# Semantic Segmentation of 3D Point Cloud to Virtually Manipulate Real Living Space

Yuki Ishikawa
*Keio University*
Yokohama, Japan
yuki_ishikawa@hvrl.ics.keio.ac.jp

Ryo Hachiuma
*Keio University*
Yokohama, Japan
ryo-hachiuma@hvrl.ics.keio.ac.jp

Naoto Ienaga
*Keio University*
Yokohama, Japan
ienaga@hvrl.ics.keio.ac.jp

Wakaba Kuno
*Keio University*
Yokohama, Japan
kuno.wakaba.9524.lilly@keio.jp

Yuta Sugiura
*Keio University*
Yokohama, Japan
sugiura@keio.jp

Hideo Saito
*Keio University*
Yokohama, Japan
hs@keio.jp

***Abstract*—This paper presents a method for the virtual manipulation of real living space using semantic segmentation of a 3D point cloud captured in the real world. We applied PointNet to segment each piece of furniture from the point cloud of a real indoor environment captured by moving a RGB-D camera. For semantic segmentation, we focused on local geometric information not used in PointNet, and we proposed a method to refine the class probability of labels attached to each point in PointNet's output. The effectiveness of our method was experimentally confirmed. We then created 3D models of real-world furniture using a point cloud with corrected labels, and we virtually manipulated real living space using Dollhouse VR, a layout system.**

***Index Terms*—furniture arrangement, semantic segmentation, virtual reality**

## I. Introduction

When moving into or renovating a room, people consider how to arrange the furniture, such as desks and chairs, in the finite space of the layout. However, it is challenging to think of an appropriate and easy-to-use layout because many factors are involved, such as ensuring the width through which people can pass between furniture pieces and overcoming restrictions on furniture placement due to the size and shape of each piece. Moreover, furniture can be very large and heavy, making it difficult to move the pieces; thus, people often do not repeatedly rearrange their furniture.

In recent years, many systems to simulate furniture arrangements have been researched and developed. MUJI Interior Simulator [1] is a virtual reality (VR) system that can create a virtual floorplan from 2D and 3D viewpoints and manually place furniture. Make it Home [2] is a VR system that automatically creates indoor 3D scenes, optimizes it, and presents plans for arranging furniture. Many methods also superimpose furniture on the real world using augmented reality (AR). Phan et al. proposed a dynamic and flexible user interface that can manipulate the arrangement and the color of virtual furniture by using multiple markers [3]. In the applications RoomCo AR [4] and Roomle 3D/AR [5], the products of affiliated furniture makers can be placed. In Roomle 3D/AR, it is also possible to experience the created space using VR. In addition, the mixed reality (MR) method uses a fingertip as an input pointer to interact with the virtual environment [6]. In all these methods, though, the furniture is newly added to the real world using a 3D CG model prepared in advance, and existing furniture cannot be moved via simulation.

Therefore, in this study, we present a method for the virtual manipulation of furniture in real living space using semantic segmentation of a 3D point cloud captured in the real world. Semantic segmentation is the task that estimates the class to which each data point belongs and labels each point by its corresponding class. We applied PointNet [7] to segment each piece of furniture from the point cloud of a real captured indoor environment.

Conventionally, many networks have used voxelized representations as 3D data for an indoor-object recognition network [8]–[10]. PointNet uses a point cloud as direct 3D data, which requires less data than a voxelized representation does, and PointNet can process a wide space of indoor data. However, because PointNet does not use local geometric information, objects may be partially labeled with incorrect classes. This causes problem when furniture is chipped or extracted with noise during segmentation. Therefore, in our proposed method, semantic segmentation via PointNet was only the first step; we then refined the class labels, focusing on local geometric information.

In this paper, we also demonstrate virtual manipulation using Dollhouse VR [11], a system that can consider the layout in practice. To date, Dollhouse VR has only been used as a 3D CG model, as with the other described simulation systems, but our proposed method of semantic segmentation uses Dollhouse VR with real-world objects to create interactive layouts.

## II. Proposed approach

The flow of the proposed approach in shown in Fig. 1. First, as pre-processing, we obtained the colorized 3D point cloud $v_i = [x_i, y_i, z_i, R_i, G_i, B_i]$ of an indoor environment via an

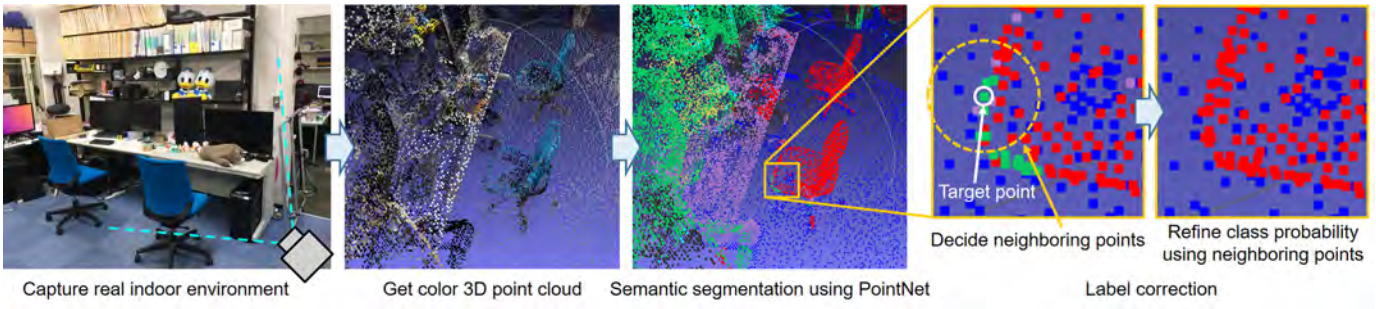



Fig. 1. Flow of the proposed approach.

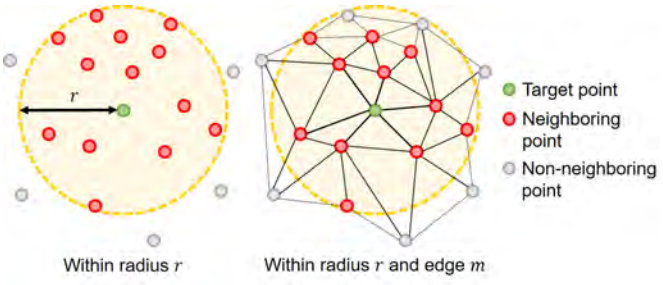



Fig. 2. Decided neighboring points.

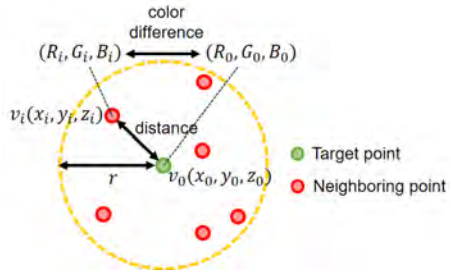



Fig. 3. Values used for refinement.

RGB-D sensor, where $i$ denotes the number of 3D points in the obtained point cloud. We then estimated the class probability $\boldsymbol{P}_i = [p_{i1}, p_{i2}, ..., p_{ik}]$ ($k$: number of class) for each 3D point $v_i$ in the 3D point cloud of an indoor environment. We trained PointNet as the estimator and applied semantic segmentation. With our proposed method, at each point in the cloud, we refined the class probability $\boldsymbol{P}_{new}$ using the class probability of its neighboring points.

### A. Neighboring 3D points decisions

The neighboring points of a target point were decided using the two methods shown in Fig. 2. On the left-hand side of Fig. 2, only the Euclidean distance between the target and each neighboring point was used. All points that fell within radius $r$ from the target were regarded as neighboring points. However, in the real world, different objects may be located near each other. In that case, the points of different objects are included among the neighboring points of a target, which may interfere with the recalculation of class probability and label correction.

Therefore, given that points included for different objects exist on different surfaces, we considered, in addition to the Euclidean distance, the adjacent relationship between points in a mesh, with each point as the vertex. This is the method on the right-hand side of Fig. 2, where all points that fell within radius $r$ and $m$ edges from the target were regarded as neighboring points. Consequently, even if points for different objects were near each other, they were not included among the target's neighboring points unless they were near each other on the surface. Thus, the accuracy of label correction increases.

### B. Class probability refinement

To refine the class probability of the target point $v_0$, we used the class probability $\boldsymbol{P}_0$ of $v_0$ and the class probability $\boldsymbol{P}_i$ $(0 \leq i \leq N)$ of each of the neighboring points $v_i$ decided by either method in Section II-A, where $N$ denotes the number of neighboring points of $v_0$. In the calculation of $\boldsymbol{P}_{new}$, which refined $\boldsymbol{P}_0$, as shown in (1), a weighted average is taken into consideration for the Euclidean distance in 3D space between $v_0$ and $v_i$ and for the Euclidean distance of RGB color space. The weight of the distance $w_i^d$ $(0 \leq w_i^d \leq 1)$ is calculated via (2), and the weight of the color distance $w_i^c$ $(0 \leq w_i^c \leq 1)$ is calculated via (3). Fig. 3 shows the relationship between the values used for these calculations.

$$\boldsymbol{P}_{new} = \frac{\sum_{i=0}^{N}(w_i^d \cdot w_i^c) \cdot \boldsymbol{P}_i}{\sum_{i=0}^{N}(w_i^d \cdot w_i^c)} \tag{1}$$

$$w_i^d = 1 - \frac{\sqrt{(x_i - x_0)^2 + (y_i - y_0)^2 + (z_i - z_0)^2}}{r} \tag{2}$$

$$w_i^c = 1 - \sqrt{\frac{(R_i - R_0)^2 + (G_i - G_0)^2 + (B_i - B_0)^2}{255^2 \cdot 3}} \tag{3}$$

According to the pre-processing of this method,$v_0$ has the label with the maximum probability among $P_0$. By attaching a label that gives the maximum probability among $\boldsymbol{P}_{new}$ calculated via the above method to $v_0$, the label is corrected.

## III. Experiments

### A. Pre-processing

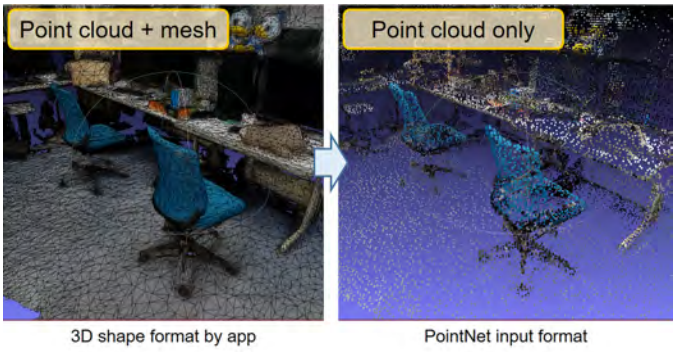



Fig. 4. Created input point cloud.

*1) Acquiring indoor 3D shapes of the real world:* We used the smartphone ZenFone AR, which can use Google Tango [12], to acquire indoor 3D shapes. Tango is a framework for AR/MR released by Google for Android. In this experiment, we used the application Constructor Developer Tool to scan the space and reconstruct the 3D model. This application can acquire 3D shapes with a mesh.

*2) Creating an input point cloud:* PointNet's input format is a colorized 3D point cloud. In the acquired 3D model, the vertex itself had no color information, and only the surface of the mesh was colored by the textured image. Thus, RGB color information was added to each vertex using this texture, and we deleted the mesh information (Fig. 4).

*3) Semantic Segmentation using PointNet:* We used PointNet to perform semantic segmentation, conducting experiments under the following conditions: CPU = Intel Core i7-6850K 3.60GHz; GPU = NVIDIA Quadro P6000, NVIDIA Quadro GV100; and RAM = 32GB. We used Area 6 of the Stanford Large-Scale 3D Indoor Spaces Dataset [13] for training. This is an annotated point cloud dataset of living spaces, including 53 rooms. Thirteen classes were used for training: ceiling, floor, wall, beam, column, window, door, table, chair, sofa, bookcase, board, and clutter.

When applying semantic segmentation by inputting the created point cloud to the trained PointNet estimator, each point of the input point cloud was estimated in terms of the 13 class probabilities and given a class label with the highest probability. Fig. 5 shows the results of PointNet's semantic segmentation. Each point of the output point cloud was given a specific color for each label. Fig. 5(b) shows the correspondence between many visible label colors and classes.

Fig. 5 demonstrates that the labeling provided roughly correct answers when viewing the entire room. However, some noise points occurred among the point clouds labeled with correct labels; thus, not all points were correctly labeled. In Fig. 6, we overlapped PointNet's output (Fig. 5(b)) with the original 3D shape, including a mesh, seen from another viewpoint. As evident in the yellow circle frames in this figure, some points were incorrectly labeled. For example, although the table was, in general, correctly labeled, parts of it were also labeled as a chair; likewise, parts of the chair were labeled as a bookcase.

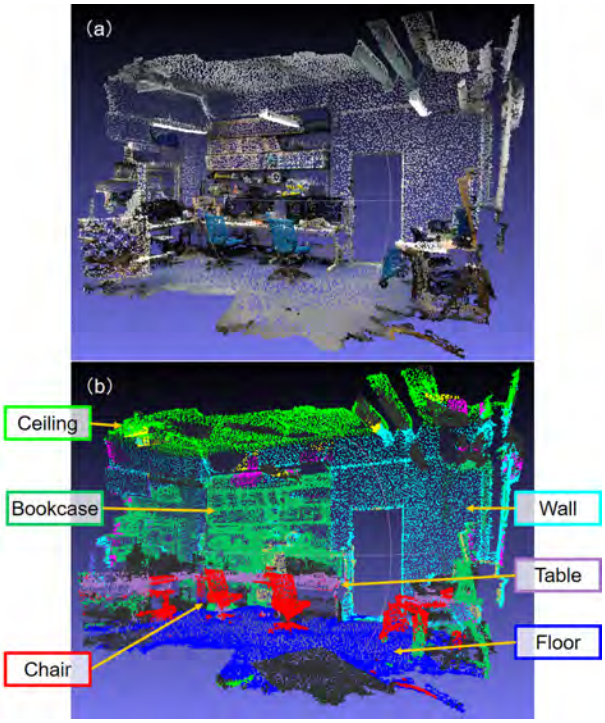



Fig. 5. Semantic segmentation using PointNet: (a) Input point cloud; (b) Output point cloud.

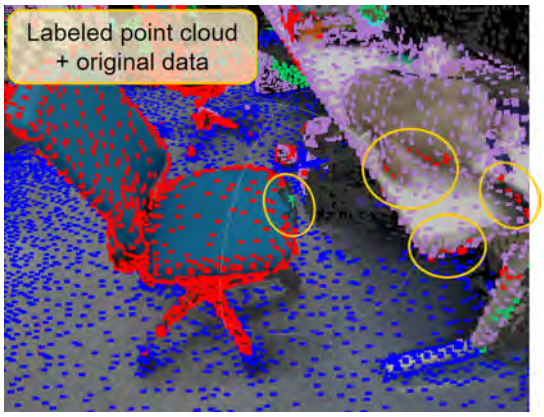



Fig. 6. Incorrect labeling.

## B. Results of Label Correction

Our proposed method of label correction was applied to the PointNet output point cloud. This point cloud had a label color. Therefore, when using the RGB value of the neighboring points to calculate the color distance from the target, we referred to PointNet's input point cloud, where each point had color information of the original 3D shape data. In our proposed method, neighboring points of each target in the point cloud were taken as points entering radius $r$ of the target or points traced within $m$ edges from the target. We set these parameters empirically, conducting experiments under the following three conditions:

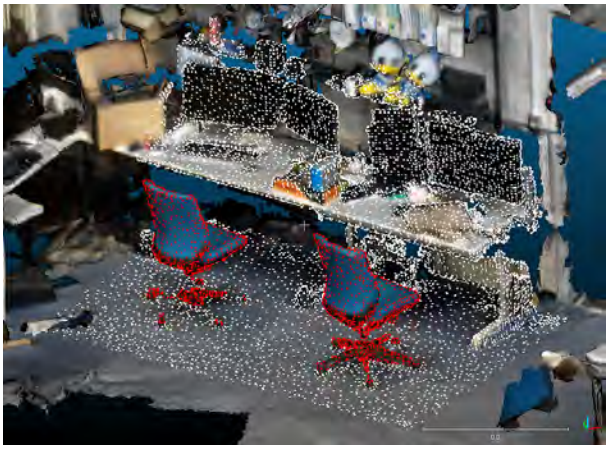


Fig. 8. Ground truth: red = chair (true); white = not chair (false).

TABLE I
PRECISION, RECALL AND F-MEASURE OF POINTNET AND OURS

| Method | | Precision[%] | Recall[%] | F-measure[%] |
|---|---|---|---|---|
| PointNet | | 87.58 | 92.37 | 89.91 |
| Ours | r=10 | 90.07 | 92.67 | 91.35 |
| | r=15 | **90.39** | 92.72 | 91.54 |
| | r=15 and m=2 | 90.32 | **94.39** | **92.31** |

- $r = 10$ [cm]
- $r = 15$ [cm]
- $r = 15$ [cm], $m = 2$

Fig. 7 shows the results of the label correction.

### C. Evaluation

We focused on the chairs as an example of furniture for virtual manipulation. A point cloud manually annotated around the two chairs in the indoor 3D space used in this experiment was set as the ground truth, and the results of label correction were compared with the PointNet output point cloud. The ground truth is a point cloud in an area of red or white color in Fig. 8: red = chair (true); white = not chair (false). Table I shows the calculated precision, recall, and F-measure of the PointNet method and of our method.

Table I demonstrates that all conditions of our proposed method were more accurate than they were for the PointNet output; thus, our proposed method was more effective. In Fig. 7, the noise that the PointNet output labeled incorrectly was reduced or eliminated using our proposed method, and the boundary line between the objects became clearer. The corner of the chair's seat in Fig. 7(b), for example, shows a reduced incorrect label range.

Comparing $r = 10$ and $r = 15$, the latter closed the hole rapidly. However, $r = 15$ lost more the correct label for the chair's legs in Fig. 7(c) than $r = 10$, they labeled the legs as floor. This is apparently due to the refined class probability, since the floor near the legs also had a large number of neighboring points in a wide range. Therefore, $r = 15$, $m = 2$ (which added the adjacent relationship of a mesh) did not close the hole in the model of the chair's seat rapidly in Fig. 7(b), but the most correct label remained on the chair's legs in Fig. 7(c). This is apparently because the legs and the floor are on different surfaces in the mesh, so the floor was not included in the neighboring points. In our proposed method, the mesh was thus effective when making decisions about neighboring points, which is also evident in the fact that the F-measure for our proposed method was the highest.

## IV. VIRTUAL MANIPULATION OF FURNITURE

As described in Section I, there are currently many systems to simulate furniture arrangements. Among them, VR methods that confirm a spatial layout from the first-person viewpoint of someone immersed in that space approximates physical sensations in the real world and is appropriate for our purposes.

In this study, we used Dollhouse VR [11] to perform virtual manipulations of furniture. Dollhouse VR is a system that can consider the layout in terms of VR, performing furniture manipulations from a bird's-eye perspective via a touch display and interactively checking the furniture arrangement from a first-person perspective in VR space via a head-mounted display (HMD). To date, Dollhouse VR has only been used as a 3D CG model, but our proposed method of semantic segmentation allows Dollhouse VR to use real-world objects interactively when considering the layout of an indoor environment.

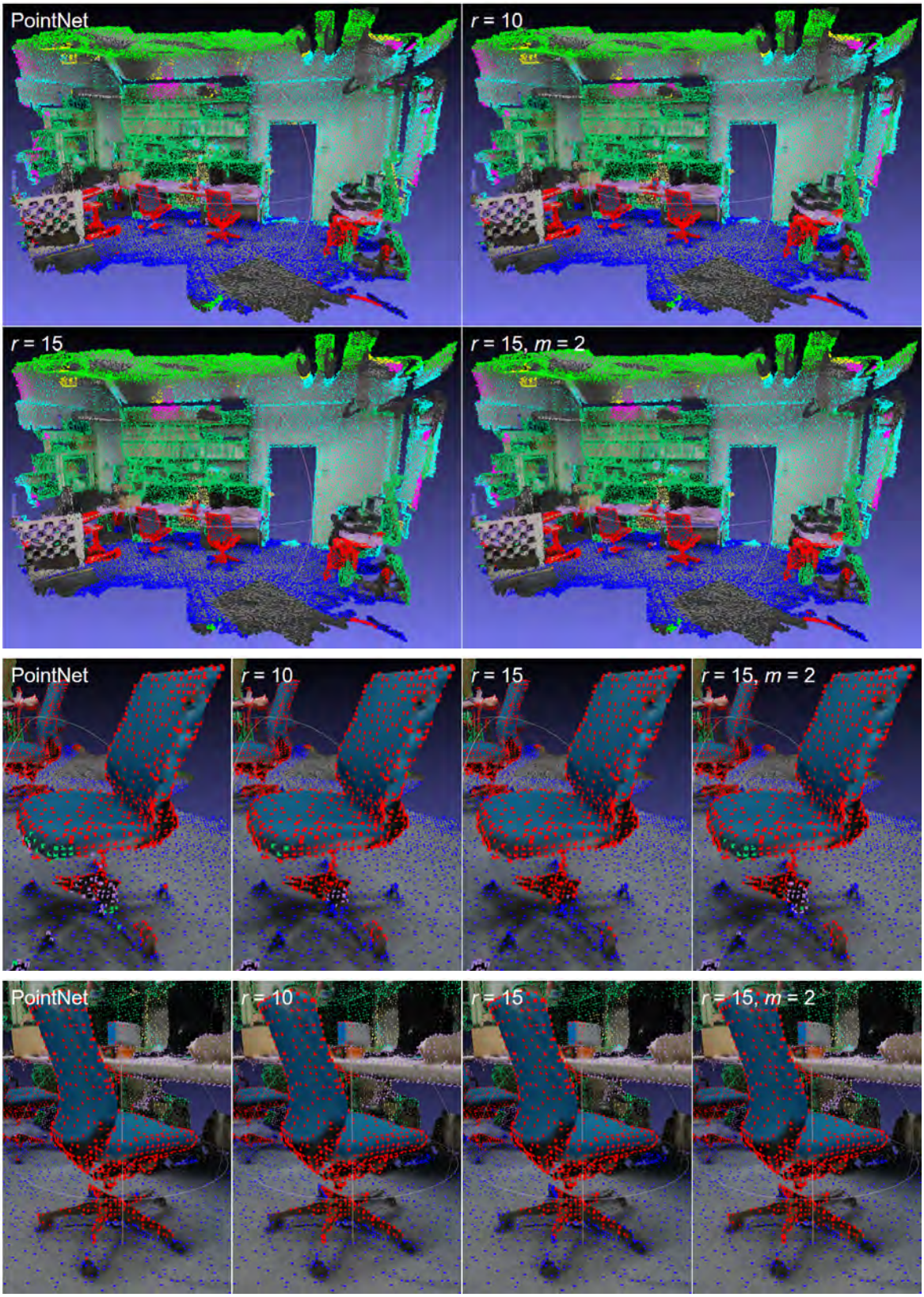
In this section, using the results from Section III, we describe experiments and results when virtually manipulating chairs as an example of furniture.

### A. Wrokflow

Fig. 9 shows the workflow. First, we divided the point cloud by label attached to each point, and acquired labeled points which we manipulate from the point cloud. However, using this process, even if multiple objects and noise are to be moved separately in a point cloud of the same label, they are regarded as the same object. Thus, we then clustered the point clouds of the same label using the k-means method. Here, it is possible to acquire furniture individually by extracting clusters in which the number of points is equal to or larger than a certain value assigned to objects to be manipulated. Finally, we overlapped the clustered results with the original 3D shape, including a mesh, and extracted surface information comprising points where $x$, $y$, $z$ coordinates coincide. Thus, an object with a mesh was created (Fig. 10).

### B. B. Simulation

Using the created object with a mesh, we performed virtual manipulations using Dollhouse VR. Here, we assumed a room renovation, and we manipulated the room's furniture. The data of the original indoor 3D space (a 3D model of the entire room) was divided into furniture to manipulate and other non-manipulated parts. Two chair objects (the furniture to manipulate) were created by the procedure described in Section IV-A. Objects not for manipulation were created by extracting the parts where the original data and the $x$, $y$, $z$ coordinates of the chair coincided and were deleted from the original data.




(c) Right side chair.

Fig. 7. Results of label correction.

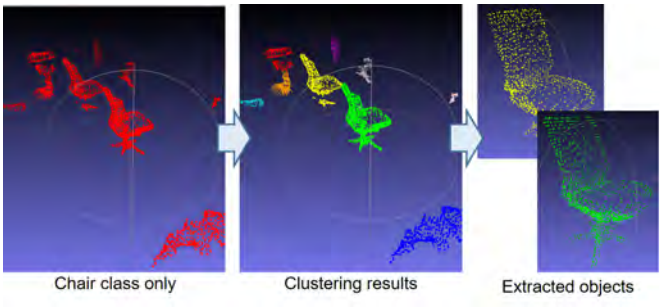



Fig. 9. Clustering by k-means method and extracting objects.

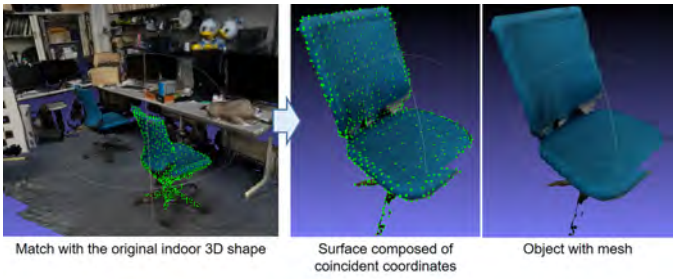



Fig. 10. Creating an object with a mesh.

Finally, these created 3D models and the player's model, as a viewpoint in VR space, were placed in Dollhouse VR. A function was added so that the player could move and rotate the chairs. This 3D model was visible on the touch display in a bird's-eye view, and on the HMD as VR space from the player's viewpoint. By swiping the chairs on the touch display, we could move and rotate them. Fig. 11 shows these virtual manipulation. The user wearing the HMD was able to experience the life-size space from the height of his / her eyes. In addition, the user could look and move around the virtual space by moving in the real world. In terms of the manipulation using touch display, the chairs could be moved and rotated by swiping them. Furthermore, the user could see the furniture being manipulated by virtual fingers on the HMD in real time.

## V. Conclusion

In this study, we proposed a method for the virtual manipulation of real living space using semantic segmentation in a 3D point cloud. For semantic segmentation, we used PointNet as the first step but then proposed a method to refine the class label attached to each point in the PointNet's output. We then experimentally confirmed its effectiveness. In addition, we created a 3D model of furniture using a point cloud to which our proposed method was applied, and we performed virtual manipulations using Dollhouse VR, a virtual layout system. To date, that system has only been used as a 3D CG model, but our method enabled layouts for real living space. When considering practical uses as an extension of this paper, researchers should conduct future studies to determine the accuracy of the 3D model's dimensions and the effectiveness of the model's appearance in terms of light and shadow.

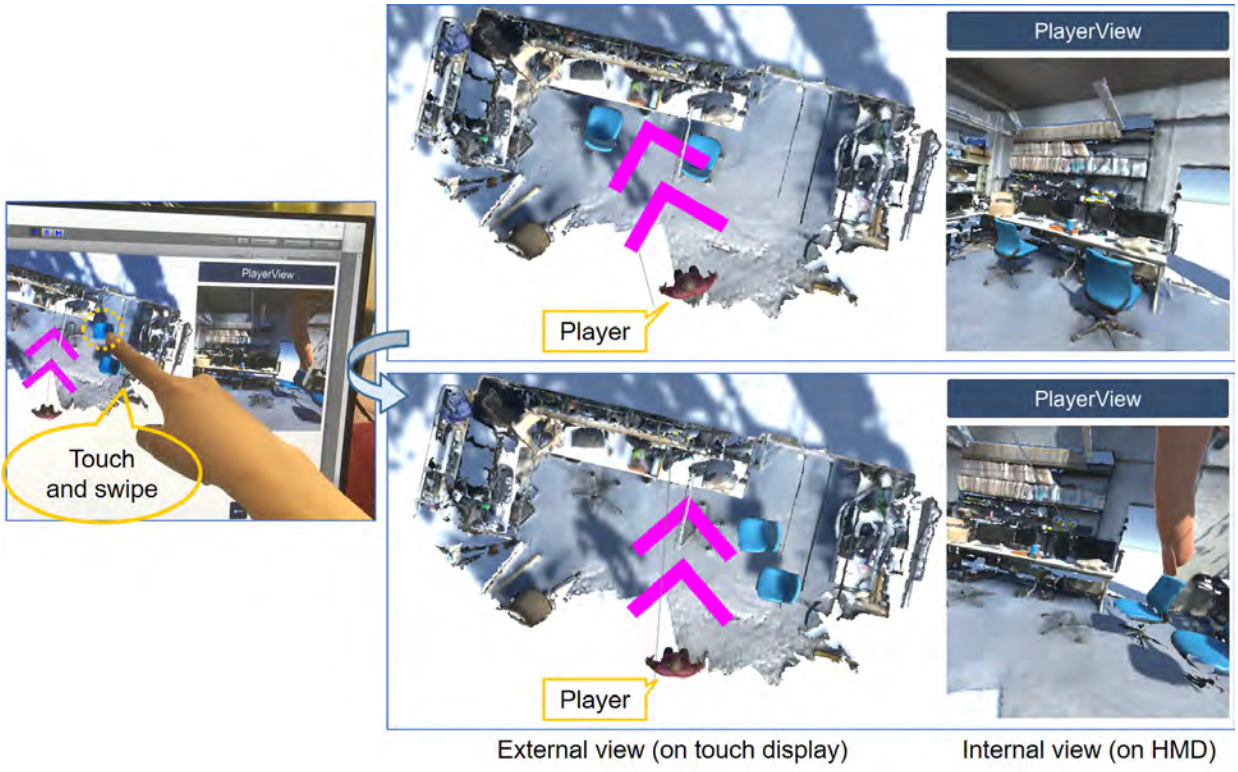



Fig. 11. Virtual manipulation of chairs.

## ACKNOWLEDGMENT

This work was supported by JST AIP-PRISM GrantNumber JPMJCR18Y2.

## REFERENCES

[1] "MUJI interior simulator," https://muji.livingstyle.jp/simulator/, accessed 2019.1.21.

[2] L. F. Yu, S. K. Yeung, C. K. Tang, D. Terzopoulos, T. F. Chan, and S. J. Osher, "Make it Home: automatic optimization of furniture arrangement," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 86:1–86:12, 2011.

[3] V. T. Phan and S. Y. Choo, "Interior design in augmented reality environment," *International Journal of Computer Applications*, vol. 5, no. 5, 2010.

[4] "RoomCo AR," https://www.roomco.jp/, accessed 2019.1.21.

[5] "Roomle 3D/AR," https://www.roomle.com/, accessed 2019.1.21.

[6] D. Herumurti, A. Yuniarti, I. Kuswardayan, W. Nurul, R. R. Hariadi, N. Suciati, and M. G. Manggala, "Mixed reality in the 3d virtual room arrangement," in *11th International Conference on Information & Communication Technology and System (ICTS2017)*. IEEE, 2017, pp. 303–306.

[7] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3d classification and segmentation," *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, vol. 1, no. 2, p. 4, 2017.

[8] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3D ShapeNets: A deep representation for volumetric shapes," in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2015)*, 2015, pp. 1912–1920.

[9] D. Maturana and S. Scherer, "VoxNet: A 3d convolutional neural network for real-time object recognition," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2015)*. IEEE, 2015, pp. 922–928.

[10] 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," in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, 2016, pp. 5648–5656.

[11] H. Ibayashi, Y. Sugiura, D. Sakamoto, N. Miyata, M. Tada, T. Okuma, T. Kurata, M. Mochimaru, and T. Igarashi, "Dollhouse VR: A multi-view, multi-user collaborative design workspace with vr technology," in *SIGGRAPH Asia 2015 Emerging Technologies*. ACM, 2015, p. 8.

[12] "Google Tango," https://get.google.com/intl/ja/tango/, accessed 2019.1.21.

[13] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. Brilakis, M. Fischer, and S. Savarese, "3d semantic parsing of large-scale indoor spaces," in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, 2016, pp. 1534–1543.