

# An Improved Local Descriptor based Object Recognition in Cluttered 3D Point Clouds

X. Liu, Y. Lu, T. Wu, T. Yuan

## Xiaoni Liu

Key Lab of Symbol Computation and  
Knowledge Engineer of Ministry of Education,  
College of Computer Science and Technology,  
Jilin University, Changchun 130012, China  
liu\_xiaoni@foxmail.com

## Yinan Lu\*

Key Lab of Symbol Computation and  
Knowledge Engineer of Ministry of Education,  
College of Computer Science and Technology,  
Jilin University, Changchun 130012, China  
\*Corresponding author: luyin@jlu.edu.cn

## Tieru Wu

College of Mathematics,  
Jilin University, Changchun 130012, China  
wutr@jlu.edu.cn

## Tianwen Yuan

College of Computer Science and Technology,  
Jilin University, Changchun 130012, China  
yuantw14@mails.jlu.edu.cn

**Abstract:** Object recognition in three-dimensional point clouds is a new research topic in the field of computer vision. Numerous nuisances, such as noise, a varying density, and occlusion greatly increase the difficulty of 3D object recognition. An improved local feature descriptor is proposed to address these problems in this paper. At each feature point, a local reference frame is established by calculating a scatter matrix based on the geometric center and the weighted point-cloud density of its neighborhood, and an improved normal vector estimation method is used to generate a new signature of histograms of orientations (SHOT) local-feature descriptor. The geometric consistency and iterative closest point method realize 3D model recognition in the point-cloud scenes. The experimental results show that the proposed SHOT feature-extraction algorithm has high robustness and descriptiveness in the object recognition of 3D local descriptors in cluttered point-cloud scenes.

**Keywords:** 3D point cloud, local feature, object recognition, noise, density variation.

## 1 Introduction

With the development of computer software theory and hardware, three-dimensional (3D) model data have become the fourth-generation multimedia data type after sound, images, and video. The 3D point cloud is an important representation of 3D model data. The purpose of object recognition in 3D point clouds is to identify the object of interest from the 3D point-cloud scenes and obtain its pose (such as position and orientation). Object recognition in 3D point clouds has become a research hotspot in the fields of interior decoration, robot arm crawling, urban planning, marine environmental monitoring, computer-aided design and manufacturing, biomedicine, smartphones, and aerospace technology.

Feature extraction is a very critical step and affects the performance of identification systems. The feature descriptors can be categorized into two groups: global and local descriptors [10]. Global descriptors compute the feature for a cluster of points representing a whole object but rather for individual points. Hence, they encode the geometry of the whole object using their  $k$ -neighborhood data. This has led to promising results in 3D model retrieval and shape classification. A pre-segmentation step is required before the computation of the global descriptors of a given object, in order to obtain the cluster that represents the object. The most popular global descriptors are the viewpoint feature histogram (VFH) [1], clustered VFH (CVFH) [2], ensemble of shape functions (ESF) [24], and global fast PFH [21]. Local descriptors are computed for individual points by using its underlying geometrical information, obtained from the query point  $k$ -neighborhood. They include point signatures (PSs) [7], spin images (SIs) [14], 3D shape context (3DSC) [8], the point feature histogram (PFH) [19], the fast PFH (FPFH) [20], the signature of histograms of orientations (SHOT) [23], the radius-based surface descriptor [3], the unique-shape context descriptor [11], the rotation-invariant feature transform [15], and the normal-aligned radial feature [6].

For a qualified local surface feature, robustness and descriptiveness are very important factors. A histogram is used to accurately describe the local area, when especially considering the spatial distribution of the neighboring points.

It is a challenging task to achieve high descriptiveness and strong robustness to various nuisances faced by the existing methods. SHOT can compete against most descriptors and can gather more information. It has two disadvantages: a large amount of computational storage is required, and the original LRF of SHOT is sensitive to the point-cloud density variation. An improved SHOT descriptor is proposed to address the noise and varying point cloud density problems in this paper. An improved scatter matrix computation based on weighted distance and point-cloud density is used for establishing an LRF and generating the normal for each point in SHOT descriptor. In an LRF, the position of geometric center is iteratively calculated based on the weighted point-cloud density. The experimental results show that the proposed feature-extraction method is robust to noise and point-cloud variations, and highly efficient for the 3D object recognition in cluttered point-cloud scenes.

The remainder of this paper is organized as follows. Section 2 briefly reviews previous studies related to point-cloud descriptors. Section 3 describes the proposed SHOT feature and a 3D object-recognition algorithm. Sections 4 and 5 present the feature-matching results and the object-recognition results for a standard dataset, respectively. Finally, we draw conclusions in Section 6.

## 2 Related work

Numerous studies on mesh and point clouds have been performed, and the methods based on mesh algorithms have achieved accurate results. SI is a famous local surface feature. Johnson et al. got the normal of a feature point to form the corresponding axis and computed 2D histograms according to two parameters, finding that the SI is robust to occlusion and clutter. To achieve a high descriptiveness, Frome et al. [8] proposed a 3DSC descriptor, as an extension of the 2D shape context method. It considered a sphere superimposed on the query point and divided the radius neighbors into smaller bins, counted the points in each bin and weighted inversely with the density of the bin. Guo et al. [9] proposed a method for constructing the LRF and extracting the rotational projection statistics (RoPS) descriptor by projecting and quantizing the rotated neighbors on the 2D  $xy$ ,  $xz$ , and  $yz$  planes. The readers may refer to a recent survey [10] for additional details.

Owing to the unprecedented levels of computing science, Aldoma and Marton et al. [1] introduced the VFH. Aldoma and Tombari et al. [2] proposed a method called as CVFH. Wohlkinger et al. [24] collected ten shape functions, which depicting the point cloud, which is called the ESF. The VFH, CVFH, and ESF descriptors all use normal method. Compared with the moderate processing time of the VFH and ESF, that of the CVFH is very high.

When the defined search radius only covers the neighborhood of the keypoint, the PFH can be used as a local descriptor in [22]. The geometrical properties of each target point were estimated by analyzing the difference between the directions of the normal in the  $k$ -neighborhood of the keypoint and building a multidimensional histogram of values. One drawback of the PFH is its high computational complexity. Rusu et al. [20] also proposed an FPFH descriptor, which was a simplified and modified version of the PFH that used less neighbors for counting the orientation diversities. Compared with the PFH, the FPFH was designed to reduce the computational complexity from  $O(nk^2)$  to  $O(nk)$  while maintaining similar descriptiveness. Further, Zhong [27] developed the intrinsic shape signature (ISS) based on the radius, elevation, and azimuth angle; the support domain was divided into several units of the same size, and the multidimensional ISS feature description was formed by counting the number of neighbors in each unit. The ISS outperformed the SI and 3DSC in the model-retrieval aspects. Tombari et al. [23] presented a SHOT local 3D descriptor that is effective and efficient for surface-matching applications. It is designed to lie at the intersection between the two classes gathering the vast majority of data in the area of 3D description, i.e., signatures and histograms. SHOT outperformed state-of-the-art local descriptors in many experiments addressing descriptor matching for object recognition, 3D reconstruction, and shape retrieval.

For local feature-based 3D object recognition in the point cloud, there involve three main steps: feature matching, hypothesis generation, and hypothesis verification. Chua and Jarvis [7] employed a PS descriptor of a scene to match against each model and then used pairs of corresponding points to calculate the rigid transformation between a candidate model and a specified scene, at the cost of considerable memory and time. Johnson and Hebert [14] matched the SI descriptor of a scene with their models to generate point correspondences, which they grouped using the hypothesis generation of geometric consistency (GC). This algorithm is not only robust to clutter and occlusion but also has robustness to complicated real scenes. Yamany and Farag [25] proposed a feature descriptor for surface signatures. Mian et al. [17] used a tensor descriptor to obtain the feature correspondences and a hypothesis model. The hypothesis model was then transformed into the scene, and the hypothesis was verified using the iterative closest point (ICP) algorithm [4]. This method is superior to the SI-based algorithm with regard to the recognition rate and efficiency. Mian et al. [18] developed a 3D object-recognition algorithm based on keypoint matching, which was used to recognize the different and unknown scales of objects.

### 3 SHOT-based 3D object recognition

#### 3.1 SHOT feature

A 3D local feature should be invariant for rigid transformations that include rotation and translation. Thus, we establish a single and repeatable improved LRF to represent the local surface. A local feature of a 3D point cloud should have a very high descriptiveness to ensure the accuracy and timeliness of the object recognition. An improved normal vector estimation method is used to generate a new SHOT local-feature description.

Guo et al. [12] proposed that the singleness and repeatability of the LRF directly affect the follow-up descriptor descriptiveness and robustness. We can see that the construction of the LRF

is essential. The original SHOT feature is robust and descriptive to noise, object occlusion, and complex scenes, but the robustness to point-cloud variations is poor. Therefore, an improved LRF is proposed to solve the problem that the original SHOT feature is sensitive to changes in the point-cloud density.

The original SHOT typically uses three orthogonal eigenvectors  $V = \{v_1, v_2, v_3\}$  obtained via eigenvalue decomposition (EVD) of the scatter matrix to represent three coordinate axes for establishing the LRF and only considers the noise in order to add the distance weight. It then directly calculates the feature point  $p$  to replace the center point  $\tilde{p}$ . The scatter matrix of the original SHOT is given as follows:

$$M = \frac{1}{\sum_{i:d_i \leq R} (R - d_i)} \sum_{i=0}^k (R - d_i)(p_i - p)(p_i - p)^T \quad (1)$$

Where  $R$  represents the radius of the support fields, and  $d_i = \|p_i - p\|$  represents the Euclidean distance between the supporting neighborhood and the feature point. The original weighted point-cloud density and the improved weighted point-cloud density are showed below, where  $p_i$  is the feature point;  $p_j$  is the supporting neighborhood point;  $\tilde{p}$  is the geometric center in different cloud densities, the selection of  $\beta$  will be described in detail in Section 4.3:

$$W_{ij} = \frac{1}{(R - \|p_j - \tilde{p}_i\|)} \quad (2)$$

$$W'_{ij} = \frac{W_{ij}}{\text{density}^\beta(p_j)} = \frac{1}{(R - \|p_j - \tilde{p}_i\|) * \text{density}^\beta(p_j)} \quad (3)$$

The scatter matrix for establishing the LRF is calculated as follows:

$$COV(p_i) = \frac{\sum_{\|p_j - p_i\| \leq R} W'_{ij} \cdot (p_j - p_i)(p_j - p_i)^T}{\sum_{\|p_j - p_i\| \leq R} W'_{ij}} \quad (4)$$

On the other hand, to maintain consistency with the original SHOT feature, we improve Lin's method [16] by means of establishing the scatter matrix to calculate the center point at different cloud densities. The improved center point is designated as the geometric center, requiring that the distance between all points in the supporting area and the geometric center, which calculated as below formula (5) is the smallest, where the distance refers to the weighted Euclidean distance. Originally, it is very sensitive to calculate the center of the scatter matrix when the point-cloud density changes, however, the effect of the geometric center on this matter is very small.

$$\arg \min \sum_{\|p_j - p_i\| < R} W'_{ij} \cdot \|p_j - \tilde{p}_i\| \quad (5)$$

The algorithm is described as follows.

Algorithm: Compute an geometric center in different cloud densities  $\tilde{p}_i$

Input: the neighborhoods  $P = \{p_1, p_2, \dots, p_n\}$  of the feature point  $p_i$  within the support radius and the threshold  $\varepsilon$

Output: geometric center  $\tilde{p}_i$

1. Initialization:  $\tilde{p}_i = p_i$

2.  $p_0 = \tilde{p}_i$

$$W'_{ij} = \frac{1}{(R - \|p_j - \tilde{p}_i\|) * \text{density}^\beta(p_j)}$$

3. Update the geometric center and the weight

$$\tilde{p}_i = \frac{\sum_{j=1}^n W_{ij} * p_j}{\sum_{j=1}^n W_{ij}}$$

4. If  $\|\tilde{p}_i - p_0\| < \varepsilon$ , output  $\tilde{p}_i$ ; otherwise, go to step 2

When the EVD is performed by the scatter matrix, the eigenvector direction is both positive and negative. Sign ambiguity is generated when an LRF is established directly by using the characteristic direction of the EVD. To solve this problem, SHOT employs disambiguated EVD [5] to make all the neighbors in the support to the center point as normal as possible.

Take the  $x$  axis as an example.

$$\begin{aligned} S_x^+ &= \{i | d_i \leq R \&\& (p_i - p) \cdot x^+ \geq 0\} \\ S_x^- &= \{i | d_i \leq R \&\& (p_i - p) \cdot x^- > 0\} \end{aligned} \quad (6)$$

Considering the same number of  $S_x^+$  and  $S_x^-$ , we first select the nearest  $k$  points around the feature point  $P_i$  ( $k = 11$ ) and then count their  $\tilde{S}_x^+$  and  $\tilde{S}_x^-$ :

$$\begin{aligned} \tilde{S}_x^+ &= \{i | i \in M(k) \&\& (p_i - p) \cdot x^+ \geq 0\} \\ \tilde{S}_x^- &= \{i | i \in M(k) \&\& (p_i - p) \cdot x^- > 0\} \end{aligned} \quad (7)$$

The final positive and negative directions of the  $x$  axis are determined by the number of statistics:  $S_x^+$ ,  $S_x^-$ ,  $\tilde{S}_x^+$ ,  $\tilde{S}_x^-$ .

$$x = \begin{cases} x^+, |S_x^+| > |S_x^-| \\ x^-, |S_x^+| < |S_x^-| \\ x^+, |S_x^+| = |S_x^-| \&\& |\tilde{S}_x^+| > |\tilde{S}_x^-| \\ x^-, |S_x^+| = |S_x^-| \&\& |\tilde{S}_x^+| < |\tilde{S}_x^-| \end{cases} \quad (8)$$

The signature of SHOT is calculated as the cosine of the angle between the normal at the point  $n_q$  and the local  $Z$  axis at the feature point  $p$  (introducing bin by the radius, the horizontal angle and the elevation division, counting the established histogram by the angle between normal vector of neighborhood and  $Z$  axis). However, in this paper, the normal of the neighborhood has not been used, but in accordance with the above calculation of the LRF method to recalculate the scatter matrix of distance weight and point-cloud density weight for each point, with the minimum eigenvalue corresponding to the feature direction to replace the normal:

$$\cos \theta = n'_q \cdot Z_p \quad (9)$$

where  $n'_q$  is the minimum eigenvalue corresponding to the feature direction, which is calculated using the improved scatter matrix and the geometric center, and  $Z_p$  is the  $Z$ -axis direction of the LRF.

### 3.2 3D object recognition

The proposed 3D object-recognition algorithm follows the framework of the object-recognition scheme based on local features, including feature-point detection, feature-description extraction, feature matching, hypothesis generation, and hypothesis verification. A block diagram of the proposed algorithm is shown in Figure 1. Then we introduce four main modules.

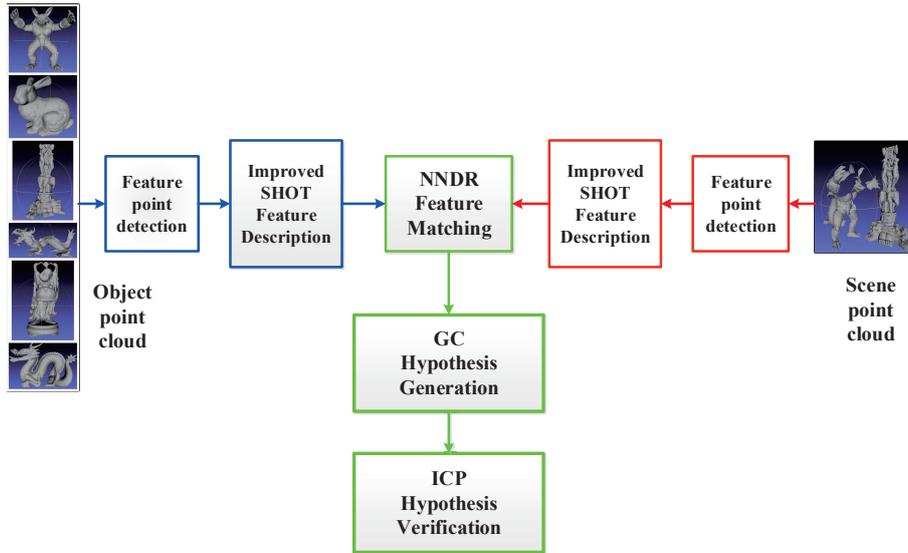


Figure 1: Block diagram of the object-recognition algorithm for 3D point-cloud scenes.

Module 1. Feature-point detection: We use the ISS method with boundary-point removal (ISS-BR) for feature-point detection in Point Cloud Library (PCL) [28] to extract the feature points in the object and scenes.

Module 2. Feature-description extraction: For each feature point, we define an LRF to generate a SHOT feature.

Module 3. Feature matching: We employ the statistical strategy of the nearest-neighbor distance ratio (NNDR); that is, when feature ratio of an object is lower than the given threshold, the two features are considered a match. The position of the object in the scenes is estimated according to the GC at the hypothesis-generation stage.

Module 4. Hypothesis verification: Because the hypothesis-generation phase can only be produced a preliminary coarse transformation, it is necessary to perform hypothesis verification which separates the correct or error hypothesis, so as to accurately calculate the rotation and translation transformation matrix from object to scene. We use the ICP hypothesis-validation method [26].

## 4 Evaluation of SHOT feature

The experimental results show that LFR in this paper has strong robustness to different point-cloud density, and our SHOT feature significantly outperforms the original SHOT features with regard to recall and precision and it also has a higher robustness to noise and point-density variation. Compared with the FPFH feature, our SHOT feature exhibits the best performance under all the levels of mesh decimation and noise.

### 4.1 Robustness to LRF

Regarding robustness, Guo et al. [9] proposed a criteria for the evaluation of the LRF. For a pair of corresponding feature points  $(k_M, k_S)$ , the established LRF is a  $3 \times 3$  matrix  $(L_M, L_S)$  that represents the directions of the three axes  $XYZ$ , and the error between the two LRFs can be expressed as

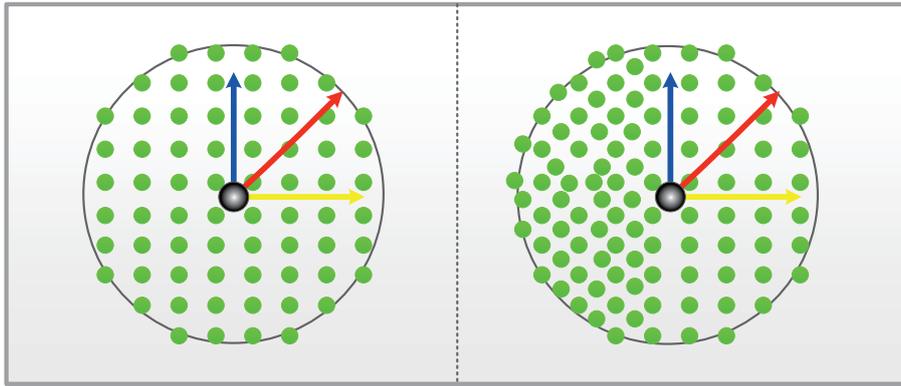


Figure 2: Established LRF for different point clouds.

$$\varepsilon = \arccos \left( \frac{\text{trace}(L_M L_S^{-1}) - 1}{2} \right) \frac{180}{\pi} \quad (10)$$

where  $\text{trace}$  indicates the trajectory of the matrix, which is the sum of the elements on the main diagonal. If the two LRFs are the same,  $L_M L_S^{-1}$  is a unit matrix,  $\text{trace}(L_M L_S^{-1})$  is 3, and the final result  $\varepsilon$  is 0. Similarly,  $\varepsilon$  can be denoted as the angle of rotation error between the two LRFs; a smaller  $\varepsilon$  yields a smaller error.  $\varepsilon$  indicates that there is no error.

## 4.2 Performance for the retrieval dataset

We use the retrieval dataset [29] to test the performance of our SHOT feature. It is well known that the retrieval dataset is a famous real-life datasets. It contains six models (Armadillo, Bunny, Thai Statue, Happy Buddha, Dragon, and Asian Dragon) and 18 scenes which taken from the Stanford 3D Scanning Repository [30]. And for creating clutter and pose variances, six models are randomly putted in a scene.

Each scene is divided into three levels of noise with standard deviations of 0.1mr, 0.3mr, and 0.5mr (where mr refers to the mesh resolution, i.e., the average distance of each point to the nearest point). Both the model and the scene are stored in the Polygon File Format (PLY) mesh model, and the transformation matrix for each model to the 3D scene is stored in the accompanying XF file. Armadillo model of the retrieval dataset and three noise scenes are shown in Figure 3. Four sample models are shown in Figure 4.

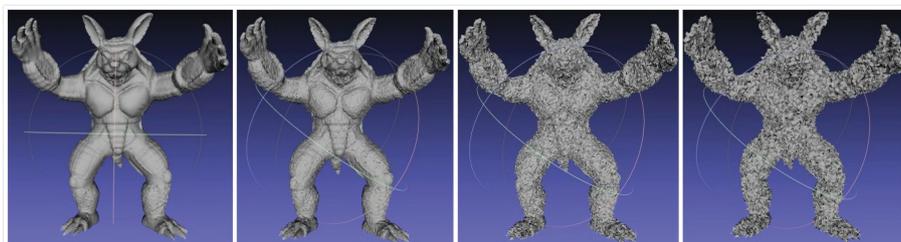


Figure 3: Armadillo model of the retrieval dataset and three noise scenes.

We get 1,000 feature points from each model to evaluate the robustness of the LRF, meanwhile, the corresponding of 1,000 feature points are also sampled to generate ground-truth correspondences. And we resample each scene to 1/2, 1/4, and 1/8 of its original mesh resolution,



Figure 4: Four sample models of the Stanford 3D Scanning Repository.

adding the three aforementioned noise scenes. Simultaneously, we select three types of LRFs - a single LRF, the LRF of the SHOT descriptor, and the LRF of the improved SHOT, which are tested with varying mesh resolutions and noise levels.

From Figure 5 we can see the average percentage of the LRF angle errors for five scenes and models at the three point-cloud densities and noise intensities. The abscissa shows the angular-error interval; e.g., 10 represents  $[0, 10)$ . For convenient statistics, we don't consider the case that the LRF angle error is greater than 90 degrees. Figure 5 shows that, for a noise with a standard deviation of 0.1mr and 0.3mr, the improved SHOT one is more stable than the original SHOT one, in particular, the original SHOT LRF has the best performance in the error interval  $[0, 10]$ , but drops to the end in the error interval  $[10, \infty)$ . On the other hand, the improved SHOT one is more strong to noise than the original SHOT one for a noise with a standard deviation of 0.5mr.

### 4.3 Descriptiveness assessment

We use the precision-recall curve (PRC), which comprises pairs of values of the recall and precision, to evaluate the descriptiveness of the 3D local-feature descriptor. The PRC is generated as follows. First, numerous keypoints are extracted from the 3D point cloud corresponding to the scene and all the models; the feature descriptors of these keypoints are then computed. Second, to determine whether the match is robust, the NNDR technique is used to perform feature matching between the two descriptors. For each feature, the match is regarded as robust if the condition  $\|N_s - N_M\|/\|N_s - N'_M\| \leq \tau$  is met, where  $N_s$  is the descriptor in the scene, and  $N_M$  and  $N'_M$  are its nearest and second-nearest neighbor in the model. The *precision* is calculated as the number of correct matches with respect to the total number of matches:

$$precision = \frac{\text{The number of correct matches}}{\text{The number of matches}} \quad (11)$$

The *Recall* is calculated as the number of correct matches with respect to the number of corresponding features between the scene and models:

$$Recall = \frac{\text{The number of correct matches}}{\text{The number of corresponding features}} \quad (12)$$

Where the value of the threshold  $\tau$  is changed from 0 to 1. To test the descriptiveness of the descriptors, the ISS-BR was used to detect the keypoints in the scene and the models.

For testing the capability of the original SHOT, our improved SHOT, and the FPFH with varying mesh resolutions, we resample each scene to 1/2, 1/4, and 1/8 of its original mesh resolution. On the other hand, for evaluating the robustness to noise, we choose noise with standard deviations of 0.1mr, 0.3mr, and 0.5mr to add to each scene. The feature-point support field is set to six times the average mesh resolution, and the RPC results at different levels of the mesh resolution and noise are shown in Figure 6.

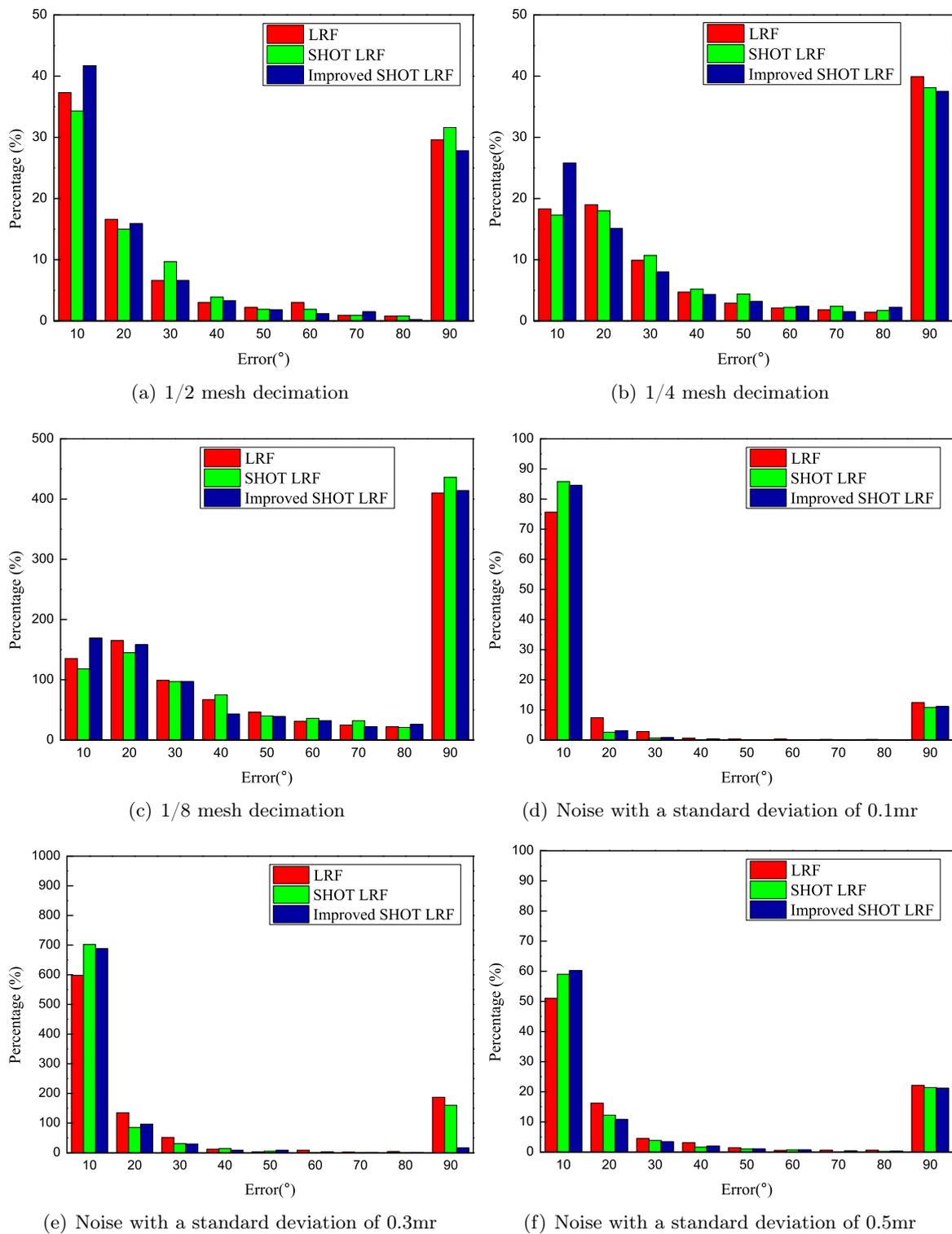


Figure 5: Comparison of different LRFs

As you can see from the graph, for varying mesh resolutions and noise levels, the improved SHOT was more descriptive and robust. Regarding the robustness to varying mesh resolutions, our SHOT feature significantly outperform the other features, and the original SHOT is the

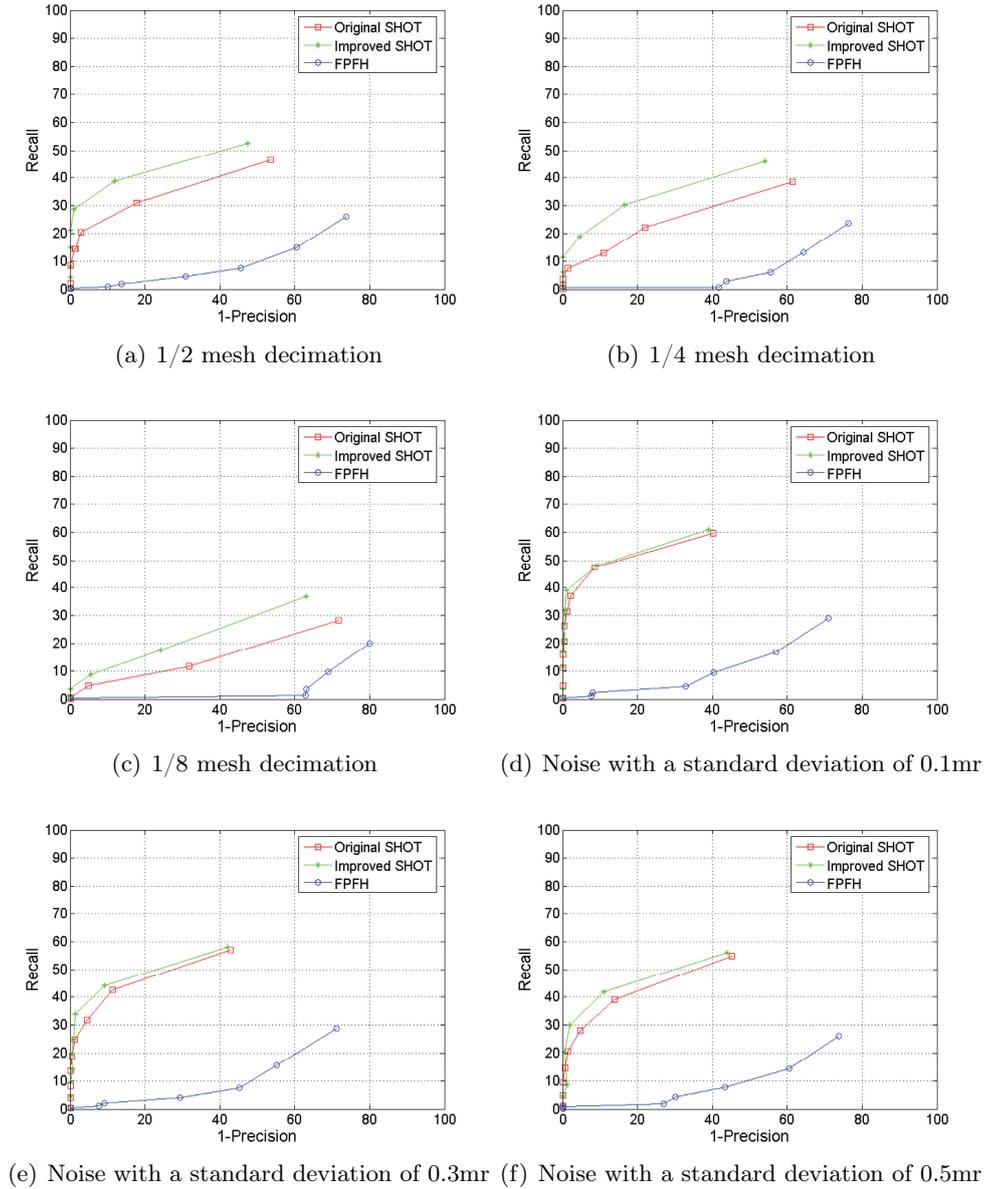


Figure 6: Recall vs. 1-precision curves with respect to the mesh resolution and noise

second. The performance of the improved SHOT feature under 1/4 and 1/8 mesh decimations is better than that of the FPFH, which is unsatisfactory. Regarding the robustness to noise, our SHOT feature exhibits the best force at all noise levels.

In Figure 7, we also show the performance of transforming the point-cloud density into a  $\beta$  square. In the case of resampling the original mesh resolution of 1/2 and noise with standard deviations of 0.1mr, the best results are obtained with  $\beta = 1$ .

## 5 Evaluation of the 3D object-recognition algorithm

To examine the performance of the 3D object-recognition algorithm with our improved SHOT feature, we perform an object-recognition experiment using the BoD1 dataset [31]. We use the improved SHOT in our object-recognition algorithm to perform experiments on this dataset using

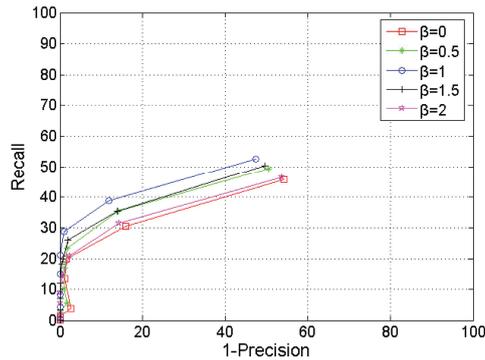


Figure 7: Performance of the SHOT features with different values of  $\beta$ .

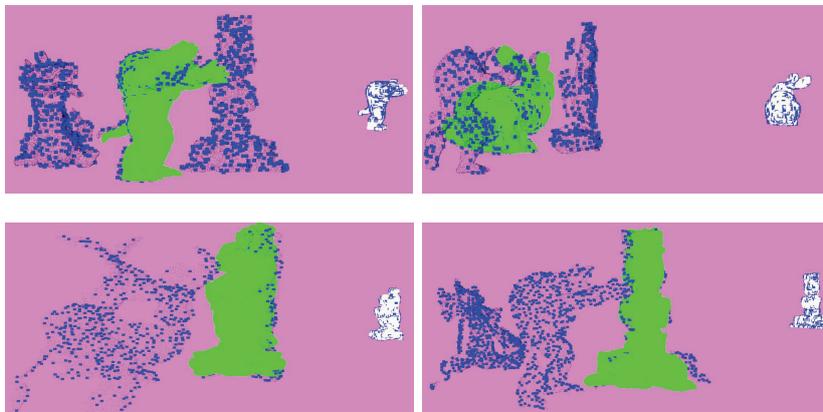


Figure 8: Correct recognition results of our algorithm

an Intel(R) Core(TM) i7-6500K 3.20-GHz Windows machine with 16 GB of RAM.

In this study, all the models from the BoD1 dataset with noise having a standard deviation of 0.1mr are resampled with the original mesh resolution of 1/2 as the experimental model library. In the recognition phase, we perform a comparative experiment with the original SHOT, our improved SHOT, and the FPFH, where we use 10 times the average mesh resolution as the radius of the calculation normal and 15 times the average mesh resolution as the radius of SHOT and FPFH descriptors. According to the descriptive experiment, an NNDR value of 0.9 was more appropriate.

Figure 8 shows the correct results of recognizing the object in the 3D cluttered point-cloud scenes via ISS-BR feature point detection with our improved SHOT feature. The black points represent the scene models of the original mesh resolution of 1/2, with added noise having a standard deviation of 0.1mr. The original object is represented by the white models on the left, and the blue points represent the feature points detected from the model and scenes. The green model indicates the locations of the objects in the scenes.

The recognition rate [13] is the ratio of the number of correct recognition to the total number of experiments:

$$\text{recognition rate} = \frac{\text{The number of correct recognition}}{\text{The total number of experiments}} \quad (13)$$

For a rigorous comparison, we perform 3D object-recognition experiments on a subset of BoD1, which contained 24 scenes of Armadillo and Bunny, 18 scenes of Dragon and Happy Buddha, and 12 scenes of Asian Dragon and Thai Statue. The recognition rates are presented

Table 1: Recognition rates for a subset of the BoD1 dataset.

Algorithm(%)	Armadillo	Bunny	Dragon	Asian Dragon	Happy,Buddha	Thai,Statue
Improved SHOT	66.7	83.3	66.7	83.3	55.6	100
SHOT	54.2	83.3	55.6	83.3	50	100
FPFH	41.7	75	44.4	75	38.9	50

in Table 1. The improved SHOT algorithm exhibits a higher recognition rate by 20.4% than the FPFH-based algorithm. The improved SHOT algorithm also outperforms the original SHOT algorithm, by 74.6% and 71.1%. The improved SHOT algorithm exhibits the highest recognition rate for all the models and achieves a recognition rate of 100% for the Thai Statue model.

The experimental results are analyzed as follows.

The original LRF of SHOT is sensitive to the point-cloud density; thus, simultaneously improving the normal and LRF to achieve the performance is more obvious. The performance of SHOT is better than that of FPFH; thus, the SHOT recognition performance is better than that of the FPFH.

Given these considerations, SHOT descriptors are the best option for time-crucial applications and for applications requiring high levels of descriptiveness and efficiency. On the other hand, for space-crucial applications, FPFH is better because of its low memory requirements for the feature storage.

Despite its benefits, the SHOT feature has disadvantages. It is designed for fixed-scale objects; once the object changes scale, the SHOT feature may face challenges. Therefore, to solve the problem of non-rigid change, future studies should improve the feature description so that it has a high degree of robustness for not only noise, varying density, and occlusion but also non-rigid changes.

## 6 Conclusion

We propose a new SHOT feature for local description. At each feature point, an LRF is established by calculating a scatter matrix according to the geometric center and the weighted point-cloud density of its neighborhood. An improved normal vector estimation method is used to generate a new SHOT local-feature descriptor. The GC and the ICP method are used to perform 3D model recognition for the point-cloud scenes. The experimental results show that the proposed SHOT feature-extraction algorithm is robust to noise and point-cloud variations. Moreover, it is highly efficient and effective for the 3D object recognition in cluttered point-cloud scenes. In recent years, Deep learning automatically learns a feature representation of 3D data. We will combine effective handcrafted 3D features with deeply learned features to further improve the performance in the future.

## Acknowledgments

This research is supported by the Specialized Research Funds for the Doctoral Program of Higher Education, China (No.20130061110054) and the National Natural Science, China (NSF61373003).

## Bibliography

- [1] Aldoma, A.; Marton, Z.; Tombari, F.; Wohlkinger, W. (2012); Tutorial: Point Cloud Library: Three-Dimensional Object Recognition and 6 DOF Pose Estimation, *Robotics and Automation Magazine IEEE*, 19, 80-91, 2012.
- [2] Aldoma, A.; Tombari, F.; Rusu, R.; Vincze, M. (2012); OUR-CVFH - Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation, *LNCS*, 7476, 113-122, 2012.
- [3] Arbeiter, G.; Fuchs, S.; Bormann, R.; Fischer, J.; Verl, A. (2012); Evaluation of 3D feature descriptors for classification of surface geometries in point clouds, *Proceedings of IROS*, 1644-1650, 2012.
- [4] Besl, P.; McKay, N. (1992); A method for registration of 3-D shapes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14, 239-256, 1992.
- [5] Bro, R.; Acar, E.; Kolda, T. (2008); Resolving the sign ambiguity in the singular value decomposition, *Journal of Chemometrics*, 22, 135-140, 2008.
- [6] Ceron, A.; Prieto, F. (2013); Evaluating and Comparing of 3D Shape Descriptors for Object Recognition, *LNCS*, 8034, 484-492, 2013.
- [7] Chua, C.; Jarvis, R. (1997); Point signatures: a new representation for 3D object recognition, *International Journal of Computer Vision*, 25, 63-85, 1997.
- [8] Frome, A.; Huber, D.; Kolluri, R.; Blow, T.; Malik, J. (2004); Recognizing Objects in Range Data Using Regional Point Descriptors, *Lecture Notes in Computer Science*, 3023, 224-237, 2004.
- [9] Guo, Y.; Sohel, F.; Bennamoun, M.; Lu, M.; Wan, J. (2013); Rotational projection statistics for 3d local surface description and object recognition, *International Journal of Computer Vision*, 105, 63-86, 2013.
- [10] Guo, Y.; Bennamoun, M.; Sohel, F.; Lu, M.; Wan, J. (2014); 3D Object Recognition in Cluttered Scenes with Local Surface Features: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36, 2270-2287, 2014.
- [11] Guo, Y.; Bennamoun, M.; Sohel, F.; Lu, M.; Wan, J. (2014); Performance Evaluation of 3D Local Feature Descriptors, *Proceedings of ACCV*, 178-194, 2014.
- [12] Guo, Y.; Sohel, F.; Bennamoun, M.; Lu, M.; Wan, J. (2013); TriSI: A Distinctive Local Surface Descriptor for 3D Modeling and Object Recognition, *Proceedings of GRAPP*, 86-93, 2013.
- [13] Guo, Y.; Sohel, F.; Bennamoun, M.; Wan, J.; Lu, M.; A Novel Local Surface Feature for 3D Object Recognition under Clutter and Occlusion, *Information Sciences*, 293, 196-213, 2014.
- [14] Johnson, A.; Hebert, M. (1999); Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21, 433-449, 1999.
- [15] Liao, S.; Chung, A.C.S. (2010); A novel multi-layer framework for non-rigid image registration, *Proceedings of ISBI*, 344-347, 2010.

- 
- [16] Lin, C.; Chen, J.; Su, P.; Chen, C. (2014); Eigen-feature analysis of weighted covariance matrices for LiDAR point cloud classification, *ISPRS Journal of Photogrammetry and Remote Sensing*, 94, 70-79, 2014.
- [17] Mian, A.; Bennamoun, M.; Owens, R. (2006); Three-dimensional model-based object recognition and segmentation in cluttered scenes, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28, 1584-1601, 2006.
- [18] Mian, A.; Bennamoun, M.; Owens, R. (2010); On the Repeatability and Quality of Key-points for Local Feature-based 3D Object Retrieval from Cluttered Scenes, *International Journal of Computer Vision*, 89, 348-361, 2010.
- [19] Rusu, R.; Blodow, N.; Marton, Z.; Beetz, M. (2008); Aligning point cloud views using persistent feature histograms, *Proceedings of IROS*, 3384-3391, 2008.
- [20] Rusu, R.; Blodow, N.; Beetz, M. (2009); Fast Point Feature Histograms (FPFH) for 3D registration, *Proceedings of ICRA*, 3212-3217, 2009.
- [21] Rusu, R.; Holzbach, A.; Beetz, M.; Bradski, G. (2009); Detecting and segmenting objects for mobile manipulation, *Proceedings of ICCVW*, 47-54, 2009.
- [22] Rusu, R. (2010); Semantic 3D Object Maps for Everyday Manipulation in Human Living Environments, *KI - Künstliche Intelligenz*, 24, 345-348, 2010.
- [23] Tombari, F.; Salti, S.; Stefano, L. (2010); Unique signatures of histograms for local surface description, *Proceedings of ECCV*, 356-369, 2010.
- [24] Wohlkinger, W.; Vincze, M. (2011); Ensemble of shape functions for 3D object classification, *Proceedings of IEEE RO-BIO*, 2987-2992, 2011.
- [25] Yamany, SM.; Farag, AA. (2002); Surface signatures: an orientation independent free-form surface representation scheme for the purpose of objects registration and matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, 1105-1120, 2002.
- [26] Zhang, Z. (1994); Iterative point matching for registration of free-form curves and surfaces, *International Journal of Computer Vision*, 13, 119-152, 1994.
- [27] Zhong, Y. (2009); Intrinsic shape signatures: A shape descriptor for 3D object recognition, *Proceedings of ICCVW*, 689-696, 2009.
- [28] PCL; [pointclouds.org/](http://pointclouds.org/).
- [29] 3D Keypoint Detection Benchmark; [www.vision.deis.unibo.it/keypoints3d/](http://www.vision.deis.unibo.it/keypoints3d/).
- [30] The Stanford 3D Scanning Repository; <http://www.vision.deis.unibo.it/keypoints3d/>.
- [31] SHOT: Unique Signatures of Histograms for Local Surface Description; [www.vision.deis.unibo.it/research/80-shot](http://www.vision.deis.unibo.it/research/80-shot).