DISCOVERING CORRESPONDENCE AMONG IMAGE SETS WITH PROJECTION VIEW PRESERVATION FOR 3D OBJECT DETECTION IN POINT CLOUDS

Tomoaki Yamazaki, Daisuke Sugimura, and Takayuki Hamamoto

Graduate School of Engineering, Tokyo University of Science, 125-8585, Tokyo, Japan

ABSTRACT

We propose a method for detecting objects that correspond to given three-dimensional (3D) point clouds in a scene. We regard the 3D object detection as a series of optimal matching of the object and scene images that are obtained by projecting point clouds into multiple viewpoints. The key novelty of the proposed method is to introduce a constraint imposed by the spatial relationship among the image-projection directions for the object point clouds, to discover the optimal matching of the projected image sets. This constraint allows to evaluate the appearance consistency of the object in multi-viewpoint scene images. Thus, image-projection directions can be effective cues to detect objects even in cluttered scenes, where previous methods are not effective. We estimate the image-projection directions for the object point clouds by applying principal component analysis to the object point clouds and hence include highly discriminative image features. Then, we back-project reliable matching results, which are retrieved from the image set correspondence, into 3D space to achieve a point-wise object detection. Experiments using public datasets demonstrate the effectiveness and performance of the proposed method.

Index Terms— 3D Object Detection, Point Clouds

1. INTRODUCTION

Three-dimensional (3D) object detection using point clouds has been widely investigated. In fact, given that point clouds provide 3D structural information, robust object detection can be achieved. Moreover, techniques for 3D object detection can be applied to a wide range of applications such as robotics, augmented reality, and virtual reality.

Different methods for 3D object detection using point clouds are available [1–13]. In [1–4], the authors proposed algorithms for object detection using 3D local feature descriptors such as: spin image (SI) [1], signature of histograms of orientations (SHOT) [2], fast point feature histogram (FPFH) [3], and etc. These algorithms perform a template matching of 3D local descriptors to detect target objects. Likewise, methods in [5–7] preliminarily apply a (semantic) segmentation of the scene point clouds to match templates with high accuracy. This segmentation allows to effectively search for objects because the target scene is (semantically) localized. However, the above mentioned methods, are not accurate in cluttered scenes, because the feature extraction and segmentation are severely compromised in such scenes.

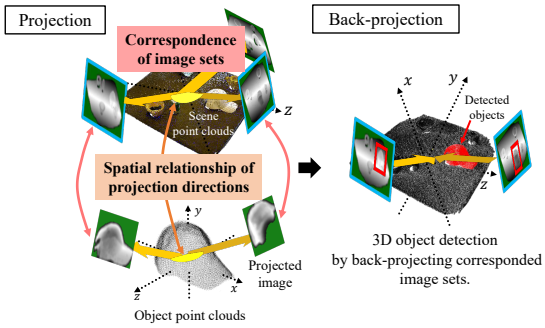


Fig. 1: Illustration of the main idea of the proposed method. We introduce a constraint based on the relationship among image-projection directions of point clouds to achieve a point-wise 3D object detection.

Unlike the template matching-based approaches, other methods [8–10] have rely on classifiers trained using 3D local descriptors [14, 15], or deep features [10], to achieve accurate object detection. However, these methods require the inclusion of many instances (features) to obtain a high accuracy.

Other previous efforts considered 3D object detection as a series of two-dimensional (2D) detection problems in different images [11–13]. Specifically, by projecting 3D point clouds into multi-viewpoint images, 2D object detection algorithms can be applied to detect 3D objects. These algorithms [11, 12] depend on projection of 3D objects obtained from computer-aided design models. However, the algorithms are limited by the information available in a model database, and hence new 3D object models should be created whenever the detection of unavailable objects is required. Likewise, Pang *et al.* [13] proposed a method for 3D object detection by using a series of images obtained from 3D point clouds projections. The method consists of a template matching between projected images of both the target object and the scene. Then, the outcomes from the template matching are back-projected into 3D space.

However, these projection-based methods hinder accurate

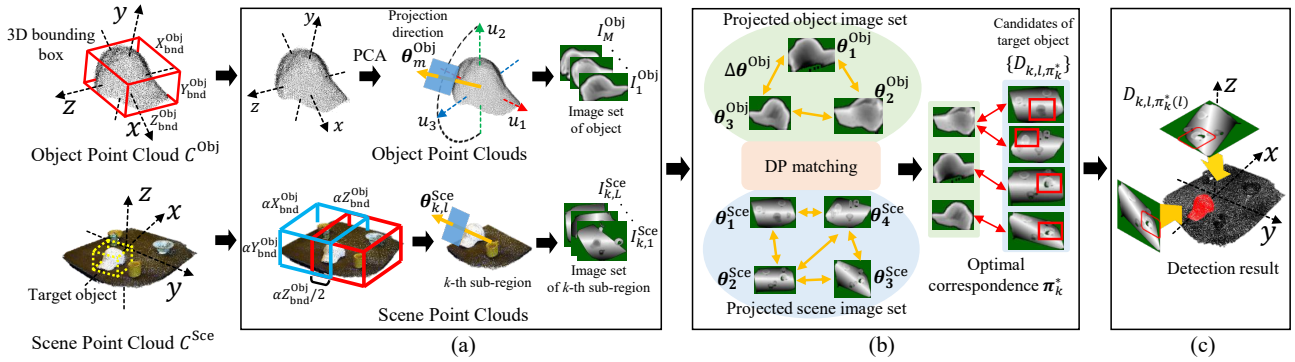


Fig. 2: Diagram of the proposed method. (a) Point clouds projection, (b) Determination of optimal correspondences among image sets, (c) 3D object detection by back-projection.

object detection, and template matching outcomes are less accurate when the projected images of the scene are similar to those of the target object. In the previous work [13], image projection directions are randomly selected, thus providing noisy observations that further degrade accuracy of the template matching. Consequently, this type of method can include false correspondences that are back-projected into 3D space, thus leading to an erroneous 3D object detection. Therefore, the solution to these problems requires (1) preventing false template matching of a single image pair and (2) estimating appropriate image-projection directions.

In this paper, we consider the 3D object detection problem as the determination of optimal correspondence among *image sets*. Unlike previous work [13] that directly uses individual correspondences of randomly projected image pairs, the simultaneous matching of projected image sets allows to evaluate the appearance consistency of the target object in multi-viewpoint scene images. Thus, the image sets provide effective cues for object detection and lead to suppressing the influences of inaccurate matching in individual images.

The main novelty in the proposed method is to introduce a constraint, which is imposed by the spatial relationship among effective image-projection directions for the object point clouds, to determine image set correspondences, as illustrated in Fig.1. We obtain the effective image-projection directions by applying a principal component analysis (PCA) to the object point clouds. Hence, the projected images include highly discriminative features; it allows to improve the accuracy of template matching. Consequently, the proposed method back-projects only reliable candidates in the region of the target object in the scene images into 3D space, thus, providing accurate object detection in point clouds.

2. METHOD OVERVIEW

Figure 2 shows a diagram of the proposed method. First, object point clouds C^{Obj} are projected toward M directions, $\{\boldsymbol{\theta}_1^{\mathrm{Obj}}, \ldots, \boldsymbol{\theta}_M^{\mathrm{Obj}}\}$, that are estimated by PCA. Hence, M projected object images, $I^{\mathrm{Obj}} = \{I_1^{\mathrm{Obj}}, \ldots, I_M^{\mathrm{Obj}}\}$, are obtained. Likewise, scene point clouds C^{Sce} are projected. The scene is preliminary divided into K sub-regions. For each sub-region, a subset of C^{Sce} is projected toward L directions. We define these projection directions as $\boldsymbol{\theta}^{\mathrm{Sce}} = \{\boldsymbol{\theta}_1^{\mathrm{Sce}}, \ldots, \boldsymbol{\theta}_K^{\mathrm{Sce}}\} = \{(\boldsymbol{\theta}_{1,1}^{\mathrm{Sce}}, \ldots, \boldsymbol{\theta}_{1,L}^{\mathrm{Sce}}), \ldots, (\boldsymbol{\theta}_{K,1}^{\mathrm{Sce}}, \ldots, \boldsymbol{\theta}_{K,L}^{\mathrm{Sce}})\}$, and thus obtain $K \times L$ projected scene images $I^{\mathrm{Sce}} = \{I_1^{\mathrm{Sce}}, \ldots, I_K^{\mathrm{Sce}}\} = \{(I_{1,1}^{\mathrm{Sce}}, \ldots, I_{1,L}^{\mathrm{Sce}}), \ldots, (I_{K,1}^{\mathrm{Sce}}, \ldots, I_{K,L}^{\mathrm{Sce}})\}$.

Then, the simultaneous correspondence between the projected object and scene image sets is determined. Finally, the point-wise 3D object detection is achieved by back-projecting the scene images that optimally correspond to those of the object into 3D space.

3. IMAGE PROJECTION OF POINT CLOUDS

3.1. Direction Estimation of Object Projection

To obtain a reliable correspondence of an image pair, we apply PCA to properly estimate the image-projection directions for C^{Obj}. In PCA, the first principal component has the largest C^{Obj} variance, which indicates that the image projection orthogonal to the direction of the first principal component contains plenty of point clouds information, thus providing highly discriminative image features.

Let the eigen vectors corresponding to the first, second and third principal components be $\mathbf{u}_1$, $\mathbf{u}_2$ and $\mathbf{u}_3$, respectively. We perform a parallel projection of C^{Obj} toward the directions that are orthogonal to $\mathbf{u}_1$. Specifically, we set $\boldsymbol{\theta}^{\mathrm{Obj}} = \{\boldsymbol{\theta}_1^{\mathrm{Obj}}, \ldots, \boldsymbol{\theta}_M^{\mathrm{Obj}}\}$ such that its elements are projected at regular angular intervals in the $\mathbf{u}_2 - \mathbf{u}_3$ 2D sub-space. The object image projected toward $\boldsymbol{\theta}_m^{\mathrm{Obj}}$ is represented as

$$I_m^{\mathrm{Obj}} = \mathrm{Proj}\left(C^{\mathrm{Obj}}; \boldsymbol{\theta}_m^{\mathrm{Obj}}\right), \tag{1}$$

where $\mathrm{Proj}(C; \boldsymbol{\theta})$ denotes an operator for parallel image projection of point clouds C to direction $\boldsymbol{\theta}$. In this way, we obtain a set of projected images corresponding to the object. The projection stages in Figure 2 illustrate the image-projection process.

3.2. Projection of Scene Point Clouds

We assume that the scene point clouds C^{Sce} are diverse and widely distributed. In that case, some of the scene point clouds can be projected onto the same pixel in a 2D image. Hence, the structural information of the scene point clouds will be partially lost in the projected image.

To overcome this situation, we divide the scene into K cuboid-shaped sub-scenes. The size of each sub-scene is set as $\alpha X^{\mathrm{Obj}}_{\mathrm{bnd}} \times \alpha Y^{\mathrm{Obj}}_{\mathrm{bnd}} \times \alpha Z^{\mathrm{Obj}}_{\mathrm{bnd}}$, where $X^{\mathrm{Obj}}_{\mathrm{bnd}}$, $Y^{\mathrm{Obj}}_{\mathrm{bnd}}$ and $Z^{\mathrm{Obj}}_{\mathrm{bnd}}$ denote the dimensions in the corresponding axes of the 3D bounding box that encloses C^{Obj}, and $\alpha > 1$ is an adjustment parameter. Each sub-scene overlaps with its neighboring sub-scenes at each axis to avoid splitting latent object point clouds in the scene. We set the overlap with respect to each axis as $\alpha X^{\mathrm{Obj}}_{\mathrm{bnd}}/2$, $\alpha Y^{\mathrm{Obj}}_{\mathrm{bnd}}/2$ and $\alpha Z^{\mathrm{Obj}}_{\mathrm{bnd}}/2$.

Based on this scene representation, the scene point clouds C^{Sce} can be represented as a series of subsets: $C^{\mathrm{Sce}} = \{C^{\mathrm{Sce}}_1, \ldots, C^{\mathrm{Sce}}_K\}$. Each subset C^{Sce}_k is projected toward directions $\{\boldsymbol{\theta}^{\mathrm{Sce}}_{k,1}, \ldots, \boldsymbol{\theta}^{\mathrm{Sce}}_{k,L}\}$. Specifically, these projection directions are selected randomly and uniformly at multiple depth layers, as indicated in [13]. The projected scene images in the k-th sub-region are represented as $I^{\mathrm{Sce}}_k = \{I^{\mathrm{Sce}}_{k,1}, \ldots, I^{\mathrm{Sce}}_{k,L}\}$, from which we obtain a set of the projected images of the scene $I^{\mathrm{Sce}} = \{I^{\mathrm{Sce}}_1, \ldots, I^{\mathrm{Sce}}_K\}$.

4. 2D OBJECT DETECTION

To aggregate candidates that belong to the target object region, we perform 2D object detection by template matching.

Similar to [13], we extract gradient-based image features from the projected images, and use them for computing a normalized cross-correlation, $f_{k,l,m}$, between I^{Obj}_m and $I^{\mathrm{Sce}}_{k,l}$. Basically, template matching extracts region $D_{k,l,m}$ from $I^{\mathrm{Sce}}_{k,l}$ with the highest $f_{k,l,m}$.

By applying the template matching to every pair of I^{Obj} and I^{Sce}_k, we obtain a set of the candidate object regions and the corresponding similarity values for k-th sub-region, $\{D_{k,l,m}\}_{1\le l\le L, 1\le m\le M}$ and $\{f_{k,l,m}\}_{1\le l\le L, 1\le m\le M}$.

5. DISCOVERING OPTIMAL CORRESPONDENCE AMONG IMAGE SETS

To improve detection performance, we estimate an optimal correspondences among image sets $\{I^{\mathrm{Obj}}_1, \ldots, I^{\mathrm{Obj}}_M\}$ and $\{I^{\mathrm{Sce}}_{k,1}, \ldots, I^{\mathrm{Sce}}_{k,L}\}$. We compute this processing for each sub-region. We define the correspondence between $I^{\mathrm{Obj}}_1, \ldots, I^{\mathrm{Obj}}_M$ and $I^{\mathrm{Sce}}_{k,1}, \ldots, I^{\mathrm{Sce}}_{k,L}$ as $\boldsymbol{\pi}_k = (\pi_k(1), \ldots, \pi_k(l), \ldots, \pi_k(L))$ where $\pi_k(l) = m$ is a correspondence between the m-th object image I^{Obj}_m and the l-th scene image $I^{\mathrm{Sce}}_{k,l}$. We obtain the optimal correspondence $\boldsymbol{\pi}^*_k$ using direct programming (DP) matching as follows:

$$\boldsymbol{\pi}^*_k = \underset{\boldsymbol{\pi}_{k,s}}{\arg\min} \left[\min_{\boldsymbol{\pi}'_{k,s}} \left\{ \sum_{l=s}^{L} \left(\underbrace{\left(1 - f_{k,l,\pi'_{k,s}(l)}\right)}_{E_1} + \underbrace{\frac{1}{2}\left|\cos(\boldsymbol{\theta}^{\mathrm{Sce}}_{k,l}, \boldsymbol{\theta}^{\mathrm{Sce}}_{k,l-1}) - \cos(\boldsymbol{\theta}^{\mathrm{Obj}}_{\pi'_{k,s}(l)}, \boldsymbol{\theta}^{\mathrm{Obj}}_{\pi'_{k,s}(l-1)})\right|}_{E_2} \right) \right\} \right], \tag{2}$$

where $\boldsymbol{\pi}_{k,s}$ is the optimal correspondence obtained when the s-th scene image $I^{\mathrm{Sce}}_{k,s}$ is set as the initial node in performing the DP matching, the first term E_1 describes the dissimilarity of template matching between $I^{\mathrm{Sce}}_{k,l}$ and $I^{\mathrm{Obj}}_{\pi'_{k,s}(l)}$ (see Sect. 4), and the second term E_2 is a constraint based on the cosine similarity to preserve the spatial relationship of the image-projection directions for C^{Obj} when searching the optimal $\boldsymbol{\pi}_{k,s}$.

6. 3D OBJECT DETECTION BY BACK-PROJECTION

We finally perform a point-wise 3D object detection by back-projecting the candidates in the target object region that correspond to object images $\{D_{k,l,\pi^*_{k(l)}}\}_{1\le k\le K, 1\le l\le L}$ into 3D space. We group the N points that belong to $\{D_{k,l,\pi^*_{k(l)}}\}$ in set $C^{\mathrm{det}} = \{c^{\mathrm{det}}_1, \ldots, c^{\mathrm{det}}_n, \ldots, c^{\mathrm{det}}_N\}$.

The back-projection from the corresponding views provide each c^{det}_n a confidence value w_n that characterizes whether it represent the target object. The confidence value w_n is calculated as $w_n = a_n \cdot b_n$, where a_n is a weight based on its number of gradient pixels in the image which c^{det}_n belongs to, as was used in [13]. The other weight b_n provides a higher value, when the projected scene image including c^{det}_n corresponds to the object image with a projection direction being close to $\mathbf{u}_3$.

We consider the point clouds that have higher confidence values w_n than threshold th as the final detection results: $\mathbf{x}^* = \{c^{\mathrm{det}}_n | w_n > th\}$.

7. EXPERIMENTS

To verify the effectiveness of the proposed method, we conducted experiments for 3D object detection using 3D Keypoint Detection Benchmark [16, 17] and RGB-D Scenes Dataset v.2 [18].

We compared our method with that proposed by Pang *et al.* [13] and methods using 3D local descriptors (SI [1], SHOT [2] and FPFH [3]). In addition, we tested our method without using the constraint based on the spatial relationship among image-projection directions (i.e., we omitted the second term E_2 in the righthand side of Eq. (2) in the minimization by DP matching), and refer to it as "Ours w/o constraint".

Table 1: Comparison results of precision rate (P), recall rate (R), and F-measure (F) for the proposed method and comparison methods on these datasets. The best and the second best scores are represented in **bold** and with underline, respectively.

Dataset	3D Keypoint Detection Benchmark [16, 17]									RGB-D Scenes Dataset v.2 [18]						Average		
Sequence	Scene1View3_0.1			Scene3View0_0.3			Scene5View3_0.1			01			05					
	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F
Ours	**0.941**	0.737	**0.825**	**0.589**	0.727	**0.628**	**0.875**	0.947	**0.908**	0.614	0.598	0.599	**0.653**	0.847	**0.707**	**0.717**	0.756	**0.716**
Ours w/o constraint	0.720	0.682	0.671	0.425	0.934	0.506	0.677	0.870	0.741	0.485	0.587	0.476	0.561	0.597	0.512	0.563	0.710	0.567
SI [1]	0.333	**1.00**	0.499	0.333	**1.00**	0.444	0.333	**1.00**	0.498	0.03	**1.00**	0.05	0.08	**1.00**	0.138	0.192	**1.00**	0.287
SHOT [2]	0.480	0.799	0.557	0.334	0.981	0.444	0.463	0.847	0.553	**0.974**	0.927	**0.949**	0.573	0.201	0.197	0.528	0.823	0.567
FPFH [3]	0.333	**1.00**	0.499	0.333	**1.00**	0.444	0.333	**1.00**	0.498	0.958	0.705	0.811	0.229	0.06	0.09	0.483	0.710	0.487
Pang *et al.* [13]	0.623	0.826	0.656	0.398	0.927	0.477	0.557	0.783	0.630	0.369	0.687	0.397	0.256	0.740	0.303	0.422	0.786	0.472

Table 2: Comparison results of computation time. The best and the second best scores are represented in **bold** and with underline, respectively.

	Ours	SI [1]	SHOT [2]	FPFH [3]	Pang *et al.* [13]
"Scene1View3_0.1" [16, 17]	3.30s	37.3s	47.1s	29.6s	**2.39s**
"05" [18]	13.1s	366s	451s	263s	**11.9s**

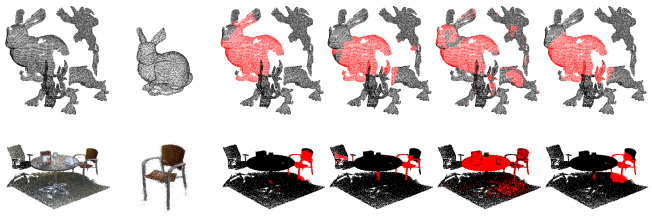
We set the parameters for the proposed method and that proposed by Pang *et al.* as follows. The number of image projections for C^{Sce} and C^{Obj} were $L = 9$ and $M = 6$, respectively. The spatial resolution of each projected image was 640×480. For the other comparison methods, we set the parameters according to those reported in PCL 1.8.0 [19].

For a quantitative evaluation, we used precision rate (P), recall rate (R) and F-measure (F), which are calculated as $\mathrm{P} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP})$, $\mathrm{R} = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$ and $\mathrm{F} = 2\mathrm{PR}/(\mathrm{P} + \mathrm{R})$, where TP, FP, and FN denote the number of true positives, false positives, and false negatives, respectively. True positives (TP) indicate the number of point clouds that are correctly detected as corresponding to the object; false positives (FPs) indicate the number of point clouds that are incorrectly detected as corresponding to the object; and false negatives (FNs) indicate the number of point clouds that are wrongly detected as scene point clouds (i.e., undetected object point clouds).

Table 1 shows the comparison results for each sequence that we tested. Figure 3 shows the visual comparisons in 3D object detection results for the sequences "Scene1View3_0.1" of the dataset [16, 17], and "05" of the dataset [18], respectively. Figure 4 shows the precision-recall curves for all the methods in the sequence "Scene1View3_0.1" of the dataset [16, 17] and "05" of the dataset [18]. We can see that our method retrieves better results than the other comparison methods.

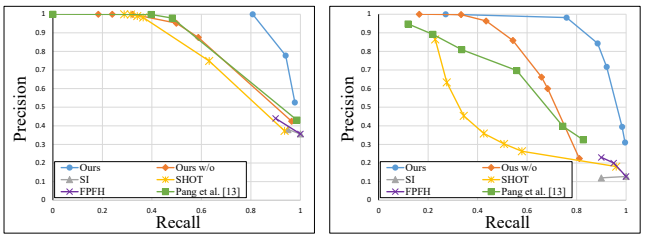
We also compared computational cost for each method. Experiments were run on a Windows PC with Intel Core i7-6700 3.40 GHz and 32 GB RAM. Table 2 shows their comparison results. We see that our result is slightly inferior to that of Pang *et al.*'s method [13], but favorably compared with the other methods.

Given these experimental results and analyses, we would like to state that our method outperforms the other state-of-the-art methods.



(a) (b) (c) (d) (e) (f)

Fig. 3: Comparison in 3D object detection results. (Upper row) Results for "Scene1View3_0.1" of the dataset [16, 17]; (Lower row) Results for "05" of the dataset [18]. (a) Input scene point clouds; (b) Input object point clouds; (c) Ours; (d) Ours w/o constraint; (e) SHOT [2]; (f) Pang *et al.* [13]. Note that the red points indicate detection results.



(a) "Scene1View3_0.1" [16, 17] (b) "05" [18]

Fig. 4: Comparison results of Precision-Recall curves for the proposed method and comparison methods.

8. CONCLUSION

We proposed a method for 3D object detection using point clouds by exploring the simultaneous correspondence of the projected object and scene image sets. We incorporated a constraint imposed by the spatial relationship among image-projection directions, which are estimated using PCA, to determine the optimal correspondence of the image sets. The simultaneous correspondence of image sets allows to obtain reliable candidates that belong to the target object region. We achieved a successful 3D object detection by back-projecting the best candidates into 3D space, as verified from the results obtained using public datasets.

9. REFERENCES

[1] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scenes," *IEEE TPAMI*, vol. 21, no. 5, pp. 433–449, 1999.

[2] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. ECCV*, 2010, pp. 356–369.

[3] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *Proc. IEEE ICRA*, 2009, pp. 3212–3217.

[4] F. Tombari, S. Salti, and L. Di Stefano, "Unique shape context for 3d data description," in *Proc. ACM workshop on 3D object retrieval*, 2010, pp. 57–62.

[5] J. Huang and S. You, "Detecting objects in scene point cloud: A combinational approach," in *Proc. IEEE 3DTV-CON*, 2013, pp. 175–182.

[6] A. Golovinskiy, V. G. Kim, and T. Funkhouser, "Shape-based recognition of 3d point clouds in urban environments," in *Proc. IEEE ICCV*, 2009, pp. 2154–2161.

[7] N. Gunji, H. Niigaki, K. Tsutsuguchi, T. Kurozumi, and T. Kinebuchi, "3d object recognition from large-scale point clouds with global descriptor and sliding window," in *Proc. ICPR*, 2016, pp. 721–726.

[8] G. Pang and U. Neumann, "Training-based object recognition in cluttered 3d point clouds," in *Proc. IEEE 3DTV-CON*, 2013, pp. 87–94.

[9] S. Song and J. Xiao, "Sliding shapes for 3d object detection in depth images," in *Proc. ECCV*, 2014, pp. 634–651.

[10] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proc. IEEE CVPR*, 2015, pp. 1912–1920.

[11] D. Chen, X. Tian, Y. Shen, and M. Ouhyoung, "On visual similarity based 3d model retrieval," in *Computer graphics forum*, 2003, vol. 22, pp. 223–232.

[12] S. Bai, X. Bai, Z. Zhou, Z. Zhang, Q. Tian, and L. J. Latecki, "Gift: Towards scalable 3d shape retrieval," *IEEE TMM*, vol. 19, no. 6, pp. 1257–1271, 2017.

[13] G. Pang and U. Neumann, "Fast and robust multi-view 3d object recognition in point clouds," in *Proc. IEEE 3DV*, 2015, pp. 171–179.

[14] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE CVPR*, 2001, pp. 511–518.

[15] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Proc. IEEE ISMAR*, 2011, pp. 127–136.

[16] A. Mian, M. Bennamoun, and R. Owens, "Three-dimensional model-based object recognition and segmentation in cluttered scenes," *IEEE TPAMI*, vol. 28, no. 10, pp. 1584–1601, 2006.

[17] A. Mian, M. Bennamoun, and R. Owens, "On the repeatability and quality of keypoints for local feature-based 3d object retrieval from cluttered scenes," *IJCV*, vol. 89, no. 2-3, pp. 348–361, 2010.

[18] K. Lai, L. Bo, and D. Fox, "Unsupervised feature learning for 3d scene labeling," in *Proc. IEEE ICRA*, 2014, pp. 3050–3057.

[19] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *Proc. IEEE ICRA*, 2011, pp. 1–4.