Pedestrian Detection from Sparse Point-Cloud using 3DCNN

Yoshiki Tatebe
Nagoya University
Graduate School of Information Science
Aichi, Japan

Daisuke Deguchi
Nagoya University
Information Strategy Office
Aichi, Japan

Yasutomo Kawanishi, Ichiro Ide, and Hiroshi Murase
Nagoya University
Graduate School of Informatics
Aichi, Japan

Utsushi Sakai
DENSO CORPORATION
Aichi, Japan

***Abstract*— This paper proposes a LIDAR-based pedestrian detection method using 3DCNN. The proposed method converts a sparse point-cloud obtained by a low-resolution LIDAR to two-channels voxel representation that consists of the 3D object probability channel and the reflection intensity channel. To evaluate the performance of the proposed method, an experiment using real-world LIDAR data was conducted. The results show that the proposed method is able to detect pedestrians more accurately than detectors trained by other conventional features.**

Keywords—LIDAR; 3DCNN; Pedestrian detection; Sparse

I. INTRODUCTION

In recent years, Advanced Driver-Assistance Systems (ADAS) are becoming popular to realize a safe and comfortable driving environment. ADAS are systems that can recognize the surrounding environment of a vehicle and warn the driver about possible dangers such as crossing pedestrians. LIDAR is one of the most popular sensors for ADAS. It is a time-of-flight camera and can measure distance to a target by illuminating the target with a pulsed laser, and measuring the reflected pulse by a sensor. A high-resolution LIDAR can be used to accurately detect a pedestrian, and is used in many researches [1]. However, it is too expensive to implement in consumer vehicles. Therefore, this paper proposes a method to detect pedestrians using a low-resolution LIDAR that is much cheaper and smaller than a high-resolution one for consumer vehicles. Since, point-clouds obtained by a low-resolution LIDAR is very sparse, it is difficult to distinguish pedestrians and similar objects such as (utility) poles and trees.

Many researches try to use a LIDAR for pedestrian detection, and some of them compute features representing a pedestrian's shape from LIDAR data [1,2]. Kidono et al. proposed the slice feature that represents the rough shape of a pedestrian combined with distribution of reflection intensities [1]. Also, to improve the detection performance of a low-resolution LIDAR, we proposed a feature extraction method utilizing multi-frames information to enhance the density of point-clouds and to capture the temporal change of point-clouds [2].

Meanwhile, deep learning is becoming to be utilized for object recognition from three-dimensional point-clouds. For example, Maturana et al. proposed VoxNet [3] that uses a representation of the distribution of a point-cloud in the voxel space, which is called an occupancy grid. Here, the occupancy grid is used as an input of a 3DCNN (3-Dimensional Convolution Neural Network). Although the results are very promising and the classification accuracy is very high, this method is designed for using high-resolution point-clouds such as CAD data. Therefore, it cannot be applied directly to extremely sparse point-clouds obtained from a low-resolution LIDAR.

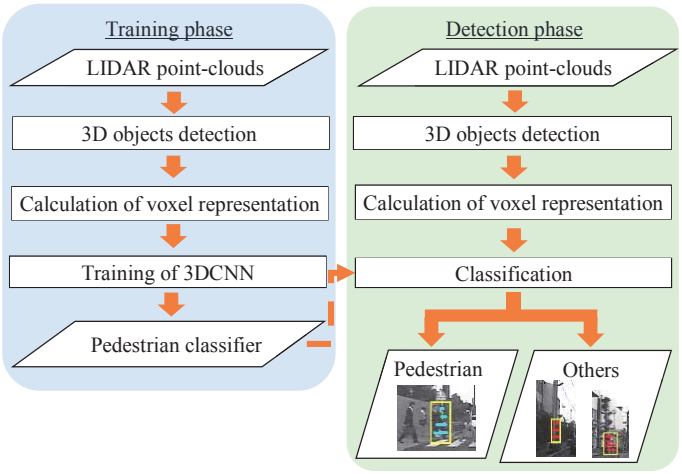

Fig. 1. Process-flow of the proposed method.

In order to combine the merits of both low-resolution LIDAR sensors and a 3DCNN, this paper proposes a novel voxel representation that can be calculated from a sparse point-cloud. A LIDAR sensor irradiates lasers at regular intervals, and measurement points are obtained if there are reflected laser beams within the irradiation interval and the width of a laser beam. The width of the laser beam increases according to the distance from the sensor because of spreading of the light. Therefore, each measured point has an ambiguity within the irradiation interval and the beam width in relation to the distance to an object. This paper utilizes this property to compute the novel voxel representation of the sparse point-cloud. In addition, the proposed method also utilizes the reflection intensities for

978-1-5386-2615-3/18/$31.00 ©2018 IEEE

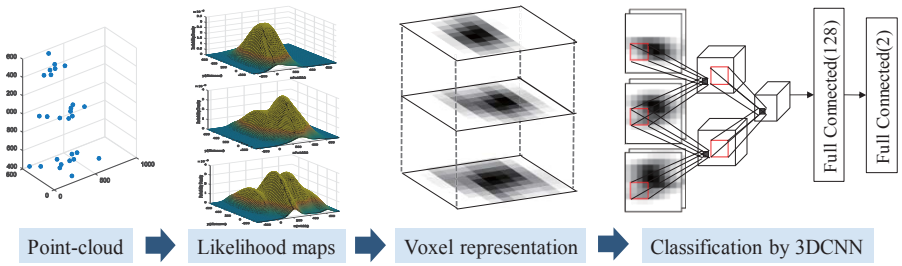

Fig. 2. Process-flow for obtaining voxel representation.

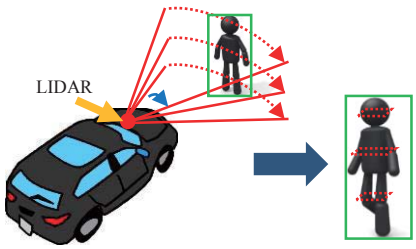

Fig. 3. Example of an $L = 3$ point-cloud.

classification, which was not considered by VoxNet. Specifically, a two-channel voxel space is constructed by calculating a 3D object likelihood voxel map and a reflection intensity likelihood voxel map as the input of a 3DCNN. The contribution and the novelty of this paper is the proposal of a novel voxel presentation that can be calculated from sparse point-clouds. By utilizing the measurement ambiguity in relation to the distance and the reflection intensity of target objects considering LIDAR characteristic, sparse point-clouds can be efficiently converted to voxel space, and it can be classified accurately using 3DCNN.

Section II describes the proposed pedestrian detection method using the novel voxel representation. Then, details of an experiment and discussions are described in Section III. Finally, this paper is concluded in Section IV.

II. Proposed Method

Figure 1 shows the overall processes of the proposed method. The first step of the proposed method is to detect 3D object candidates from point-clouds. This is achieved by evaluating the 3D structure of a point-cloud and the difference of the reflection intensity between 3D objects and the road surface. The next step is to convert the sparse point-clouds of each candidate to a voxel representation. Figure 2 shows the flow of this process. To cope with the phenomena of LIDAR described in Section I, kernel density estimation is used for obtaining the likelihood maps, where the size of the kernel changes according to the distance to objects. Finally, the voxel representation is obtained by sampling from these maps. In the training phase, to increase the number of training samples, data augmentation is performed by perturbation and rotation of each point-cloud based on the LIDAR characteristics. By inputting a two-channel voxel representation to 3DCNN, each candidate is classified. More details are described in the following sections.

A. Detection of 3D object candidates

In the proposed method, 3D objects are detected and extracted from point-clouds by applying a 3D object detection and tracking algorithm [4]. A point-cloud obtained by this algorithm is henceforth called as a *candidate point-cloud*.

In this paper, L represents the number of horizontal scans hitting an object, and each candidate point-cloud is split by each scan. Figure 3 shows an example when $L = 3$. A candidate point-cloud can be represented as

$$\begin{aligned} \mathcal{Q} &= \{\mathcal{Q}_l\}_{l=1}^{L}, \\ \mathcal{Q}_l &= \{\boldsymbol{x}_{l,i}\}_{i=1}^{N_l}, \end{aligned} \tag{1}$$

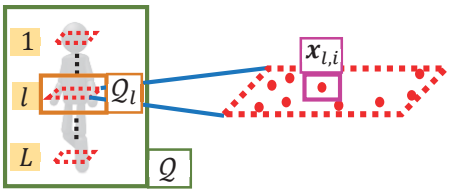

Fig. 4. Structure of a candidate point-cloud.

where $\mathcal{Q}$ is the whole candidate point-cloud, $\mathcal{Q}_l$ is a point-cloud obtained by the l-th horizontal scan, and $\boldsymbol{x}_{l,i}$ is the i-th data of the l-th scan with coordinates (x, y, z) obtained from the LIDAR. Figure 4 is a graphical representation of this structure. A reflection intensity of a laser beam at $\boldsymbol{x}_{l,i}$ is represented as $r_{l,i} = \boldsymbol{I}(\boldsymbol{x}_{l,i})$.

B. Conversion from sparse point-clouds to voxel representation

The proposed method calculates a two-channel voxel the representation $(\mathcal{V}^d, \mathcal{V}^r)$ from the 3D object likelihood and the reflection intensity of an object. These are obtained from $\mathcal{Q}$ of (1). Figure 2 shows the flow for converting the point-cloud to a voxel representation in the case of $L = 3$. This process consists of two steps. In the first step, the 3D object likelihood map $\mathcal{M}_l^d$ and the reflection intensity likelihood map $\mathcal{M}_l^r$ are calculated from each $\mathcal{Q}_l$. In the second step, both likelihood maps are converted into a voxel representation to form a two-channel voxel representation.

1) Generation of likelihood maps: Since a point-cloud obtained from a low-resolution LIDAR is extremely sparse, it is difficult to convert it directly into the voxel space. Therefore, the kernel density estimation based on the distance and the reflection intensity of each point is utilized to generate likelihood maps. Details of this process is described below.

Using measurement data $\boldsymbol{x}_{l,i}$, the 3D object likelihood map $\mathcal{M}_l^d$ can be formulated as

$$\sum_{i=1}^{N_l} g(\mathbf{P}\boldsymbol{x}_{l,i}, \Sigma_{l,i}), \tag{2}$$

where g is a 2D Gaussian distribution function, and $\mathbf{P}$ is a projection matrix to the road surface. That is, a likelihood map is a mixture of Gaussian kernels $g(\mathbf{P}\boldsymbol{x}_{l,i}, \Sigma_{l,i})$ whose parameters are the projected point $\mathbf{P}\boldsymbol{x}_{l,i}$ and a different variance-covariance matrix $\Sigma_{l,i}$. Here, $\Sigma_{l,i}$ depends on the distance $d_{l,i} = \|\boldsymbol{x}_{l,i}\|$, and formulated as follows:

$$\Sigma_{l,i} = \begin{pmatrix} \sigma_{l,i}^{x} & 0 \\ 0 & \sigma_{l,i}^{y} \end{pmatrix}, \quad (3)$$

$$\sigma_{l,i}^{x} = \tan\alpha \times d_{l,i}, \quad (4)$$

$$\sigma_{l,i}^{y} = E\left(d_{l,i}\right). \quad (5)$$

Both $\sigma_{l,i}^{x}$ and $\sigma_{l,i}^{y}$ are calculated based on $d_{l,i}$ and the scan errors of the LIDAR. Specifically, $\sigma_{l,i}^{x}$ indicates the horizontal scan error, and is obtained from two sensor characteristics (the width of a laser beam and the measurement error). $\sigma_{l,i}^{y}$ indicates the scan error along the depth direction, and is also obtained from a sensor characteristic.

By using normalized reflection intensity $r_{l,i}^{*}$, the reflection intensity likelihood map $\mathcal{M}_l^{r}$ is calculated as

$$\sum_{i=1}^{N_l} r_{l,i}^{*} \times g\left(\mathrm{P}\boldsymbol{x}_{l,i}, \Sigma_{l,i}\right), \quad (6)$$

where

$$r_{l,i}^{*} = r_{l,i} \times d_{l,i}^{2}. \quad (7)$$

2) *Conversion to voxel representation:* A two-channel voxel representation $(\mathcal{V}^{d}, \mathcal{V}^{r})$ is obtained by sampling the likelihood maps at a certain interval. Specifically, each voxel value is formulated as

$$\mathcal{V}(j,k,l) = \int_{\Omega_{j,k}} \mathcal{M}_l(\boldsymbol{x}) d\boldsymbol{x}, \quad (8)$$

where the integration domain $\Omega_{j,k}$ is defined as

$$\Omega_{j,k} = \{\boldsymbol{x} + \overline{\boldsymbol{x}} \mid \gamma \|\boldsymbol{x} - (j,k,0)^{T}\|_{\infty} < 0.5\}. \quad (9)$$

Here, $\|.\|_{\infty}$ is L_{∞}-norm, and $\overline{\boldsymbol{x}}$ is calculated as

$$\overline{\boldsymbol{x}} = \frac{1}{|\mathcal{Q}|} \sum_{\boldsymbol{x} \in \mathcal{Q}} \mathrm{P}\boldsymbol{x}. \quad (10)$$

γ is the sampling interval which corresponds to the size of each voxel. Here, $\gamma = 0.1$ is used. This is equal to a sampling from $10\ \mathrm{cm} \times 10\ \mathrm{cm}$ region. Finally, the voxel representation has $10 \times 10 \times L \times 2$ dimensions.

C. Data augmentation

In order to increase the training data, perturbation and rotation of a point-cloud are performed against each training sample. In this process, the number of the training data is increased by 10 times. The amount of perturbation is controlled by the scan error characteristic of the LIDAR. Data augmentation by rotation is performed within the range of -10 to 10 degrees. By combining these perturbation and rotation, various patterns of training samples are generated.

D. Training of 3DCNN

The proposed method constructs a 3DCNN for each L. The same network architecture is used for each L, and the network parameters are as follows.

TABLE I. NUMBER OF POINT CLOUDS FOR EACH NUMBER OF SCAN HITS L

	Number of horizontal scan hits L			
	$L = 3$	$L = 4$	$L = 5$	$L = 6$
Pedestrian	1,134	5,343	7,016	8,258
Others	7,143	7,360	8,477	14,719

TABLE II. EVALUATED METHODS

Method	***Conventional***	***Proposed 1***	***Proposed 2***	***Proposed 3***
Feature	Handmade [1,2]	Voxel	Voxel	Voxel
Classifier	SVM	3DCNN	2DCNN	3DCNN

- 3D Convolutional layer $(32, 3 \times 3 \times 2, 1)$
- 3D Convolutional layer $(32, 3 \times 3 \times 2, 1)$
- Full Connected layer (128)
- Full Connected layer (2)

Here, a 3D Convolution layer is represented by parameters of (f, k, s) where the number of filter is f, the filter size is k, and the stride is s. Full Connected layer (n) has n outputs. LeRU [5] is used as an activation function, the batch size is 128, and the epoch is 100.

E. Classification by 3DCNN

The voxel representation of a candidate point-cloud described in Section II-B is used as an input of the 3DCNNs, and then each candidate is classified whether it is a pedestrian or not. The output of this network is the detection result of the proposed method.

III. EXPERIMENTS AND DISCUSSIONS

To evaluate the effectiveness of the proposed method, experiments were conducted using point-clouds obtained from a low-resolution LIDAR.

A. Experimental setup

The experiment was conducted using point-clouds obtained from a low-resolution LIDAR, collected in a real-world environment. By applying a 3D object detection and tracking algorithm to the collected data, candidate point-clouds were obtained. All of extracted positive samples and negative samples (pole, tree, and so on) were used as the input of the proposed method. TABLE I shows the result of aggregating the data by each L. The performance of the proposed method was evaluated by two-fold cross validation. To ensure fair comparison, point-clouds obtained from the same object were only included in either of training or test samples. We evaluated the proposed method by ROC (Receiver Operating Characteristic) curve (False Positive Rate (FPR) vs. True Positive Rate (TPR)) and its partial AUC (Area Under the Curve), which is calculated from the partial area of an ROC curve.

The low-resolution LIDAR used in the experiment was equipped on top of the rear-view mirror. The LIDAR could obtain the distance to a target at 10 fps. The vertical detectionangle was 6 degrees (1 degree pitch), and the horizontal

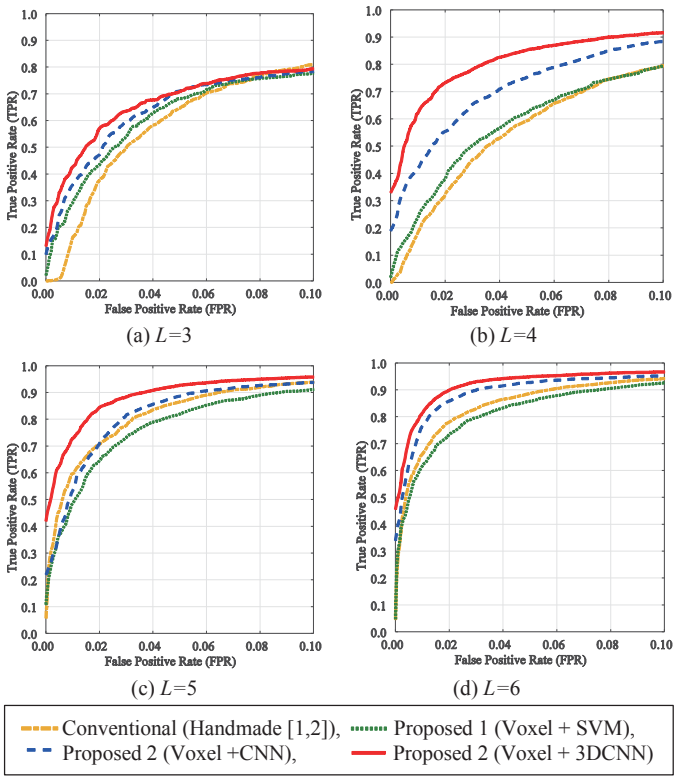

Fig. 5. ROC curves of each method by the number of horizontal scan hits *L*.

detection angle was 40 degrees (0.1 degrees pitch). On the other hand, based on the sensor characteristics of the LIDAR, $\alpha = 0.1$ degrees in (4) and $\sigma_{l,i}^{y}$ in (5) were provided by a preliminary experiment.

Four methods were evaluated in this experiment as shown in TABLE II. Here, Handmade represents conventional handmade features proposed by K. Kidono et al [1]. and our previous work [2].

B. Results and discussions

Figure 5 shows ROC curves (FPR < 10 %) of the experimental results, and TABLE III shows partial AUC (FPR < 10 %). It can be seen that TPR of proposed method 3 was higher than the other methods. Therefore, we confirmed that the proposed method 3 could detect pedestrians more accurately in low FPR (very important in pedestrian detection). Figure 6 shows a sample of a True Positive (TP) and a False Positive (FP).

As seen in Fig. 5, the voxel representation combined with SVM (Proposed 1) achieved relatively good results. This suggests that the voxel representation has a potential to describe the conventional handmade features. For example, the slice feature is obtained from the 3D object likelihood channel, and the distribution of reflection intensity can also be obtained from the reflection intensity channel.

Proposed method 3 showed significant improvement in comparison with the results from the conventional method. Therefore, the combination of the proposed voxel representation and 3DCNN is very effective to classify a sparse point-cloud.

Finally, the detection accuracy of proposed method 3 was higher than that of proposed method 2. From this result, we confirmed that it is effective for pedestrian detection to consider the 3D structure of a point -cloud.

TABLE III. PARTIAL AUC OF EACH METHOD

Method	Partial AUC by *L*			
	***L* = 3**	***L* = 4**	***L* = 5**	***L* = 6**
Conventional	0.564	0.535	0.805	0.836
Proposed 1	0.601	0.565	0.757	0.807
Proposed 2	0.629	0.698	0.809	0.884
Proposed 3	0.658	0.802	0.881	0.914

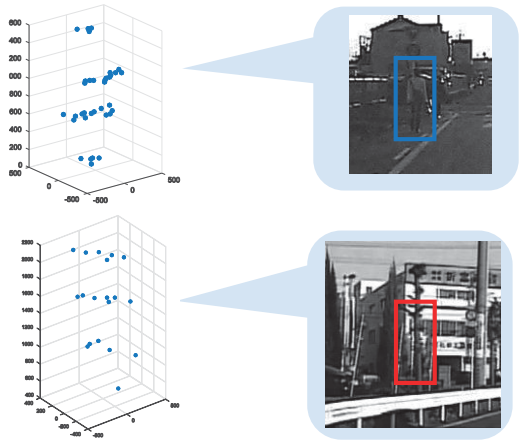

Fig. 6. Sample of a True Positive and a False Positive

IV. CONCLUSION

This paper proposed a novel voxel representation that can be calculated from a sparse point-cloud to achieve pedestrian detection using 3DCNN. Specifically, to detect pedestrians accurately even if a point-cloud is sparse, two likelihood maps are calculated considering the distance and the reflection intensity of target objects. Then, point-clouds are converted into voxel representation using these likelihood maps. The results showed that 3DCNN + the proposed voxel representation could detect pedestrians more accurately than the conventional method. In addition, the effectiveness of 3D convolution was confirmed.

Future work includes the utilization of multi-frame information, improvement of the 3D object detection method and the network architecture.

ACKNOWLEDGMENT

Parts of this research were supported by MEXT, Grant-in-Aid for Scientific Research.

REFERENCES

[1] K. Kidono, et al. “Pedestrian recognition using high-definition LIDAR.” Proc. 2011 IEEE Intelligent Vehicles Symposium, pp. 405-410, June 2011.

[2] Y. Tatebe, et al. “Can we detect pedestrians using low-resolution LIDAR?” Proc. Int. Conf. on Computer Vision Theory and Applications 2017, pp. 157-164, Feb. 2017.

[3] D. Maturana, et al. “Voxnet: A 3D convolutional neural network for real-time object recognition.” Proc. 2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 922-928, Sept. 2015.

[4] T. Ogawa, et al. “Pedestrian detection and tracking using in-vehicle LIDAR for automotive application.” Proc. 2011 IEEE Intelligent Vehicles Symposium, pp. 734-739, June 2011.

[5] V. Nair, and G. E. Hinton. “Rectified linear units improve restricted Boltzmann machines.” Proc. 2010 Int. Conf. on Machine Learning, pp. 807-814, June 2010.