A 3D Recognition System with Local-Global Collaboration

Kai Sheng Cheng, Huei Yung Lin(✉), and Tran Van Luan

Department of Electrical Engineering,
Advanced Institute of Manufacturing with High-Tech Innovation,
National Chung Cheng University, 168 University Road, Chiayi 62102, Taiwan
ram4996@yahoo.com.tw, hylin@ccu.edu.tw, tranvanluan07118@gmail.com

Abstract. To the best of increasing robotic vision in 3D conceptual for recognizing this living world, this paper proposed a 3D recognition system by combining the local feature and global verification technique. To approach this, we modified the state-of-art methods and organized it as a robust hybrid flow. Another contribution to this paper, we release the finest parameters to the Kinect sensor as well as the dataset. In the proposed framework, we expect the pre-process can deal with range filtering, noise reduction, and point cloud refinement. After this, the captured point cloud is more reliable and better to describe the object surface. The Second part is focused on recognition and pose estimation. We here refer two robust methods, SHOT descriptor and Hough Voting, one for the local feature generation and the other contributes to the object alignment. Finally, through the ICP to refine the pose matrix, we remove the false positive while verifying the good instance. Moreover, we design a keypoint selective mechanism after the hypothesis verification stage back into local conception.

1 Introduction

The human population growth is rising, and a huge increase in demand on many products for daily use or consumer electronics is probably inevitable. To face this problem, the industry needs to produce a large amount of standardized products. Mass production uses assembly line to make copies of product quickly, which involves foods, medicine, 3C electronics, apparels and vehicles, etc. Usually a standard factory contains a modern automobile assembly line, but the machinery mass production line is very expensive to ensure its products to be successful output the profit. Due to the high cost of machinery line or partially completed products not well fitting to the robotic arm, some factories employ tremendous manpower to work on each individual step. To increase the product yield rate or produce special material in a factory, employees sometimes need to stay in a hard strict environment and wear anti-dust cloth in a disinfect or high radiation exposure zone.

The high labor cost implies the robotic automation is a key prospect to the growth of manufacturing. Consequently, many research programs focus on

© Springer International Publishing AG 2017
C.-S. Chen et al. (Eds.): ACCV 2016 Workshops, Part II, LNCS 10117, pp. 19–33, 2017.
DOI: 10.1007/978-3-319-54427-4_2

manufacturing applications. The leading research organization Robotics Virtual Organization (Robotics VO) in the US and a large cooperative research center euRobotics AISBL in Europe are currently developing the new technologies for industrial manufacturing. The main research topics include accurate indoor object positioning systems for robotic manipulators (positioning the object), sensor based safety systems, the interaction between human and robot (machine vision), higher levels of realism in filtering system (3D segmentation), reactive planning and controllable in real industrial factory or workshop safety (machine learning).

This work deals with the problem of object recognition and its 3D pose estimation. It is an important issue on visual servoing and provides the information for the robotic manipulator to interact with the target object. In the literature, many 3D recognition techniques have been proposed. The state-of-the-art recognition systems usually adopt two strategies: (1) Use a 2D affine patch dataset with 2D local features to find the correspondences in 2D scenes or 3D point clouds [1–4]. (2) Use a 3D point cloud as a model with local/global feature to find the correspondences in the 3D scene [5–9]. The former is based on 2D invariant local features. It provides the system for recognition from free viewpoints with non-rigid changes. The benefit of these systems is the model can be generalized into multiple 3D viewpoints which link the features between patches and the scene. However, the 2D patch cannot represent all possible 3D conception. The latter is to match the object in a scene by its 3D model. Recent hardware advance allows direct 3D data acquisition from the real scene, and a variety of applications can be developed. These systems use 3D features to group correspondences or generated 3D model to indicate the object (the scene might be 2D). By using the SIFT descriptor in a cluttered environment, Hsiao *et al.* take different viewpoints of a 3D model and the 2D image of the scene for object recognition [6]. Their approach shows the robustness on finding the pose matrix, but at the cost of losing accuracy. Gomes *et al.* [5] propose a recognition system for real-time acquisition by extracting the keypoints in different radii for each level (distance). In [7], Drost *et al.* vote the matched descriptor in an accumulator space, and a point pair scheme is designed for reducing the matching computation requirement. In [8,9], an ideal local pipeline is presented following the steps: keypoint extraction, description, match, correspondence grouping, absolute orientation, ICP, hypothesis verification. All the steps can be roughly partitioned into: pre-processing (for the input data grabbed by the sensor), recognition and pose estimation (including keypoint extraction, description, matching), and post-processing (refining the pose estimation results and evaluating the final outputs).

The approach presented in this paper is similar to the pipeline based on the local feature concept, but with a global verification technique and few pre/post stages. Our system takes a 3D point cloud as input, and assumes the data points are acquired from a single viewpoint. The model datasets of the objects are built under the same environment settings. All keypoints are extracted by normal estimation before SHOT descriptor generation. The local reference frame and support space are computed for each keypoint. For data matching and pose

estimation, we use KdTree, FLANN [10] to find the model-scene correspondences of the keypoint pairs. Hough voting is then performed to derive the pose in terms of a rotation matrix and a translation vector. The verification step further refines the pose by ICP, which is able to minimize the mismatch in cluster or occlusion scenes. Finally, a global hypotheses verification is carried out to minimize the false positive and maximize the true positive, followed by filtering the outliers by matched keypoints.

The contribution of this paper contains the formulation and implementation of robust 3D recognition from the real scenes. 3D features are formulated through normal estimation and eigen value decomposition (EVD), and the interpretation is separated into two parts: local feature and global verification. We analyze the techniques for extracting point features, and categorizing the state-of-the-art methods using *signatures* and *histograms*. Due to the way it works for global verification through segmentation and clustering, there are differences compared to the local pipeline. The proposed system architecture in 3D recognition combines the local and global verification to complement the disadvantages. The experiments show the robustness behavior in an environment containing multiple dissimilar objects without suffering from occlusion or clustered scenes.

2 The Unique Signatures of Histograms

The unique signatures of histograms for surfaces and texture description (SHOT) extend the exist works from [11–13], which highlight two major approaches using signatures and histograms. The signature method encodes an invariant by describing the 3D surface into a neighborhood around a given point. It localizes each trait value into coordinate bins, and is highly descriptive due to its individual localized information in the support area. However, small noise can potentially perturb the descriptor. In the histogram concept, the trait value is given according to the specific quantized domain as accumulated count. It is based on local topological entities which map into a histogram. Compared to signatures, histograms gain the robustness, but trading the descriptive accuracy by compressing the trait value into each bin.

For the signature based 3D descriptors, Novatnack and Nishino propose a method based on geometric scale-space to analyze the scale invariant of a range image [14]. The feature normal is encode within the support to ensure the local shape descriptor can be derived and deployed with different global scales. In [15], it indicates the signature is given by the 3D coordinate of each vertex within a support in the local reference frame. Continuing the 2D feature point research, the SIFT descriptor is extended to a hybrid scheme for depth images [16], and the SURF descriptor is adopted for 3D data to compute Haar wavelets as signature trait [17]. For the histogram based 3D descriptors, the spin image computes the 2D image histogram with a volume by measuring a plane spinning around the surface normal [18]. The same concept is used in local surface patches [19] and shape indices [20]. In the 3D shape context, a real full local reference frame that modifies the concept from the spin image and accumulates a 3D histogram

to each feature points with a radii around the center is then proposed. Point Feature Histograms (PFH) [21] and Fast Point Feature Histograms (FPTH) [22] accumulate the 3D information into histogram bins that contain three angular values with the normal area overlap among relevant points. More recently, MeshHoG (MH) uses the same hybrid structure as SHOT [23]. It combines the signature and histogram with a unique local reference frame as well as the color information.

SHOT descriptor is generated based on an encoded histogram of normal points, with a local support space. To simulate the inherent signature, a set of local histogram is computed as a 3D sphere with accumulated support. The SHOT signature structure accumulated in the 3D grid is aligned with the axes defined by its local reference frame. Thus, the descriptor performs as a mixture produced by histograms and signatures. In SHOT descriptor, the points are accumulated into several bins from local histograms according to the angle between the point normal and local axis. Several coarser bins can be created by interpolation on normal directions, azimuth planes, elevation planes, and sphere radii. Since each plane contains descriptive information from the local histogram, the sphere grid performs a coarse partition with proper units of descriptor. The sphere grid indicates 32 partitioning volumes from 8 azimuth, 2 elevation, 2 radial divisions. On the other hand, by combining a proper number of bins from internal histograms (11 bins), the total descriptor length is 352. In SHOT descriptor, it is important to avoid boundary effects due to the local histogram.

3 Global Hypotheses Verification

We apply SHOT descriptor to transform feature points to the local reference frame. 3D Hough voting [8] is then performed after point feature registration. In general, Hough voting can be a pose estimation stage for the model (off-line) and the scene. For a reference point C^M in the model coordinates, we can find an exact match C^S in the scene. We give the same EVD process to obtain the local reference frames for the model and the scene, so we can assume the feature points in the model is defined as F_i^M with the centroid C^M. A vector $V_{i,G}^M$ describing the relationship between F_i^M and C^M can be written as

$$V_{i,G}^M = C^M - F_i^M \tag{1}$$

For a global vector $V_{i,G}^M$, we can then find a term $R_{G,L}^M$ representing the rotation invariant to transform to a local vector $V_{i,L}^M$. The relation can be written as

$$V_{i,L}^M = R_{G,L}^M \cdot V_{i,G}^M \tag{2}$$

where $R_{G,L}^M$ is given by

$$R_{G,L}^M = [L_{i,x}^M \;\; L_{i,y}^M \;\; L_{i,z}^M]^\top \tag{3}$$

Once the rotation matrix from the model and the scene, R_L^{MS}, is derived, $V_{i,L}^S$ can be transformed into the global reference frame, and the equation are given by

$$V_{i,G}^S = R_{LG}^S \cdot V_{i,L}^S + F_j^S \tag{4}$$

$$R_{G,L}^S = [L_{j,x}^S, L_{j,y}^S, L_{j,z}^S] \tag{5}$$

3D Hough voting picks one or more object poses which are higher than a threshold associated with a similar surface in the scene. Global Hypotheses Verification (GHV) is introduced as an additional step to further verify and reject false positives (false detection). First, we consider some notations about GHV after SHOT recognition pipeline. Assume the model set in the library contains m point clouds, $M = \{M_1, \ldots, M_m\}$, and a scene point cloud, S. For a general case, a scene might include several sets of the models. The pose estimation produces the transformation T given by the SHOT pipeline. It relates each model instance to the scene S with 6 DOFs. A pair (M_{h_i}, T_{h_i}), where h_i is a subset from the recognition hypotheses $H = \{h_1, \cdots, h_2\}$, is given by the previous recognition process. In each cue, it tries to determine and minimize the cost function value. The GHV method is designed to maximize the correct recognition items (TPs) belonging to the instance set H, and remove the wrong recognition items (FPs). In addition, a boolean term $X = \{x_0, \cdots, x_n\}$ denotes the ICP converges or not. It considers the case of partial occlusion or rotation in the scene because the model descriptor might be different from the scene or not fit exactly. In the occlusion case, a model might not be visible, i.e., self-occlusion or occluded by the scene parts. We use the binary term X to indicate the corresponding hypothesis is false or valid ($x_i = 0/1$).

Here we introduce the cues in GHV process and adopted in our implementation [24].

Cue (1) Scene Fitting: We assume a model point set $M_{h_i}^v$ has been calculated, and determine the scene fitting points corresponding to the model points. The cue is for examining how the points are explained under a threshold based on the Euclidean distance. For each ICP process, the local fitting measure is given by

$$\omega_{hi}(P) = \delta(p, q) \tag{6}$$

where q represents the model with a pose T and is denoted as $q = N(p, M_{h_i}^v)$, and $\delta(p, q)$ represents the scene point set obtained from 3D Hough voting and is defined by

$$\delta(p, q) = \begin{cases} (-\frac{||p-q||_2}{\rho_e} + 1)(n_p \cdot n_q), & ||p - q||_2 \leq \rho_e \\ 0, & \text{elsewhere} \end{cases} \tag{7}$$

We can take $\delta(p, q)$ as a local alignment of surfaces. Equation (7) checks the normal direction by (n_p, n_q), and it is expected to have two normals in the same direction. If (p, q) distance is smaller than a threshold ρ_e, a weight value (0 to 1)

is assigned according to the normal direction to examine where the scene-model fitting is accepted. We conclude the contribution of Cue 1 by

$$\Omega_x(p) = \sum_{i=1}^{n} \omega_{hi}(p) \cdot x_i \tag{8}$$

All $\omega_{hi}(p)$ will be explained if the ICP term $x_i = 1$, and thus $\Omega_x(p) > 0$. If a point $p \in M_{h_i}^v$ but is not fitted in any scene point set according to Eq. (7), we denote it as ϕ_{h_i}.

Cue (2) Multiple Assignment: This cue gives a function for examining the term in Cue 1 by subtraction. The equation is given as follow:

$$\Lambda_X(p) = \begin{cases} \sum\limits_{i=1}^{n} sgn(\omega_{h_i}(p)), \; sgn(\omega_{h_i}(p)) > 1 \\ 0, \qquad\qquad\qquad \text{elsewhere} \end{cases} \tag{9}$$

where

$$sgn(X) = \begin{cases} -1, \; X < 0 \\ 0, \quad X = 0 \\ 1, \quad X > 0 \end{cases} \tag{10}$$

Cue (3) Cost Function: The cost function concludes Cue 1 and Cue 2 to increase the number of recognized instances as many as possible. It can be simply described as

$$\zeta(X) = f_S(X) + \lambda \cdot f_M(X) \tag{11}$$

where λ is a constant regularization value, and f_S, f_M are

$$f_S(X) = \sum_{p \in S} (\Lambda_X(p) - \Omega_x(p)) \tag{12}$$

$$f_M(X) = \sum_{n=1}^{n} |\phi_{h_i}| \cdot x_i \tag{13}$$

4 System Development and Implementation

In the pre-process stage, we capture 3D point cloud data by Kinect V1 and V2 through Kinect SDK 1.8 and 2.0, respectively. According to the sensor depth range (V1: 1.2–3.5 m, V2: 0.5–4.0 m), we fix a filter range of 1.8 m to remove the background points. For uniform down-sampling of the large point cloud, we set the radius as 0.01 for the model and 0.0125 for the scene. The local features are extracted by normal estimation and SHOT after the pre-processing stage, followed by Hough voting for pose estimation. We set the parameters for the local reference frame radius as 0.08 and the clustering threshold as 10.0. The example model is extracted from the scene exactly, thus the rotation is an identity matrix and the translation vector is zero.

After pose estimation, we fix the pose to minimize the Hough voting errors by ICP. Due to the inherent property of ICP, there are mis-voted cases caused by occlusion or the object placed in a pose but different from the model set. In this cases, we refine the pose from the identity rotation matrix and zero translation vector. The parameters for ICP maximum number of iteration and correspondence distance are 5 and 0.005, respectively. Note that Hough voting gives a rough transformation after 5 ICP iterations no matter the pose converges or not. The global verification process is then carried out after the ICP refinement. It is used to justify the final pose a good or bad instance.

Figure 1 shows an example of hypothesis verification. An offline model dataset is displayed on the right. The red, cyan and violet poses indicate the production of Hough voting, convergence of ICP, and the point correspondences, respectively. The following parameters are used: clutter regularizer, 5.0; inliner threshold, 0.2; clutter radius, 0.015; regularizer value, 3.0; and normal radius: 0.05. The system verifies the actual (true positive) instance as the green pose. In the final step, we pick the highest keypoint matched instance as our result. For example, if there are four instances with matched keypoints $M <30, 20, 75, 70>$, we pick the highest (75) but also giving a threshold K (say, 15). It means that the number of matched points under 75 but greater than 60 is still a good instance. All libraries and codes are built using PCL 1.7.2 [25] on a PC with Intel i7-4790 processor.

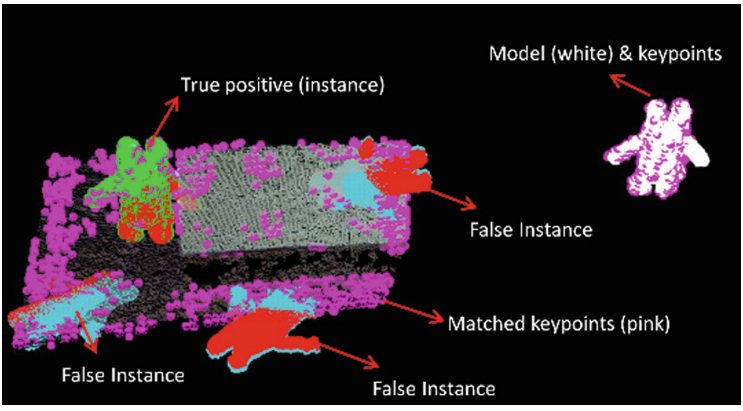


Fig. 1. An example of global hypothesis verification. (Color figure online)

5 Experiments

To evaluate the proposed technique, three datasets are generated for several interested scenarios. Five objects (Alien, Bear, Cbox, Crab, Sulley) are included individually in the occlusion and rotation datasets as shown in Fig. 2, and the cluster dataset contains many objects in the clustered scenes. In the experiments, the objects are placed at about 1 m away from a fixed viewpoint camera.

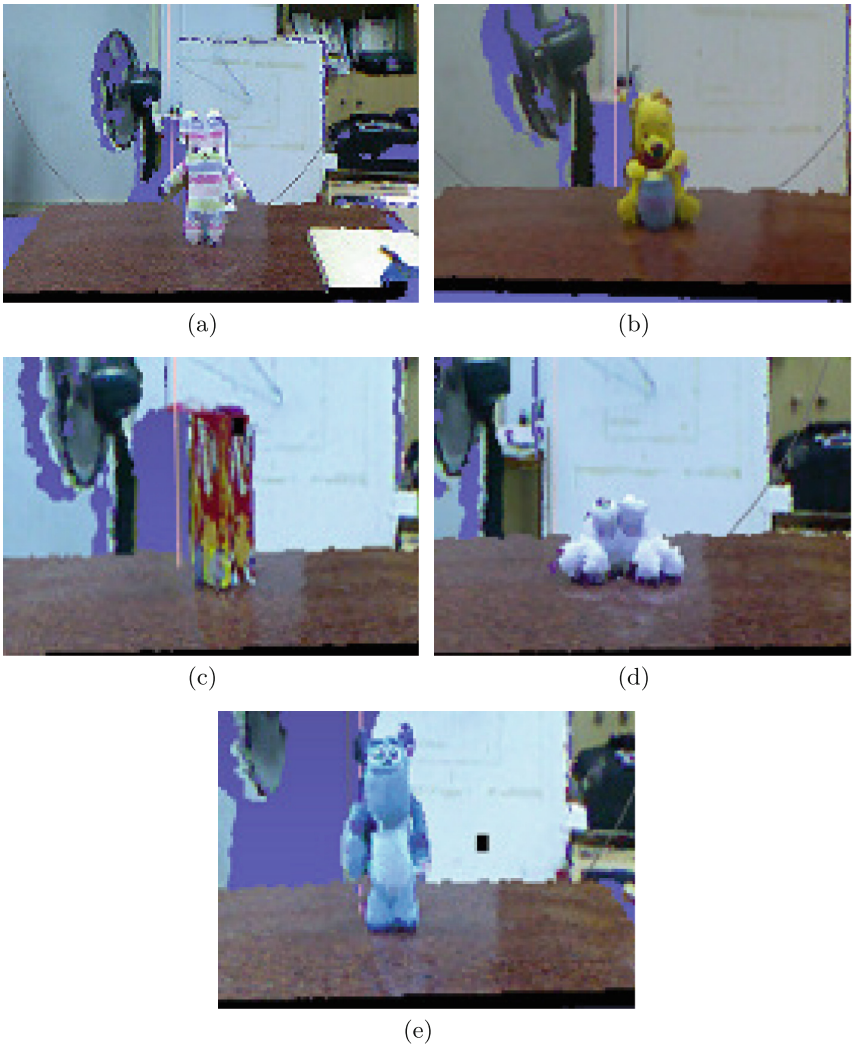


Fig. 2. The test objects used in the experiments. (a) Alien, (b) Bear, (c) Cbox, (d) Crab, and (e) Sulley.

– **Occlusion dataset:** In this dataset, we examine the system limitation by occluding the object with different levels. Five datasets of the scene are collected without any object rotation. Each dataset contains only a single object and can be described by the percentage of occlusion. The non-occluded point cloud is used as the benchmark to calculate the occlusion percentage for the occluded scenes.

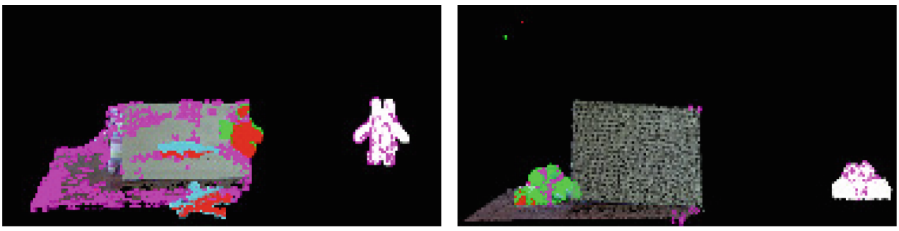
- **Rotation dataset:** In this dataset, we rotate the object with different angles but keep the viewpoint still. The scenes only contain a single object without any occlusion event. The non-rotation scene is used as the benchmark dataset.
- **Cluster dataset:** In the cluster dataset, additional objects are placed in the scene. We also investigate the recognition results with different distances between objects and the scene.

Table 1 shows the occlusion experiment results. Due to the sensor frame rate and accuracy (including the point cloud density), Kinect V2 generally works better than Kinect V1. This also illustrates how the system can perform with

Table 1. The occlusion test results with Kinect V1 and V2

Occ. (V1%,V2%)		Kinect V1		Kinect V2	
		<Reg, Match, GHV>	Corr	<Reg, Match, GHV>	Corr
Alien	(6%, 4%)	<1,1,1>	68	<1,1,1>	141
	(10%, 7%)	<4,1,1>	56	<1,1,1>	134
	(22%, 17%)	<4,1,1>	40	<1,1,1>	112
	(41%, 33%)	**<3, 0, 1>**[a]	31	<1,1,1>	62
	(53%, 51%)	*False*		<1,1,1>	25
Bear	(5%, 12%)	<1,1,1>	107	<1,1,1>	224
	(11%, 21%)	<1,1,1>	109	<1,1,1>	136
	(21%, 30%)	<1,1,1>	76	<1,1,1>	72
	(28%, 47%)	*False*		*False*	
	(48%, 53%)	*False*		*False*	
CBox	(11%, 15%)	<2,1,2>	73	<1,1,1>	116
	(23%, 23%)	<1,1,1>	47	<1,1,1>	94
	(33%, 33%)	<1,1,1>	42	<1,1,1>	53
	(39%, 42%)	*False*		<1,1,1>	34
	(50%, 55%)	*False*		*False*	
Crab	(7%, 11%)	<1,1,1>	94	<1,1,1>	63
	(15%, 19%)	<1,1,1>	56	<1,1,1>	44
	(24%, 30%)	*False*		*False*	
	(34%, 40%)	*False*		*False*	
	(52%, 50%)	*False*		*False*	
Sulley	(7%, 6%)	<1,1,1>	134	<1,1,1>	222
	(14%, 12%)	<1,1,1>	80	<2,1,1>	151
	(24%, 29%)	<1,1,1>	64	<1,1,1>	99
	(34%, 40%)	*False*		<1,1,1>	58
	(50%, 54%)	*False*		*False*	

[a]In this case, <3,0,1>, the system gives 3 Reg instances but without any true positives. It also verifies a wrong instance as a good result.



(a) (b)

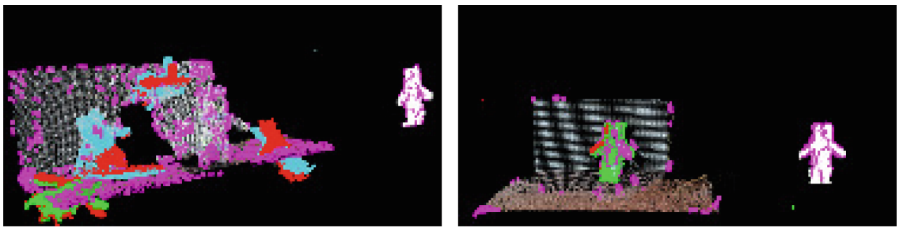
Fig. 3. (a) Alien fails to match in the occlusion dataset. (b) An example of Crab scene dataset.

a noisy or incomplete point cloud input. We expect the occlusion experiment to have the following specification: (1) The system can keep recognizing an object until it is occluded by a specific percentage. (2) Once the system recognizes an object successfully, the triplet <Reg, Match, GHV> should be at least (Reg $\geq$ GHV $\geq$ Match = 1), where <Reg>: recognized instance, <Match>: true matched, <GHV>: instance verified successfully in global hypothesis verification, and Corr: correspondences of the recognized instance. (3) The correspondence keypoint belongs to good instances should decrease till the system reaches its limitation.

Some special cases in the occlusion dataset are given as follows. In Alien V1, the object is occluded by 41% and the system gives <3,0,1>. Three recognized instances are found, with no true positives but a good verification. This is due to V1 sensor can only sense a model without the z-axis information, so the system treats it as a flat surface. On the other hand, Alien V1 dataset outputs more recognized instances than the V2 dataset. In Crab dataset, the object can only be recognized while the occluded region is no more than 20%. This is because the Crab dataset can only show the front view surface less than other datasets. Figure 3 illustrates the special cases in Alien V1 and Crab datasets.

In Bear dataset, the column Corr of Kinect V1 gives similar values for 5% and 11% of occlusion. This is due to the infrared sensor accuracy. In CBox dataset, the system gives the output <2,1,2> for 11% of occlusion. This is caused by two flat areas of the model surface, so that several instances are obtained by the Hough voting process but only one good instance is verified. In Sulley dataset, Kinect V2 gives <2,1,1> output for 12% of occlusion. It indicates that one recognized instance is bad and filtered out by the verification stage. To summarize, the proposed technique is able to recognize the object and filter out bad instances for the occlusion dataset. Moreover, Kinect V2 shows the best results in the point cloud noise reduction and provides almost all <1,1,1> for the match triplet.

In the rotation experiment, we expect the object can be recognized by the system after it is rotated. Let the front view be defined as the 90° direction, and the object is rotated to the left or the right by every 20°. Without the Kinect sensor noise, the correspondences for recognizing the instances should decrease



(a) (b)

Fig. 4. (a) The case of Alien V1 50° fails to recognize. (b) The example of Alien V2 50°.

Table 2. The rotation test results with Kinect V1 and V2

Rotate degree		Kinect V1		Kinect V2	
		<Reg, Match, GHV>	Corr	<Reg, Match, GHV>	Corr
Alien	70°	<5,1,1>	72	<1,1,1>	130
	50°	<5,0,1>	38	<1,1,1>	87
	30°	<1,0,0>	29	<1,1,1>	48
	110°	<6,1,1>	72	<1,1,1>	149
	130°	<8,1,1>	46	<1,1,1>	113
	150°	<2,1,0>	45	<1,1,1>	104
CBox	70°	<1,1,1>	60	<1,1,1>	144
	50°	<1,1,1>	71	<1,1,1>	116
	30°	<2,1,2>	51	<1,1,1>	60
	110°	<1,1,1>	142	<1,1,1>	97
	130°	<1,1,1>	53	<1,1,1>	42
	150°	*False*		*False*	
Crab	70°	<1,1,1>	105	<1,1,1>	112
	50°	*False*		<1,1,1>	51
	30°	*False*		<1,1,1>	26
	110°	*False*		<1,1,1>	141
	130°	*False*		<1,1,1>	54
	150°	*False*		*False*	
Sulley	70°	<1,1,1>	107	<1,1,1>	261
	50°	<1,1,1>	59	<1,1,1>	146
	30°	<1,1,1>	28	<1,1,1>	60
	110°	<1,1,1>	108	<1,1,1>	302
	130°	<1,1,1>	72	<1,1,1>	208
	150°	<1,1,1>	39	<1,1,1>	123

* Bear dataset cannot be recognized by the system in this experiment.

when the rotation angle is increased. Some fail cases are as follows. In Alien V1 dataset, the object surface is incomplete due to the Kinect V1 sensor accuracy issue. Particularly, the system gives an erroneous output <5,0,1> for the rotation angle of 50°. In Bear and Crab datasets, the objects can not be recognized using Kinect V1. Furthermore, Bear dataset fails with Kinect V2 either, and thus the results are not shown in the table. This is mainly due to the self-occlusion of the tall object during rotation which makes the surface more difficult to model. Figure 4 shows the recognition results of the Alien scene at 50° captured by Kinect V1 and V2.

In general, as illustrated in Table 2, Kinect V2 gives better recognition results than V1 due to the noise issue mentioned in the occlusion experiment. Thus, there are more recognition instances shown in Alien V1 dataset. In the rotation experiment, the difficulty is to deal with the vanishing and emerging parts of the object surface. Although some Kinect V1 datasets give good recognition results, the instances are estimated by ICP with a verification process, and more system computation is required.

In the last experiment, we set up three clustered scenes with all objects placed randomly in front of the camera, as shown in Fig. 5. The objective is to recognize Alien, Bear, CBox, Sulley in a scene. The objects in Scene 1 are placed with more occlusion, Scene 2 contains fairly separated objects, and Scene 3 describes an extremely clustered environment. Table 3 shows the results of Cluster dataset, where Alien and Sulley are recognized with good verification. Alien in the clustered scenes is almost not occluded by other objects, but Sulley is placed at different locations with variable revelation. Notice that, for CBox object, Scene 3 shows more surface than Scene 1 but the system gives false

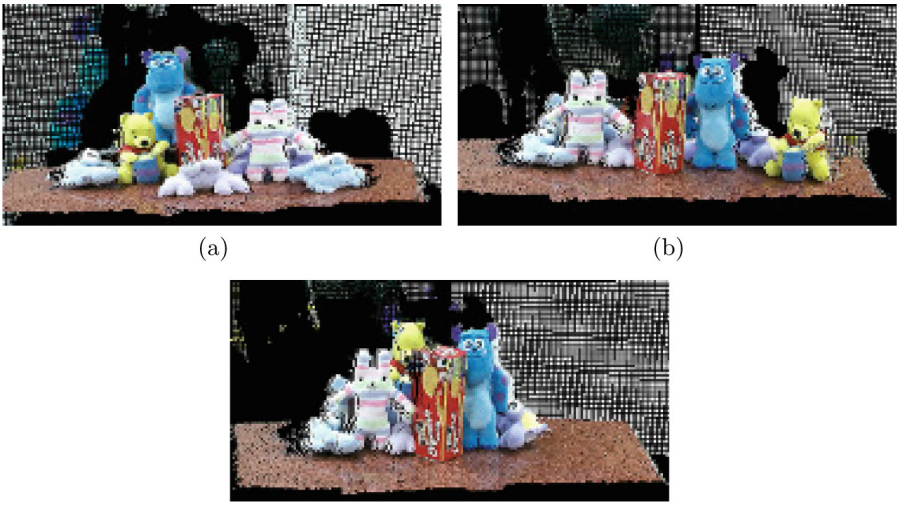


Fig. 5. The dataset with clustered scenes, (a) Scene 1, (b) Scene 2, (c) Scene 3.

Table 3. Cluster dataset test results using Kinect V2 only.

Cluster scene		Kinect V2	
		<Reg, Match, GHV>	Corr
Alien	Scene 1	<1,1,2>	44
	Scene 2	<1,1,1>	78
	Scene 3	<1,1,1>	60
Bear	Scene 1	<1,1,1>	63
	Scene 2	<1,1,1>	112
	Scene 3	*False*	
Cbox	Scene 1	<1,1,1>	43
	Scene 2	<1,1,1>	39
	Scene 3	*False*	
Sulley	Scene 1	<1,1,2>	75
	Scene 2	<1,1,1>	111
	Scene 3	<1,1,1>	55

output. This is due to left part of CBox cannot be estimated for the distance from Kinect V2 sensor and only an incomplete model is obtained.

6 Conclusion

In this paper, we propose a structural hybrid technique for the 3D recognition system. It fully builds using the 3D concept based on local features with global verification of the output instances. Our system takes a 3D point cloud as input, and assumes the data points are acquired from a single viewpoint. The model datasets of the objects are built under the same environment settings. The proposed system architecture in 3D recognition combines the local and global verification to complement the disadvantages. The experiments show the robustness behavior in an environment containing multiple dissimilar objects without suffering from occlusion or clustered scenes. Our system is able to adapt in a general environment and provide better recognition results by verifying good instances in the experiments. The future work will focus on three major issues of the 3D recognition techniques, (1) sensor accuracy: to deal with the resolution of the model and the scene, (2) partial model capability: to increase the recognition rate with partially acquired scenes, and (3) computation requirement: to apply GPU on Hough voting and hypothesis verification, which are not supported by OpenMP.

Acknowledgement. The support of this work in part by the Ministry of Science and Technology of Taiwan under Grant MOST 104-2221-E-194-058-MY2 is gratefully acknowledged.

References

1. Lowe, D.G.: Local feature view clustering for 3D object recognition. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Kauai, HI, USA, 8–14 December 2001, pp. 682–688 (2001)
2. Ponce, J., Lazebnik, S., Rothganger, F., Schmid, C.: Towards true 3D object recognition. In: International Conference on Computer Vision and Pattern Recognition (CVPR), Washington, pp. 4034–4041 (2004)
3. Toshev, A., Makadia, A., Daniilidis, K.: Shape-based object recognition in videos using 3D synthetic object models. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Miami, Florida, USA, 20–25 June 2009, pp. 288–295 (2009)
4. Hetzel, G., Leibe, B., Levi, P., Schiele, B.: 3D object recognition from range images using local feature histograms. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Kauai, HI, USA, 8–14 December 2001, pp. 294–299 (2001)
5. Gomes, R.B., da Silva, B.M.F., de MedeirosRocha, L.K., Aroca, R.V., Velho, L.C.P.R., Gonçalves, L.M.G.: Efficient 3D object recognition using foveated point clouds. Comput. Graph. **37**, 496–508 (2013)
6. Hsiao, E., Collet, A., Hebert, M.: Making specific features less discriminative to improve point-based 3D object recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010, pp. 2653–2660 (2010)
7. Drost, B., Ulrich, M., Navab, N., Ilic, S.: Model globally, match locally: efficient and robust 3D object recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010, pp. 998–1005 (2010)
8. Tombari, F., di Stefano, L.: Object recognition in 3D scenes with occlusions and clutter by Hough voting. In: Fourth Pacific-Rim Symposium on Image and Video Technology (PSIVT), pp. 349–355 (2010)
9. Aldoma, A., Marton, Z., Tombari, F., Wohlkinger, W., Potthast, C., Zeisl, B., Rusu, R.B., Gedikli, S., Vincze, M.: Tutorial: point cloud library: three-dimensional object recognition and 6 DOF pose estimation. IEEE Robot. Automat. Mag. **19**, 80–91 (2012)
10. Muja, M., Lowe, D.G.: Fast approximate nearest neighbors with automatic algorithm configuration. In: International Conference on Computer Vision Theory and Application (VISSAPP), Lisboa, Portugal, 5–8 February 2009, pp. 331–340 (2009)
11. Hoppe, H., DeRose, T., Duchamp, T., McDonald, J.A., Stuetzle, W.: Surface reconstruction from unorganized points. In: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH, Chicago, IL, USA, 27–31 July 1992, pp. 71–78 (1992)
12. Mitra, N.J., Nguyen, A., Guibas, L.J.: Estimating surface normals in noisy point cloud data. Int. J. Comput. Geom. Appl. **14**, 261–276 (2004)
13. Tombari, F., Salti, S., Stefano, L.: Unique signatures of histograms for local surface description. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6313, pp. 356–369. Springer, Heidelberg (2010). doi:10.1007/978-3-642-15558-1_26
14. Novatnack, J., Nishino, K.: Scale-dependent/invariant local 3D shape descriptors for fully automatic registration of multiple sets of range images. In: European Conference on Computer Vision, Marseille, France, 12–18 October 2008, pp. 440–453 (2008)

15. Mian, A.S., Bennamoun, M., Owens, R.A.: On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. Int. J. Comput. Vis. (IJCV) **89**, 348–361 (2010)
16. Darom, T., Keller, Y.: Scale-invariant features for 3-D mesh models. IEEE Trans. Image Process. **21**, 2758–2769 (2012)
17. Knopp, J., Prasad, M., Willems, G., Timofte, R., Gool, L.J.V.: Hough transform and 3D SURF for robust three dimensional classification. In: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, 5–11 September 2010, pp. 589–602 (2010)
18. Johnson, A.E., Hebert, M.: Using spin images for efficient object recognition in cluttered 3D scenes. IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) **21**, 433–449 (1999)
19. Dorai, C., Jain, A.K.: COSMOS - a representation scheme for 3D free-form objects. IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI) **19**, 1115–1130 (1997)
20. Chen, H., Bhanu, B.: 3D free-form object recognition in range images using local surface patches. Pattern Recogn. Lett. **28**, 1252–1262 (2007)
21. Rusu, R.B., Blodow, N., Marton, Z.C., Beetz, M.: Aligning point cloud views using persistent feature histograms. In: International Conference on Intelligent Robots and Systems, 22–26 September 2008, pp. 3384–3391 (2008)
22. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (FPFH) for 3D registration. In: IEEE International Conference on Robotics and Automation (ICRA), Kobe, Japan, 12–17 May 2009, pp. 3212–3217 (2009)
23. Zaharescu, A., Boyer, E., Horaud, R.: Keypoints and local descriptors of scalar functions on 2D manifolds. Int. J. Comput. Vis. (IJCV) **100**, 78–98 (2012)
24. Aldoma, A., Tombari, F., Stefano, L., Vincze, M.: A global hypotheses verification method for 3D object recognition. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7574, pp. 511–524. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33712-3_37
25. PointClouds.org: Point cloud library (2014). http://pointclouds.org/

http://www.springer.com/978-3-319-54426-7

Computer Vision – ACCV 2016 Workshops
ACCV 2016 International Workshops, Taipei, Taiwan,
November 20-24, 2016, Revised Selected Papers, Part II
Chen, C.-S.; Lu, J.; Ma, K.-K. (Eds.)
2017, XV, 640 p. 335 illus., Softcover
ISBN: 978-3-319-54426-7