Pattern Recognition Letters 128 (2019) 148–154

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

3D object recognition and pose estimation for random bin-picking using Partition Viewpoint Feature Histograms

Deping Li [a,b,c], Ning Liu [a,b,c,*], Yulan Guo [d,e], Xiaoming Wang [a], Jin Xu [c]

[a] College of Information Science and Technology, Jinan University, Guangzhou 510632, PR China
[b] Department of Electronic Engineering, College of Information Science and Technology, Jinan University, Guangzhou 510632, PR China
[c] Robotics Research Institute of Jinan University, Guangzhou 510632, PR China
[d] College of Electronic Science, National University of Defense Technology, Changsha 410073, PR China
[e] School of Electronics and Communication Engineering, Sun Yat-sen University, Guangzhou 510275, PR China

ARTICLE INFO

Article history:
Received 29 August 2018
Revised 28 June 2019
Accepted 20 August 2019
Available online 27 August 2019

Keywords:
Point cloud
Feature descriptor
Bin-picking

ABSTRACT

3D object recognition and pose estimation are challenging tasks in industrial scenarios. In this paper, we propose an accurate and robust algorithm for object recognition and 6DOF pose estimation for bin-picking applications. We first split the whole point cloud of the object into four parts according to the bounding box of the point cloud. The surface shape characteristics are extracted from these four parts using an extended fast point feature histogram and an extended viewpoint direction component. These surface shape characteristics of four parts are further concatenated to generate a Partition Viewpoint Feature Histogram (PVFH) descriptor. Comparison with the state-of-the-art global descriptors have demonstrated the effectiveness of PVFH. Experimental results on industrial parts demonstrate that the PVFH feature descriptor ensures more accurate pose estimation, and higher computational efficiency.

1. Introduction

3D object recognition and pose estimation are well-studied problems in computer vision due to their numerous applications in scene understanding, robotics, and virtual reality. Previous studies have proposed efficient algorithms for object recognition and pose estimation in household environments [15,16]. However, these algorithms are unsuitable for industrial scenarios [20]. Compared to daily objects in household environments, objects in industrial scenario have two distinctive features. First, these objects usually share the same color. Second, many industrial objects are composed of shape primitives such as cylinders, spheres and planes. Features of points on these shape primitives are very similar. Consequently, it is more difficult for these algorithms to recognize and localize industrial objects than daily objects.

The Viewpoint Feature Histogram (VFH) [28] descriptor is an effective 3D feature descriptor for object recognition and pose identification in bin-picking tasks with six-Degrees-Of-Freedom (6DOF). The VFH descriptor is robust against a large degree of surface noise and missing depth information. However, pose estimation may fail when objects are symmetrically placed against to the viewpoint [6]. To address this limitation, a Partition Viewpoint Feature Histogram (PVFH) descriptor is proposed in this paper. In the PVFH descriptor, the whole point cloud of an object is split into four parts according to the bounding box of the whole point cloud. Each point cloud is then used to build a Darboux coordinate system to characterize the surface of the object. The characteristic features of these four parts are finally concatenated to form the PVFH descriptor. Evaluation results demonstrate that the PVFH feature descriptor achieves more accurate pose estimation, and higher computational efficiency.

In the remainder of this paper, we first discuss related work in Section 2. Section 3 describes the system overview of the recognition pipelines of our proposed method. Section 4 introduces the VFH descriptor and the PVFH descriptor. Section 5 presents experimental analyses of our proposed algorithm. Conclusions are drew in Section 6.

2. Related work

Several state-of-the-art object recognition and pose estimation algorithms have been proposed in the literature. Multimodal-LINE (LINEMOD) was proposed by exploiting both depth and color images [15]. Hinterstoisser et al. [16] improved LINEMOD to achieve an average recognition rate of 96.60% and a speed of 119 ms/frame

* Corresponding author.
E-mail address: tliuning@jnu.edu.cn (N. Liu).

https://doi.org/10.1016/j.patrec.2019.08.016
0167-8655/

on their ACCV3D dataset. Rios-Cabrera et al. [27] proposed Discriminatively Trained Templates (DTT) based on LINEMOD. These algorithms can achieve a high recognition rate and speed. However, if color information is unavailable, their performance declines [20].

One promising pose estimation algorithm was proposed by [8]. The algorithm combines an efficient voting scheme with point pair features without use of color information. Another advantage of the algorithm is that this algorithm is robust to occlusion. Since then, several algorithms were proposed based on it. Choi et al. [7] used boundary points with directions and boundary line segments to match planar industrial objects for perform bin picking. Birdal et al. [5] incorporated a coarse-to-fine segmentation, a weighted Hough voting, an interpolated recovery of pose parameters and an occlusion-aware ranking method into the original algorithm [8]. Hinterstoisser et al. [17] introduced a more effective sampling strategy with improved the pre-processing and post-processing steps. This method achieved good results on daily objects of the ACCV3D dataset [16] and the Occlusion Dataset [19]. Wu et al. [35] also performed bin picking based on [8].

3D keypoint descriptors are also used for pose estimation on data with only depth information. A set of descriptors are available in Point Cloud Library (PCL) [31]. Frome et al. [10] extended 2D Shape Context (SC) method to 3D point clouds to propose a 3DSC descriptor. Flint et al. [9] introduced a THRIFT descriptor by extending the 2D Scale Invariant Feature Transform (SIFT) [22] descriptor to 3D space. Tombari et al. [33] introduced a unique and unambiguous LRF and proposed the Signature of Histograms of OrienTations (SHOT). Guo et al. [14] proposed a method for the construction of LRF and the extraction of Rotational Projection Statistics (RoPS) descriptor by projecting and quantizing the rotated neighbors on the 2D *xy*, *xz*, and *yz* planes. These descriptors can achieve a relatively high descriptiveness and stability[12,13], but take a lot of computation time.

Several algorithms have also been proposed to recognize objects by decomposing point clouds into geometric primitives [25]. Liu et al. [21] developed a multi-flash camera to estimate depth edges. Edges are then matched with object templates using directional chamfer matching. Schnabel et al. [32] detected planes, spheres, cylinders, cones and tori in the presence of outliers and noise using the RANSAC method. Holz et al. [18] detected shape and contour primitives to achieve object recognition. However, this algorithm is only applicable to objects that can be described by contour and shape primitives. It cannot be applied to arbitrary synthetic objects.

3. System overview

The process of PVFH based object recognition and pose estimation method is shown in Fig. 1. There are two phases in our process: offline training and online identification.

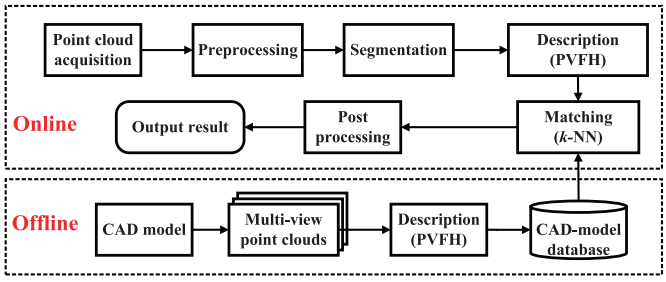

Fig. 1. The block diagrams of PVFH based object recognition and pose estimation pipeline.

3.1. Offline training

In this section, we will describe the process to build the CAD-model database. During offline training, the 3D meshes of objects are transformed to partial point clouds to simulate the data produced by a 3D sensor. To obtain multi-view point clouds of distinguishable viewpoints, several virtual cameras are uniformly placed around the CAD model of the object on a bounding sphere with a radius large enough to enclose the CAD model of the object. To obtain uniform sampling around the object, the sphere is generated using an icosahedron as the initial shape and each triangular face of the icosahedron is then subdivided with four equilateral triangles. This is done recursively for each face until the desired level of recursion is reached. The level of recursion indicates the number of triangles for the representation of the approximated sphere. Virtual cameras are finally placed at the vertices or the polygon centers of these triangles, and partial views of the mesh are obtained by sampling the depth buffers using the graphic card.

There are two main parameters that govern this process: 1) the level of recursion and 2) the resolution of synthetic depth images. In [1], the level of recursion is set to one (80 views are generated) and a resolution of 150×150 gives an appropriate level of detail over the object. However, 80 views are insufficient for rotationally symmetric objects. To obtain enough distinguishable viewpoints, we rotate the partial point clouds around z axis with a set of angles to obtain new partial point clouds. In our experiments, the level of recursion is set to one and the angle is set to 36°. Once the views are generated for each model and the partial point clouds are rotated, the PVFH features are computed on these partial views and the descriptors are stored for future use. Moreover, a transformation between the coordinates of each model and its view coordinates is also saved. This transformation can be used to transform the view to the model. Therefore, the full 3D model can be obtained by applying the transformation for each view and fusing these transformed views together.

3.2. Online identification

The online identification phase consists of several steps, including point cloud acquisition, preprocessing, segmentation, description, matching, and post processing.

3.2.1. Point cloud acquisition

The point cloud data can be acquired by different techniques such as stereo vision [20], structured light [26], or Time-of-Flight (TOF) [11]. In this work, we use structured light to capture point cloud data. Structured light can provide dense and highly accurate 3D data of the scene without addressing the correspondence problems (which typically present in passive triangulation methods such as stereo vision) and without the need for scanning or moving parts [3].

3.2.2. Preprocessing

The raw point cloud produced by a 3D sensor usually requires preprocessing before being handed to object recognition algorithms. Due to the large number of points in a point cloud, some filtering methods are required to reduce the number of points. In this paper, we applied a pass-through filter and a voxel grid filter (which are available in PCL) on raw point cloud to reduce undesirable data. The pass-through filter is used to remove points with the z values that are too large or too small. The voxel grid filter is used to reduce the density of points in the point cloud. Specifically, a voxel grid is first generated with a specified unit size over the point cloud and all points falling inside each voxel are then represented by the centroid of these points.

3.2.3. Segmentation

Objects have to be segmented from the scene for the PVFH based object recognition method. Segmentation is a well investigated topic and several techniques are available [23]. In this work, we use an effective method based on the extraction of a dominant scene plane and a Euclidean clustering step on the remaining points after dominant plane removal. The clustering step is guided by a threshold t, which indicates how close two points are considered to belong to the same object. Therefore, for a successful segmentation, the method requires that different objects are far away from each other with a minimal distance t. This method has a strong prior due to its assumption on a dominant plane, such as a table or a floor. Nevertheless, such an assumption is commonly used in robotic scenarios where the structure of human-made environment is exploited. PCL provides several components to perform point cloud segmentation, more details of this segmentation method can be found in [29].

3.2.4. Description and matching

The outcome of segmentation represents the potential objects in the scene. The shape and geometry information of each object is described by a PVFH descriptor. Afterwards, the PVFH histograms are independently compared against those obtained during the offline training stage. More specifically, a descriptor of PVFH is compared with all histograms in database using the Euclidean distance and the best k-Nearest Neighbor (k-NN) are retrieved. The k NN represent the k-best similar views in the offline training set obtained by the PVFH feature descriptor and the distance metric. Several neighbors rather than a single NN are usually retrieved, each retrieved object yielding an object hypothesis. The 6DOF pose associated to each object hypothesis can be greatly improved through pose estimation and post processing.

3.2.5. Post processing

The postprocessing stage is used to improve the recognition result. One first step, the center points of the objects in the scene and each matched object instance are computed and a center point translation matrix is obtained. Therefore, the partial view candidates obtained by matching can be aligned using the translation matrix. This alignment is further refined using a surface registration method such as the Iterative Closes Point (ICP) [4] algorithm. As a postprocessing step, ICP algorithm is used to find a precise alignment between the source point cloud and the target point cloud. After ICP refinement, we use a hypothesis verification method [2] especially conceived for global pipelines to reject false hypotheses. The hypothesis verification method is based on reordering the k candidates list using a metric based on the number of inliers between the object cluster in the scene and the aligned training views. The number of inliers can be efficiently computed by counting the number of points on the object of scene that have a neighbors on the candidate views within a certain distance (e.g., 5 mm).

4. The PVFH descriptor

In this section, we first describe in details the VFH [30] descriptor, then we introduce the novel PVFH descriptor, which characterizes an object' s pose and enhances the system' s ability to identify objects with mirrored poses.

4.1. VFH descriptor

The VFH descriptor [30] represents four angular distributions of surface normals, including the extended FPFH [28] and a view feature, as shown in Fig. 2. Let p_c and p_i denote the gravity center of a point cloud and a point on the point cloud, respectively. n_c is

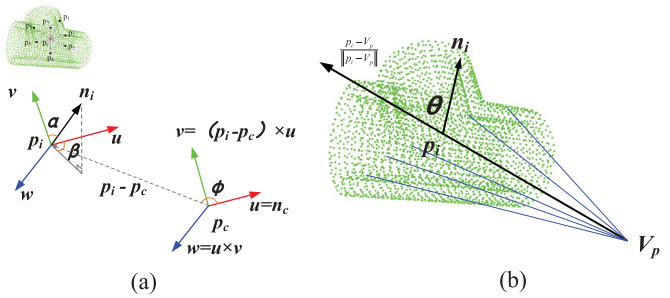

Fig. 2. Object description. (a) extended FPFH and (b) viewpoint feature. α is the angle between the point normal n_i and the v axes, ϕ is the angle between the vector of $(p_i - p_c)$ and the u axis, β is the angle between the projection of point normal n_i on the uw plane and the u axis.

the vector located at p_c, its coordinate is equal to the average of all surface normals. n_i is the surface normal at point p_i. The Darboux coordinate frame (u, v, w) of each point p_i is defined as:

$$u = n_c \tag{1}$$

$$v = \frac{p_c - p_i}{\|p_c - p_i\|} \times u \tag{2}$$

$$w = u \times v \tag{3}$$

The extended FPFH component, including the normal angular deviations $cos(\alpha_i)$, $cos(\phi_i)$, β_i, and the viewpoint direction component for each point p_i and its normal n_i are given by

$$cos(\alpha_i) = v \cdot n_i \tag{4}$$

$$cos(\phi_i) = u \cdot \left(\frac{p_i - p_c}{\|p_i - p_c\|} \right) \tag{5}$$

$$\beta_i = arctan\left(\frac{w \cdot n_i}{u \cdot n_i} \right) \tag{6}$$

$$\theta_i = n_i \cdot \left(\frac{p_i - V_c}{\|p_i - V_c\|} \right) \tag{7}$$

where p_i is point on the point cloud, p_c is the center of point cloud, n_i is the normal of the point p_i, u, v, w is the Darboux coordinate frame (as shown in Eq. (1)), V_c is the viewpoint.

For the components of extended FPFH including $cos(\alpha_i)$, $cos(\phi_i)$, β_i, a histograms of 45 bins is computed, and a histogram of 128 bins is generated for θ_i. Finally, a histogram of 263 bins is generated to describe the point cloud. Fig. 3 illustrate the VFH descriptor of an object. The histogram bins of 0 to 134 is used for the extended FPFH component, the bins of 180 to 307 is used for the viewpoint component. Note that, the histogram bins of 135 to 179

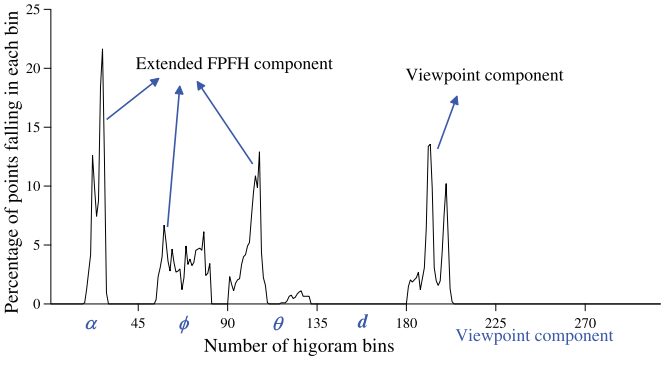

Fig. 3. Example of the VFH descriptor of an object.

represent the distance features between points and the centroid, and their default values are set to zero in PCL.

Although VFH has been used for object recognition and pose estimation in mobile manipulation and grasping applications, some limitations are faced by accurate 3D pose estimation. For example, the surfaces of an object might be flat, so their VFHs may generate false-positive results in some symmetric poses, and VFH is invariant to rotations around the camera's view direction, so it may generate false results.

4.2. Portion viewpoint feature histogram descriptor

To address this problem, we propose a PVFH descriptor based on VFH. The Darboux coordinate frame (u, v, w) of the VFH descriptor is generated from the whole point cloud, which is sensitive to noise and occlusions (e.g., missing parts of the object). To enhanced the descriptive power, we split the whole point cloud into four parts, as shown in Fig. 4. First, a tight-fitting axis-aligned bounding box (AABB) of the whole point cloud is obtained by PCA, and the X and Y axes dimensions of the AABB are obtained by computing the minimum and maximum coordinate values along the X and Y axis. Then, the AABB is split into four equal spaces along the X and Y axes, respectively. The points that fall inside different spaces belong to different parts. The define of the four part as follows: the first part is located in the region with $p_x < \frac{X_L}{2}$ and $p_y < \frac{Y_L}{2}$(i.e., the red points in Fig. 4), the second part is located in the region with $p_x \geq \frac{X_L}{2}$ and $p_y < \frac{Y_L}{2}$ (i.e., the blue points in Fig. 4), the third part is located in the region with $p_x < \frac{X_L}{2}$ and $p_y \geq \frac{Y_L}{2}$ (i.e., the green points in Fig. 4), the fourth part is located in the region with $p_x \geq \frac{X_L}{2}$ and $p_y \geq \frac{Y_L}{2}$ (i.e., the white points in Fig. 4). Where p_x and p_y represent the x and y value of an point, respectively; X_L and Y_L represent the X and Y dimensions of AABB, respectively. In our 3D vision system, the structured light is projected to XY plane, so we split the bounding box into four equal spaces along X axis and Y axis.

Then, we define a Darboux coordinate systems $D = (u_i, v_i, w_i)$ for each part of the point cloud according to Eqs. (1)–(3). Given $D = (u_i, v_i, w_i)$ and Eqs. (4)–(7), the normal angular deviations of α, ϕ, β and θ can be computed for the four parts of the point cloud. Finally, the PVFH histogram is generated by concatenating all these normal angular deviations of α, ϕ, β and θ. The PVFH histogram is defined as:

$$H = \{\alpha_1, \phi_1, \beta_1, \alpha_2, \phi_2, \beta_2, \alpha_3, \phi_3, \beta_3, \alpha_4, \phi_4, \beta_4, \theta_1, \theta_2, \theta_3, \theta_4\} \quad (8)$$

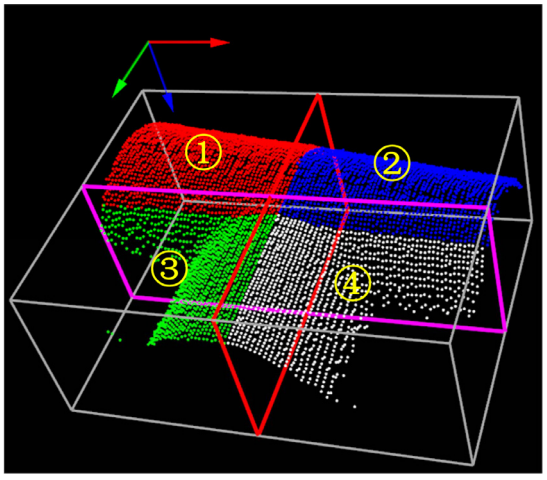

Fig. 4. An illustration of point cloud division. Red, greed, blue arrows represent the positive directions of X, Y, Z, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

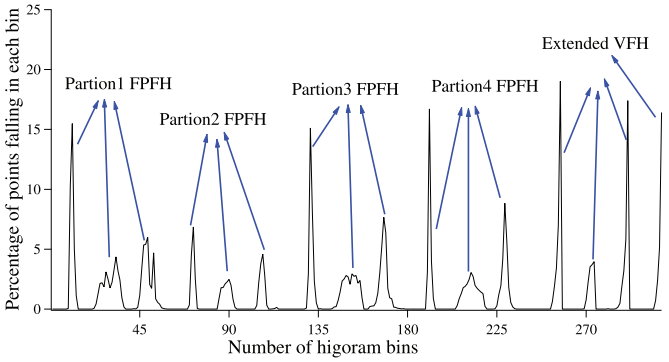

Fig. 5. An illustration of the PVFH histogram of an object.

For VFH descriptor, each angle in the extended FPFH takes 45 bins, and the viewpoint feature takes 128 bins. In this work, the number of bins used for each normal angular deviation is 20 and the viewpoint component is 17. Therefore, 308 bins are used for PVFH. Fig. 5 illustrates the PVFH descriptor with five components: four extended FPFH components and an extended viewpoint direction component.

PVFH descriptors encode feature information of the four parts of point clouds in corresponding bins. Therefore, PVFH descriptors can better handle occlusions as long as some parts of the object are visible. More importantly, the four parts of point clouds contain the orientation information around the XY plane. Consequently, PVFH descriptors are discriminative for symmetry objects. Fig. 6 shows the performance of VFH and PVFH to identify objects with mirror symmetry poses. Fig. 6(a) shows a point cloud where the normal direction of the object surface is the same as the viewpoint direction. Fig. 6(b)–(c) show the VFH and PVFH histograms of that point cloud. Fig. 6(d)–(g) show two point clouds with yaw angles of $+30^\circ$ and -30° from the normal direction of an object surface in the viewpoint direction, respectively. Fig. 6(e)–(f) respectively show the VFH and PVFH histograms of the point cloud with a yaw angles of $+30^\circ$, respectively. Fig. 6(h)–(i) show the VFH and PVFH histograms of the point cloud with a yaw angle

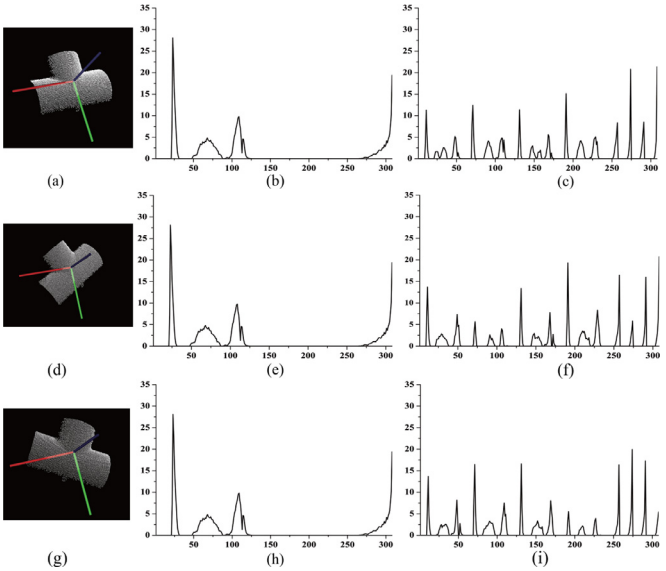

Fig. 6. An comparison of VFH and PVFH descriptors of point clouds with three poses. (a) the normal direction of the object surface is the same as the viewpoint direction. (d) and (g) point clouds with yaw angles of $+30^\circ$ and -30°. (b), (c) and (h) show the VFH histogram of point clouds given in (a), (d), (g), respectively. (d), (f) and (i) show the PVFH histogram of point clouds given in Figs. in (a), (d), (g), respectively.

of −30°, respectively. From Fig. 6(b), (e) and (h), it can be found that there is no obvious difference for the VFH histograms of the three point clouds shown in Fig. 6(a), (d) and (h). This indicates that the VFH descriptor may produce incorrect pose estimation results for object grasping because these VFH descriptor are similar even although their underlying point clouds have different poses. In contrast, from Fig. 6(c), (f) and (i), we can find that the difference between these PVFH histograms generated from the point clouds with three different poses are very obvious. Specifically, the histogram of the 25th bin, 75th bin,180th bin, and the viewpoint direction component (i.e., the bins of 250 to 307) have significant differences. PVFH heavily relies on the object segmentation results, so the performance of PVFH is affected by segmentation quality.

5. Experiment

In this section, several experiments were carried out to evaluate the performance of our proposed method. First we compare our method to the VFH descriptor [30] and the CVFH descriptor [2] on the 3DNet datasets [34]. Then we test the robustness of the proposed method with respect to different levels of Gaussian noise. To test the object recognition and pose estimation performance of PVFH on 3D industrial products, we also performed object recognition and pose estimation experiments the industrial products under various conditions. The object recognition experiments were conducted on a PC with an Intel® Core™ i5-6500 CPU and 8 GB DDR III RAM.

5.1. Evaluation on the 3DNet dataset

5.1.1. 3DNet dataset

The 3DNet dataset is a free dataset for object class recognition and 6DOF pose estimation from point clouds [34]. 3DNet provides a large-scale hierarchy of CAD-model databases, which have 10, 60 and 200 object classes. The 3DNet dataset also contains thousands of scenes captured by an RGB-D sensor. In our experiments, 10 object classes and 1612 point clouds were used.

5.1.2. Results

The recognition results achieved by VFH, CVFH and PVFH features with respect to different numbers of nearest neighbors are shown in Fig. 7. The first number in each row of the legend represents the rank-1 recognition rate while the second number represents the rank-15 recognition rate. Overall, PVFH achieves the best rank-1 and rank-15 recognition rates, followed by CVFH and VFH. The rank-1 recognition rates are 46.9%, 51.1% and 56.0% for VFH, CVFH and PVFH, respectively. Meanwhile, the rank-15 recognition rates are 71.9%, 71.4% and 83.0% for VFH, CVFH and PVFH, respectively. For objects with mirror symmetry, like apples, bowls and bottles, the recognition rate of PVFH is improved as compared to VFH and CVFH. Experiment results indicate that PVFH is more robust to objects with mirror symmetry since the point cloud of an object is divided into four parts and the VFHs of 4 parts are concatenated.

5.2. Robust to Gaussian noise

In order to evaluate the robustness to Gaussian noise of the proposed descriptors, we add Gaussian noises with increasing standard deviation of 0.1, 0.2, 0.3, 0.4 and 0.5 mr to the scene object on 3DNet dataset. For a given scene object, Gaussian noise is independently added to the *X, Y* and *Z* axis.

Table 1 show the recognition rate with different levels of Gaussian noise for VFH, CVFH, PVFH descriptor. Compared to VFH and CVFH, PVFH descriptor achieves the best robustness with respect to Gaussian noise. This can be contributed to block description. Block description can increase the descriptiveness for object and the robustness for noise.

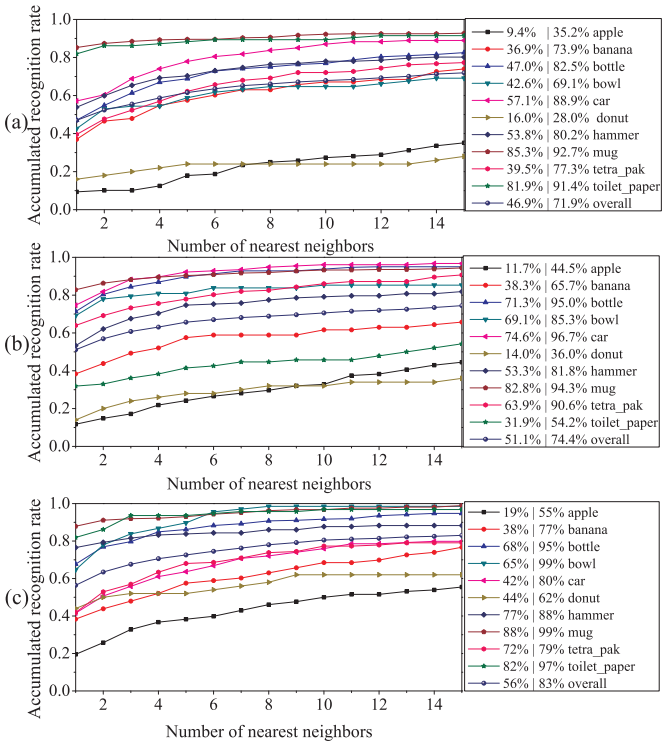

Fig. 7. Recognition rate of VFH (a), CVFH (b) and PVFH (c) with respect to numbers of nearest neighbors achieved on the 3DNet dataset.

Table 1
The recognition rate with different levels of Gaussian noise.

Noise	VFH	CVFH	PVFH
0.1mr	37.1%	35.7%	43.2%
0.2mr	36.5%	35.7%	42.8%
0.3mr	36.4%	35.6%	42.7%
0.4mr	35.9%	35.1%	43.0%
0.5mr	35.3%	35.4%	42.6%

5.3. Evaluation on point cloud of industrial products

In this section, we test the object recognition and pose estimation performance of PVFH on point cloud of 3D industrial products. In this experiment, we use three-way pipes with random poses as the objects for recognition.

5.3.1. Test data

In this experiment, the test data was captured by a 3D acquisition system. The 3D acquisition system consists of a digital micromirror device projector (TI DLP lightcrafter 4500) and a monochrome camera (PointGrey Flea3 FL3-U3-13Y3) equipped with a c-mount lens (Computar 8mm F1.4 M0814-MP2). The test dataset includes 75 objects, which were segmented from 14 scenes using a Euclidean clustering segmentation method [29].

5.3.2. Implementation details

In this experiment, we first build a pose database based on the CAD model (Section 3.1). In the test stage, a query descriptor is extracted from the object. Query and training descriptors are matched using FLANN [24]. Then, for each matched object instance, a coarse pose is computed using the alignment which transforms

Table 2
The description time.

Point Number	VFH descriptor time (ms)	PVFH descriptor time (ms)
443	0.22	0.29
2747	1.29	1.34

the query and the training partial views. The coarse pose is then refined using the ICP algorithm.

5.3.3. Results

Table 2 shows the average point number and the average feature extraction time of VFH and PVFH. When the average point number is 443, the average feature extraction time of the VFH descriptor is 0.22 ms, and the PVFH descriptor is 0.29 ms. When the average point number is 2747, the average feature extraction time of the VFH descriptor is 1.29 ms, and the PVFH descriptor is 1.34 ms. There is no clear difference between the average feature extraction time of VFH and PVFH. Fig. 8(a) shows the comparative results achieved by VFH and PVFH. It is shown that PVFH achieves better recognition rates than VFH. Therefore, fewer hypotheses are required for verification to find the correct object, and the recognition time is reduced. Fig. 8(b) shows the pose refinement time of ICP with respect to increasing number of nearest neighbors. The pose refinement time includes the time for ICP pose refinement for all matched object instances. From Fig. 8(b), we see that the ICP pose refinement time increases with the number of nearest neighbors, and the time of PVFH is lower than VFH. Fig. 8(c) shows the ICP errors achieved by VFH and PVFH after pose refinement. The ICP error represents the standard Root Mean Square (RMS) error between the final surface pairs after ICP pose refinement. The ICP error decreases with the number of nearest neighbors. It is obvious that the PVFH feature achieves lower ICP errors than VFH. Fig. 8(d) shows the feature matching time of FLANN under different numbers of nearest neighbors. The average matching time of the VFH descriptors is 2.6ms, and that of the PVFH descriptors is 2.7ms. There is no clear difference between the average matching time of VFH and PVFH. The average matching time has no correlation

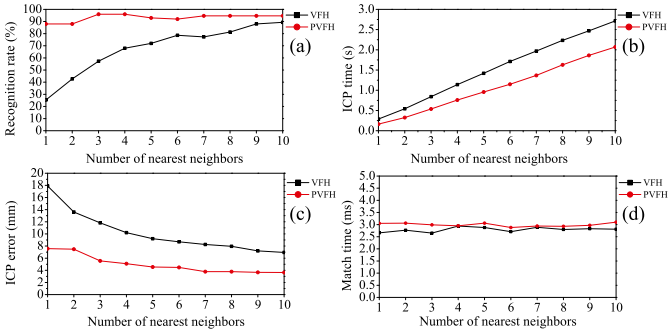

Fig. 8. Performance evaluation results. (a) recognition rate, (b) ICP time, (c) ICP error, (d) matching time.

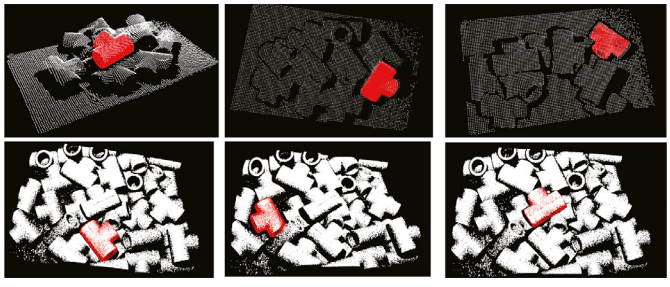

Fig. 9. An illustration of the object recognition results achieved by PVFH.

with the numbers of nearest neighbors. Fig. 9 illustrates the object recognition results of PVFH. The first row in Fig. 9 corresponds to the object with the average point number of 443, and the second row corresponds to the object with the average point number of 2747.

6. Conclusions

This paper proposes a 3D object recognition and pose estimation system for bin-picking. To ensure accurate pose estimation for symmetrical objects around the direction of viewpoint, we propose a PVFH descriptor. The PVFH descriptor characterizes the pose of an object and improves the object recognition performance under mirrored poses. Experimental results on the 3DNet dataset show that PVFH achieves better recognition rates than two existing global descriptors. Experimental results on the dataset of three-way pipes show that PVFH produces higher recognition rates and lower ICP errors with less computational time. Therefore, PVFH is suitable for real-time object recognition and 6DOF pose estimation.

Declaration of Competing Interest

The authors declared that they have no conflicts of interest to this work. We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (no. 61775172, 61602499), the Natural Science Foundation of Guangdong Province, China under grant (no. 2018030310482), the Jinan University special fund for science and technology platform construction (no. 17817003), the National Postdoctoral Program for Innovative Talents (no. BX201600172) and Fundamental Research Funds for the Central Universities (no. 18lgzd06).

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.patrec.2019.08.016.

References

[1] A. Aldoma, F. Tombari, L.D. Stefano, M. Vincze, A global hypotheses verification method for 3D object recognition, in: European Conference on Computer Vision, Springer, 2012, pp. 511–524, doi:10.1007/978-3-642-33712-3_37.
[2] A. Aldoma, M. Vincze, N. Blodow, D. Gossow, S. Gedikli, R.B. Rusu, G. Bradski, Cad-model recognition and 6DOF pose estimation using 3D cues, in: IEEE International Conference on Computer Vision Workshops, IEEE, 2011, pp. 585–592, doi:10.1109/iccvw.2011.6130296.
[3] T. Bell, S. Zhang, Towards superfast 3D optical metrology with digital micromirror device (DMD) platforms, in: Emerging Digital Micromirror Device Based Systems and Applications VI, 8979, International Society for Optics and Photonics, 2014, p. 897907, doi:10.1117/12.2035270.
[4] P.J. Besl, N.D. Mckay, A Method for Registration of 3-D Shapes, IEEE Comput. Soc., 1992.
[5] T. Birdal, S. Ilic, Point pair features based object detection and pose estimation revisited, in: International Conference on 3D Vision, IEEE, 2015, pp. 527–535, doi:10.1109/3dv.2015.65.
[6] C.-S. Chen, P.-C. Chen, C.-M. Hsu, Three-dimensional object recognition and registration for robotic grasping systems using a modified viewpoint feature histogram, Sensors 16 (11) (2016) 1969, doi:10.3390/s16111969.
[7] C. Choi, Y. Taguchi, O. Tuzel, M.-Y. Liu, S. Ramalingam, Voting-based pose estimation for robotic assembly using a 3D sensor, in: IEEE International Conference on Robotics and Automation, Citeseer, 2012, pp. 1724–1731, doi:10.1109/icra.2012.6225371.
[8] B. Drost, M. Ulrich, N. Navab, S. Ilic, Model globally, match locally: Efficient and robust 3D object recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 998–1005, doi:10.1109/cvpr.2010.5540108.
[9] A. Flint, A. Dick, A. Van Den Hengel, Thrift: Local 3D structure recognition, in: Dicta, IEEE, 2007, pp. 182–188, doi:10.1109/dicta.2007.4426794.

[10] A. Frome, D. Huber, R. Kolluri, T. Bülow, J. Malik, Recognizing objects in range data using regional point descriptors, in: European Conference on Computer Vision, Springer, 2004, pp. 224–237, doi:10.1007/978-3-540-24672-5_18.
[11] S. Fuchs, S. Haddadin, M. Keller, S. Parusel, A. Kolb, M. Suppa, Cooperative bin-picking with time-offlight camera and impedance controlled dlr lightweight robot iii, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2010, pp. 4862–4867, doi:10.1109/iros.2010.5651046.
[12] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, 3D object recognition in cluttered scenes with local surface features: a survey, IEEE Trans. Pattern Anal. Mach. Intell. 36 (11) (2014) 2270–2287, doi:10.1109/TPAMI.2014.2316828.
[13] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, N.M. Kwok, A comprehensive performance evaluation of 3D local feature descriptors, Int. J. Comput. Vis. 116 (1) (2016) 66–89, doi:10.1007/s11263-015-0824-y.
[14] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, J. Wan, Rotational projection statistics for 3D local surface description and object recognition, Int. J. Comput. Vis. 105 (1) (2013) 63–86, doi:10.1007/s11263-013-0627-y.
[15] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, V. Lepetit, Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes, in: IEEE International Conference on Computer Vision, IEEE, 2011, pp. 858–865, doi:10.1109/iccv.2011.6126326.
[16] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, N. Navab, Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes, in: Asian Conference on Computer Vision, Springer, 2012, pp. 548–562, doi:10.1007/978-3-642-37331-2_42.
[17] S. Hinterstoisser, V. Lepetit, N. Rajkumar, K. Konolige, Going further with point pair features, in: European Conference on Computer Vision, Springer, 2016, pp. 834–848, doi:10.1007/978-3-319-46487-9_51.
[18] D. Holz, S. Holzer, R.B. Rusu, S. Behnke, Real-time plane segmentation using rgb-d cameras, in: Robot Soccer World Cup, Springer, 2011, pp. 306–317, doi:10.1007/978-3-642-32060-6_26.
[19] A. Krull, E. Brachmann, F. Michel, M. Ying Yang, S. Gumhold, C. Rother, Learning analysis-by-synthesis for 6D pose estimation in rgb-d images, in: IEEE International Conference on Computer Vision, IEEE, 2015, pp. 954–962, doi:10.1109/iccv.2015.115.
[20] M. Li, K. Hashimoto, Curve set feature-based robust and fast pose estimation algorithm, Sensors 17 (8) (2017) 1782, doi:10.3390/s17081782.
[21] M.-Y. Liu, O. Tuzel, A. Veeraraghavan, Y. Taguchi, T.K. Marks, R. Chellappa, Fast object localization and pose estimation in heavy clutter for robotic bin picking, Int. J. Robot. Res. 31 (8) (2012) 951–973, doi:10.1177/0278364911436018.
[22] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110, doi:10.1023/b:visi.0000029664.99615.94.
[23] A.K. Mishra, Y. Aloimonos, Visual segmentation of simple objects for robots, Robotics (2012) 1–8.
[24] M. Muja, D.G. Lowe, Fast approximate nearest neighbors with automatic algorithm configuration., VISAPP (1)2, 2, 2009.
[25] M. Nieuwenhuisen, D. Droeschel, D. Holz, J. Stückler, A. Berner, J. Li, R. Klein, S. Behnke, Mobile bin picking with an anthropomorphic service robot, in: IEEE International Conference on Robotics and Automation, IEEE, 2013, pp. 2327–2334, doi:10.1109/icra.2013.6630892.
[26] J.-K. Oh, K. Baek, D. Kim, S. Lee, Development of structured light based bin–picking system using primitive models, in: Frontiers of Assembly and Manufacturing, Springer, 2010, pp. 141–155, doi:10.1007/978-3-642-14116-4_12.
[27] R. Rios-Cabrera, T. Tuytelaars, Discriminatively trained templates for 3D object detection: A real time scalable approach, in: IEEE International Conference on Computer Vision, IEEE, 2013, pp. 2048–2055, doi:10.1109/iccv.2013.256.
[28] R.B. Rusu, N. Blodow, M. Beetz, Fast point feature histograms (FPFH) for 3D registration, in: IEEE International Conference on Robotics and Automation, Citeseer, 2009, pp. 3212–3217, doi:10.1109/robot.2009.5152473.
[29] R.B. Rusu, N. Blodow, Z.C. Marton, M. Beetz, Close-range scene segmentation and reconstruction of 3D point cloud maps for mobile manipulation in domestic environments, in: IEEE International Conference On Intelligent Robots and Systems, IEEE, 2009, pp. 1–6, doi:10.1109/iros.2009.5354683.
[30] R.B. Rusu, G. Bradski, R. Thibaux, J. Hsu, Fast 3D recognition and pose using the viewpoint feature histogram, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2010, pp. 2155–2162, doi:10.1109/iros.2010.5651280.
[31] R.B. Rusu, S. Cousins, 3D is here: point cloud library (PCL), in: IEEE International Conference on Robotics and Automation, IEEE, 2011, pp. 1–4, doi:10.1109/icra.2011.5980567.
[32] R. Schnabel, R. Wessel, R. Wahl, R. Klein, Shape recognition in 3D point-clouds(2008).
[33] F. Tombari, S. Salti, L. Di Stefano, Unique signatures of histograms for local surface description, in: European Conference on Computer Vision, Springer, 2010, pp. 356–369, doi:10.1007/978-3-642-15558-1_26.
[34] W. Wohlkinger, A. Aldoma, R.B. Rusu, M. Vincze, 3DNet: large-scale object class recognition from cad models, in: IEEE International Conference on Robotics and Automation, IEEE, 2012, pp. 5384–5391, doi:10.1109/icra.2012.6225116.
[35] C.-H. Wu, S.-Y. Jiang, K.-T. Song, Cad-based pose estimation for random bin-picking of multiple objects using a rgb-d camera, in: International Conference on Control, Automation and Systems, IEEE, 2015, pp. 1645–1649, doi:10.1109/iccas.2015.7364621.