Sensors **2014**, *14*, 24156-24173; doi:10.3390/s141224156

OPEN ACCESS
sensors
ISSN 1424-8220
www.mdpi.com/journal/sensors

Article

Recognizing Objects in 3D Point Clouds with Multi-Scale Local Features

Min Lu [1,2], Yulan Guo [1,*], Jun Zhang [1], Yanxin Ma [1] and Yinjie Lei [3]

[1] College of Electronic Science and Engineering, National University of Defense Technology, Changsha, Hunan 410073, China; E-Mails: lumin@nudt.edu.cn (M.L.); zhj64068@sina.com (J.Z.); newmyxin@163.com (Y.M.)

[2] Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada

[3] College of Electronics and Information Engineering, Sichuan University, Chengdu 610064, China; E-Mail: leolyj@gmail.com

* Author to whom correspondence should be addressed; E-Mail: yulan.guo@nudt.edu.cn; Tel.: +86-731-8457-6349.

External Editor: Assefa M. Melesse

Received: 26 August 2014; in revised form: 21 November 2014 / Accepted: 3 December 2014 / Published: 15 December 2014

Abstract: Recognizing 3D objects from point clouds in the presence of significant clutter and occlusion is a highly challenging task. In this paper, we present a coarse-to-fine 3D object recognition algorithm. During the phase of offline training, each model is represented with a set of multi-scale local surface features. During the phase of online recognition, a set of keypoints are first detected from each scene. The local surfaces around these keypoints are further encoded with multi-scale feature descriptors. These scene features are then matched against all model features to generate recognition hypotheses, which include model hypotheses and pose hypotheses. Finally, these hypotheses are verified to produce recognition results. The proposed algorithm was tested on two standard datasets, with rigorous comparisons to the state-of-the-art algorithms. Experimental results show that our algorithm was fully automatic and highly effective. It was also very robust to occlusion and clutter. It achieved the best recognition performance on all of these datasets, showing its superiority compared to existing algorithms.

Keywords: object recognition; point cloud; local feature; clutter; occlusion

---

1. Introduction

Object recognition is an active research topic in the area of computer vision [1,2]. It has a number of applications, including robotics, forensics, surveillance and remote sensing [3–5]. With the rapid development of 3D point cloud acquisition techniques, point clouds have became increasingly popular and available [6–9]. The aim of 3D object recognition is to correctly identify objects in a point cloud and estimate their 3D pose (*i.e.*, location and orientation) [10,11]. Although many algorithms have been proposed in the area of 3D object recognition, it is still very challenging to recognize objects in complex scenes in the presence of clutter and occlusion [10–12].

The existing 3D object recognition techniques can broadly be classified into global and local feature-based algorithms [4,11]. Global feature-based algorithms describe the whole surface of an object by a single descriptor. They require the scene point cloud to be pre-processed by a suitable 3D segmentation algorithm for the purpose of extracting individual object instances in the presence of clutter and/or occlusions [13]. They are frequently investigated in the area of shape classification and model retrieval [14]. In contrast, local feature-based algorithms have attracted more interests due to their robustness to clutter and occlusion [11,15,16]. Specifically, they first identify a number of keypoints in a scene and then extract a feature descriptor for each keypoint. These feature descriptors of the scene are finally matched against these feature descriptors of 3D models to get the recognition results.

Many existing local feature-based 3D object recognition algorithms use a set of single-scale features to represent a point cloud [10,17–21]. However, choosing an appropriate scale for a keypoint is very difficult. A large-scale feature contains sufficient information of the local surface of a keypoint at the cost of its high sensitivity to occlusion and clutter. On the contrary, a small-scale feature is very robust to occlusion and clutter. It however suffers from low descriptiveness. Scientific evidence from both physics and biological vision shows that multi-scale features are much more desirable with many attractive properties [22]. In this paper, we propose a multi-scale feature representation algorithm, which encodes an object by a set of local surface features with multiple scales. Experimental results show that our algorithm improves the 3D object recognition performance by a large margin compared to the state-of-the-art.

Once the scene and models are represented by local features, feature correspondences are established by matching scene features against model features. Three matching techniques have been proposed in the literature: threshold based, nearest neighbor (NN)-based and nearest neighbor distance ratio (NNDR)-based techniques [11,23,24]. In the case of threshold-based matching, two features are matched if the distance between their descriptors is less than a threshold. In the case of NN-based matching, two features are matched if the model feature descriptor is the nearest neighbor to the scene feature descriptor and if the distance between them is less than a threshold. In the case of NNDR matching, two features are matched if the model feature descriptor is the nearest neighbor to the scene feature descriptor and if the distance ratio between the first and second nearest neighbor is less than a threshold. The NNDR-based matching technique outperforms the other two techniques in terms of matching precision due to the fact that it penalizes the descriptors, which have many similar matches [23]. The NNDR-based matching technique has been widely used in image mosaic, 3D modeling, object recognition and 3D mapping [24,25]. One common limitation of all of these matching techniques is that an appropriate

threshold should be determined before hand. The threshold is usually tuned by training experiments and is, therefore, dependent on the training data. This paper proposes a coarse-to-fine matching technique, which does not rely on selecting any specific threshold.

In this paper, we present an effective algorithm to recognize 3D objects in point clouds using multi-scale local surface features. Specifically, the contributions of the paper are as follows.

(i) We present a multi-scale feature representation algorithm to encode each scene/model. It first detects a number of keypoints in each scene/model and then extracts several feature descriptors with different scales at each keypoint. The proposed multi-scale feature representation is able to capture both the fine and coarse structures of a local surface.

(ii) We introduce a coarse-to-fine feature matching technique to establish feature correspondences between an input scene and models. It first uses a low threshold to produce a small number of accurate feature correspondences. It then increases the threshold to boost the number of feature correspondences. The proposed technique maintains a high level accuracy of feature matching while increasing the number of feature correspondences.

(iii) We develop a 3D object recognition framework based on the multi-scale feature representation and coarse-to-fine feature matching algorithms. The proposed method was tested on two publicly available datasets. Experimental results show that our method achieved high recognition rates. It was robust to clutter and occlusion and outperformed the state-of-the-art methods.

The rest of this paper is organized as follows. Section 2 describes the multi-scale feature representation algorithm. Section 3 introduces the coarse-to-fine feature matching algorithm and the 3D object recognition framework. Section 4 presents the experimental results for 3D object recognition in cluttered scenes with a comparison to existing techniques. Section 5 concludes the paper.

2. Multi-Scale Feature Representation

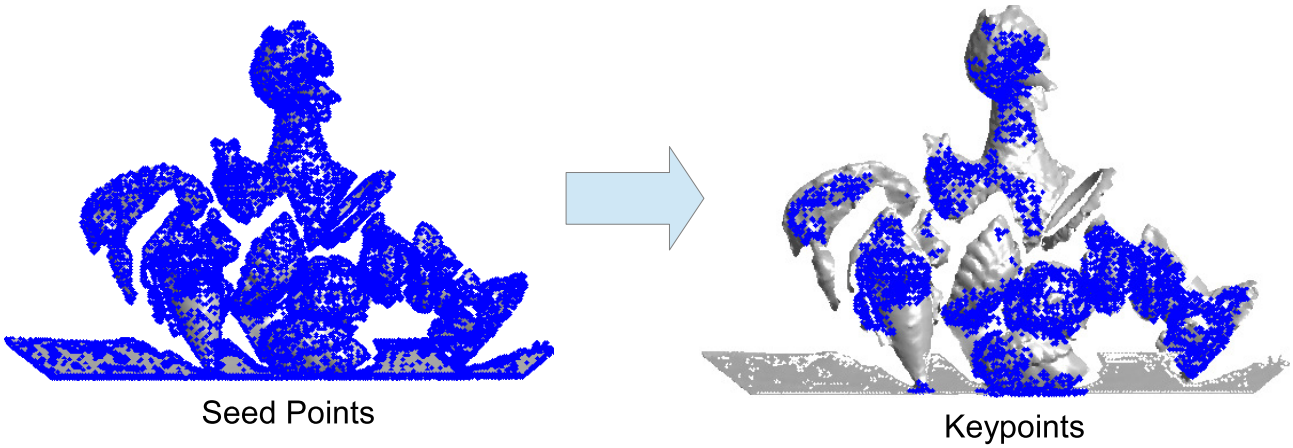
In this section, we present a multi-scale algorithm for object representation. The algorithm consists of two modules, *i.e.*, keypoint detection and feature description.

2.1. Keypoint Detection

The task of keypoint detection is to identify a set of interest points, which are distinctive and repeatable under a number of variations, including viewpoint changes, sensor noise, occlusion, clutter and point density variation [26]. In this paper, we detect keypoints based on our previous work [4]. Given a point cloud $\mathbf{P}$, it is first converted into a triangular mesh $\mathbf{M}$. The mesh is then decimated to obtain a low resolution mesh $\widehat{\mathbf{M}}$. In this paper, we used the MATLAB function 'reducepatch' to reduce the number of faces of the original mesh while attempting to preserve the overall shape of the object. For all vertices of the decimated mesh $\widehat{\mathbf{M}}$, their nearest points in the original mesh $\mathbf{M}$ are selected as seed points. The seed points are further pruned by a resolution control strategy [21] and a boundary checking algorithm [4,27]. In order to further improve the repeatability of keypoints, seed points with symmetric local surfaces are removed. The remaining seed points are finally considered keypoints. The whole process of keypoint detection is illustrated in Figure 1. It is clear that most of the keypoints are detected

in the areas with large surface variation. Moreover, no keypoint can be found on the planar surface. This is important since the local planar surface contains very poor geometric information and is, therefore, not discriminative enough for feature description and object recognition.

Figure 1. An illustration of the keypoint detection process.



2.2. Feature Description

Once a set of keypoints have been detected from a point cloud $\mathbf{P}$, the next step is to describe the neighborhood of each keypoint with a feature descriptor. The descriptor projects the local surface around the keypoint into a proper feature space [26]. Descriptiveness and robustness are two critical qualifications for a local feature descriptor [11]. A number of feature descriptors have been proposed in the literature, including spin image [20], point's fingerprint [28], 3D shape context (3DSC) [29], snapshot [30], variable-dimensional local shape descriptors (VD-LSD) [19], Mesh Histogram of Oriented Gradients (MeshHOG) [31], exponential map (EM) [32] and rotational projection statistics (RoPS) [4,18]. For a comprehensive survey on local feature descriptors, the reader should refer to [11]. As reported in [4,11,33], RoPS achieves superior performance for feature matching in terms of precision and recall. It is also very robust to a set of variations, including Gaussian noise, shot noise, varying mesh resolutions and holes [4]. Therefore, we choose the RoPS algorithm for feature description in this paper. Different from our previous work [4,33] where fixed-scale RoPS features are used for 3D object recognition, this paper proposes a multi-scale RoPS feature representation. The process for generating multi-scale RoPS descriptors for a keypoint is illustrated in Figure 2.

Given a keypoint $\boldsymbol{p}$, multiple scales $\{r_1, r_2, \cdots, r_{n_s}\}$ are used to define the support radii of the keypoint. In our previous work, it is shown that the feature descriptors with a single scale of 15-times mesh resolution (mr) present the best overall performance. In order to encode more information around a keypoint, we use four different scales around 15 mr to achieve multi-scale representation in this paper (see more details in Section 3.5). For a selected scale r_k, a local surface $\mathcal{L}_k$ is cropped from the mesh by using a sphere of radius r_k centered at the keypoint $\boldsymbol{p}$. Assume that the local surface $\mathcal{L}_k$ consists of n_t

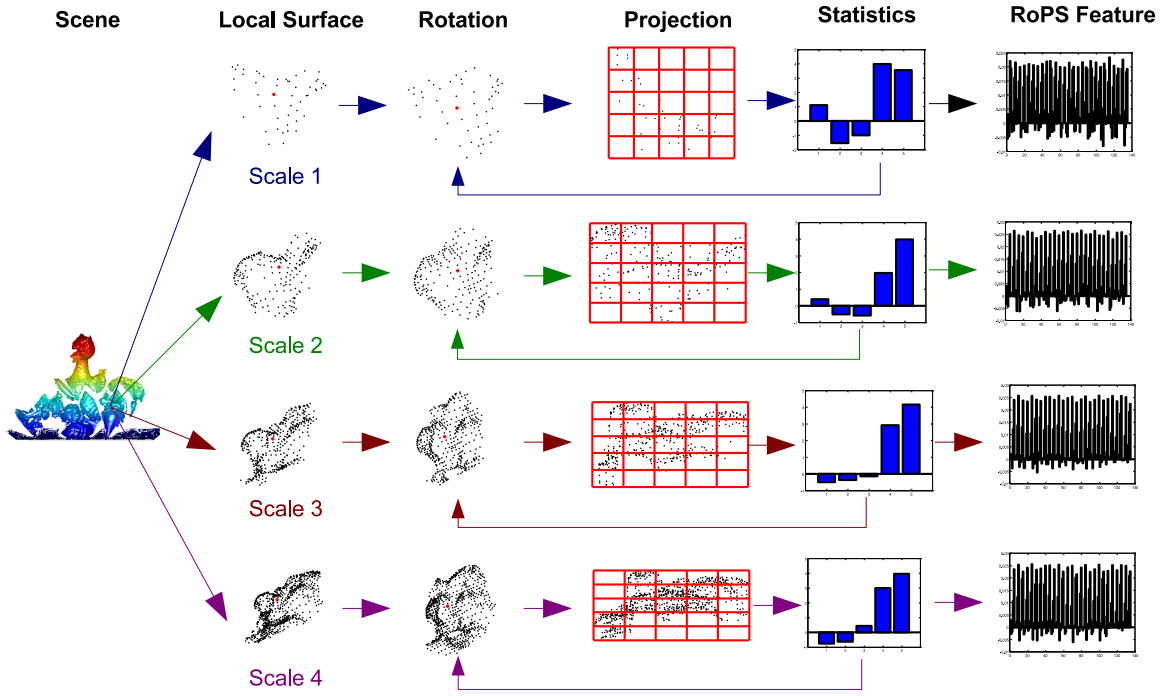
triangles and n_v vertices and each triangle $\mathcal{T}_i$ contains vertices $\boldsymbol{p}_{i1}$, $\boldsymbol{p}_{i2}$ and $\boldsymbol{p}_{i3}$, the scatter matrix $\mathbf{C}_i$ of each triangular face is calculated using all of the points lying within the triangle [4].

$$\begin{aligned}\mathbf{C}_i &= \frac{\int_0^1 \int_0^{1-v} (\boldsymbol{p}_c(v,u)-\boldsymbol{p})(\boldsymbol{p}_c(v,u)-\boldsymbol{p})^{\mathrm{T}} du dv}{\int_0^1 \int_0^{1-s} dt ds} \\ &= \frac{1}{12}\sum_{j=1}^{3}\sum_{k=1}^{3}\left(\boldsymbol{p}_{ij}-\boldsymbol{p}\right)\left(\boldsymbol{p}_{ik}-\boldsymbol{p}\right)^{\mathrm{T}} + \frac{1}{12}\sum_{j=1}^{3}\left(\boldsymbol{p}_{ij}-\boldsymbol{p}\right)\left(\boldsymbol{p}_{ij}-\boldsymbol{p}\right)^{\mathrm{T}}\end{aligned}, \tag{1}$$

where:

$$\boldsymbol{p}_c\left(v,u\right) = \boldsymbol{p}_{i1} + v(\boldsymbol{p}_{i2}-\boldsymbol{p}_{i1}) + u\left(\boldsymbol{p}_{i3}-\boldsymbol{p}_{i1}\right) \tag{2}$$

Figure 2. An illustration of the multi-scale rotational projection statistics (RoPS) feature description process (figure best seen in color).



The scatter matrices of the n_t triangles are then summed into an overall scatter matrix $\mathbf{C}$. Next, an eigenvalue decomposition is applied to the overall scatter matrix to result in three eigenvectors:

$$\mathbf{CV} = \mathbf{EV} \tag{3}$$

here, the diagonal entries of the matrix $\mathbf{E}$ correspond to the eigenvalues $\{\lambda_1, \lambda_2, \lambda_3\}$ of the scatter matrix $\mathbf{C}$, and the matrix $\mathbf{V}$ consists of the three orthogonal eigenvectors $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3\}$ of the scatter matrix $\mathbf{C}$.

Finally, a sign disambiguation technique is performed on the three eigenvectors $\{\boldsymbol{v}_1, \boldsymbol{v}_2, \boldsymbol{v}_3\}$, resulting in three orthogonal and unambiguous vectors. That is, each unambiguous vector points in the direction of the scatter vectors [4]. These vectors are used to form a unique and repeatable local reference frame (LRF) for the local surface $\mathcal{L}_k$.

Once the LRF for keypoint $\boldsymbol{p}$ with scale r_k is generated, the local surface $\mathcal{L}_k$ is aligned with the LRF to achieve its invariance with respect to rigid transformations (*i.e.*, rotations and translations), resulting in a transformed local surface $\mathcal{L}_k^{'}$. That is:

$$\mathcal{L}_k^{'} = \mathbf{R}_{lrf}\left(\mathcal{L}_k - \boldsymbol{p}\right) \tag{4}$$

where $\mathbf{R}_{lrf}$ is the rotation matrix defined by the LRF at the keypoint $\boldsymbol{p}$.

In order to encode the complete information of the local surface from different viewpoints, these points on $\mathcal{L}_k^{'}$ are rotated along the three coordinate axes (*i.e.*, the x-, y- and z-axes). Along each axis, the points are rotated by a set of angles $\{\theta_1, \theta_2, \cdots, \theta_{n_\theta}\}$, resulting in a resulted surface $\mathbf{R}_x(\theta)\mathcal{L}_k^{'}$. The rotation matrix $\mathbf{R}_x(\theta)$ along the x-axis is defined as:

$$\mathbf{R}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{5}$$

The rotation matrix $\mathbf{R}_y(\theta)$ along the is the y-axis is defined as:

$$\mathbf{R}_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{6}$$

The rotation matrix $\mathbf{R}_z(\theta)$ along the is the z-axis is defined as:

$$\mathbf{R}_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

Each rotation angle θ is defined between zero and 90 degrees. There is a tradeoff between the completeness and redundancy of the descriptor when selecting an appropriate number of rotation angles. Specifically, the descriptor with a larger number of rations encodes more information of the local surface. However, the information redundancy represented in the descriptor is also significantly higher. In our work, three rotations are used along each coordinate axis to achieve optimal overall performance.

For each rotation, these points are projected onto the three coordinate planes (*i.e.*, the xy, yz and xz planes). The projection process is defined as a mapping from a 3D space to a 2D space $\psi : \mathbb{R}^3 \rightarrow \mathbb{R}^2$. A distribution matrix $\mathbf{D}$ is then obtained on each plane by counting the number of points falling into the bins of a $L \times L$ lattice. The value of L determines both the descriptiveness and the robustness of the extracted descriptor. That is, a smaller value of L encodes more detail of the local surface with higher sensitivity to varying mesh resolutions. In our previous work [4], it is demonstrated that $L = 5$ provides the best overall performance. Consequently, $L = 5$ is used in this paper. The distribution matrix $\mathbf{D}$ can further be encoded with few low-dimensional statistics. Different combinations of several statistics have been investigated in [4]; it is shown that the combination of five statistics (including four central moments [34] and one Shannon entropy [35]) achieves the best experimental performance. These statistics are invariant to rotations and translations.

The moment μ_{mn} with the order $m + n$ is calculated as:

$$\mu_{mn} = \sum_{u=1}^{L}\sum_{v=1}^{L}\left(u - \bar{u}\right)^m\left(v - \bar{v}\right)^n \mathbf{D}\left(u, v\right) \tag{8}$$

The entropy e is defined as:

$$e = -\sum_{u=1}^{L}\sum_{v=1}^{L} \mathbf{D}(u,v)\log(\mathbf{D}(u,v)) \tag{9}$$

The total statistics of the distribution matrices on all planes (*i.e.*, the xy, yz and xz planes) with all rotations (*i.e.*, $\theta_1, \theta_2, \cdots, \theta_{n_\theta}$) are finally concatenated to form an overall RoPS feature descriptor. In order to represent the point cloud with multi-scale features, n_s feature descriptors $\{\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots, \boldsymbol{f}_{n_s}\}$ are generated for each keypoint. That is, the feature descriptor $\boldsymbol{f}_k$ is generated for keypoint $\boldsymbol{p}$ with scale r_k.

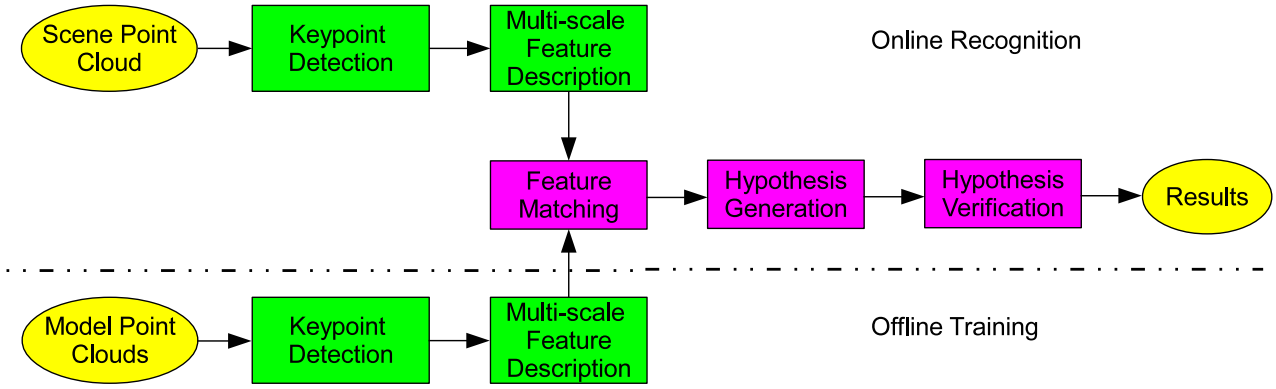
3. Object Recognition

In this section, we propose a novel 3D object recognition framework based on multi-scale feature representation and coarse-to-fine feature matching techniques.

3.1. Object Recognition Framework

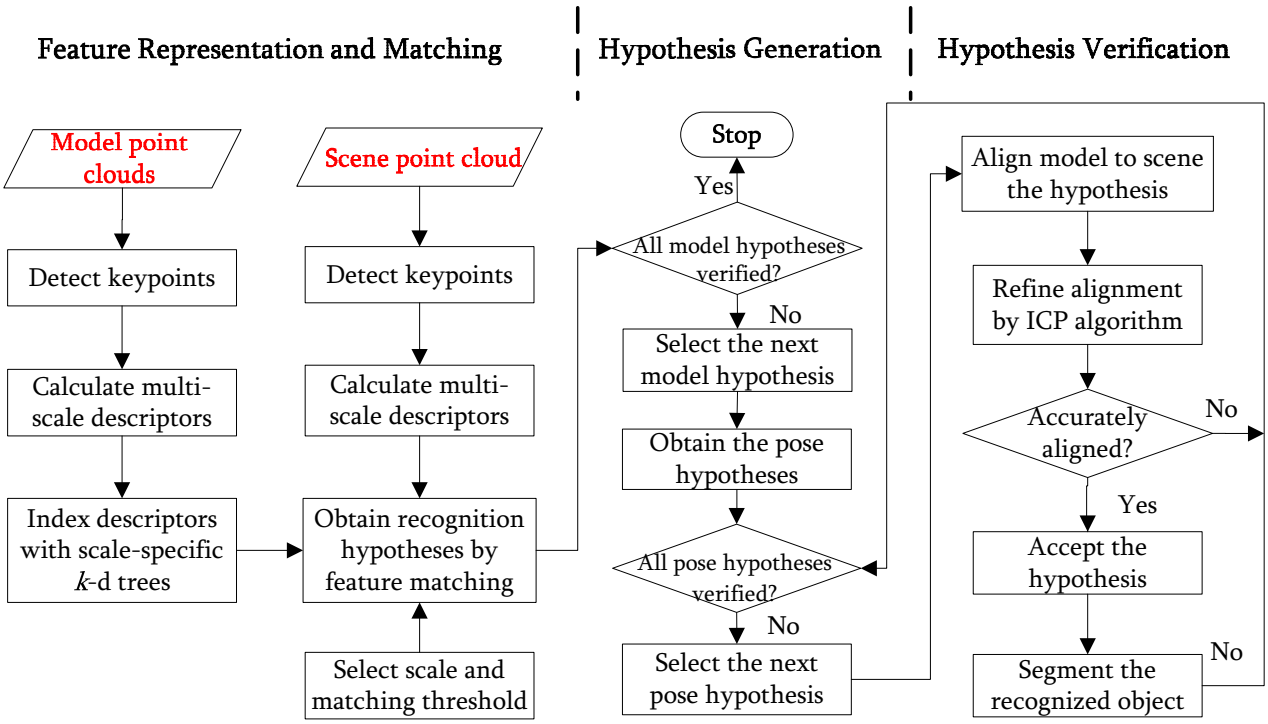
The pipeline of the 3D object recognition algorithm is shown in Figure 3, which consists of two major phases: offline training and online recognition. The flowchart of the 3D object recognition algorithm is presented in Figure 4.

Figure 3. The pipeline of the 3D object recognition algorithm (figure best seen in color).



During the phase of offline training, a model library $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, \cdots, \mathcal{M}_{N_m}\}$, which contains N_m models for the 3D objects of interest, are constructed. For each model, $\mathcal{M}_i$, n_m keypoints are first detected by uniform sampling and then pruned by a resolution control strategy [4,21]. For each keypoint $\boldsymbol{p}_m^i$, its LRF $\mathbf{F}_{mk}^i$ and RoPS descriptor $\boldsymbol{f}_{mk}^i$ for each scale r_k are calculated. In order to enable efficient feature matching during online recognition, n_s scale-specific k-d trees are separately constructed to index the RoPS descriptors of all models. That is, for each scale r_k, all RoPS descriptors that correspond to the scale r_k are indexed with a k-d tree.

Figure 4. The flowchart of the 3D object recognition algorithm.



During the phase of online recognition, n_p keypoints are detected from the scene $\mathcal{S}$ using the technique presented in Section 2.1. For each keypoint $\boldsymbol{p}_s$, its LRF $\mathbf{F}_{sk}$ and RoPS descriptor $\boldsymbol{f}_{sk}$ for each scale r_k are generated. Consequently, the scene is represented by a set of multi-scale RoPS descriptors $\{\boldsymbol{f}_{sk}\}\,(s = 1, 2, \cdots, n_p.k = 1, 2, \cdots, n_s)$, where n_p is the number of scene keypoints and n_s is the number of scales for each keypoint. We then propose a multi-scale feature matching strategy to produce a set of recognition hypotheses $\mathcal{H} = \{h_1, h_2, \cdots, h_{n_h}\}$. Each hypothesis h_l is defined by a pair $(\mathcal{M}_{h_l}, \mathcal{T}_{h_l})$, where $\mathcal{M}_{h_l}$ is the model hypothesis and $\mathcal{T}_{h_l}$ is the pose hypothesis, which is used to transform $\mathcal{M}_{h_l}$ to $\mathcal{S}$. Given the hypotheses $\mathcal{H}$, a hypothesis verification module is used to distinguish true hypotheses from false hypotheses, which further improves the rate of true positives while reducing the rate of false positives.

3.2. Feature Matching

We assume that the scene features are $\{\boldsymbol{f}_{sk}\}\,(s = 1, 2, \cdots, n_p, k = 1, 2, \cdots, n_s)$ and the model features are $\left\{\boldsymbol{f}^i_{mk}\right\}\,(i = 1, 2, \cdots, N_m, m = 1, 2, \cdots, n_m, k = 1, 2, \cdots, n_s)$, where n_p is the number of scene keypoints, N_m is the number of models, n_m is the number of model keypoints and n_s is the number of scales for each keypoint. For each scene feature $\boldsymbol{f}_{sk}$ with scale r_k, it is matched against all model features in the library that have the same scale. Here, feature matching is performed using the previously-constructed k-d tree in order to speed up the process.

A scene feature $\boldsymbol{f}_{sk}$ and a model feature $\boldsymbol{f}^i_{mk}$ are matched if $\boldsymbol{f}^i_{mk}$ is the nearest neighbor to $\boldsymbol{f}_{sk}$ and if the distance ratio between the first and the second nearest neighbors is below a threshold τ_f. In this paper, several values have been used for the threshold τ_f to perform coarse-to-fine feature matching and object recognition (see Section 3.5 for more details). The scene feature $\boldsymbol{f}_{sk}$ and its matched model feature $\boldsymbol{f}^i_{mk}$

are considered a feature correspondence $\left(\boldsymbol{f}_{sk}, \boldsymbol{f}^{i}_{mk}\right)$. Each feature correspondence $\left(\boldsymbol{f}_{sk}, \boldsymbol{f}^{i}_{mk}\right)$ gives a vote to the i-th model. Then, the transformation (*i.e.*, pose estimation) $\mathbf{T}^{i}_{smk}$ between the i-th model and the scene is calculated. The pose estimation $\mathbf{T}^{i}_{smk}$ consists of a rotation matrix $\mathbf{R}^{i}_{smk}$ and a translation vector $\boldsymbol{t}^{i}_{smk}$, that is:

$$\mathbf{R}^{i}_{smk} = \left(\mathbf{F}_{sk}\right)^{\mathrm{T}} \mathbf{F}^{i}_{mk} \tag{10}$$

$$\boldsymbol{t}^{i}_{smk} = \boldsymbol{p}_{s} - \mathbf{R}^{i}_{smk}\boldsymbol{p}^{i}_{m} \tag{11}$$

where $\boldsymbol{p}_{s}$ is the scene keypoint, $\boldsymbol{p}^{i}_{m}$ is the keypoint of the i-th model, $\mathbf{F}_{sk}$ is the LRF at the scene keypoint $\boldsymbol{p}_{s}$ with the scale r_{k} and $\mathbf{F}^{i}_{mk}$ is the LRF at the model keypoint $\boldsymbol{p}^{i}_{m}$ with the scale r_{k}.

3.3. Hypothesis Generation

For a given scale r_{k} and matching threshold τ_{f}, a set of feature correspondences can be generated. The models that have received votes from the feature correspondences are considered model hypotheses. For each model hypothesis $\mathcal{M}_{h_{l}}$, its associated pose estimations are then grouped into several clusters using the technique proposed in [4]. We calculate the cluster center $(\mathbf{R}_{c}, \boldsymbol{t}_{c})$ for each cluster as the average value of all rotations and translations which fall in that cluster. Each cluster center is considered a pose hypothesis for the model hypothesis $\mathcal{M}_{h_{l}}$. Note that, more than one cluster (*i.e.*, pose hypotheses $\mathcal{T}_{h_{l}}$) can be generated for each model hypothesis $\mathcal{M}_{h_{l}}$.

3.4. Hypothesis Verification

Given the scene $\mathcal{S}$ and hypotheses $\mathcal{H} = \{h_{1}, h_{2}, \cdots, h_{n_{h}}\}$, each hypothesis $h_{l} = (\mathcal{M}_{h_{l}}, \mathcal{T}_{h_{l}})$ is verified as follows. First, the model $\mathcal{M}_{h_{l}}$ is aligned with the scene $\mathcal{S}$ using the pose hypothesis $\mathcal{T}_{h_{l}}$. The alignment is further refined with an iterative closest point (ICP) algorithm [36]. The residual error ε of the ICP process is selected as a measure for the alignment. In addition, we define a visible proportion α as another measure, that is:

$$\alpha = \frac{n_{closest}}{n_{scene}} \tag{12}$$

where $n_{closest}$ is the number of closest point pairs between $\mathcal{S}$ and $\mathcal{M}_{h_{l}}$ and n_{scene} is the number of points in the scene $\mathcal{S}$.

The two measures ε and α are used to determine whether the hypothesis can be accepted or not. Ideally, for an object that is not occluded and its pose is accurately estimated, the residual error ε is zero and the visible proportion α is one. In practice, two thresholds τ_{ε} and τ_{α} are used to determine an acceptable hypothesis. In order to accept as many correct hypotheses as possible while reducing false positives, a flexible thresholding scheme is used in this paper to perform hypothesis verification. Specifically, two groups of thresholds $(\tau_{\varepsilon 1} = 0.75\ \mathrm{mr}, \tau_{\alpha 1} = 0.04)$ and $(\tau_{\varepsilon 2} = 1.5\ \mathrm{mr}, \tau_{\alpha 2} = 0.2)$ are adopted, where 'mr' denotes the average mesh resolution. These thresholds were determined by a tuning experiment, and the same values were applied to all experiments presented in the paper. The hypothesis $(\mathcal{M}_{h_{l}}, \mathcal{T}_{h_{l}})$ is accepted only if $\varepsilon < \tau_{\varepsilon 1}$ and $\alpha > \tau_{\alpha 1}$ or $\varepsilon < \tau_{\varepsilon 2}$ and $\alpha > \tau_{\alpha 2}$. Otherwise, the hypothesis is rejected.

3.5. Coarse-to-Fine Recognition

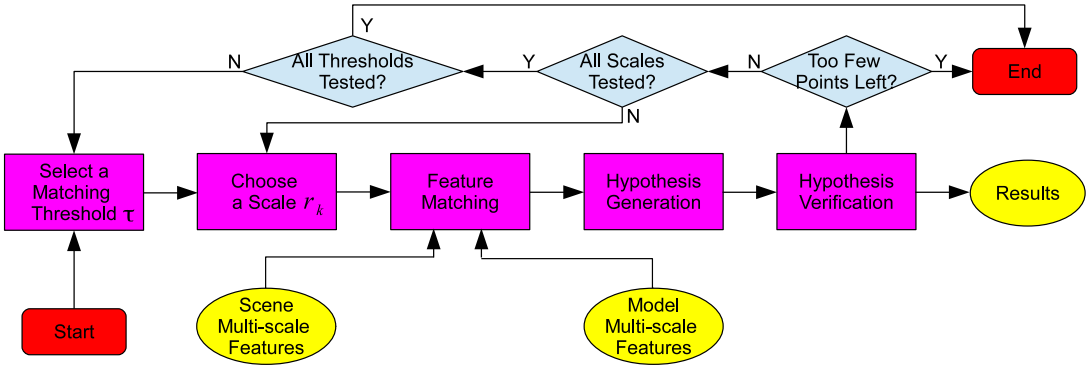
Most existing algorithms generate feature descriptors using a single scale and perform feature matching using a pre-defined threshold τ_f. They however have a number of limitations. First, it is difficult to choose an appropriate scale for a fixed-scale feature-based 3D object recognition algorithm. That is, feature descriptors with a large scale are very sensitive to occlusion and clutter (which is common in most object recognition scenarios). In contrast, feature descriptors with a small scale lack rich descriptiveness. Second, although many adaptive-scale keypoint detection algorithms have been proposed in the literature (e.g., Mesh Difference of Gaussians (MeshDoG) [31], keypoint quality-adaptive scale (KPQ-AS) [27] and salient points (SP) [37]), their scale repeatability is low [26]. For example, the scale repeatability of MeshHoG, KPQ-AS and SP algorithms is, respectively, 41%, 51% and 43%, when tested on the University of Western Australia (UWA) laser scanner dataset [26]. Consequently, the performance of object recognition is adversely affected by the errors of scale estimation. Third, the pre-defined threshold τ_f is data-dependent and very difficult to determine. That is, although selecting a strict threshold can produce highly accurate feature correspondences, the number of feature correspondences may be too few to perform effective object recognition. On the contrary, selecting a loose threshold would produce lots of false feature correspondences, which not only increases the computational time, but also deteriorates the accuracy of object recognition.

In this paper, a coarse-to-fine algorithm is proposed to solve these problems. The algorithm is illustrated in Figure 5. Multiple scales (*i.e.*, 5 mr, 10 mr, 15 mr and 20 mr) and different matching thresholds τ_f (*i.e.*, 0.7, 0.8, 0.9 and 1.0) are used in the algorithm. Here, 'mr' stands for the average mesh resolution. Note that the values of the aforementioned thresholds were selected by a tuning experiment and were applied to the experiments on all datasets (Section 4). First, the algorithm uses large-scale features (with a scale of 20 mr) and a strict threshold (with a value of 0.7) to perform feature matching (Section 3.2), hypothesis generation (Section 3.3) and hypothesis verification (Section 3.4). If part of the hypotheses are accepted by the algorithm, the instances of these model hypotheses are recognized from the scene, and the scene points that belong to these model hypotheses are removed. Once all resulting hypotheses are verified, the object recognition algorithm then proceeds to features with a smaller scale, while keeping the matching threshold fixed. The aforementioned feature matching, hypothesis generation and hypothesis verification modules are then repeated. Once features of all scales for a fixed matching threshold are tested, the algorithm proceeds to a looser matching threshold. The aforementioned process continues until either too few points left in the scene for recognition or all scales and thresholds have been tested.

The strengths of the coarse-to-fine algorithm are as follows. First, the objects with large visible parts in the scene can be recognized with a high priority, since the object recognition algorithm starts with matching of large-scale features. Consequently, most of these highly visible objects can be segmented from the scene within a few iterations, which significantly reduces the overall recognition time. Second, the object with small visible parts in the scene can also be easily recognized due to the reason that most of the large objects have already been segmented in advance. That is, there are very few clutter points left in the scene. As a result, the recognition rate is increased. Third, the feature correspondences are sequentially verified by the order of their distinctiveness. That is, the most distinctive feature

correspondences are verified with a strict threshold before these less distinctive ones. As a result, the computational efficiency is improved, and the number of correct feature correspondences is also increased.

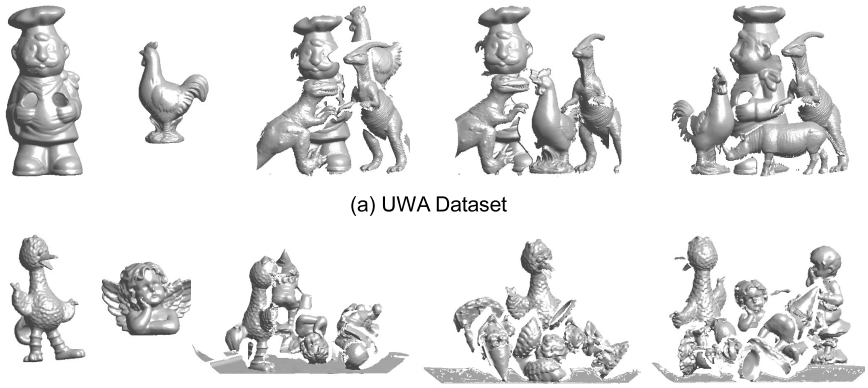
Figure 5. Coarse-to-fine object recognition algorithm (figure best seen in color).



4. Experimental Results

The proposed algorithm is tested on two publicly available datasets, *i.e.*, the University of Western Australia (UWA) dataset [10] and the Queen's LiDAR dataset [19,38]. Some example images of the two datasets are shown in Figure 6.

Figure 6. Example images of the University of Western Australia (UWA) and Queen's datasets.



4.1. Recognition Results on the UWA Dataset

The UWA dataset is currently regarded as the most popular benchmark for 3D object recognition [4,10,11,13,19,27,32]. It consists of five models and 50 scenes. Each scene contains four or five of the models in the presence of occlusion and clutter. Specifically, four or five models were first selected and randomly placed on a table, and a point cloud was then acquired by the triangulation based Konica Minolta Vivid 910 scanner from a single viewpoint. The total number of instances of each object in all scenes is shown in Table 1. The recognition rate of each object on the UWA dataset is also presented in Table 1. It can be observed that chef, chicken and T-rex achieved the highest recognition

rate of 100%. Besides, Parasaurolophus and rhino also obtained a high recognition rate of more than 96%. Specifically, both of them had only one instance left in the scene that was not correctly recognized. The failure of these two instances was due to being highly occluded. The overall recognition rate of the five objects is 99.1%. The results clearly confirm that the proposed coarse-to-fine recognition algorithm is capable of recognizing objects in complex scenes in the presence of multiple objects, occlusion, clutter and noise.

Table 1. Recognition rate of each object on the UWA dataset.

	Chef	**Chicken**	**Parasaurolophus**	**T-Rex**	**Rhino**	**Overall**
Number of Instances	50	48	45	45	28	216
Recognition Rate (%)	100	100	97.8	100	96.4	99.1

In order to further analyze the robustness of our algorithm with respect to occlusion and clutter, we present the recognition rates of the five objects on the 50 scenes in Figure 7, as a function of occlusion and clutter. The results reported by the EM-based algorithm [32] are also shown in Figure 7. According to the definitions presented in [10,32], occlusion is calculated as:

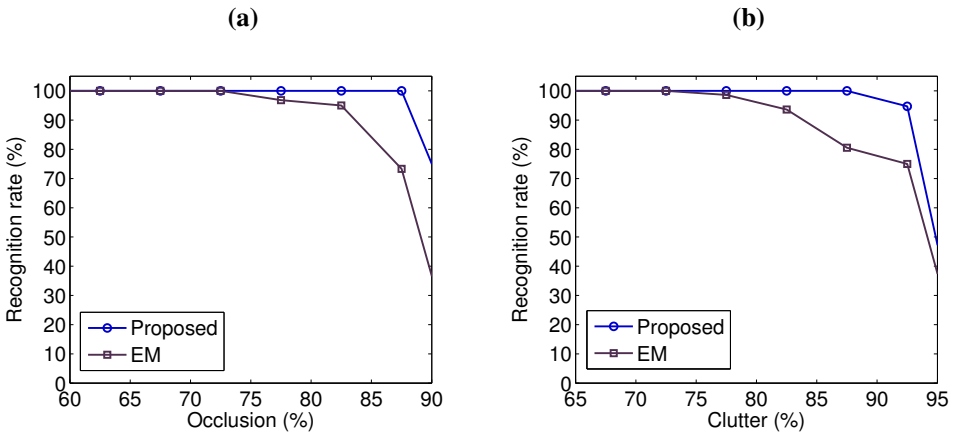
$$\text{occlusion} = 1 - \frac{\text{model surface patch area in scene}}{\text{total model surface area}} \tag{13}$$

Clutter is calculated as:

$$\text{clutter} = 1 - \frac{\text{model surface patch area in scene}}{\text{total surface area of scene}} \tag{14}$$

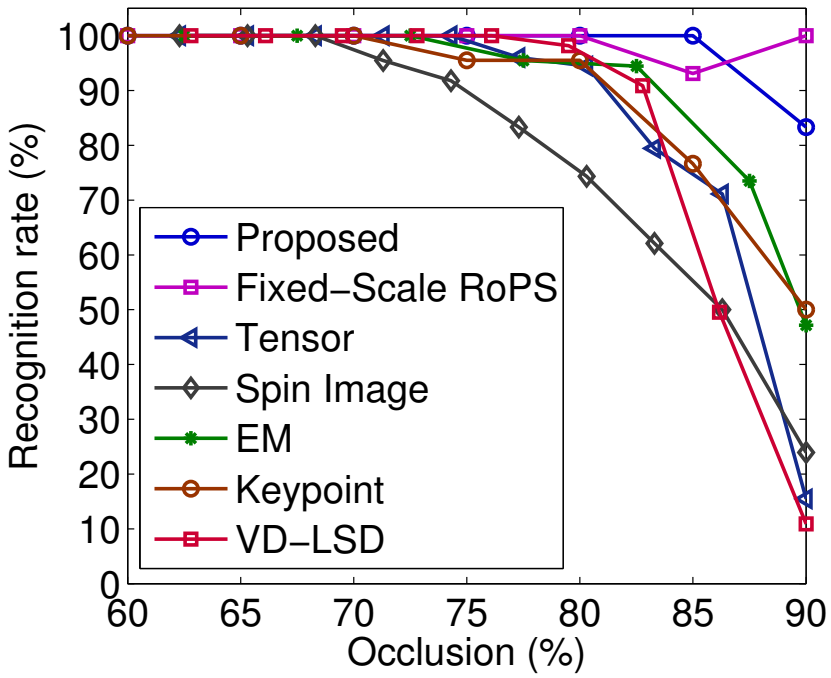
We can observe that our algorithm is very robust to occlusion and clutter. It achieved a high recognition rate of 100% with up to 87.5% occlusion and 87.5% clutter. Its recognition rate was still as high as 94.7% with up to 92.5% clutter. Our algorithm clearly outperformed the EM-based algorithm [32], especially on the scenes with high values of occlusion and clutter. These comparative results further demonstrate the effectiveness of our algorithm for object recognition in the presence of significant occlusion and clutter.

Figure 7. Recognition rates of the five objects on the 50 scenes of the UWA dataset. (**a**) Recognition rates with respect to occlusion; (**b**) Recognition rates with respect to clutter.



In order to perform a rigorous and fair comparison with the state-of-the-art algorithms [4,10,32], we compare our results with the recognition results presented in [4,10,32] on exactly the same dataset of cluttered scenes. That is, we excluded the model rhino from our recognition results. That is because the spin image-based algorithm failed to recognize the rhino in any of these scenes (as discussed in [10]). Figure 8 shows the recognition rates of the remaining four objects with respect to occlusion. The results reported by tensor [10], spin image [10], keypoint [27], VD-LSD [19], EM [32] and fixed-scale RoPS [4] -based algorithms are also presented in Figure 8. The recognition rate of our algorithm in this case was 100% with up to 84% occlusion. In contrast, the recognition rates of tensor [10], spin image [10], EM [32] and fixed-scale RoPS [4] -based algorithms were respectively 96.6%, 87.7%, 97.5% and 98.8%, with up to 84% occlusion. The proposed algorithm obtained the best overall recognition rate of 99.5%, followed by the fixed-scale RoPS-based algorithm (with an overall recognition rate of 98.9%).

Figure 8. Recognition rates of the four objects on the 50 scenes of the UWA dataset with respect to occlusion (figure best seen in color).



The superior performance of our algorithm is due to several facts. First, our RoPS feature outperforms the state-of-the-art local surface features in terms of recall and precision [4]. It is also very robust to a set of variations, including clutter, occlusion, noise and varying mesh resolutions, as demonstrated in [4]. Consequently, both fixed-scale and multi-scale RoPS feature-based algorithms achieved better performance compared to the others (as shown in Figure 8). Second, multi-scale RoPS features are capable of encoding both coarse and fine structures of an object. They are, therefore, more effective for the purpose of object recognition compared to their fixed-scale counterparts (as shown in Figure 8). Specifically, large-scale features are more suitable for the efficient recognition of objects with small occlusion. In contrast, small-scale features are more appropriate for the robust recognition of objects with large occlusion. An illustration of the scene with high occlusion (*i.e.*, the chicken model) is shown in Figure 9. Finally, our coarse-to-fine recognition algorithm uses multiple thresholds for feature

matching rather than a single threshold. It therefore, produces more hypotheses and ultimately improves the recognition accuracy.

Figure 9. An illustration of the scene with occlusion of an object (*i.e.*, the chicken model) by more than 85%. EM, exponential map; VD-LSD, variable-dimensional local shape descriptors.

4.2. Recognition Results on Queen's LIDAR Dataset

The Queen's LIDAR dataset is composed of five models and 80 scenes. Each scene was generated by placing one, three, four or five of the models in a scene and was scanned from a single viewpoint using a Konica-Minolta Vivid 3D scanner [19]. The objects in each scene are highly cluttered, where the clutter includes both other objects and background [19]. We first tested our algorithm on the full dataset, which contains 80 scenes. Table 2 shows our recognition rates of the five objects on the dataset, with a direct comparison to the results achieved by EM [32] and fixed-scale RoPS [4] -based algorithms. It is clear that our algorithm achieved a recognition rate of 100% for all objects in that dataset. The second best place is taken by the fixed-scale RoPS-based algorithm, with an average recognition rate of 95.4%. In contrast, the performance achieved by the EM-based algorithm is relatively low, with an average recognition rate of 82.4%. It can be inferred that the proposed multi-scale RoPS feature-based algorithm further improves the performance of 3D object recognition compared to the fixed-scale RoPS feature-based algorithm [4].

Table 2. Recognition rates on the full Queen's dataset. The best results are in bold face.

Algorithm	Angel (%)	Big-Bird (%)	Gnome (%)	Kid (%)	Zoe (%)	Average (%)
Proposed	**100**	**100**	**100**	**100**	**100**	**100**
Fixed-Scale RoPS [4]	97.9	100	97.7	95.8	85.4	95.4
EM [32]	77.1	87.5	87.5	83.3	76.6	82.4

In order to have a fair comparison with the results reported by EM-, VD-LSD-, 3DSC-, spin image- and fixed-scale RoPS-based algorithms, we tested our coarse-to-fine 3D object recognition algorithm on the same dataset as [4,19,32]. The selected dataset is actually a subset of the full Queen's LiDAR dataset. The subset dataset contains only 55 scenes. Each scene consists of three, four or five objects. Our recognition rates of the five objects on this subset dataset are shown in Table 3. We also present the results reported by fixed-scale RoPS-, EM-, VD-LSD with scalar quantization (VD-LSD-(SQ)) , VD-LSD with vector quantization (VD-LSD-(VQ)) , 3DSC-, spin image- and spin image spherical-based algorithms in Table 3. Similar to the results achieved on the full dataset, our coarse-to-fine algorithm obtained a recognition rate of 100% for all objects on this subset dataset. It is better than the fixed-scale RoPS-based algorithm by a margin of 4.6% in the average recognition rate. This observation fully indicates that the proposed multi-scale RoPS-based coarse-to-fine algorithm outperforms the algorithm that uses fixed-scale RoPS features and a single matching threshold. As compared with other algorithms, the advantage of the proposed algorithm is even more significant. That is, the average recognition rates reported by all other algorithms (except fixed-scale RoPS) are less than 85%.

Table 3. Recognition rates on a subset of the Queen's dataset. The best results are in bold face. 3DSC, 3D shape context.

Algorithm	Angel (%)	Big-Bird (%)	Gnome (%)	Kid (%)	Zoe (%)	Average (%)
Proposed	**100**	**100**	**100**	**100**	**100**	**100**
Fixed-Scale RoPS [4]	97.4	100	97.4	94.9	87.2	95.4
EM [32]	NA	NA	NA	NA	NA	81.9
VD-LSD (SQ) [19]	89.7	100.0	70.5	84.6	71.8	83.8
VD-LSD (VQ) [19]	56.4	97.4	69.2	51.3	64.1	67.7
3DSC [19]	53.8	84.6	61.5	53.8	56.4	62.1
Spin Image [19]	53.8	84.6	38.5	51.3	41.0	53.8
Spin Image Spherical [19]	53.8	74.4	38.5	61.5	43.6	54.4

5. Conclusions

In this paper, we proposed a coarse-to-fine algorithm for 3D object recognition in point clouds. We used multi-scale RoPS features to represent an object and performed 3D object recognition based on feature matching, hypothesis generation and hypothesis verification. We employed several different matching thresholds to conduct feature matching in order to further improve the object recognition performance. The proposed algorithm was evaluated on two publicly available datasets. Experimental results show that our algorithm outperformed existing algorithms in terms of recognition rates. It is shown that the algorithm is also very robust to occlusion and clutter. In our future work, we aim to investigate the challenging task for the recognition of geometrically featureless objects/scenes (e.g., with planar surfaces). One of the prospective solutions is to integrate geometric and photometric information for object recognition.

Acknowledgments

This research is supported by two funds (Nos. 61471371 and 61403265) from the National Natural Science Foundation of China (NSFC).

Author Contributions

Min Lu and Yulan Guo contributed equally to this work and should be considered co-first authors. Min Lu was responsible for the theoretical work of this paper, and Yulan Guo and Yanxin Ma implemented the algorithm and conducted most of the experiments. The draft of this article was initially written by Min Lu and Yulan Guo and further revised by Jun Zhang and Yinjie Lei. Yinjie Lei also contributed to the experimental setup.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Aggarwal, A.; Kirchner, F. Object Recognition and Localization: The Role of Tactile Sensors. *Sensors* **2014**, *14*, 3227–3266.
2. Astua, C.; Barber, R.; Crespo, J.; Jardon, A. Object Detection Techniques Applied on Mobile Robot Semantic Navigation. *Sensors* **2014**, *14*, 6734–6757.
3. Guo, Y.; Wan, J.; Lu, M.; Niu, W. A Parts-based Method for Articulated Target Recognition in Laser Radar Data. *Opt. Int. J. Light Electron Opt.* **2013**, *124*, 2727–2733.
4. Guo, Y.; Sohel, F.; Bennamoun, M.; Lu, M.; Wan, J. Rotational Projection Statistics for 3D Local Surface Description and Object Recognition. *Int. J. Comput. Vis.* **2013**, *105*, 63–86.
5. Lei, Y.; Bennamoun, M.; Hayat, M.; Guo, Y. An Efficient 3D Face Recognition Approach using Local Geometrical Signatures. *Pattern Recognit.* **2014**, *47*, 509–524.
6. Rusu, R.B.; Cousins, S. 3D is here: Point cloud library (PCL). In Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011; pp. 1–4.
7. Sui, C.; Haque, S.; Togneri, R.; Bennamoun, M. A 3D audio-visual corpus for speech recognition. In Proceedings of the 14th Australasian International Conference on Speech Science and Technology, Sydney, Australia, 3–6 December 2012; pp. 125–128.
8. Cahalane, C.; McElhinney, C.; Lewis, P.; McCarthy, T. Calculation of Target-Specific Point Distribution for 2D Mobile Laser Scanners. *Sensors* **2014**, *14*, 9471–9488.
9. Paulus, S.; Dupuis, J.; Riedel, S.; Kuhlmann, H. Automated Analysis of Barley Organs Using 3D Laser Scanning: An Approach for High Throughput Phenotyping. *Sensors* **2014**, *14*, 12670–12686.
10. Mian, A.; Bennamoun, M.; Owens, R. Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 1584–1601.

11. Guo, Y.; Bennamoun, M.; Sohel, F.; Lu, M.; Wan, J. 3D Object Recognition in Cluttered Scenes with Local Surface Features: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 2270–2287.
12. Shah, S.A.A.; Bennamoun, M.; Boussaid, F.; El-Sallam, A. A Novel Local Surface Description for Automatic 3D Object Recognition in Low Resolution Cluttered Scenes. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Sydney, Australia, 2–8 December 2013; pp. 638–643.
13. Aldoma, A.; Tombari, F.; di Stefano, L.; Vincze, M. A global hypotheses verification method for 3D object recognition. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 511–524.
14. Osada, R.; Funkhouser, T.; Chazelle, B.; Dobkin, D. Shape distributions. *ACM Trans. Graph.* **2002**, *21*, 807–832.
15. Shah, S.A.A.; Bennamoun, M.; Boussaid, F.; El-Sallam, A. 3D-DIV: A Novel Local Surface Descriptor for Feature Matching and Pairwise Range Image Registration. In Proceedings of the 2013 20th IEEE International Conference on Image Processing, Melbourne, Australia, 15–18 September 2013; pp. 2934–2938.
16. Meng, X.; Yin, Y.; Yang, G.; Xi, X. Retinal identification based on an improved circular Gabor filter and scale invariant feature transform. *Sensors* **2013**, *13*, 9248–9266.
17. Guo, Y.; Sohel, F.; Bennamoun, M.; Wan, J.; Lu, M. Integrating Shape and Color Cues for Textured 3D Object Recognition. In Proceedings of the 8th IEEE Conference on Industrial Electronics and Applications, Melbourne, Australia, 19–21 June 2013.
18. Guo, Y.; Bennamoun, M.; Sohel, F.; Wan, J.; Lu, M. 3D Free Form Object Recognition using Rotational Projection Statistics. In Proceedings of the IEEE 14th Workshop on the Applications of Computer Vision, Tampa, FL, USA, 15–17 January 2013; pp. 1–8.
19. Taati, B.; Greenspan, M. Local shape descriptor selection for object recognition in range data. *Comput. Vis. Image Underst.* **2011**, *115*, 681–694.
20. Johnson, A.E.; Hebert, M. Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* **1999**, *21*, 433–449.
21. Zhong, Y. Intrinsic shape signatures: A shape descriptor for 3D object recognition. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Kyoto, Japan, 27 September–4 October 2009; pp. 689–696.
22. Hou, T.; Qin, H. Efficient computation of scale-space features for deformable shape correspondences. In Proceedings of the European Conference on Computer Vision, Heraklion, Crete, Greece, 5–11 September 2010; pp. 384–397.
23. Mikolajczyk, K.; Schmid, C. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1615–1630.
24. Lowe, D. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110.
25. Guo, Y.; Sohel, F.; Bennamoun, M.; Lu, M.; Wan, J. TriSI: A Distinctive Local Surface Descriptor for 3D Modeling and Object Recognition. In Proceedings of the 8th International Conference on Computer Graphics Theory and Applications, Barcelona, Spain, 21–24 February 2013; pp. 86–93.

26. Tombari, F.; Salti, S.; di Stefano, L. Performance Evaluation of 3D Keypoint Detectors. *Int. J. Comput. Vis.* **2013**, *102*, 198–220.
27. Mian, A.; Bennamoun, M.; Owens, R. On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. *Int. J. Comput. Vis.* **2010**, *89*, 348–361.
28. Sun, Y.; Abidi, M. Surface matching by 3D point's fingerprint. In Proceedings of the 8th IEEE International Conference on Computer Vision, Vancouver, BC, Canada, 7–14 July 2001; Volume 2, pp. 263–269.
29. Frome, A.; Huber, D.; Kolluri, R.; Bülow, T.; Malik, J. Recognizing objects in range data using regional point descriptors. In Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic, 11–14 May 2004; pp. 224–237.
30. Malassiotis, S.; Strintzis, M. Snapshots: A novel local surface descriptor and matching algorithm for robust 3D surface alignment. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 1285–1290.
31. Zaharescu, A.; Boyer, E.; Horaud, R. Keypoints and Local Descriptors of Scalar Functions on 2D Manifolds. *Int. J. Comput. Vis.* **2012**, *100*, 78–98.
32. Bariya, P.; Novatnack, J.; Schwartz, G.; Nishino, K. 3D Geometric Scale Variability in Range Images: Features and Descriptors. *Int. J. Comput. Vis.* **2012**, *99*, 232–255.
33. Guo, Y.; Sohel, F.; Bennamoun, M.; Wan, J.; Lu, M. An Accurate and Robust Range Image Registration Algorithm for 3D Object Modeling. *IEEE Trans. Multimed.* **2014**, *16*, 1377–1390.
34. Hu, M. Visual pattern recognition by moment invariants. *IRE Trans. Inf. Theory* **1962**, *8*, 179–187.
35. Shannon, C. A mathematical theory of communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423.
36. Besl, P.J.; McKay, N.D. A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **1992**, *14*, 239–256.
37. Castellani, U.; Cristani, M.; Fantoni, S.; Murino, V. Sparse points matching by combining 3D mesh saliency with statistical descriptors. *Comput. Graph. Forum* **2008**, *27*, 643–652.
38. Taati, B.; Bondy, M.; Jasiobedzki, P.; Greenspan, M. Variable dimensional local shape descriptors for object recognition in range data. In Proceedings of the 11th IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil, 14–21 October 2007; pp. 1–8.

© 2014 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/4.0/).