International Journal of Control, Automation and Systems 16(6) (2018) 2888-2895
http://dx.doi.org/10.1007/s12555-018-0182-y
ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

View-point Invariant 3D Classification for Mobile Robots Using a Convolutional Neural Network

Jiyoun Moon*, Hanjun Kim, and Beomhee Lee

Abstract: 3D object classification is an important component in semantic scene understanding for mobile robots. However, many current systems do not consider the practical issues such as object representation from different viewing positions of mobile robots. A novel 3D object representation is introduced using cylindrical occupancy grid and 3D convolutional neural network with row-wise max pooling layer. Due to the rotationally invariant characteristics of this method, robots can successfully classify 3D objects regardless of starting positions of object modelling. Experimental results on publicly available benchmark dataset show the significantly improved performance compared with other conventional algorithms.

Keywords: 3D object classification, cylindrical CNN, mobile robots, view-point invariant.

1. INTRODUCTION

3D object classification is one of the most fundamental problems in semantic scene understanding that improves the capabilities of mobile robots to carry out high-level manipulation and navigation tasks [1]. As 3D sensors such as RGB-D cameras and LiDAR are increasingly common in robotic systems, 3D object classification for robots is becoming an important and challenging problem to understand their surroundings. To fully utilize the 3D information of surrounding objects, deep learning whose performance heavily relies on large amount of data needs to be developed. In particular, convolutional neural network (CNN) achieves significantly better performances than conventional methods using machine learning algorithms on hand crafted features [2–4]. Generally, the performance of 3D object classification using CNN greatly depends on the input representation. Since most of the state-of-the-art classification methods are not robust to object rotation, it will generate a problem that objects are represented quite differently according to the starting view position for 3D object modeling.

In this paper, we propose a novel 3D object classification method that is invariant to starting position of 3D object modelling while maintaining geometric information. First, at different starting positions, a mobile robot collects 3D point cloud data using an RGB-D camera and conducts 3D object modelling. Second, the modelled objects are converted into volumetric representation in the form of binary cylindrical occupancy grid. Then, 3D CNN with *row-wise max-pooling* (*RWMP*) layer learns rotation-invariant features which is robust to object classification regardless of the starting position for modelling. The effectiveness of the proposed method is proved by comparing the performance of the proposed method to other conventional algorithms on both publicly open dataset and the dataset gathered from the real-world environment.

2. RELATED WORK

With the development of augmented reality in autonomous vehicles, the use of LiDAR and RGBD camera has been increased recently, where 3D data is gradually becoming common [5]. Thus, collision avoidance and mapping problem using 3D information are widely researched, while object classification and recognition using 3D data is less investigated [6]. To efficiently deal with rich source of 3D information, hand crafted feature descriptors like scale-invariant feature transform (SIFT) or speeded-up robust features (SURF) are commonly used along with a machine learning classifier [7, 8]. Also, 3D shape descriptors like binary signatures of histograms of orientations (B-SHOT) [9] and shift-invariant ring feature [10] are developed as a common practice. However, those performances can vary greatly depending on 3D input data.

Recently, algorithm based hand-crafted methods have been replaced by deep learning, which summarizes the core features or contents from a large amount of complex data through a combination of nonlinear transformation.

Manuscript received March 23, 2018; revised July 10, 2018; accepted August 13, 2018. Recommended by Associate Editor Dong-Joong Kang under the direction of Editor Euntai Kim. This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. 2017R1A2B2002608).

Jiyoun Moon, Hanjun Kim, and Beomhee Lee are with Automation and Systems Research Institute, Department of Electrical Engineering, Seoul National University, Seoul, Korea (e-mails: {jiyounmoon, k3k5good, bhlee}@snu.ac.kr).
* Corresponding author.

In particular, CNN which generates general purpose descriptors using trainable filters and pooling operations showed excellent performances. It outperformed conventional methods in different areas of computer vision and others such as visual object recognition [11–13], natural language process [14], human tracking [15, 16], and audio classification [17, 18]. However, 3D object classification using CNN has problems lacking in memory and computational complexity burden. To overcome this limitation, the 3D convolutional neural network (3D CNN) that learns features automatically with volumetric representation and multi-view image was proposed and outperformed 3D object classification.

Multi-view convolutional neural network (MVCNN) generally extracts 2D data in the form of image from a 3D registered object and performs object classification [19]. It has advantages that we can leverage massive image databases like ImageNet and pre-trained neural networks like VGGNet. A similar method, deepPano performs object classification based on the panoramic image generated along the surface of a 3D object [20]. Even though deepPano is not able to classify objects using pre-trained neural networks, it has advantage that it can contain all surface information of an object from only one image.

The well-known approaches to volumetric representation are VoxNet [5], 3DShapeNets [21], and VoxelNet [22]. VoxNet converts 3D data into the form of binary rectangular occupancy grid. 3DShapeNets represents each voxel grid as surface, occluded, or free space using visual surface and depth map. VoxelNet performs voxel partition and converts 3D data within each voxel into a vector representation. Recent works use octrees algorithm [23, 24] or sparse voxel grids [25] to create occupancy grid faster and more efficient. These methods have the advantage of utilizing both LiDAR and RGBD point cloud data.

Between these volumetric representation and multi-view image methods, the performance of multi-view image was proved to be better [26]. However, it has limitations in that a mobile robot cannot utilize the 2.5D data in other tasks besides object classification since it loses geometric information. For rectangular volumetric representation, it also has limitations in practical approach that the same object can be represented quite differently according to the position where a robot starts modelling a 3D object, thus the object classification performance is degraded.

3. APPROACH

This paper has three main components: 3D object data gathering and modelling using a mobile robot, a volumetric representation in the form of binary cylindrical occupancy grid which has rotation-invariant characteristic, and an object classification using 3D CNN. We describe each detailed analysis below.

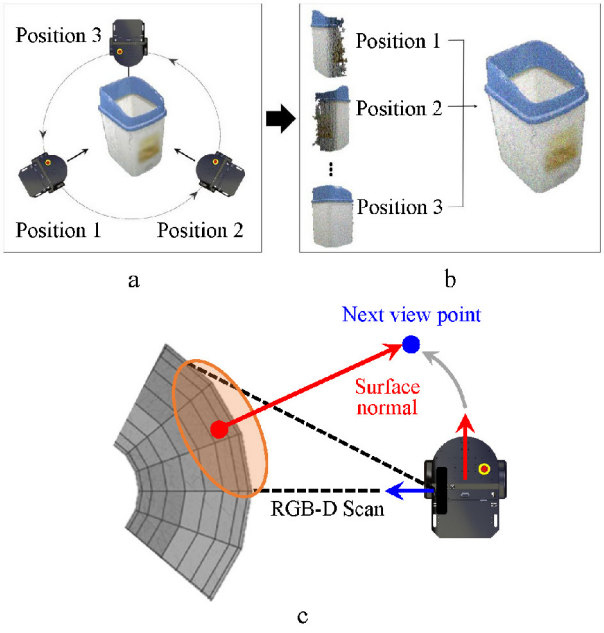

Fig. 1. 3D object data gathering and modelling with a mobile robot: (a) A robot moving around the object for data gathering (b) 3D object modelling based on various view-point data (c) Next view-point selection based on surface normal.

3.1. 3D object data gathering and modelling with a mobile robot

3D object modelling is a process to construct an object in a 3D virtual space for the purpose of an object recognition and classification. To perform 3D object modeling, a robot needs to gather 3D information of an entire object by observing the object at various point of views with sensors. The observation position should be selected by a robot considering the characteristics of the sensors mounted on the robot and the shape of an object to efficiently collect the data of the object.

To create a 3D object model, a mobile robot needs to move around the object in order to gather overall information about the 3D object, as shown in Fig. 1(a). The information of the 3D object can be gathered with high sensor accuracy when the RGB-D camera and the observed surface are in the perpendicular direction [27]. Therefore, the mobile robot selects the next position using the surface characteristics of the object by finding the direction where the camera and the surface of object are perpendicular to each other, as shown in Fig. 1(c). Principle component analysis (PCA) is used to estimate the normal vector of the object surface. As a result, the mobile robot observes and gathers the data of the object while moving along the surface of the object.

Once the mobile robot gathers the data at various viewpoints, 3D object modelling is performed as shown in

Fig. 1(b). To perform modelling, we find spatial transformations that align the data at each view-point into a globally consistent model using the iterative closest point (ICP). The ICP registration algorithm is widely used for 3D data such as point cloud and RGB-D, especially when we can get only one direction of object data at each time step [28]. Sensor data from odometry and RGB-D data are fused into one to get a more accurate model. To verify the proposed approach that is robust to object representation problem, the mobile robot performs data gathering and modelling of 3D objects at various starting positions.

3.2. Volumetric representation in the form of binary cylindrical occupancy grid

To construct a cylindrical occupancy grid, first, each of 3D data is projected onto the cylindrical coordinate whose axis is parallel to the principle axis of 3D modelled object. The upright-orientation assumption allows us to define $\mathbf{x}_0 = (x_0, y_0, h_0)^T$ as the principle axis by the weighted average of all point cloud data. In projection process, each point cloud $\mathbf{x} = (x, y, h)^T$ is mapped to a cylindrical coordinate as $\mathbf{c} = (r, \theta, h)^T$: $r = \sqrt{x^2 + y^2}$, $\theta = \text{atan2}(y, x) \in (0, 2\pi]$. With fixed h, an ideal basis in cylindrical coordinates can take separate form as radial part $P(r)$ and angular part $\Psi(\theta)$. While $P(r)$ can be determined by the context, the optimal choice for the angular part is defined as the fourier basis $\Psi_m = e^{im\theta}$ where m is an integer [29]. A simple rotation behavior function $J(\theta)$ on an object in terms of fourier basis is as follows:

$$J(\theta) = \sum_{m=-\infty}^{\infty} a_m \Psi_m(\theta) = \sum_{m=-\infty}^{\infty} a_m e^{im\theta}, \tag{1}$$

where $a_m = \frac{1}{2\pi} \int_0^{2\pi} J(\theta) e^{im\theta} d\theta$. If the function $J(\theta)$ is rotated by an angle α, then:

$$J(\theta - \alpha) = \sum_{m=-\infty}^{\infty} a_m e^{im(\theta-\alpha)} = \sum_{m=-\infty}^{\infty} (a_m e^{-im\alpha}) e^{im\theta}, \tag{2}$$

which shows the shift property. This is an important characteristics to extract rotation-invariant features in the latter part.

Occupancy grid is used to efficiently represent a large amount of point cloud data as free and occupied space. Each point $\mathbf{c} = (r, \theta, h)$ is mapped to discrete grid coordinate $\mathbf{g} = (i, j, k)$. Assuming the ideal beam sensor model, 3D ray tracing is used to calculate the states of each grid. At first, the initial value of each grid is set to $g^0_{ijk} = 0$ and it is updated as follows:

$$g^t_{ijk} = min(g^{t-1}_{ijk} + z^t, 1), \tag{3}$$

where z^t is range measurements in time step t. If it hits, $z^t = 1$ which means occupied. If it pass through, $z^t = 0$ which means free. Through this process, the volumetric representation in the form of binary cylindrical occupancy grid is shown in Fig. 2(a).

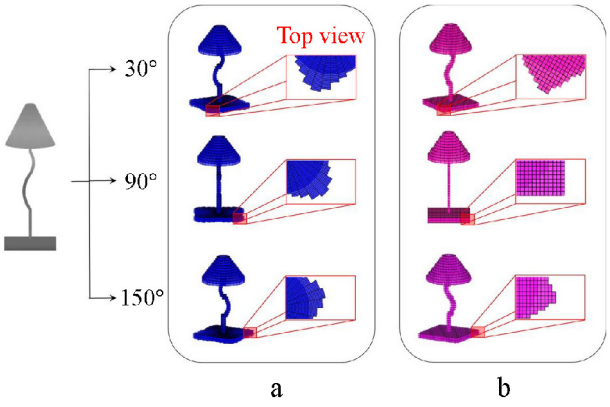

Fig. 2. Volumetric representation of a 3D object as it rotates: (a) Cylindrical occupancy grid (b) Rectangular occupancy grid.

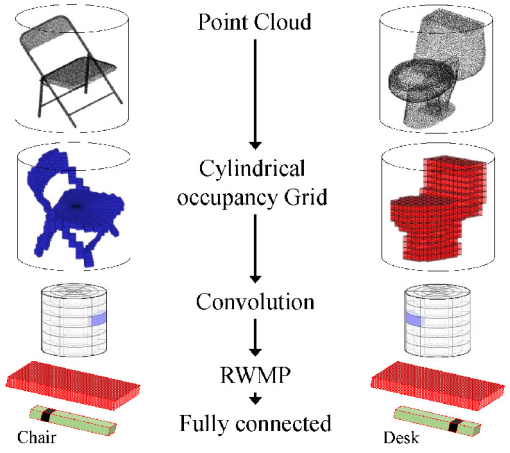

Fig. 3. Cylindrical 3D CNN with a RWMP layer.

Fig. 2(b) shows conventional rectangular occupancy grid which is a rectangular (cartesian) coordinate transformed into a grid coordinate. By the top view that shows the enlarged part of the highlighted in object, it can be seen that the cylindrical occupancy grid only performs shift while preserving the shape as the object rotates. On the other hand, the shape of rectangular occupancy grid changes as it rotates.

3.3. Object classification using cylindrical 3D CNN

The 3D CNN is mostly used in 3D object recognition, object generation problem, and video frame analysis. The process of 3D CNN can be divided into two parts, feature extraction and classification [30]. As in Fig. 3, the feature extraction part consists of convolution layer that preserves connectivity of the input data and pooling layer which helps to obtain translation-invariant features. At the

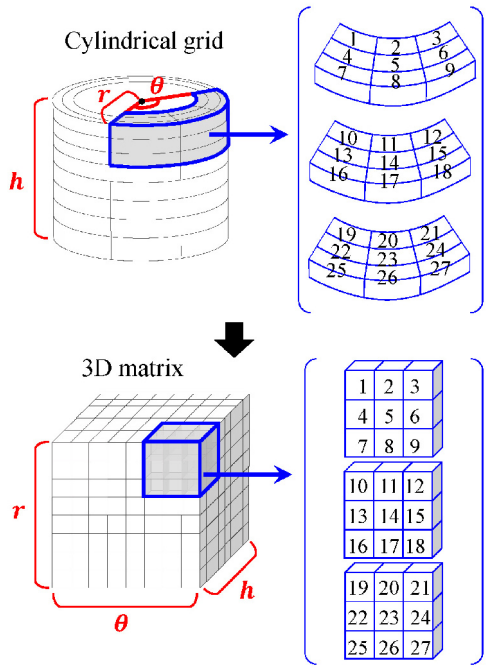

Fig. 4. Conversion process of cylindrical grid information to 3D matrix.

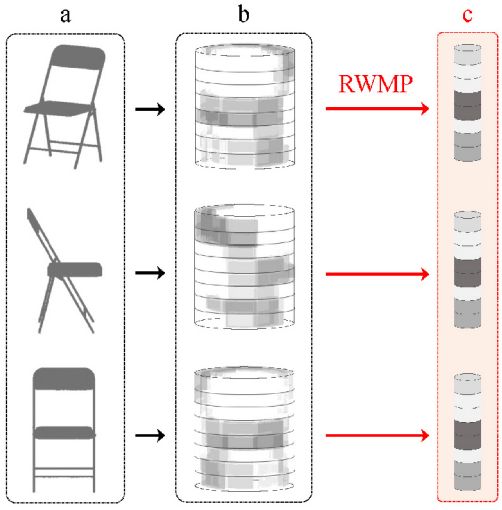

Fig. 5. RWMP layer operation: (a) 3D object of the chair but rotated to different angles, (b) convolutional feature map for 3D cylindrical occupancy grid, (c) output of RWMP layer.

end, fully connected layer classify the input data.

The input to cylindrical 3D CNN is a cylindrical occupancy grid $\mathbf{g} \in \{0,1\}^{p \times q \times r}$ and the output is the corresponding label $\mathbf{y} \in \mathbb{R}^n$. The process of network training is to find the weights $\mathbf{w}_{ij}^{pqr}$ and bias b_{ij} in the j-th feature map in the i-th layer with the dataset $(\mathbf{g}^{(1)}, \mathbf{y}^{(1)}), ..., (\mathbf{g}^{(n)}, \mathbf{y}^{(n)})$. To perform 3D convolution operation, we converted cylindrical grid information into 3D matrix as shown in Fig. 4. Using the 3D matrix, the value $v_{ij}^{r\theta h}$ is achieved by convolving a 3D kernel at position (r, θ, h) as follows:

$$v_{ij}^{r\theta h} = \sigma\left(b_{ij} + \sum_{p=0}^{P_{i-1}} \sum_{q=0}^{Q_{i-1}} \sum_{r=0}^{R_{i-1}} \mathbf{w}_{ij} v_{i-1}^{(r+p)(\theta+q)(h+r)}\right), \quad (4)$$

where P_{i-1}, Q_{i-1}, and R_{i-1} are the height, width, and depth of the 3D kernel. A rectified linear unit (ReLU) is applied to all, but the last layer, to act as an activation function σ. The parameter $\mathbf{w}$ is updated by the stochastic gradient descent (SGD) method minimizing the loss function L, which is defined as the multinomial cross-entropy. In the last layer, a softmax classifier is used to output prediction probabilities $g^{(i)} \in [0,1]$. Dropout is added by 0.4 after each layer to reduce overfitting [31].

Unlike most of 3D CNN architecture, a cylindrical 3D CNN takes a special layer named RWMP which takes the maximum value at each row at the same height and is concatenated into the output vector, as shown in Fig. 5. Then, the output vector is inserted between convolution layer and fully connected layer. Therefore, 3D CNN with a cylindrical occupancy grid is rotation-invariant, which means the features remain the same even though the object rotates. This is because RWMP layer is not affected by the shift while a cylindrical occupancy grid just shifts as the object rotates.

4. EXPERIMENTAL RESULTS

To evaluate the performance of 3D object classification that is robust to rotation of the object, experiments were performed using two datasets. One is the dataset with 6 categories that we collected using a mobile robot. The other is ModelNet10 dataset [32] with 10 categories which are usually used as the benchmark for object classification. Both datasets consist of objects that are commonly found in indoor environments like offices, bedrooms, and bathrooms. Fig. 6 and Fig. 7 show some samples of objects in each dataset.

We used Pioneer 3-DX robot, Asus Xtion PRO RGB-D camera, laptop, and intelligent space to gather our dataset. A laptop is located on the Pioneer 3-DX robot for collecting and processing data. The RGB-D camera is mounted on the upper left wheel of the robot. The process of collecting information of all objects is done in the intelligent space that can estimate the location of the robot.

On the intelligent space, several CCD cameras are attached as shown in Fig. 8. The position and orientation information of the robot is obtained through the visual tag that is attached on the center of the robot. The CCD cameras used in our experiments have a high S/N of 58dB. Also, we used visual tags that are robust to translation and

Fig. 6. Samples of our dataset.

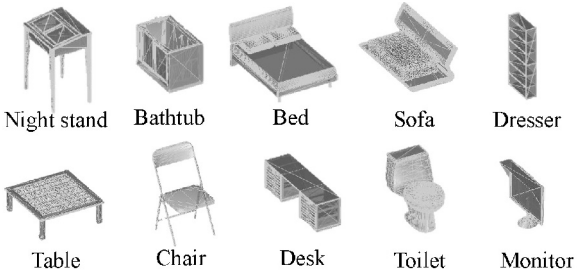

Fig. 7. Samples of ModelNet10 dataset.

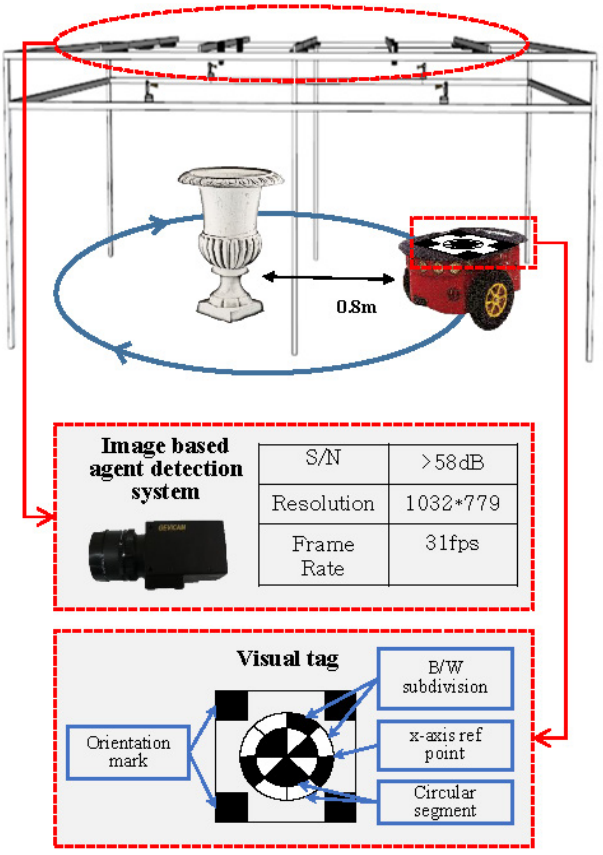

Fig. 8. Image based agent detection system in intelligent system.

rotation. Therefore, the position and location of the robot estimated by the intelligent space are reliable. We use this data as the ground truth. During the dataset acquisition, the speed of the robot was 5cm/s. The distance between camera and the object was 0.8m. The time interval was

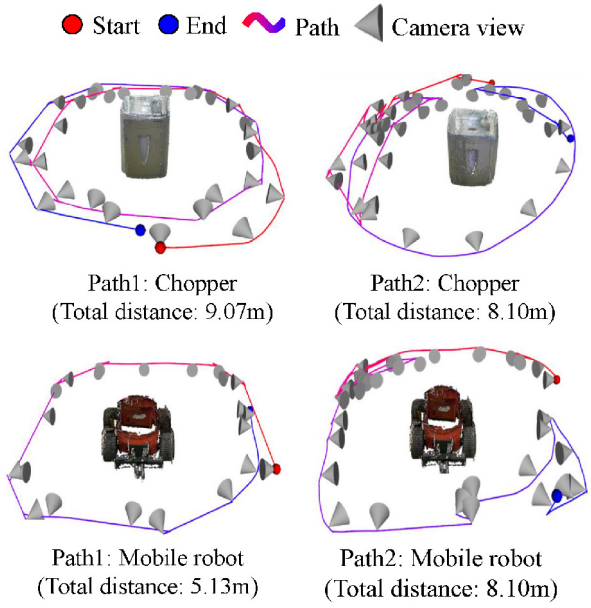

Fig. 9. Robot's movements based on selected observation position using PCA.

0.5 second. Fig. 9 shows the path of the robot moving along the selected observation positions using PCA. In order to verify the proposed method, we gathered the dataset at different starting positions for each object using a mobile robot.

To evaluate proposed method that is robust to object rotation, which means that object classification does not depend on the starting position of the mobile robot's observation and modelling. We manually modified the CAD models in ModelNet10 and our dataset into the form that is proper for our implementation. Each 3D object was projected into $32 \times 32 \times 32$ binary cylindrical occupancy grid, where the conversion time took 0.131 seconds. As shown in Fig. 10, the object maintains the shape of the occupancy grid, which is not affected by the rotation. We used GPU-acceleration to increase learning speed of neural networks. Cylindrical 3D CNN with four convolution layers, one RWMP layer and three fully connected layers showed the best performance in object classification. This network takes 0.096 seconds for the test. For comparison, we used same architecture for the rectangular 3D CNN. Details of the network architecture are specified in Table 1.

Using the network architecture in Table 1, Fig. 11 shows the classifier performance of each category in the form of a confusion matrix. Although most 3D objects are successfully classified, the results show that few objects such as table and desk, or nightstand and dresser, which have similar shapes are misclassified. Table 2 shows the object classification performance of the proposed method through comparison with 3D CNN using rectangular oc-

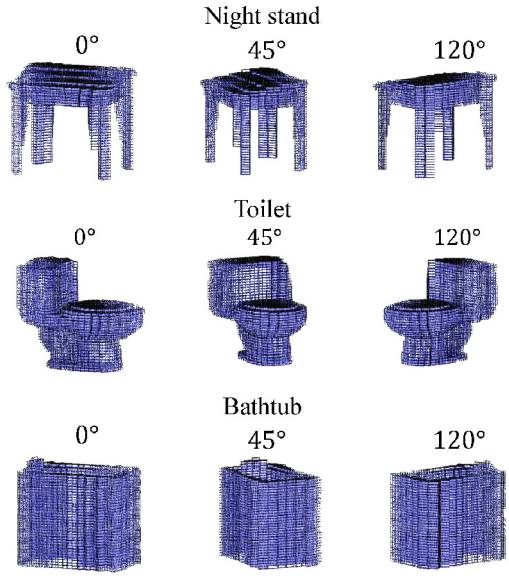

Fig. 10. Volumetric representation in the form of binary cylindrical occupancy grid.

Table 1. Architecture of a cylindrical 3D CNN.

Layer type	Filter Size	Output size	Number of parameters
Convolution	$3 \times 3 \times 3$	$30 \times 30 \times 30$	448
Convolution	$3 \times 3 \times 3$	$30 \times 30 \times 30$	448
ReLU	-	$30 \times 30 \times 30$	-
Convolution	$5 \times 5 \times 5$	$26 \times 26 \times 26$	64,032
ReLU	-	$26 \times 26 \times 26$	-
Convolution	$3 \times 3 \times 3$	$24 \times 24 \times 24$	55,360
ReLU	-	$24 \times 24 \times 24$	-
Convolution	$3 \times 3 \times 3$	$22 \times 22 \times 22$	221,312
ReLU	-	$22 \times 22 \times 22$	-
RWMP	$1 \times 22 \times 1$	$22 \times 1 \times 22$	
Fully Connected	-	1024	63,439,872
Fully Connected	-	512	524,800
Fully Connected	-	10	5,130

cupancy grid. To evaluate the robustness of object rotation, we trained both datasets at 0° and tested classification at 0°, 45°, 90°, and 135° rotated objects. We also trained both datasets with the same network in Table 1 except pooling layer and checked the rotation invariant property of RWMP layer.

Table 2 shows the proposed method marked in bold font which achieved higher classification accuracies at almost all rotated degrees in ModelNet10. The object classification accuracy of the untrained rotation angle showed a large difference between two methods. In particular, the performance of object classification is greatly improved compared to that of the conventional algorithm as the difference between the trained angle and tested angle increases. This is because cylindrical occupancy grid maintains the shape depending on the angle, unlike rectangular occupancy grid. Table 2 also shows the classification results using our dataset. This verified that the cylindrical occupancy grid performed better than the existing method even though the performance of our approach at 0° was slightly degraded. Therefore, the results show that our method outperforms existing rectangular algorithms.

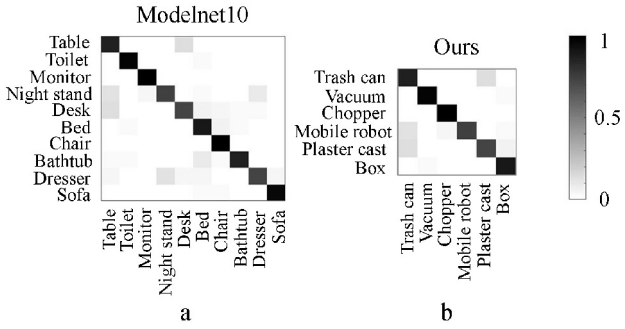

Fig. 11. Confusion matrices: (a) archieved on modelnet 10 dataset, (b) archieved on our dataset.

Table 2. Classification accuracies as a object rotates.

Occupancy grid	Degree	ModelNet10		Ours	
		Max Pooling	RWMP	Max Pooling	RWMP
Rectangular	0°	0.82	**0.82**	0.91	**0.92**
	45°	0.28	**0.45**	0.16	**0.26**
	90°	0.43	**0.41**	0.33	**0.46**
	135°	0.20	**0.29**	0.17	**0.33**
Cylindrical	0°	0.88	**0.87**	0.93	**0.91**
	45°	0.58	**0.86**	0.33	**0.89**
	90°	0.28	**0.84**	0.33	**0.91**
	135°	0.35	**0.83**	0.16	**0.88**

5. CONCLUSION

This paper proposed a new 3D object classification method that considers object representation problem which is robust to observation positions for mobile robots. First, a mobile robot starts gathering 3D object data in different observation positions using a 3D sensor. Then, the modelled 3D object is represented in the form of cylindrical occupancy grid which has the shift property as the object rotates. Eventually, classification is done by a cylindrical 3D CNN with an RWMP layer that extracts rotation-invariant features. We evaluated classification accuracies using ModelNet10 dataset and dataset that is gathered by a mobile robot. As a result, we verified that our algorithm classified the objects with higher success rates than the other conventional algorithm in both datasets.

REFERENCES

[1] I. Kostavelis and A. Gasteratos, "Semantic mapping for mobile robotics tasks: a survey," *Robotics and Autonomous Systems*, vol. 66, pp. 86-103, 2015.

[2] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," *Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshops,* pp. 512-519, 2014.

[3] R. Socher, B. Huval, B. Bath, C. D. Manning, and A. Y. Ng, "Convolutional-recursive deep learning for 3d object classification," *Advances in Neural Information Processing Systems*, pp. 656-664, 2012.

[4] L. A. Alexandre, "3d object recognition using convolutional neural networks with transfer learning between input channels," *Intelligent Autonomous Systems 13*, pp. 889-898, 2016.

[5] D. Maturana and S. Scherer, "Voxnet: a 3D convolutional neural network for real-time object recognition," *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 922-928, 2015.

[6] V. Hegde and R. Zade, "Fusionnet: 3D object classification using multiple data representations," arXiv preprint arXiv:1607.05695, 2016.

[7] J. Behley, V. Steinhage, and A. B. Cremers, "Performance of histogram descriptors for the classification of 3D laser range data in urban environments," *Proc. of IEEE International Conference on Robotics and Automation*, pp. 4391-4398, 2012.

[8] A. Teichman, J. Levinson, and S. Thrun, "Towards 3d object recognition via classification of arbitrary object tracks," *Proc. of IEEE International Conference on Robotics and Automation*, pp. 4034-4041, 2011.

[9] S. M. Prakhya, B. Liu, and W. Lin, "B-shot: a binary feature descriptor for fast and efficient keypoint matching on 3d point clouds," *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1929-1934, 2015.

[10] S. Bu, P. Han, Z. Liu, K. Li, and J. Han, "Shift-invariant ring feature for 3d shape," *The Visual Computer*, vol. 30, no. 6-8, pp. 867-876, 2014.

[11] F. J. Huang, Y. L. Boureau, and Y. LeCun, "Unsupervised learning of invariant feature hierarchies with applications to object recognition," *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1-8, 2007.

[12] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," *Proc. of 26th Annual International Conference on Machine Learning*, pp. 609-616, 2009.

[13] M. Norouzi, M. Ranjbar, and G. Mori, "Stacks of convolutional restricted boltzmann machines for shift-invariant feature learning," *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 2735-2742, 2009.

[14] R. Collobert and J.Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," *Proc. of 25th International Conference on Machine Learning*, pp. 160-167, 2008.

[15] J. Fan, W. Xu, Y. Wu, and Y. Gong, "Human tracking using convolutional neural networks," *IEEE Trans. on Neural Networks*, vol. 21, no. 10, pp. 1610-1623, 2010.

[16] M. Yang, F. Lv, W. Xu, and Y. Gong, "Detection driven adaptive multi-cue integration for multiple human tracking," *Proc. of IEEE International Conference on Computer Vision*, pp. 1554-1561, 2009.

[17] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," *Advances in Neural Information Processing Systems*, pp. 1096-1104, 2009.

[18] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," *Proc. of IEEE International Conference on Acoustics, Speech and Signal*, pp. 131-135, 2017.

[19] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view convolutional neural networks for 3D shape recognition," *Proc. of IEEE International Conference on Computer Vision*, pp. 945-953, 2015.

[20] B. Shi, S. Bai, Z. Zhou, and X. Bai, "Deeppano: deep panoramic representation for 3D shape recognition," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2339-2343, 2015.

[21] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: a deep representation for volumetric shapes," *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1912-1920, 2015.

[22] Y. Zhou and O. Tuzel, "Voxelnet: end-to-end learning for point cloud based 3d object detection," arXiv preprint arXiv:1711.06396, 2017.

[23] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Generative and discriminative voxel modeling with convolutional neural networks," arXiv preprint arXiv:1608.04236, 2016.

[24] G. Riegler, A. O. Ulusoy, and A. Geiger, "Octnet: learning deep 3d representations at high resolutions," *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 3577-3586, 2017.

[25] M. Engelcke, D. Rao, D. Z. Wang, C. H. Tong, and I. Posner, "Vote3deep: fast object detection in 3D point clouds using efficient convolutional neural networks," *Proc. of IEEE International Conference on Robotics and Automation*, pp. 1355-1361, 2017.

[26] C. R. Qi, H. Su, M. NieSSner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and multi-view cnns for object classification on 3D data," *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 5648-5656, 2016.

[27] C. V. Nguyen, S. Izadi, and D. Lovell, "Modeling kinect sensor noise for improved 3D reconstruction and tracking," *Proc. of IEEE International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, pp. 524-530, 2012.

[28] S. Li, J. Wang, Z. Liang, and L. Su, "Tree point clouds registration using an improved icp algorithm based on kd-tree," *Proc. of IEEE International Conference on Geoscience and Remote Sensing Symposium*, pp. 4545-4548, 2016.

[29] K. Liu, H. Skibbe, T. Schmidt, T. Blein, K. Palme, T. Brox, and O. Ronneberger, "Rotation-invariant hog descriptors using fourier analysis in polar and spherical coordinates," *International Journal of Computer Vision*, vol. 106, no. 3, pp. 342-364, 2014.

[30] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 221-231, 2013.

[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929-1958, 2014.

[32] K. V. Vishwanath, D. Gupta, A. Vahdat, and K. Yocum, "Modelnet: towards a datacenter emulation environment," *Proc. of IEEE International Conference on Peer-to-Peer Computing*, pp. 81-82, 2009.

Jiyoun Moon received her Bachelor's of Science in Robotics from Kwangwoon University in August 2014. Her major research interests include Natural language process, Semantic scene understanding, and Mission planning.

Hanjun Kim received his Bachelor's of Science in Electrical and Computer Engineering from Seoul National University in February 2015. His major research interests include SLAM, Reinforcement Learning, and Semantic scene understanding.

Beomhee Lee received the B.S. and M.S. degrees in Electronics Engineering from Seoul National University, in 1978 and 1980, respectively, and the Ph.D. degree in Computer Information, and control engineering from the University of Michigan, Ann Arbor, MI, USA in 1985. Since then, he had been associated with the School of Electrical Engineering at Purdue University as an Assistant Professor until 1987.