D. McAllester, D. Ramaman,"A Discriminatively Trained, Multi scale, Deformable Part Model", Computer Vision and Pattern Recognition 2008.

[3] X. Wang, T. X. Han, S. Yan, "An HOG-LBP Human Detection with Partial Occlusion" , International Conference on Computer Vision, 2009.

[4] W. Nam, B. Han, J. H. Han, "Improving Object Localization Using Macrofeature Layout Selection" , International Conference on Computer Vision Workshop on Visual Surveillance, 2011.

[5] J. Marin, D. Vazquez, A. Lopez, J. Amores, B. Leibe, "Random Forests of Local Experts for Pedestrian Detection" , International Conference on Computer Vision, 2012.

[6] W. Ouyang, X. Wang, "Joint Deep Learning for Pedestrian Detection", Computer Vision and Pattern Recognition, 2013.

[7] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, "Gradient-Based Learning Applied to Document Recognition" , Proceedings of the IEEE, 1998.

[8] G. E. Hinton, N. Srivastava, A. Krizhevsky, S. Ilya, R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors" , Clinical Orthopaedics and Related Research, vol. abs/1207.0, 2012.

[9] I. Goodfellow, D. Warde-Farley, M. Mirza, A. C. Couville, Y. Bengio, "Maxout Network", International Conference on Machine Learning, pp.1319-1327, 2013.

[10] P. Viola, M. Jones,"Robust Real-Time Face Detection" , Computer Vision and Pattern Recognition, 2004.

[11] A. Krizhevsky, S. Ilva, G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Network" , Advances in Neural Information Processing System 25, pp.1097-1105, 2012.

[12] C. C. Wang, C. Thorpe, S. Thrun,"Online Simultaneous Localization And Mapping with Detection And Tracking of Moving Objects:Theory and Results from a Ground Vehicle in Crowded Urban Areass" , International Conference on Robotics and Automation, 2003.

[13] R. Girshick, J. Donahue, T. Darrell, J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation" , Computer Vision and Pattern Recognition, 2014.

[14] D. H. Hubel, T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cats visual cortex", The Journal of Physiology, pp.106-154, 1962.

[15] D. Scherer, A. Müller, B. Sven "Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition", International Conference on Artificial Neural Networks, Vol. 6354 of Lecture Notes in Computer Science, pp.92-101, 2010.

[16] Y. Bureau, J. Ponce, Y. LeCun, "A Theoretical Analysis of Feature Pooling in Visual Recognition",Neural Information Processing System, 2011.

[17] D. E. Rumelhart, G. E. Hinton, R. J. Williams, "Learning representations by back-propagating errors" , Neurocomputing, pp. 696-699, 1988.

[18] P. Dollar, Z. Tu, P. Perona, S. Belongie, "Integral Channel Feature" , British Machine Vision Conference, 2009.

[19] P. Dollar, R. Appel, W. Kienzle, "Crosstalk Cascades for Frame-Rate Pedestrian Detection" , European Conference on Computer Vision, 2012.

[20] W. R. Schwartz, A. Kembhavi, D. Harwood, L. S. Davis,"Human Detection Using Partial Least Squares Analysis" ,International Conference on Computer Vision, 2009.

[21] D. Park, D. Ramanan, C. Fowlkes, "Multi Resolution Models for Object Detection" , European Conference on Computer Vision, 2010.

[22] P. Dollar, S. Belongie, P. Perona,"The Fastest Pedestrian Detection" , British Machine Vision Conference, 2010.

[23] R. Benenson, M. Mathias, T. Tuytelaars, L. V. Gool,"Seeking the Strongest Rigid Detector" , Computer Vision and Pattern Recognition, 2013.

[24] P. Sermanet, K. Kavukcuoglu, S. Chintala, Y. LeCun,"Pedestrian Detection with Unsupervised Multi-stage Feature Learning" , Computer Vision and Pattern Recognition, pp.3626–3633, 2013.

[25] W.Ouyang, X. Wang,"A Discriminative Deep Model for Pedestrian Detection with Occlusion Handling" , Computer Vision and Pattern Recognition, 2012.

[26] P. Dallar, R. Appel, S. Belongic, P. Perona,"Fast Feature Pyramids for Object Detection" , Pattern Analysis and Machine Intelligence, 2014.

[27] G. Chen, Y. Ding, J. Xiao, T. Han,"Detection Evolution with Multi-order Contextural Co-occurrence" , Computer Vision and Pattern Recognition, 2013.

[28] P. Luo, Y. Tian, X. Wang, X. Tang,"Switchable Deep Network for Pedestrian Detection", Computer Vision and Pattern Recognition, 2014.

[29] P. Sabzmeydani, G. Mori,"Detecting Pedestrians by Learning Shapelet Features" , Computer Vision and Pattern Recognition, 2007.

[30] C. Wojek, B. Schiele,"A Performance Evalution of Single and Multi-Feature People Detection" , German Association for Pattern Recognition, 2009.