Int J Comput Vis (2016) 116:66–89
DOI 10.1007/s11263-015-0824-y

A Comprehensive Performance Evaluation of 3D Local Feature Descriptors

Yulan Guo[1,2] · **Mohammed Bennamoun**[2] · **Ferdous Sohel**[2] · **Min Lu**[1] · **Jianwei Wan**[1] · **Ngai Ming Kwok**[3]

Received: 28 March 2014 / Accepted: 3 April 2015 / Published online: 16 April 2015
© Springer Science+Business Media New York 2015

Abstract A number of 3D local feature descriptors have been proposed in the literature. It is however, unclear which descriptors are more appropriate for a particular application. A good descriptor should be descriptive, compact, and robust to a set of nuisances. This paper compares ten popular local feature descriptors in the contexts of 3D object recognition, 3D shape retrieval, and 3D modeling. We first evaluate the descriptiveness of these descriptors on eight popular datasets which were acquired using different techniques. We then analyze their compactness using the recall of feature matching per each float value in the descriptor. We also test the robustness of the selected descriptors with respect to support radius variations, Gaussian noise, shot noise, varying mesh resolution, distance to the mesh boundary, keypoint localization error, occlusion, clutter, and dataset size. Moreover, we present the performance results of these descriptors when combined with different 3D keypoint detection methods. We finally analyze the computational efficiency for generating each descriptor.

干扰

1. 针对descrption的特征分三个部分进行测试;
2. 针对不同的应用，与其结合的测试;
3. 一些复杂度的理论计算

Communicated by M. Hebert.

✉ Yulan Guo
yulan.guo@nudt.edu.cn

[1] College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, Hunan, People's Republic of China

[2] School of Computer Science and Software Engineering, The University of Western Australia, Perth, WA 6009, Australia

[3] School of Mechanical and Manufacturing Engineering, The University of New South Wales, Sydney, NSW 2052, Australia

Keywords Performance evaluation · Local feature descriptor · Keypoint detector · 3D object recognition · 3D shape retrieval · 3D modeling · 3D surface

1 Introduction

Local features have proven to be very successful in many vision tasks such as 3D object categorization and recognition (Matei et al. 2006; Mian et al. 2006a; Shang and Greenspan 2010; Lai et al. 2011; Guo et al. 2013b), 3D modeling and scene reconstruction (Mian et al. 2006b; Guo et al. 2014c), 3D model retrieval and shape analysis (Bronstein et al. 2011; Gao and Dai 2014), and 3D biometrics (Lei et al. 2014; Bennamoun et al. 2015). Local features have been extensively investigated over the last few decades with the aim of designing descriptors which are distinctive and robust to occlusions and clutter (Mian et al. 2006a). A local feature based algorithm typically involves two major phases: keypoint detection and feature description (Tombari et al. 2013). In the keypoint detection phase, keypoints with rich information content are first identified and their associated scales (spatial extents) are determined (Mian et al. 2010; Tombari et al. 2013). In the feature description phase, the local geometric information around a keypoint is extracted and stored in a high-dimensional vector (i.e., feature descriptor) (Guo et al. 2013b). Finally, the feature descriptors of one surface are matched against the feature descriptors of other surfaces of interest to yield point-to-point feature correspondences (Tombari et al. 2010b, 2013; Guo et al. 2014a).

1. 关键点检测;
2. 关键点的特征描述

A large variety of 3D keypoint detectors and local feature descriptors have been proposed in the literature (Bronstein et al. 2010; Tombari et al. 2013). It is widely agreed that the evaluation of 3D keypoint detectors and local feature descriptors is very important (Tombari et al. 2013). Sev-

既然是evaluation的算法，就需要给出evaluation的评价指标，并给出这两种评价指标的的"含义"

eral 3D keypoint detector evaluations can be found in the literature, e.g., Bronstein et al. (2010), Salti et al. (2011), Boyer et al. (2011), Tombari et al. (2013). Descriptiveness and robustness are considered to be two of the most important attributes for a 3D local feature descriptor (see more in Sect. 4.3) (Restrepo and Mundy 2012). A feature descriptor is **descriptive** if it is capable of encapsulating the predominant 占主要成分的 information of the underlying surface. That is, it should provide sufficient descriptive richness to distinguish one local surface from another. A feature is **robust** if it is insensitive to a number of disturbances which can affect the data, e.g., noise and variations in the mesh resolution (Tombari et al. 2013).

足够的描述性

Although a large number of feature descriptors have been proposed, they were exclusively designed for a specific application scenario (e.g., object recognition, and shape retrieval) and they have only been tested on a limited number of datasets (collected specifically for that particular application). It is therefore, very challenging for developers to choose an appropriate descriptor for their particular application. When compared to the performance evaluations of 2D keypoint detectors (Schmid et al. 2000; Mikolajczyk and Schmid 2004; Mikolajczyk et al. 2005; Moreels and Perona 2007), 2D local feature descriptors (Mikolajczyk and Schmid 2005; Moreels and Perona 2005, 2007; Burghouts and Geusebroek 2009), and 3D keypoint detectors (Bronstein et al. 2010; Boyer et al. 2011; Salti et al. 2011, 2012; Tombari et al. 2013; Filipe and Alexandre 2014), only a very limited number of performance evaluations were conducted on 3D local feature descriptors (Guo et al. 2014b). Most of these evaluation articles tested only a small number of 3D local feature descriptors and for a specific application domain. Moreover, their datasets were limited in size and did not cover a sufficient variety of 3D objects and scanners.

registration? ——找一找有没有这样的算法啊?
我需要对一些相关的survey进行一些比较

表述自己要做什么

In this paper, we present a comprehensive comparison and analysis of the state-of-the-art 3D local feature descriptors by extensively testing their performance on eight popular datasets. Our comparison is grounded on an established methodology that was previously adopted in the evaluation of 2D local feature descriptors in Mikolajczyk and Schmid (2005). Our datasets contain a large variety of scene types acquired with different imaging techniques (e.g., Minolta Vivid, Stereo, Space-time, and Microsoft Kinect). The performances of these descriptors on these different datasets are analyzed and discussed. We also evaluate these descriptors in three different application contexts (namely, 3D object recognition, 3D shape retrieval, and 3D modeling), with a particular focus on 3D object recognition under occlusion and clutter. Moreover, we test the robustness of these descriptors with respect to a set of disturbances including support radius, Gaussian noise, shot noise, varying mesh resolutions, distance to the mesh boundary, keypoint localization error, occlusion and clutter. In addition, the combined performance of these feature descriptors with different keypoint detectors is also presented and analyzed. Finally, the computational complexity of these descriptors is also compared and discussed.

如果不出意外，我们的可能是用在SLAM的数据集上面

各种结合的算法，放到我的论文里面应该说就是optimization方法

2 Related Work

This section presents a brief overview of existing work on the performance evaluation of 3D local feature descriptors.

Bronstein et al. (2010) tested Heat Kernel Signatures (HKS) (Sun et al. 2009) and spin image (SI) (Johnson and Hebert 1999) in the context of 3D shape retrieval. Their results show that HKS performs better than SI. Boyer et al. (2011) evaluated four local feature descriptors including Mesh-HoG (Zaharescu et al. 2009), Scale-Invariant Spin Image (SISI) (Darom and Keller 2012), local depth SIFT (Darom and Keller 2012), and generalized HKS (GHKS) in the context of 3D shape retrieval. Alexandre (2012) evaluated both local and global feature descriptors on a clutter-free dataset for 3D object and category recognition. They concluded that the features which combined color and shape information achieved the best performance compared to the features which only used shape information. Kim and Hilton (2013) proposed a framework for 2D/3D multi-modal data registration. They also evaluated four 3D local feature descriptors (i.e., SI (Johnson and Hebert 1999), 3D shape context (3DSC) (Frome et al. 2004), fast point feature histogram (FPFH) (Rusu et al. 2009)), and signature of histogram of orientations (SHOT) (Tombari et al. 2010b; Salti et al. 2014) for the registration of 3D data from different sources. For indoor scenes, FPFH works slightly better compared to the others. For outdoor scenes, SHOT and FPFH achieved the best performance. Restrepo and Mundy (2012) presented a performance evaluation of four 3D local feature descriptors (i.e., SI, 3DSC, SHOT, and FPFH) on probabilistic volumetric models of large-scale urban scenes that were acquired from multi-view aerial imagery. The descriptors were evaluated in terms of accuracy for object classification using the Bag-of-Words technique. Experimental results revealed that FPFH produced a high recall while being compact and fast to compute. Salti et al. (2012) investigated the effectiveness of the combinations between seven 3D local feature descriptors and six keypoint detectors. Experimental results showed that intrinsic shape signatures (ISS) is the most effective 3D keypoint detector, while SHOT is amongst the best 3D local feature descriptors.

做了什么，结果怎么样

However, none of the listed papers above exhaustively tested the performance of 3D local feature descriptors. *First*, a very limited number of descriptors were evaluated in each paper and their coverage of existing descriptors was not sufficient. *Second*, these descriptors were mainly tested under a particular application scenario while their performance in other application contexts remains unclear. *Third*, the

覆盖范围

robustness of existing 3D local feature descriptors was not fully analyzed. In this work, we address these limitations by testing and comparing the state-of-the-art 3D local feature descriptors on a set of datasets, which cover the scenarios of 3D object recognition, 3D shape retrieval, and 3D modeling. Note that, although Salti et al. (2012) compared the performance of different local feature descriptors in three different application contexts, our work tested six additional local feature descriptors including Local Surface Patch (LSP) (Chen and Bhanu 2007a), THRIFT (Flint et al. 2008), Point Feature Histograms (PFH) (Rusu et al. 2009), Fast Point Feature Histograms (FPFH) (Rusu et al. 2009), Tri-Spin-Images (TriSI) (Guo et al. 2013c), and Rotational Projection Statistics (RoPS) (Guo et al. 2013b). Moreover, the robustness of these feature descriptors is not covered by Salti et al. (2012) and is fully investigated in this paper.

3 3D Local Feature Descriptors

A number of 3D local feature descriptors have been constructed to encode the information of a local surface. Among these approaches, many algorithms use histograms to represent different characteristics of the local surface. Specifically, they describe the local surface by accumulating geometric or topological measurements (e.g., point numbers) into histograms according to a specific domain (e.g., point coordinates, geometric attributes). We categorize these algorithms into "spatial distribution histogram" and "geometric attribute histogram" based descriptors.

3.1 Spatial Distribution Histogram based Descriptors

These descriptors represent the local surface by generating histograms according to the spatial distributions (e.g., coordinates) of the points on the surface. They usually start with the construction of a Local Reference Frame/Axis (LRF/A) for the keypoint, and partition the 3D support region into several bins according to the LRF/A. They then generate a histogram for the local surface by accumulating the spatial distribution measurements (e.g., number of points) in each spatial bin.

Spin Image (SI) (Johnson and Hebert 1998, 1999) The surface normal $\boldsymbol{n}$ at the keypoint $\boldsymbol{p}$ is used as the LRA at the keypoint, and each point $\boldsymbol{q}_i$ in the support region is then represented with two parameters α and β. Here, α and β are the in-plane and out-plane distances of the point $\boldsymbol{q}_i$ to the keypoint, respectively. The $\alpha-\beta$ space is then discretized into a 2D array accumulator. Finally, the SI descriptor is generated by accumulating the points in the support region into each bin of the 2D array, as illustrated in Fig. 1a. The dimension of the SI descriptor is d_{si}^2, where d_{si} is the number of bins along each dimension of the $\alpha - \beta$ space. The SI descriptor has been successfully used in many applications with a number of variants (Ruiz-Correa et al. 2001; Dinh and Kropac 2006; Assfalg et al. 2007; Darom and Keller 2012).

3D Shape Context (3DSC) (Frome et al. 2004) 3DSC uses the surface normal $\boldsymbol{n}$ at the keypoint $\boldsymbol{p}$ as its LRA. First, a 3D spherical grid is placed at $\boldsymbol{p}$, with the north pole of the grid being aligned with the surface normal $\boldsymbol{n}$. Next, the support region is divided into several bins along the radial, azimuth and elevation dimensions. The divisions are logarithmically spaced along the radial dimension and linearly along the other two dimensions, as shown in Fig. 1b. The 3DSC descriptor is generated by counting the weighted number of points falling into each bin of the 3D grid. The dimension of 3DSC is $d_{sc\_r} \times d_{sc\_a} \times d_{sc\_e}$, where d_{sc_r}, d_{sc_a} and d_{sc_e} are respectively the numbers of bins along the radial, azimuth and elevation axes. One major limitation of 3DSC is that d_{sc_a} descriptors have to be calculated at each model keypoint due to its ambiguity along the azimuth dimension (Frome et al. 2004; Tombari et al. 2010b).

Unique Shape Context (USC) (Tombari et al. 2010a) USC is an extension of 3DSC which avoids computing multiple descriptors at a given keypoint. First, an LRF is constructed for each keypoint using the technique presented in Tombari et al. (2010b). Next, the local surface is aligned with the LRF in order to provide invariance to rigid transformations (i.e., rotations and translations). The support region of the keypoint is then divided into several bins, as shown in Fig. 1c. Finally, a USC descriptor is generated by accumulating the weighted sum of the points falling into each bin, which is analogous to the approach used in 3DSC. USC improves 3DSC in terms of memory footprint and efficiency (Tombari et al. 2010a). The dimension of USC is the same as 3DSC.

Rotational Projection Statistics (RoPS) (Guo et al. 2013a,b) RoPS is based on a novel, unique and repeatable LRF. First, an LRF is constructed for each keypoint and the local surface is aligned with the LRF to achieve invariance to rigid transformations. The points on the local surface are then respectively rotated around the *three* coordinate axes (i.e., x, y and z). For each rotation, the points in the support region are further projected onto the *three* coordinate planes (i.e., xy, yz and xz). A distribution matrix is generated for each plane by dividing the plane into several bins and counting the number of points falling into each bin. The distribution matrix is subsequently encoded with *five* statistics. Finally, the RoPS descriptor is generated by concatenating all these statistics of all rotations and projections, as shown in Fig. 1e. The dimension of the RoPS descriptor is $3 \times 3 \times 5 \times d_{rops\_r}$, where d_{rops_r} is the number of rotations around each axis.

Tri-Spin-Image (TriSI) (Guo et al. 2013c, 2015) It uses a similar technique as in Guo et al. (2013b) to construct its LRF. Once the LRF is defined for a keypoint, the local surface is aligned with the LRF. Next, a spin image is generated using the x axis as its LRA and the SI descriptor

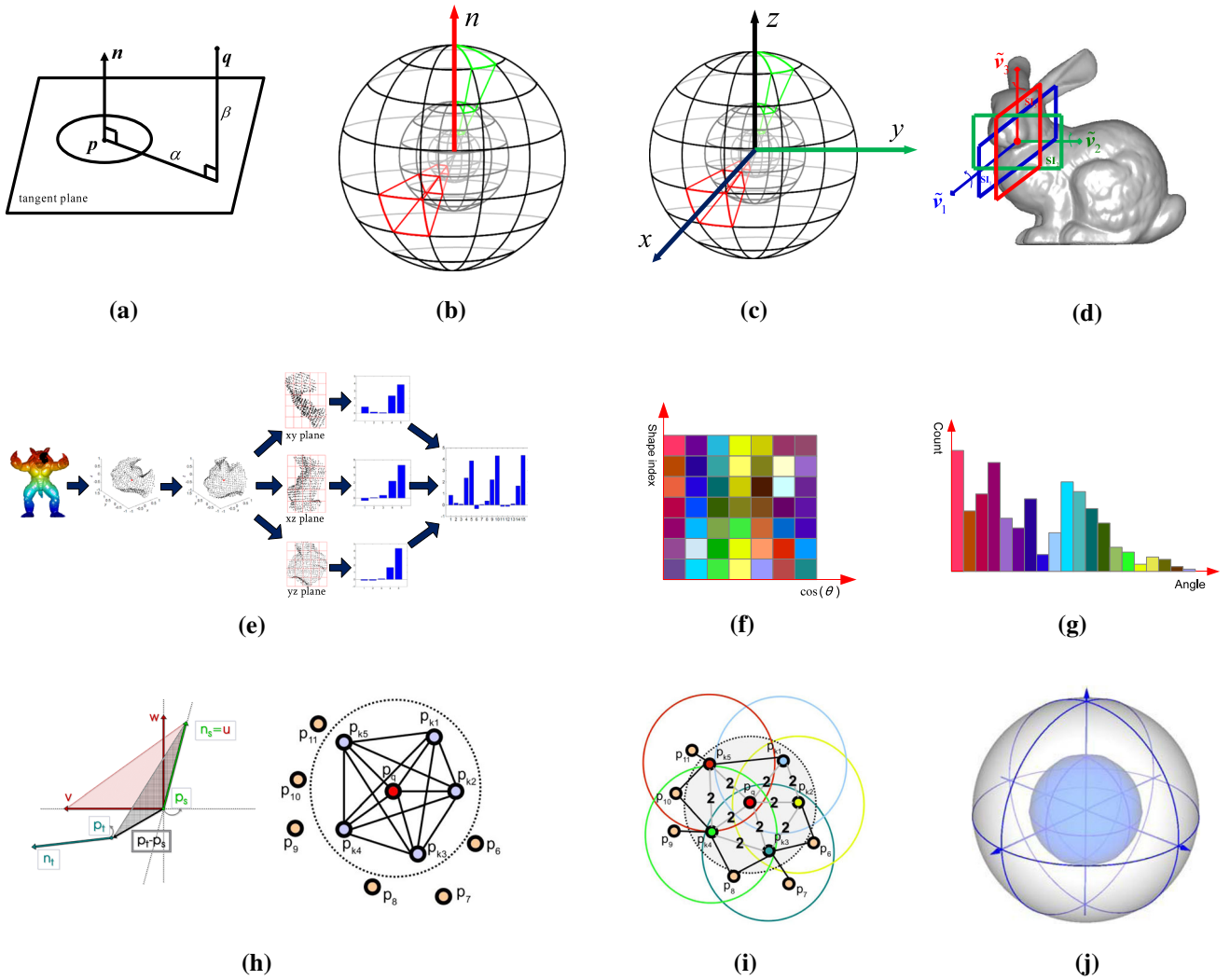

Fig. 1 A schematic illustration of the selected descriptors. **a** SI **b** 3DSC **c** USC **d** TriSI **e** RoPS **f** LSP **g** THRIFT **h** PFH **i** FPFH **j** SHOT (Color figure online)

(Johnson and Hebert 1999) procedure is then adopted. In addition, another two spin images are generated using the y and z-axes as the LRAs of these spin images, as shown in Fig. 1d. The three spin images are then concatenated to form the TriSI descriptor. TriSI significantly improves the descriptiveness and robustness compared to SI (Guo et al. 2013c). The dimension of the TriSI descriptor is $3d_{trisi}^2$, where d_{trisi} is the number of bins along each dimension.

Other Methods The 3D Tensor descriptor (Mian et al. 2006a) uses a pair of points and their surface normals to construct its LRF. The local surface is aligned with the LRF and the support region is divided into several cubic bins to define a 3D grid. The area of intersection of the local surface with each bin of the grid is recorded in the 3D Tensor. That is, each element of the 3D Tensor is equal to the total surface area of the local surface intersecting the grid bin that corresponds to that tensor element. Other methods in this category include spin image signature (Assfalg et al. 2007), multi-resolution spin image (Dinh and Kropac 2006), and asymmetry patterns shape context (Sukno et al. 2013).

3.2 Geometric Attribute Histogram based Descriptors

These descriptors represent the local surface by generating histograms according to the geometric attributes (e.g., normals, curvatures) of the points on the surface.

Local Surface Patch (LSP) (Chen and Bhanu 2007a,b): The shape index (Koenderink and Doorn 1992) of each point $\boldsymbol{q}_i$ in the support region of the keypoint $\boldsymbol{p}$, and the cosine of the angle between the surface normal of $\boldsymbol{q}_i$ and the normal at the keypoint $\boldsymbol{p}$ are calculated. The LSP descriptor is a 2D histogram, which is formed by accumulating points in particular bins along the two dimensions (i.e., the shape index value, and the cosine of the angle between the surface normals), as shown in Fig. 1f. The dimension of LSP is $d_{lsp\_s} \times d_{lsp\_a}$, where d_{lsp_s} and d_{lsp_a} are respectively the numbers of bins

along the dimensions of the shape index and the surface normals.

THRIFT (Flint et al. 2007, 2008) It is a 1D histogram of the deviation angles between the surface normal at the keypoint $\boldsymbol{p}$ and the surface normals at the neighboring points $\{\boldsymbol{q}_i\}$, as shown in Fig. 1g. The contribution of each neighboring point $\boldsymbol{q}_i$ to a particular bin of the histogram is determined by two factors: the density of point samples and the distance from the neighboring point to the keypoint. The dimension of THRIFT is d_{thrift}, where d_{thrift} is the number of bins of the histogram.

Point Feature Histogram (PFH) (Rusu et al. 2008) PFH is a multi-dimensional histogram over several features of point pairs in the support region. For each pair of the points in the support region, a Darboux frame is first defined using the surface normals and point positions (as shown in Fig. 1h). Next, four features are calculated for each point pair using the Darboux frame, the surface normals, and their point positions. PFH is generated by accumulating points in particular bins along the four dimensions. The dimension of PFH is d_{pfh}^4, where d_{pfh} is the number of bins along each dimension. In a later work (Rusu et al. 2009), one feature (i.e., the distance between any two points) is excluded from the histogram of PFH in order to improve its robustness with respect to variations in the point densities. Consequently, the dimension of the modified PFH becomes d_{pfh}^3.

Fast Point Feature Histogram (FPFH) (Rusu et al. 2009) The generation of a FPFH descriptor consists of two steps. In the first step, a simplified point feature histogram (SPFH) is generated for each point by calculating the relationships between the point and its neighbors (as shown in Fig. 1i). This is different from PFH, where the relationships between all pairs of points in the support region are calculated. In SPFH, the descriptor is generated by concatenating three separate histograms along three feature dimensions. That is, one histogram is generated along each dimension. This is also different from PFH, where a joint histogram is generated along three different dimensions. In the second step, FPFH is constructed as the weighted sum of the SPFH of the feature point and the SPFHs of the points in the support region. The dimension of FPFH is $3d_{fpfh}$, where d_{fpfh} is the number of bins along each dimension.

Signature of Histogram of Orientations (SHOT) (Tombari et al. 2010b; Salti et al. 2014) The descriptor encodes the histograms of the surface normals in different spatial locations. First, an LRF is constructed for the keypoint $\boldsymbol{p}$, and its neighboring points in the support region are aligned with the LRF. Next, the support region is divided into several volumes along the radial, azimuth and elevation axes, as shown in Fig. 1j. For each volume, a local histogram is generated by accumulating point counts into bins according to the angles between the normals at the neighboring points within the volume and the normal at the keypoint. Finally, the SHOT descriptor is generated by concatenating all the local histograms. The dimension of the SHOT descriptor is $d_{shot\_r} \times d_{shot\_a} \times d_{shot\_e} \times d_{shot\_h}$, where d_{shot_r}, d_{shot_a} and d_{shot_e} are respectively the number of divisions along the radial, azimuth and elevation dimensions, and d_{shot_h} is the number of bins in each local histogram.

Other Methods: Taati and Greenspan (2011) proposed a set of variable-dimensional local shape descriptors (VD-LSD). An eigenvalue decomposition is first performed on the covariance matrix of each point to obtain several invariant properties for that point. The VD-LSD is then generated by accumulating the neighboring points into histogram bins according to their invariant properties. Lo and Siebert (2009) developed a 2.5D SIFT algorithm by extending the classic SIFT method from 2D grey-scale images to depth images. Bayramoglu and Alatan (2010) introduced the SI-SIFT algorithm using SIFT to extract descriptors from the shape index values rather than the depth values from the depth image.

Evaluation, 这里主要是列举索要研究的对象。

自己可以在此基础上先进行列举，然后看一下自己想要做什么，然后再在此基础上进行添加。

4 Experimental Setup

After a brief review of the major 3D local feature descriptors and their characteristics, we now proceed to carry out a comprehensive comparison. In this section, we first describe the datasets and the evaluation criteria used in our tests. We also present the implementation details for the evaluated descriptors.

4.1 Datasets

We evaluate the descriptors on eight publicly available datasets (Guo et al. 2014d). Figure 2 shows some examples of models and scenes taken from these datasets. The details of these datasets are listed in Table 1. We first test the 3D local feature descriptors on the same datasets (i.e., the first five datasets in Table 1) following the evaluation of 3D keypoints in Tombari et al. (2013). Therefore, our work is able to provide the possibility to select an appropriate combination of 3D keypoint detectors and feature descriptors for a particular application based on their respective performance on a benchmark dataset. We then test the descriptors on three additional datasets to further support our findings. These datasets are selected based on the following considerations.

这里有点像训练集和测试集的意思，

用一部分的dataset进行评估，用另一部分的dataset进行performance的验证

(i) Diverse Acquisition Techniques: The *Retrieval* and *Random views* datasets were synthetically built using models taken from the Stanford 3D scanning repository (Curless and Levoy 1996). The *Laser scanner* and *2.5D Views* datasets (Mian et al. 2006a) were acquired with a laser scanner (i.e., Minolta Vivid 910). The *LIDAR* dataset (Taati and Greenspan 2011) was also captured by a Minolta Vivid scanner, but with a relatively lower image quality com-

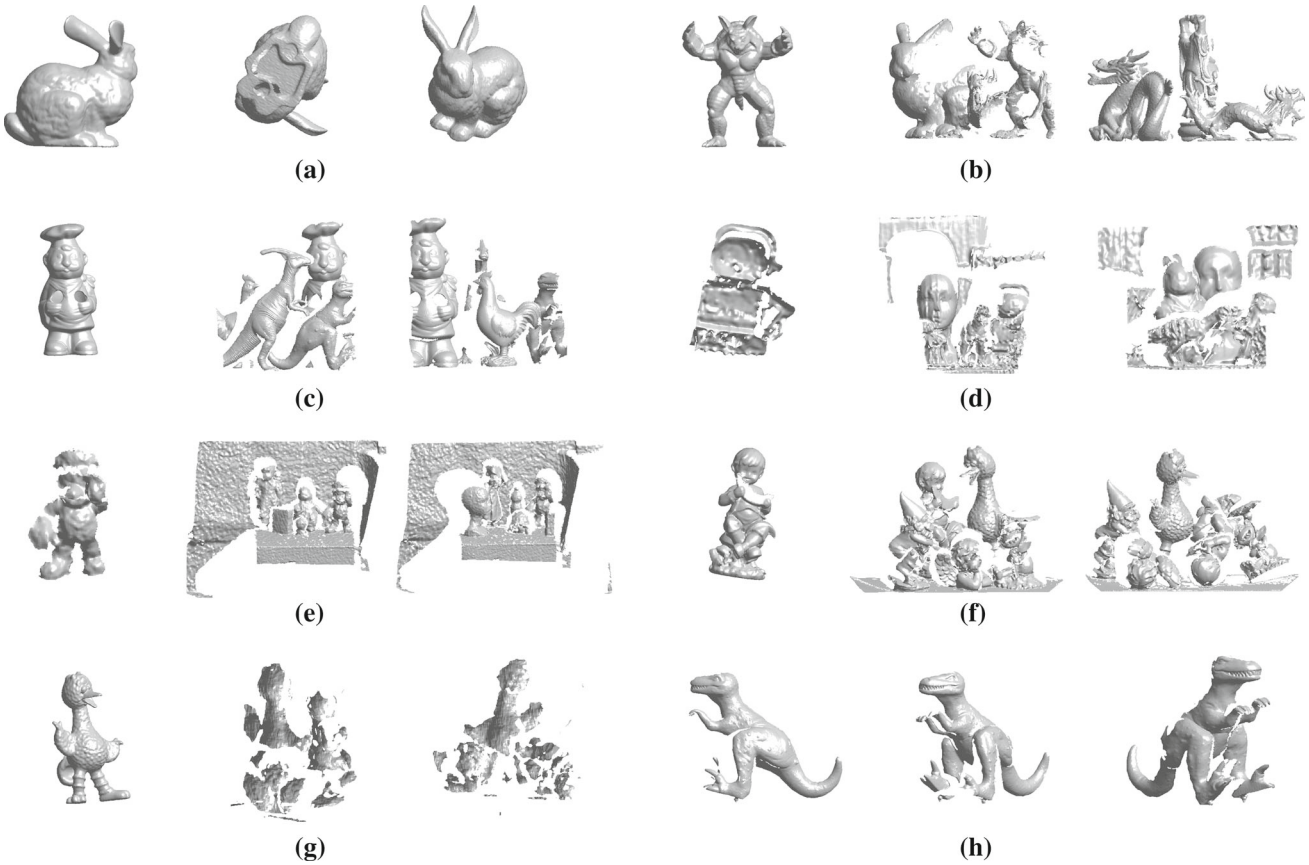

Fig. 2 Examples of models and scenes from the datasets. One model and two scenes are shown for each dataset. **a** Retrieval. **b** Random views. **c** Laser scanner. **d** Space time. **e** Kinect. **f** LIDAR. **g** Dense stereo. **h** 2.5D views

Table 1 Datasets used in the evaluation

No.	Dataset name	Acquisition	Quality	Occlusion	Clutter	Model	Scene	#Models	#Scenes	Scenario
1	Retrieval	Synthetic	High	N	N	3D	3D	6	18	Retrieval
2	Random views	Synthetic	High	Y	Y	3D	2.5D	6	36	Recognition
3	Laser scanner	Minolta vivid	High	Y	Y	3D	2.5D	5	10	Recognition
4	Space time	SpaceTime stereo	Medium	Y	Y	2.5D	2.5D	6	12	Recognition
5	Kinect	Microsoft kinect	Low	Y	Y	2.5D	2.5D	27	17	Recognition
6	LIDAR	Minolta vivid	Low	Y	Y	3D	2.5D	5	10	Recognition
7	Dense stereo	Bumblebee	Very Low	Y	Y	3D	2.5D	3	12	Recognition
8	2.5D views	Minolta vivid	High	Y	N	-	2.5D	-	75	Modeling

pared to *Laser scanner* and *2.5D Views*. The *Dense stereo* dataset (Taati and Greenspan 2011) was collected with a low-resolution Bumblebee stereo camera. The *Space time* dataset was obtained using the SpaceTime stereo acquisition technique (Tombari et al. 2013). The *Kinect* dataset (Tombari et al. 2013) was generated using a low-cost commercial scanner (Microsoft Kinect v1).

(ii) Different Application Scenarios: The focus of this paper is on the selection of the 3D feature descriptors that achieve the best performance in the context of 3D object recognition. We also, however, analyze the performance of the descriptors for 3D shape retrieval and 3D modeling. The aim of 3D object recognition is to correctly identify objects that are present in a scene and recover their poses (Guo et al. 2013b). Each scene is a 2.5D surface mesh, which consists of several objects, while each model can be a 2.5D mesh or a full 3D mesh of an isolated object. Objects in a scene can be affected by both occlusion and clutter. The aim of 3D shape retrieval is to search for similar 3D models in a gallery given a probe 3D shape (Tangelder and Veltkamp 2004). Both the probe (scene) and the gallery (model) shapes are commonly represented by full 3D meshes. In

this case, occlusion and clutter are absent in either the scene or the model. The aim of 3D modeling is to register and integrate a set of meshes of an object that are acquired from different viewpoints in order to construct a complete model of the object (Guo et al. 2014c). Each scene is a 2.5D mesh scanned from an isolated object. Models are not needed for the 3D modeling scenario. Objects can be subject to occlusion in each scene, but there is no clutter.

Each scene in the *Random views, Laser scanner*, *LIDAR* and *Dense stereo* datasets is a 2.5D mesh acquired by a scanner from a specific viewpoint, whereas their models are full 3D meshes. Therefore, these datasets are suitable to compare the performance of 3D local feature descriptors in the case of object recognition. The *Space time* and *Kinect* datasets represent a different object recognition scenario where 2.5D models (rather than full 3D models) are used to recognize their instances in cluttered 2.5D views.

The *Retrieval* dataset focuses on the 3D shape retrieval scenario, where a full model is used to create each scene without any occlusion or clutter. These scenes include rigid transformations and synthetic noise. The main purpose of employing this dataset is to address a shape retrieval scenario using the same objects as *Random Views*. Therefore, the impact of the application contexts (e.g., recognition *vs* retrieval) on the performance of the descriptors can be highlighted (Tombari et al. 2013).

The *2.5D Views* dataset addresses a 3D modeling scenario where each scene is a 2.5D mesh of one object without any clutter. The scenes in the *2.5D Views* dataset were scanned from the same objects as the *Laser Scanner* dataset. Therefore, the impact of the application context (i.e., recognition *vs* modeling) on the performance of the descriptors can also be highlighted here.

(iii) Various Image Qualities: These selected datasets contain large variations in the image quality. For example, the quality of the surface meshes in *Retrieval* and *Random views* are very high since they were synthetically generated from high-resolution models. *Laser scanner* and *2.5D Views* contain images with a medium level quality. In contrast, the image quality of the other datasets is relatively low because they were acquired with low-resolution scanners. The variation in the image quality enables us to analyze the performance of the existing descriptors under different levels of image qualities.

选取dataset的理由：
1. 从不同的获取设备；
2. 不同的应用场景；3. 不同的图像质量。 —— 能否更加深入地挖掘一下，这些只是表面现象，其对应的点云的内部特征是什么呢？—— 比如说sparse/dense，noise之类的一些

4.2 Ground-Truth

There are six datasets in the context of object recognition: Each dataset consists of a set of models $\mathcal{M} = \left\{\mathbf{M}_{j=1}^{N_m}\right\}$ and a set of scenes $\mathcal{S} = \left\{\mathbf{S}_{k=1}^{N_s}\right\}$. Each scene is a 2.5D point-cloud/mesh (acquired by a scanner from a specific viewpoint) that contains a subset of the models. The ground-truth rigid transformation (i.e., rotation $\boldsymbol{R}_{jk}$ and translation $\boldsymbol{t}_{jk}$) between each model $\mathbf{M}_j$ and its instance in the scene $\mathbf{S}_k$ is known a priori. In the synthetic datasets (e.g., *Random views*), the ground-truth transformation is obtained during the process of simulation (Tombari et al. 2013). In the non-synthetic datasets, the ground-truth transformation is recorded during the process of pointcloud/mesh acquisition. The reader is referred to Mian et al. (2006a); Tombari et al. (2010b); Taati and Greenspan (2011); Tombari et al. (2013) for more details regarding the generation of these ground-truth transformations.

The *retrieval* dataset in 3D shape retrieval context: The dataset comprises 6 models $\mathcal{M} = \left\{\mathbf{M}_{j=1}^{6}\right\}$ and 18 scenes $\mathcal{S} = \left\{\mathbf{S}_{k=1}^{18}\right\}$. Each scene is a 3D mesh which was created by applying a random rigid transformation to a selected 3D model. The ground-truth transformation is therefore known during the process of simulation.

The *2.5D View* dataset in the 3D modeling context: The dataset contains a set of scenes $\mathcal{S} = \left\{\mathbf{S}_{k=1}^{N_s}\right\}$ from four objects. Each scene is a 2.5D mesh which contains a separate object and was acquired by a scanner from a specific viewpoint. The ground-truth transformation between any pair of meshes of the same object is computed in two steps. First, a coarse transformation is obtained using manually selected point correspondences. The transformation is then applied to one mesh such that the two meshes are roughly aligned. Second, the iterative closest point (ICP) algorithm (Besl and McKay 1992) is used to refine the transformation between the roughly aligned meshes. The composition of the coarse and fine transformations results in an accurate transformation between the two meshes.

评价标准的参考：ground truth 是如何生成的

4.3 Evaluation Criteria

We test the selected descriptors in terms of descriptiveness, robustness, scalability, and efficiency. We also tested the combined performance of these descriptors with different keypoint detectors. Their definitions are given in the following.

4.3.1 Descriptiveness

We use the *Precision-recall* curve (PRC) to evaluate the descriptiveness of a feature descriptor. The PRC is commonly used for the evaluation of local feature descriptors (in both 2D images and 2.5D meshes), for example in Ke and Sukthankar (2004); Mikolajczyk and Schmid (2005); Flint et al. (2008); Tombari et al. (2010b); Guo et al. (2013b). The PRC is more suitable for evaluating feature descriptors compared to another popular criterion (i.e., receiver operating characteristics (ROC)) which is well-suited for evaluating classifiers

(Ke and Sukthankar 2004; Tombari et al. 2010b; Mikolajczyk and Schmid 2005).

The PRC is generated as follows. First, a number of keypoints are detected from both the scene and all the models. A feature descriptor is then computed for each keypoint using the method under consideration. Second, the nearest neighbor distance ratio (NNDR) technique (Lowe 2004; Mikolajczyk and Schmid 2005) is used to perform feature matching. Specifically, for each feature $\boldsymbol{f}_i^S$ in the scene, its nearest and second nearest neighbors (denoted by $\boldsymbol{f}_i^M$ and $\boldsymbol{f}_{i'}^M$) in the models are selected. The ratio between the two distances is calculated as $\left\|\boldsymbol{f}_i^S - \boldsymbol{f}_i^M\right\| / \left\|\boldsymbol{f}_i^S - \boldsymbol{f}_{i'}^M\right\|$. If the distance ratio is less than a threshold τ, the two features $\boldsymbol{f}_i^S$ and $\boldsymbol{f}_i^M$ are considered a match, as given in Lowe (2004); Moreels and Perona (2007); Tombari et al. (2010b); Guo et al. (2013b). Further, if $\boldsymbol{f}_i^M$ comes from the same object as $\boldsymbol{f}_i^S$, and the distance between the keypoint of $\boldsymbol{f}_i^M$ and the ground-truth corresponding point of $\boldsymbol{f}_i^S$ is less than half of the support radius, the match is assumed correct (similar to the test used in Mikolajczyk and Schmid (2003, 2005); Flint et al. (2008)). Otherwise, it is assigned a false match. The *precision* is calculated as the number of correct matches with respect to the total number of matches:

$$Precision = \frac{\text{The number of correct matches}}{\text{The number of matches}}. \tag{1}$$

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

$$Recall = \frac{\text{The number of correct matches}}{\text{The number of corresponding features}}. \tag{2}$$

The value of the threshold τ is varied from 0 to 1 to obtain the PRC. To test the descriptiveness of the descriptors, the ISS method with boundary point removal (ISS-BR) was used to detect the keypoints in the scene and the models. As reported in Salti et al. (2012), ISS achieves the best performance compared to other keypoint detectors when used in conjunction with feature descriptors. This conclusion is also demonstrated in Sect. 5.5. Since the same procedure is applied to all methods, we believe the comparison is fair and unbiased. Moreover, in order to further demonstrate the results of existing feature descriptors when combined with different 3D keypoints, we provide an additional experiment, which is discussed in Sects. 4.3.4 and 5.5. Note that, once the keypoints are detected, the same keypoints are used for all descriptors.

Since there is no model in the *2.5D View* dataset, we take each pair of meshes that have an overlap of more than 50% as a scene-model pair. That is, we treat one view as a scene and another view as a model, and follow the same method described above to calculate the PRC results. For all these datasets, we plot the average of the PRC results over the number of model-scene pairs of each dataset to show aggregated results.

4.3.2 Robustness

We test the robustness of each feature description method with respect to a set of sources of interferences that may affect the performance. These sources include Gaussian noise, shot noise, varying mesh resolutions, support radius, distance to the mesh boundary, keypoint localization errors, occlusion and clutter.

Support Radius We use different support radii to define the neighboring local surface of each keypoint. For a given radius ρ, points within a radius of ρ of the keypoint constitute the neighborhood of that keypoint. It should be noted that in the case of 3D data, "scale" corresponds to the "support radius" (Tombari et al. 2013). In this paper, the support radius ρ is defined based on a global measure such that the extracted descriptor is less sensitive to the sampling resolution of the data. Following the approach proposed in Zaharescu et al. (2012), the support radius ρ is calculated as:

$$\rho = \sqrt{\frac{\alpha A_m}{\pi}}. \tag{3}$$

where A_m is the total area of the 3D surface of an object, α is a parameter to control the size of the support radius. For full 3D models (e.g., those in the *Retrieval* and *Laser scanner* datasets), the total area A_m is computed as the sum of all triangle areas of the mesh. For 2.5D models (e.g., those in the *Kinect* and *2.5D Views* datasets) acquired from a single viewpoint, the total area A_m is estimated as 4.5 times the overall area of the 2.5D mesh. Here, 4.5 is the average ratio between the area of a 2.5D single-view mesh and the area of its corresponding 3D model for a number of tested objects. Seven different support radii with α ranging from 0.08 to 1.4 % are used to produce the results presented in Sect. 5.3.1.

Gaussian Noise: We add five levels of Gaussian noise with standard deviations of $\frac{\rho}{150}$, $\frac{2\rho}{150}$, $\frac{3\rho}{150}$, $\frac{4\rho}{150}$, and $\frac{5\rho}{150}$ to each scene, where ρ denotes the support radius. For a given standard deviation, Gaussian noise is independently added to the x, y, and z-axes of each scene point. An illustration of a scene with three levels of Gaussian noise is shown in Fig. 3a–c. The robustness results with respect to different levels of Gaussian noise are presented in Sect. 5.3.2.

Shot Noise: We add five levels of shot noise with outlier ratios of 0.2, 0.5, 1.0, 2.0, and 5.0 % to each scene. Given an outlier ratio γ, a ratio γ of the total points in each scene is first selected and a displacement with an amplitude of $\frac{4\rho}{3}$ is then added to each selected point along its normal direction

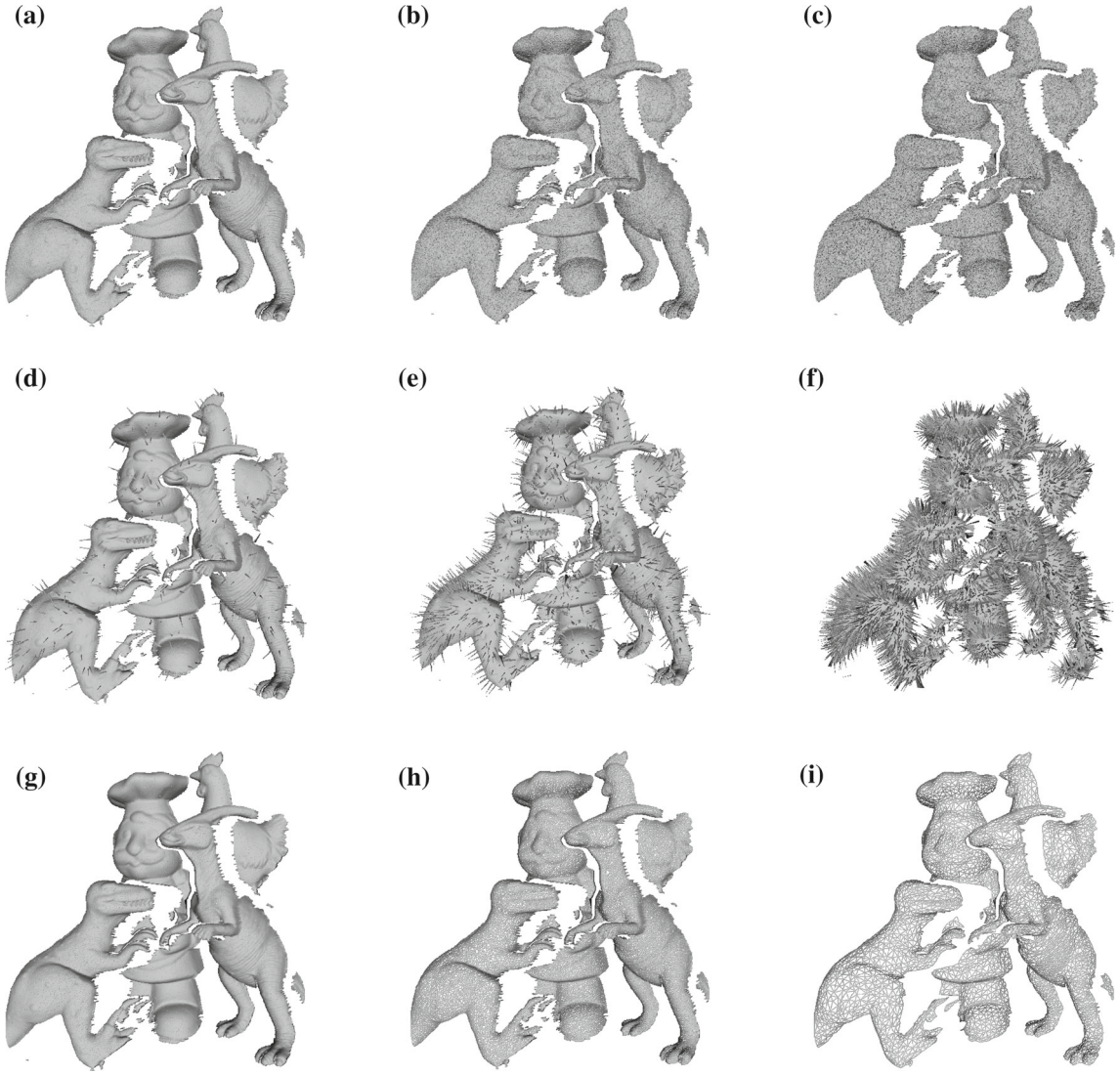

Fig. 3 An illustration of a scene with different levels of Gaussian noise, shot noise, and mesh resolutions. **a** Gaussian noise—Level 1 **b** Gaussian noise—Level 3 **c** Gaussian noise—Level 5 **d** Shot noise—Level 1 **e** Shot noise—Level 3 **f** Shot noise—Level 5 **g** Decimation—Level 1 **h** Decimation—Level 3 **i** Decimation—Level 5

(the same as in Zaharescu et al. (2012)). Note that, shot noise usually lies along the viewing line of a given point. However, the displacement along the normal direction can well approximate the real shot noise, especially for complete 3D meshes (Zaharescu et al. 2012). An illustration of a scene with three levels of shot noise is shown in Fig. 3d–f. The robustness results with respect to different levels of shot noise are presented in Sect. 5.3.3.

Varying Mesh Resolutions We resample each scene at five levels such that only 1/2, 1/4, 1/8, 1/16, and 1/32 of their original points are left in the resampled scene. An illustration of a scene with three levels of mesh resolutions is shown in Fig. 3g–i. The robustness results with respect to varying mesh resolutions are presented in Sect. 5.3.4.

Distance to the Mesh Boundary We first extract the boundary points of each scene (as shown in Fig. 4), and then calculate the shortest distance d_b of each keypoint to the boundary points. We classify the scene keypoints into 6 groups according to their distances to the mesh boundary. Each group contains keypoints with a range of distances. For example, the 2nd group contains keypoints with distances d_b larger than $\frac{\rho}{5}$ and less than $\frac{2\rho}{5}$. The 6th group contains keypoints with distances d_b larger than ρ (ρ is the support radius). The robustness results with respect to different distances to the mesh boundary are presented in Sect. 5.3.5.

Keypoint Localization Error For each pair of corresponding points $\left(\boldsymbol{p}_i^M, \boldsymbol{p}_i^S\right)$ in each scene-model pair, we randomly select another scene point $\boldsymbol{p}_{i'}^S$ such that the distance between

Fig. 4 An illustration of a scene with boundary points. The boundary points are shown in *red* (Color figure online)

$\boldsymbol{p}_i^S$ and $\boldsymbol{p}_{i'}^S$ is less than a threshold τ_d. We use these new corresponding points $(\boldsymbol{p}_i^M, \boldsymbol{p}_{i'}^S)$ to produce the RPC results. Six different distance thresholds τ_d (i.e., $\frac{\rho}{15}$, $\frac{3\rho}{15}$, $\frac{5\rho}{15}$, $\frac{7\rho}{15}$, $\frac{9\rho}{15}$, and $\frac{11\rho}{15}$) are used in the tests. The robustness results with respect to different keypoint localization errors are presented in Sect. 5.3.6.

Occlusion and Clutter In order to analyze the robustness of the selected feature descriptors with respect to occlusion and clutter, we first calculate the occlusion and clutter of each local surface around a keypoint of the scene. Following a similar approach to the one used in Johnson and Hebert (1999); Mian et al. (2006a), the occlusion and clutter for each local surface in a scene is calculated as:

$$\text{Occlusion} = 1 - \frac{\text{The area of model local surface in scene}}{\text{The area of the model local surface}}, \tag{4}$$

and

$$\text{Clutter} = 1 - \frac{\text{The area of model local surface in scene}}{\text{The area of the scene local surface}}, \tag{5}$$

Note that, in Johnson and Hebert (1999); Mian et al. (2006a), the overall occlusion and clutter are calculated for each object that is present in the scene. However in this paper, the occlusion and clutter are calculated for each local surface around a keypoint (since we are investigating the robustness of a feature descriptor that is defined on a local surface rather than the whole surface of an object). The keypoints in the scene are then classified into several different groups according to their occlusion and clutter. The robustness results with respect to different occlusion and clutters are presented in Sect. 5.3.7.

In order to avoid the influence of the keypoint detection algorithms on the robustness results, N_f (N_f=1000 in this evaluation) keypoints are randomly selected from each scene without keypoint detection, their corresponding points in the models are then extracted using the known rigid transformations between the scene and models (as in Tombari et al. (2010b); Guo et al. (2013b)). Then, $2N_f$ feature descriptors are computed using the method under consideration, and the PRC curve for each descriptor is generated. Further, the area under the PRC curve (denoted by AUC_{pr}) for each descriptor is calculated. AUC_{pr} is a simple and aggregated metric to measure how an algorithm performs over the whole precision-recall space (Davis and Goadrich 2006). A perfect descriptor would produce a recall equal to 1 for any precision. In this ideal case, the AUC_{pr} is 1. In order to further avoid the influence of the feature descriptiveness on the robustness results (due to the significant variations in the absolute values of the descriptiveness metrics between different descriptors), the AUC_{pr} of each descriptor is normalized by its maximum value under different levels of a specific nuisance (e.g., Gaussian noise). Consequently, the normalized AUC_{pr} clearly indicates the robustness of a selected descriptor, without any influence caused by the descriptiveness of the descriptors and the accuracy of the keypoint detectors.

4.3.3 Scalability

In order to test the performance of the feature descriptors on large datasets and their scalability with respect to different sizes of the model dataset, we constructed a new dataset which consists of 52 models. The models in the dataset were collected from a number of sources including the *Laser scanner*, *LIDAR*, *Kinect* datasets and the Ca' Foscari Venezia dataset (Rodolà et al. 2013). The scenes in the dataset are taken from the *Laser scanner* dataset. The performance of the selected descriptors with respect to different sizes of model datasets is presented in Sect. 5.4.

4.3.4 Combination with 3D Keypoint Detectors

Although the focus of this paper is on the evaluation of local feature descriptors, we also assess the performance of the selected descriptors combined with different 3D keypoint detectors. The purpose is to demonstrate the effect of different detectors on the performance of feature descriptors. We use uniform sampling (Rusu and Cousins 2011), Harris3D (Sipiran and Bustos 2011), ISS (Zhong 2009), and ISS with boundary point removal (ISS-BR) to detect the keypoints in each scene and model. We then use each selected feature description method to generate a descriptor around

each keypoint. The performance of each detector-descriptor combination is also measured with PRC. The main purpose of this experiment is to investigate the overall performance of a feature descriptor when combined with different keypoint detectors. An evaluation of these detector-descriptor pairs on all datasets is not the main focus of this paper. We therefore, only present the results of this experiment on the *Laser scanner* dataset for readability.

4.3.5 Efficiency

A thorough evaluation of the selected descriptors is provided in terms of computational efficiency. For each method, we calculate the average time on several scenes to generate 1000 descriptors in each scene. Since the computational time for the feature generation is related to the number of points in the support region, we calculate the time for generating a feature descriptor with respect to various number of points in the support region (i.e., from 10^2 to 10^5 points). Note that, a major factor that would affect the computational time is the number of points in the local surface (rather than the properties of the dataset). We therefore, only present the results of this experiment on the *Laser Scanner* dataset in order to improve the readability. 给出评价指标以及计算方式，为什么这样计算。

4.4 Implementation Details

In the following, we present the implementation details of the algorithms for normal and curvature estimation, and local feature description. We also describe the selection of descriptors.

4.4.1 Normal and Curvature Estimation

这里需要进一步排除，需要hack到算法内部进行评价

Surface normals and curvatures are commonly used in many 3D local feature descriptors (Bariya et al. 2012). Since the triangular mesh is the most popular approximation of a continuous surface, we evaluate the existing 3D feature descriptors in this discrete format (Tombari et al. 2013). For the datasets (e.g., *LIDAR*) which only provide point-cloud representations of the scenes and models, we convert these pointclouds into triangular meshes using the method described in Guo et al. (2013b). Let $\boldsymbol{I} = \{\boldsymbol{V}, \boldsymbol{F}\}$ be the data structure of a mesh comprising vertices $\boldsymbol{V}$ and triangular faces $\boldsymbol{F}$, where vertices (or points) $\boldsymbol{V}$ are the 3D coordinates of each point and the triangular faces $\boldsymbol{F}$ are the index numbers of the points which make up the individual faces. The normal of a triangular face can be calculated from the equation of the plane $n_1x+n_2y+n_3z+a = 0$, where the normal of the triangular face is $[n_1, n_2, n_3]^{\mathrm{T}}$. Since the triangular face is defined by three points, the equation of the normal can easily be solved (Mian et al. 2010). Given the normal of each triangular face, the normal of a point is then determined as the mean value of the normals of all triangular faces sharing that point, the same as in Mian et al. (2010); Zaharescu et al. (2012). The surface normals are reoriented towards the the outside of the objects in order to resolve any ambiguity related to the direction of the surface normals. Besides the surface normals, the mean/Gaussian curvatures and shape index values of a mesh are calculated using the algorithm proposed by Chen and Schmitt (1992). Note that, other algorithms for normal and curvature estimation are also available in the literature, e.g., (Meek and Walton 2000). They are however out of the scope of this paper (examining the best normal and curvature estimation algorithm is not our focus). Moreover, since the same algorithms for the normal and curvature estimation are used for all descriptors, we can consider that the test is unbiased.

4.4.2 Selected Descriptors

We use 10 different descriptors for our performance evaluation. These descriptors were briefly described in Sect. 3 and include SI (Johnson and Hebert 1999), 3DSC (Frome et al. 2004), LSP (Chen and Bhanu 2007a), THRIFT (Flint et al. 2008), PFH (Rusu et al. 2008), FPFH (Rusu et al. 2009), SHOT (Tombari et al. 2010b), USC (Tombari et al. 2010a), RoPS (Guo et al. 2013b), and TriSI (Guo et al. 2013c). These descriptors are selected based on the criteria of popularity, state-of-the-art performance, and their category (based on their underlying concept). Other methods presented in Sect. 3 have specific requirements that make their inclusion in this comparison infeasible. Specifically, the 3D Tensor descriptor (Mian et al. 2006a) is defined at the center of two points rather than a single point of the input mesh, and it is therefore difficult to generate 3D Tensor descriptors at a set of given keypoints. VD-LSD (Taati and Greenspan 2011) needs a computationally expensive training phase for each model (i.e., 5–6 h per model (Taati and Greenspan 2011)). Moreover, the resulting descriptor is model (object) dependent, and it is not optimal for other models other than the training model. 2.5D SIFT (Lo and Siebert 2009) and SI-SIFT (Bayramoglu and Alatan 2010) can only work on depth images with a lattice structure. Since it is impossible to obtain a single depth image for a full 3D model, these methods cannot be tested in the context of 3D object recognition and 3D shape retrieval.

3DSC, PFH, FPFH, SHOT, and USC were implemented in C++. They are available in the Point Cloud Library (PCL) Version 1.7.1 (Rusu and Cousins 2011; Aldoma et al. 2012a), while the others were implemented in Matlab 2011b. All these descriptors were tested on a Windows 7 platform. Unless otherwise stated, all the parameters of these descriptors were fixed during the experiments and for all datasets. In a similar manner to Mikolajczyk et al. (2005); Salti et al. (2012); Alexandre (2012); Restrepo and Mundy (2012); Tombari et al. (2013), the proposed default parameters in

Table 2 Selected 3D local feature descriptors

No.	Name	Length	Parameters	Implementation
1	SI (Johnson and Hebert 1999)	225	$d_{si} = 15$	Matlab
2	3DSC (Frome et al. 2004)	1980	$d_{sc\_r} = 15, d_{sc\_a} = 12, d_{sc\_e} = 11$	PCL
3	LSP (Chen and Bhanu 2007a)	578	$d_{lsp\_s} = 17, d_{lsp\_a} = 34$	Matlab
4	THRIFT (Flint et al. 2007)	32	$d_{thrift} = 32$	Matlab
5	PFH (Rusu et al. 2008)	125	$d_{pfh} = 5$	PCL
6	FPFH (Rusu et al. 2009)	33	$d_{fpfh} = 11$	PCL
7	SHOT (Tombari et al. 2010b)	352	$d_{shot\_a} = 8, d_{shot\_r} = 2,$ $d_{shot\_e} = 2, d_{shot\_h} = 11$	PCL
8	USC (Tombari et al. 2010a)	1980	$d_{sc\_r} = 15, d_{sc\_a} = 12, d_{sc\_e} = 11$	PCL
9	RoPS (Guo et al. 2013b)	135	$d_{rops\_r} = 3$	Matlab
10	TriSI (Guo et al. 2013c)	675	$d_{trisi} = 15$	Matlab

the original articles or PCL implementations were used for all selected descriptors. The values of these parameters are listed in Table 2. The only tuned parameter is d_{thrift}, the number of bins of the histogram in the THRIFT descriptor. d_{thrift} was set to 32 after tuning the performance of THRIFT on an independent training dataset. For a descriptor with fixed parameters, the support radius of the local surface determines not only the descriptiveness of the descriptor but also its robustness to occlusion and clutter. The support radius for all descriptors was set according to Eq. 3 with an α equal to 0.54 % throughout this paper (except Sect. 5.3.1 where it was varied to assess the robustness of the descriptors). All experiments were conducted on a computer with 3.5 GHz Intel Core i7-2700K CPU and 16GB of RAM.

5 Performance Evaluation

In this section, we present and discuss the experimental results of our evaluation. First, the descriptiveness of the descriptors was tested on eight different datasets (Sect. 5.1). Second, the overall compactness of these descriptors over the eight datasets was analyzed (Sect. 5.2). Third, the robustness of descriptors was investigated on the *Laser scanner* dataset with respect to a number of factors (Sect. 5.3). Fourth, the scalability of the descriptors with respect to different numbers of models was studied (Sect. 5.4). Fifth, the combined performances of the selected feature descriptors with several different keypoint detectors were analyzed (Sect. 5.5). Finally, the computational efficiency of these descriptors was presented (Sect. 5.6).

5.1 Descriptiveness

In this section, we use PRC (Sect. 4.3.1) to evaluate the descriptiveness of the selected descriptors on the eight datasets of Sect. 4.1.

5.1.1 Retrieval Dataset

The *Retrieval* dataset contains 18 scenes and 6 models. The scene meshes with the lowest level of noise are used in this experiment. Figure 5a shows the PRC results of the selected descriptors on this dataset. USC achieves the best recall results, followed by SHOT, RoPS, FPFH, PFH and TriSI. 3DSC gives a moderate performance. Note that, the recall achieved by USC is much higher than 3DSC on the same dataset. This clearly demonstrates that the use of an LRF in USC is able to reduce the memory requirements and the computational complexity of 3DSC, and improve the matching accuracy of 3DSC (Tombari et al. 2010a). Besides, SI is inferior to 3DSC but outperforms LSP and THRIFT. THRIFT produces very low scores on this dataset. The ranking for SHOT, FPFH, and SI is consistent with the results reported in Salti et al. (2014).

5.1.2 Random Views Dataset

The *Random views* dataset contains 36 scenes and 6 models. The scene meshes with the lowest level of noise are used in this experiment. The PRC results of the selected descriptors on this dataset are shown in Fig. 5b. SHOT achieves the best performance, closely followed by PFH, FPFH and RoPS. The next discriminative descriptor is TriSI. 3DSC achieves slightly better results compared to SI and USC. Similar to the case of the *Retrieval* dataset, LSP and THRIFT give the lowest scores. Note that, *Retrieval* and *Random views* have the same models, with the major difference that the *Random Views* dataset contains occluded objects and clutter. Comparing the results in Fig. 5a and b, three observations can be made. *First*, the recall on *Random views* is significantly lower than the recall of *Retrieval* due to the more challenging conditions caused by occlusions and clutter (see Fig. 2a and b). *Second*, when comparing the difference of the performance of each descriptor on these two datasets, USC, TriSI, RoPS,

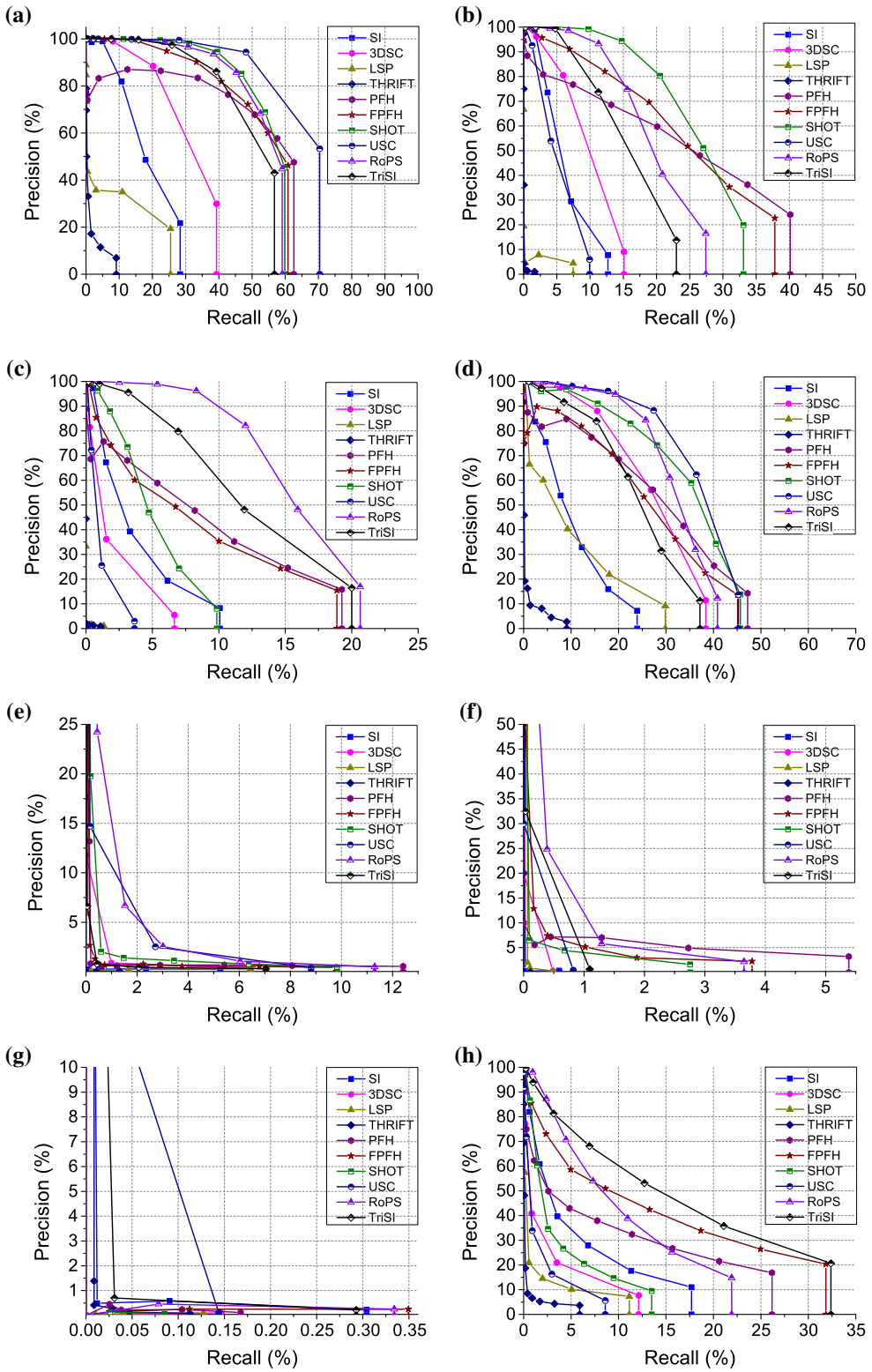

Fig. 5 Descriptiveness of the selected descriptors of Sect. 3 on the eight datasets of Sect. 4.1. **a** Retrieval **b** Random views **c** Laser scanner **d** Space time **e** Kinect **f** LIDAR **g** Dense Stereo **h** 2.5D views (Color figure online)

and SHOT have a larger drop compared to the other descriptors. This is because the TriSI, RoPS, and SHOT descriptors are very sensitive to occlusions and clutter (further results in Sects. 5.3.1, 5.3.5, and 5.3.7). *Third*, the rankings of these descriptors on these two datasets are similar except for USC. USC is more suitable for 3D shape retrieval compared to 3D object recognition .

5.1.3 Laser Scanner Dataset

Figure 5c shows the PRC results of the selected descriptors on the *Laser scanner* dataset. RoPS achieves the best result, showing a significant improvement over the other descriptors. The next best performing descriptor is TriSI, followed by PFH and FPFH that have a similar performance. Note that, FPFH significantly reduces the computational complexity of the feature generation compared to PFH, while maintaining a similar performance in terms of feature matching accuracy. SHOT produces moderate results, which are better than those of SI and 3DSC. 3DSC performs much better than USC on this dataset at the cost of an increased computational complexity and storage requirement.

5.1.4 Space Time Dataset

Figure 5d gives the PRC results of the selected descriptors on the *Space Time* dataset. We can see that USC outperforms all the other descriptors, closely followed by SHOT and RoPS. 3DSC also achieves acceptable results on this dataset. It can be concluded that shape context style descriptors (such as 3DSC and USC) are more suitable for applications with *Space Time*. The next best performing descriptors are PFH, FPFH, and TriSI. All of these descriptors produce very close responses. The difference between FPFH and PFH is small. A similar performance is also achieved by SI and LSP, with a lower recall compared to TriSI. The ranking for SHOT, FPFH, and SI is consistent with the results reported in Salti et al. (2014).

5.1.5 Kinect Dataset

The performance of the descriptors on the *Kinect* dataset is shown in Fig. 5e. The recall on this dataset is lower than the recall on the *Laser scanner* and *Space time* datasets by a large margin. This is caused by the low quality of the data acquired with the Kinect sensor which is very noisy and spiky, and comes with a low depth resolution. All descriptors achieve very low recall on this dataset. RoPS produces the best performance compared to other descriptors, followed by USC, SHOT, PFH and FPFH. The recall of FPFH is lower than PFH, while LSP works slightly better than SI. We also observe that THRIFT does not work well on this dataset.

5.1.6 LIDAR Dataset

Figure 5f shows the PRC results of the selected descriptors on the *LIDAR* dataset. The recall on this dataset is very low for all descriptors. The overall performance of these descriptors on this dataset is even worse than the performance on the *Kinect* dataset (Sect. 5.1.5). It can be seen that the best performance attributes to RoPS, followed by PFH, FPFH, TriSI, and SHOT. USC performs better than 3DSC, the same as on the *Space Time* and *Kinect* datasets (see Fig. 5d and e). The achieved recall of SI, LSP and THRIFT on this dataset is very low. The ranking for RoPS, 3DSC and SI is also consistent with the results reported in Taati and Greenspan (2011); Guo et al. (2013b) that used the same dataset, where the recognition rates achieved by the RoPS, 3DSC, and SI based algorithms are 95.4, 62.1, and 35.4 %, respectively.

5.1.7 Dense Stereo Dataset

In Fig. 5g, we present the PRC results of these descriptors on the *Dense stereo* dataset. The quality of the data acquired with the dense stereo technique (Bumblebee sensor in this case) is very poor. The stereo images have much higher noise levels compared to the *LIDAR* dataset (Taati and Greenspan 2011). Consequently, the recalls achieved by all descriptors are extremely low, with the largest value below 0.4 %. Meanwhile, the differences in performance between all descriptors are very small. It can be noticed that a relatively better performance is achieved by USC, TriSI, THRIFT, and SI.

5.1.8 2.5D Views Dataset

Figure 5h shows the PRC results of these descriptors on the *2.5D Views* dataset. TriSI gives the best results, followed by FPFH and RoPS. PFH produces a much lower score compared to FPFH although the former requires more computational time. SI and SHOT give a moderate performance, followed by 3DSC and USC. Besides, the scores of LSP and THRIFT are amongst the lowest.

Both *2.5D Views* and *Laser scanner* were acquired with Minolta Vivid 910. The major difference between these two datasets is that *Laser scanner* contains both occlusions and clutter while *2.5D Views* only contains occlusions. Compared to the results reported on *Laser scanner* (Fig. 5c), several observations can be drawn. *First*, the overall rankings of these descriptors are similar on the two datasets. RoPS and TriSI give the best results, while 3DSC, USC, LSP, and THRIFT achieve relatively low scores. *Second*, RoPS achieves the best performance on *Laser scanner* while TriSI performs best on *2.5D Views*. It can be concluded that RoPS is more suitable for the case of 3D object recognition rather than 3D modeling. Similarly, PFH is more suitable for 3D object recognition while FPFH is more suitable for 3D modeling.

Table 3 The AUC_{pr} results of the descriptors of Sect. 3 on the eight datasets of Sect. 4.1

Descriptor / Dataset	SI	3DSC	LSP	THRIFT	PFH	FPFH	SHOT	USC	RoPS	TriSI
Retrieval	0.18494	0.30731	0.08053	0.01388	0.48490	*0.52107*	*0.54452*	**0.63877**	*0.53190*	0.49252
Random views	0.06177	0.09558	0.00478	0.00044	*0.23110*	*0.24480*	**0.26045**	0.05055	*0.19597*	0.15498
Laser scanner	0.03130	0.01959	0.00020	0.00013	*0.08167*	*0.07724*	0.04830	0.01029	**0.15010**	*0.11585*
Space time	0.10091	0.26879	0.10048	0.00753	*0.27320*	0.25585	*0.33830*	**0.36522**	*0.31886*	0.24042
Kinect	0.00014	0.00099	0.00015	0.00001	*0.00125*	0.00101	*0.00213*	*0.00391*	**0.00638**	0.00083
LIDAR	0.00010	0.00049	0.00064	0.000000	*0.00284*	*0.00209*	0.00145	0.00125	**0.00491**	*0.00196*
Dense stereo	*0.00005*	0.00000	0.00000	*0.00007*	0.00000	0.00001	0.00000	**0.00012**	0.00001	*0.00008*
2.5D views	0.05308	0.02596	0.01427	0.00375	*0.08668*	*0.13429*	0.03769	0.01704	*0.10133*	**0.16109**
Average	0.05403	0.08984	0.02513	0.00323	0.14521	*0.15454*	*0.15410*	0.13589	**0.16368**	*0.14597*
Median	0.04219	0.02277	0.00271	0.00029	*0.08418*	*0.10576*	0.04300	0.01366	*0.12572*	**0.13541**

The best performance is reported in bold face, and the top 4 results for each dataset are shown in italic

Third, 3DSC achieves a better performance than USC on these two datasets.

5.1.9 Descriptiveness Overall Performance

In order to directly compare the performance of these descriptors on each dataset, the AUC_{pr} metrics of all these descriptors on the eight datasets are reported in Table 3. In order to further evaluate the overall performance of these descriptors, we also present the average and median AUC_{pr} of the descriptors over all datasets. Several conclusions can be summarized as follows.

First, RoPS is the best performing descriptor. Specifically, RoPS achieves the best performance on the *Laser scanner*, *Kinect*, and *LIDAR* datasets. USC performs best on the *Retrieval*, *Space Time* and *Dense stereo* datasets. Overall, RoPS has the highest average value of AUC_{pr} across all these datasets. It outperforms USC by a large margin, with median values of AUC_{pr} equal to 0.12572 and 0.01366, respectively. Other good feature descriptors in terms of average and median AUC_{pr} include SHOT, TriSI, PFH, and FPFH. In contrast, LSP and THRIFT are the descriptors with the lowest overall performance on all these datasets.

Second, the performance of these descriptors depends on the dataset. It is clear that RoPS, FPFH, and PFH are the descriptors with the best performance on the high-resolution datasets (i.e., *Retrieval*, *Random views*, *Laser scanner*, and *2.5D views*). Besides, RoPS, USC, PFH, and TriSI are the top descriptors when tested on the low resolution datasets (i.e., *Space time*, *Kinect*, *Dense stereo*, and *LIDAR*). RoPS and PFH achieved good results (shown in italic in Table 3) on both low and high resolution datasets.

Third, TriSI, SI, RoPS, FPFH, and PFH generally show a more stable performance across datasets compared to all the others. In contrast, the performance of THRIFT, USC, LSP, 3DSC, and SHOT varies significantly, as revealed by the large differences between their average and median values of the AUC_{pr}. This conclusion corroborates with the results in Restrepo and Mundy (2012) and Kim and Hilton (2013).

Fourth, the AUC_{pr} results of all descriptors on *Kinect*, *LIDAR* and *Dense Stereo* datasets are very low. This indicates that the design of effective descriptors for high noise and low-resolution data requires more attention.

5.2 Compactness

These selected feature descriptors have different lengths of floating-point numbers (see Table 2). The length of a feature affects the memory footprint and the computational efficiency during feature matching. In this work, we initially measure the compactness of a feature as the average value of AUC_{pr} per float. The term *compactness* is defined as

$$Compactness = \frac{\text{Average value of } AUC_{pr}}{\text{Length of the descriptor}}. \quad (6)$$

The term *compactness* represents the overall descriptiveness of each floating-point number in a descriptor vector. We use the average value of AUC_{pr} in Table 3 to calculate the *compactness* of each descriptor over all these datasets. The length of each descriptor is given in Table 2, and the *compactness* results are shown in Fig. 6. FPFH is the most compact feature descriptor. It achieves high-level performance in terms of precision and recall (see Table 3) with a very short descriptor (i.e., the length of the descriptor is 33). RoPS comes second with the highest AUC_{pr} value and a relatively short descriptor (with only 135 floats). PFH achieves a close score of *compactness* compared to RoPS. Besides, SHOT, SI, and TriSI achieve a medium performance in terms of compactness. On the other hand, although USC and 3DSC perform well in terms of precision and recall (see Table 3), their compactness is very low. 3DSC achieves the lowest

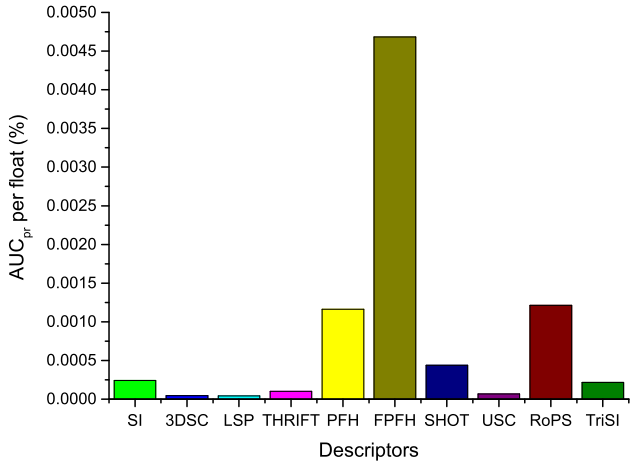

Fig. 6 The compactness of the selected descriptors on these datasets (Color figure online)

compactness compared to all the other descriptors. This is because the lengths of the USC and 3DSC descriptors are larger than the others by orders of magnitude. That is, the lengths of a typical 3DSC and USC descriptor are 1980. It can therefore, be concluded that FPFH, RoPS, and PFH are quite suitable for applications with strict limits on the computational complexity and storage requirements (e.g., robots and mobile phones).

5.3 Robustness

In this section, we use AUC_{pr} to evaluate the robustness of the selected descriptors with respect to a set of disturbances including support radius, Gaussian noise, shot noise, varying mesh resolutions, distance to the mesh boundary, keypoint localization error, occlusion and clutter. In order to reduce the number of charts and improve readability, we only present experimental results on the *Laser Scanner* dataset. The *Laser Scanner* dataset is selected because it is one of the most frequently used datasets in 3D computer vision (Mian et al. 2006a, 2010; Bariya et al. 2012; Tombari et al. 2013; Aldoma et al. 2012b; Guo et al. 2013b).

5.3.1 Support Radius

The support radius affects both the feature's descriptiveness and its robustness to occlusions and clutter (Guo et al. 2013b). That is, a large support radius enables a descriptor to encapsulate more information of the local surface and therefore provides higher descriptiveness. On the other hand, a large support radius increases the sensitivity to occlusion and clutter. The performances of these selected descriptors with respect to different support radii are shown in Fig. 7a. Seven support radii were used in the experiments.

Two major observations can be made from the results in Fig. 7a. *First*, the normalized AUC_{pr} results of RoPS, TriSI, PFH, and FPFH improve rapidly when the support radius is increased. Their performances then decrease with an increase in the support radius once a peak value is reached. This is because these descriptors are highly sensitive to occlusion and clutter (as further demonstrated in Sect. 5.3.7). They produce the best performance when an optimal trade-off is achieved between their descriptiveness and sensitivity. *Second*, for the descriptors that are less sensitive to occlusion and clutter (e.g., SI, USC, and 3DSC), their performances increase consistently with an increase in the support radius. This is because, the major factor influencing their performance is the encapsulated information of the underlying local surface rather than the mesh boundary (as further explained in Sect. 5.3.5).

5.3.2 Gaussian Noise

Figure 7b shows the normalized AUC_{pr} results of these descriptors with respect to different levels of Gaussian noise. The performances of all descriptors decrease rapidly when the standard deviation of the Gaussian noise increases. SHOT, USC, TriSI, and RoPS are the most robust descriptors with respect to low-level Gaussian noise. When the noise level is high (with a standard deviation of more than $\frac{2\rho}{150}$), USC outperforms the other descriptors by a large margin in terms of robustness. The normalized AUC_{pr} value of USC is very stable with a standard deviation of the Gaussian noise ranging from $\frac{2\rho}{150}$ to $\frac{5\rho}{150}$. In contrast, LSP, THRIFT, FPFH, and PFH are very sensitive to Gaussian noise, their AUC_{pr} values drop significantly when the standard deviation of the Gaussian noise increases to $\frac{\rho}{150}$. This is because these descriptors rely on the first-order and/or second-order surface derivatives (i.e., surface normal, shape index) that are prone to noise.

5.3.3 Shot Noise

The normalized AUC_{pr} results of these descriptors with respect to different levels of shot noise are presented in Fig. 7c. TriSI is highly robust to shot noise, its normalized AUC_{pr} value is very stable under all levels of shot noise. Its robustness is due to the fact that an adaptive outlier-rejection technique is used during the generation of the TriSI descriptor. USC and SHOT achieve a close performance compared to TriSI. The other descriptors are more vulnerable to shot noise. PFH and RoPS are highly sensitive to shot noise, their performance deteriorates dramatically even with a low level of shot noise. The performance of FPFH also drops sharply in the presence of shot noise. Overall, the spatial distribution histogram based descriptors (including TriSI, USC, SI, and 3DSC) are more robust to shot noise compared to the

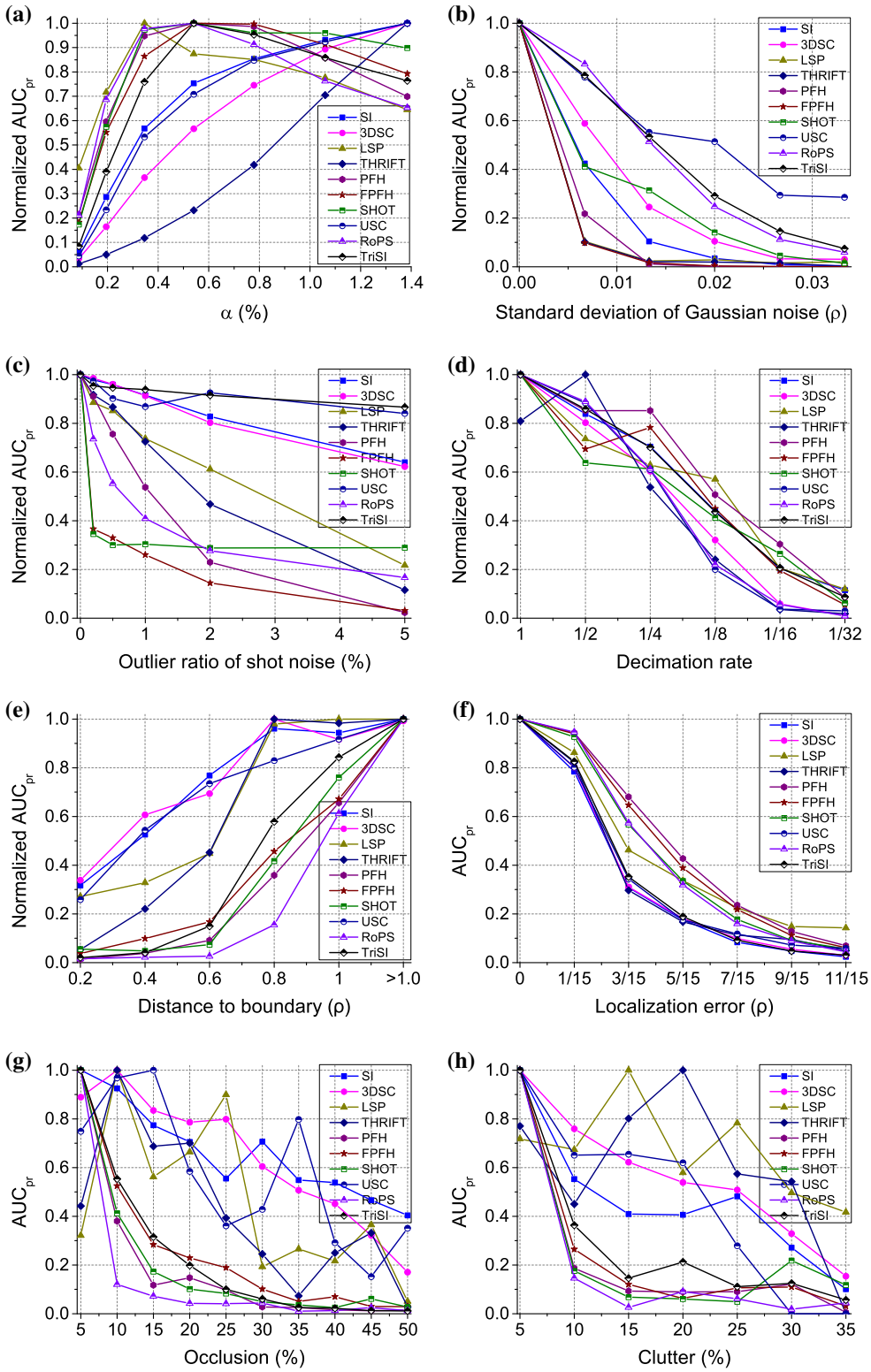

Fig. 7 Robustness of the selected descriptors of Sect. 3 on the *Laser scanner* dataset. **a** Support radius **b** Gaussian noise **c** Shot noise **d** Varying mesh resolutions **e** Distance to the mesh boundary **f** Keypoint localization error **g** Occlusion **h** Clutter (Color figure online)

geometric attribute histogram based descriptors (including FPFH and PFH). This is because the variation in the spatial distribution of a point cloud is minor in the presence of a small number of outliers. However, the geometric attributes (e.g., surface normals) can vary significantly in the presence of outliers. From Fig. 7b and c, it is also clear that FPFH and PFH descriptors are sensitive to both Gaussian and shot noise. SI is sensitive to Gaussian noise, but it is robust to shot noise.

5.3.4 Varying Mesh Resolutions

Figure 7d shows the normalized AUC_{pr} results of these descriptors with respect to varying mesh resolutions. PFH, FPFH, SI, TriSI are robust to varying mesh resolutions. Their drop in performance with respect to varying mesh resolutions is smaller compared to other descriptors. In contrast, THRIFT and USC are very sensitive to varying mesh resolutions.

5.3.5 Distance to the Mesh Boundary

The points distance from the boundary is an important attribute that can significantly affect the performance of a descriptor. A scene with boundary points is illustrated in Fig. 4 where the boundary points are shown in red. The performance of the selected descriptors with respect to the distance to the boundary is shown in Fig. 7e. We can observe that the performance of RoPS is significantly boosted by eliminating points that are close to the mesh boundary. Specifically, the normalized AUC_{pr} of RoPS is increased from about 0.6 to about 1.0 by removing points with distances less than 1 ρ to the mesh boundary. Similarly, the normalized AUC_{pr} results of PFH, SHOT, FPFH, and TriSI are also significantly improved by removing boundary points. In contrast, SI, 3DSC, and USC are more robust to boundary points. 3DSC achieves the best robustness performance compared to all the other descriptors when tested on keypoints with distances less than 0.5 ρ to the boundary. Since the points close to the boundary include occlusions and clutter (as shown in Fig. 4), it can be concluded that TriSI, FPFH, PFH, RoPS, and SHOT are sensitive to occlusions and clutter. In contrast, SI, 3DSC, and USC are very robust to occlusions and clutter. This is consistent with the conclusions drawn in Sects. 5.3.1 and 5.3.7.

5.3.6 Keypoint Localization Error

In this section, we investigate the influence of the keypoint localization error on the performance of descriptors. Figure 7f displays the normalized AUC_{pr} results with respect to different levels of keypoint localization errors. As expected, the recall decreases with increasing keypoint localization errors. The performance of TriSI, SI, and 3DSC drops faster compared to all the other descriptors, especially for keypoints with small localization errors. This indicates that these three descriptors are highly sensitive to the accuracy of the keypoint localization. Besides, the performance of USC and THRIFT are also significantly affected in the presence of keypoint localization errors. LSP, RoPS, and SHOT achieve a medium level of robustness performance. In contrast, PFH and FPFH are the most robust descriptors with respect to keypoint localization errors. Overall, the spatial distribution histogram based descriptors (including TriSI, SI, and 3DSC) are more sensitive to keypoint localization errors compared to the geometric attribute histogram based descriptors. This is because the former uses spatial distribution measurements (e.g., number of points) in a set of partitioned regions to represent the local surface. Therefore, the extracted feature descriptor is prone to keypoint localization errors.

5.3.7 Occlusion and Clutter

The normalized AUC_{pr} results with respect to different levels of occlusion and clutter are shown in Fig. 7g and h. It shows that the RoPS descriptor is highly sensitive to both occlusion and clutter, its normalized AUC_{pr} value drops from 1.0 to about 0.1 when occlusion is increased from 5 to 10 %. Besides, SHOT, PFH, FPFH, and TriSI are also vulnerable to occlusion and clutter. In contrast, 3DSC, SI, and USC are just robust to occlusion; while THRIFT, 3DSC, SI, and LSP are robust to clutter. It is clear that the descriptors with a higher distinctive power are more sensitive to occlusion and clutter compared to those with a lower distinctive power. That is because the former includes more details of the local surface, which makes it more vulnerable to changes of that surface under occlusion and clutter. The absolute AUC_{pr} values of all of the selected descriptors are very low when occlusion (or clutter) is more than 10 %. It can therefore be concluded that occlusion and clutter are two major factors for the performance deterioration of existing 3D local feature descriptors. Note that, the selected feature descriptors are more sensitive to clutter compared to occlusion. That is because the presence of clutter introduces not only missing data, but also additional information to the local surface which does not belong to the object.

5.3.8 Robustness Overall Performance

In order to comprehensively analyze the robustness of the selected feature descriptors with respect to different nuisances, the most robust and sensitive descriptors in each case are listed in Table 4. The following conclusions can be drawn:

First, USC and TriSI are the most robust descriptors. USC is robust with respect to several nuisances including Gaussian noise, shot noise, and distance to the mesh boundary. TriSI is robust to Gaussian noise, shot noise, and varying mesh

Table 4 The most robust and sensitive 3D local feature descriptors with respect to different nuisances

	Most robust descriptors	Most sensitive descriptors
Gaussian noise	SHOT, USC, TriSI, RoPS	LSP, THRIFT, FPFP, PFH
Shot noise	TriSI, USC, SHOT	FPFP, RoPS, PFH
Varying mesh resolutions	PFH, FPFH, TriSI, SI	3DSC, USC
Distance to the mesh boundary	SI, 3DSC, USC	RoPS, PFH, SHOT, FPFP
Keypoint localization error	PFH, FPFH	TriSI, SI, 3DSC
Occlusion	3DSC, SI, USC	RoPS, PFH, SHOT, FPFH, TriSI
Clutter	THRIFT, 3DSC, SI, LSP	RoPS, PFH, SHOT, FPFH, TriSI

resolutions. In contrast, PFH and FPFH are the most sensitive descriptors. They are vulnerable to Gaussian noise, shot noise, and distance to the mesh boundary.

Second, RoPS and TriSI achieve a more balanced performance considering both descriptiveness and robustness (see Tables 3 and 4). Specifically, RoPS achieves the highest descriptiveness score compared to the other descriptors (Tables 3).

5.4 Scalability

Figures 8a and b show the AUC_{pr} and the normalized AUC_{pr} results with respect to varying numbers of models in the dataset. It can be seen from Fig. 8a that RoPS achieves the best AUC_{pr} results, followed by TriSI. The superiority of the RoPS descriptor is highly significant when tested on a small dataset (e.g., with a number of models that is less than 20). When the number of models increases, the difference in performance between RoPS and TriSI becomes smaller. Regarding the scalability, it is clear from Fig. 8b that TriSI, USC, and 3DSC achieve a very stable performance with respect to varying numbers of models. Their normalized AUC_{pr} results remain almost the same when the number of models increases. SI and RoPS achieve an acceptable scalability with respect to varying numbers of models. In contrast, the performance of THRIFT drops dramatically when the number of models increases. The AUC_{pr} value of THRIFT tested on a dataset with 10 models is lower than 40 % of its corresponding AUC_{pr} value when tested on a dataset with 4 models. The scalability of FPFH, LSP and PFH is also very low. Note that, the ranking of the selected descriptors remains unchanged when tested on different datasets with varying numbers of models (see Fig. 8a). Therefore, the most appropriate feature descriptor for a particular application can be selected based on a pilot test on a small-size dataset.

5.5 Combination with 3D Keypoint Detectors

In order to demonstrate the influence of a keypoint detector on the performance of feature descriptors, the PRC results of these descriptors combined with four different keypoint detectors (i.e., Uniform Sampling, Harris3D, ISS, ISS-BR) are shown in Fig. 9a–d, respectively. Several observations can be drawn from these figures.

First, the overall performance of these descriptors combined with uniform sampling is the lowest compared to those achieved when combined with Harris3D, ISS and ISS-BR. This is reasonable since uniform sampling does not consider the geometric characteristics of the underlying local surface while performing keypoint detection. Consequently, the repeatability of the resulting keypoints is relatively low, and the feature matching performance is decreased. This also reveals the fact that adopting an appropriate keypoint detection method can boost the performance of feature matching (Alexandre 2012). *Second*, the overall performance of these

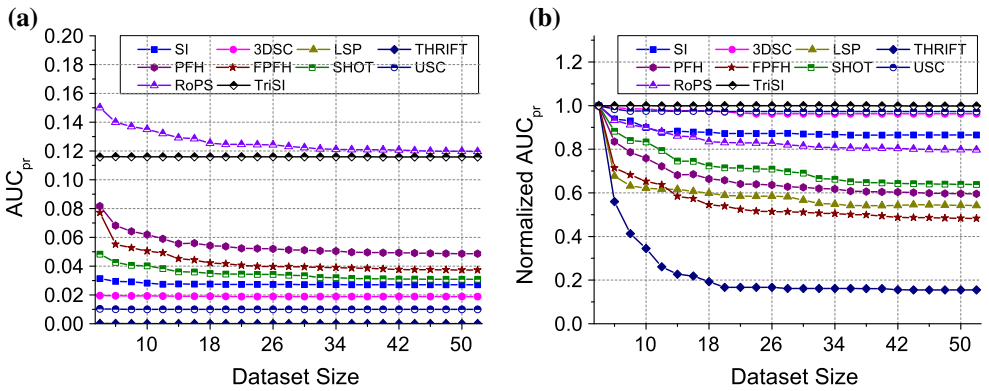

Fig. 8 The scalability of the selected descriptors. **a** AUC_{pr} **b** normalized AUC_{pr} (Color figure online)

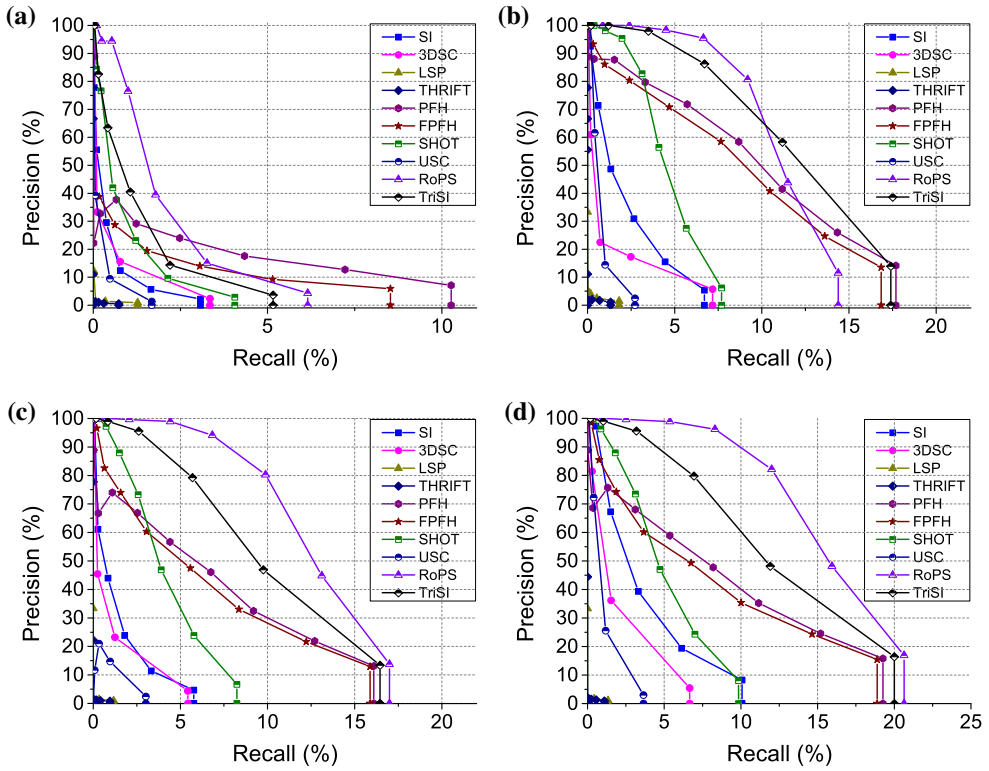

Fig. 9 Performance of the selected descriptors of Sect. 3 on the Laser scanner dataset combined with different 3D keypoint detectors. **a** Uniform Sampling **b** Harris3D **c** ISS **d** ISS-BR (Color figure online)

descriptors combined with Harris3D and ISS is comparable. That is due to the reason that these two detectors achieve similar keypoint detection performance. From Fig. 9c and d, it can further be noticed that the performance of these descriptors combined with ISS-BR is better compared to the combination with ISS. This observation is in line with the conclusion drawn in Sect. 5.3.5. That is, the performance of most descriptors can be improved by eliminating keypoints which are close to the mesh boundary. *Third*, the rank of these descriptors remains almost the same when combined with Harris3D, ISS and ISS-BR detectors. The only exception happens between TriSI and RoPS. TriSI achieved the best performance compared to the other descriptors when combined with Harris3D detector, while RoPS is the best descriptor when combined with ISS and ISS-BR detectors. The next best performing descriptors are PFH and FPFH. The score of SHOT, 3DSC, SI and USC is relatively low, while LSP and THRIFT demonstrate insufficient performance.

In order to further investigate the influence of 3D keypoint detectors, we first randomly extracted 1000 feature points from each scene without keypoint detection. We then generated their corresponding feature points in the models using known rigid transformations between the scene and models. The scene features were then matched against the model features to generate a PRC curve for each selected descriptor (see Sect. 4.3.2 for more details). Note that, the

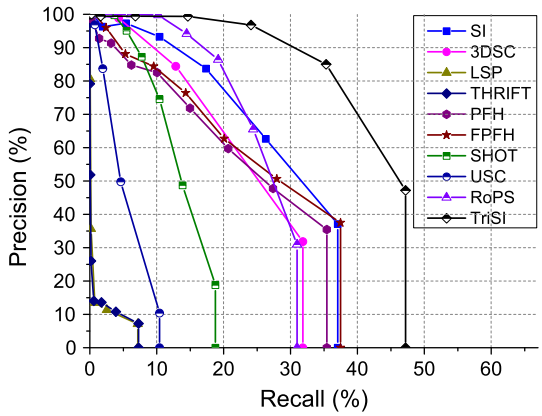

Fig. 10 Descriptiveness of the selected descriptors of Sect. 3 on the Laser scanner dataset with ground-truth keypoints (Color figure online)

corresponding feature points in the scene and the models were detected from exactly the same physical positions (without any keypoint localization error). The PRC results are shown in Fig. 10. It is clear that the performance of all these descriptors with extracted keypoints using the methods in Rusu and Cousins (2011); Sipiran and Bustos (2011); Zhong (2009) (as shown in Fig. 9) is lower compared to their performance with ground-truth corresponding feature points (as shown in Fig. 10). This is because keypoint localization errors significantly decrease the score of the feature matching. Note

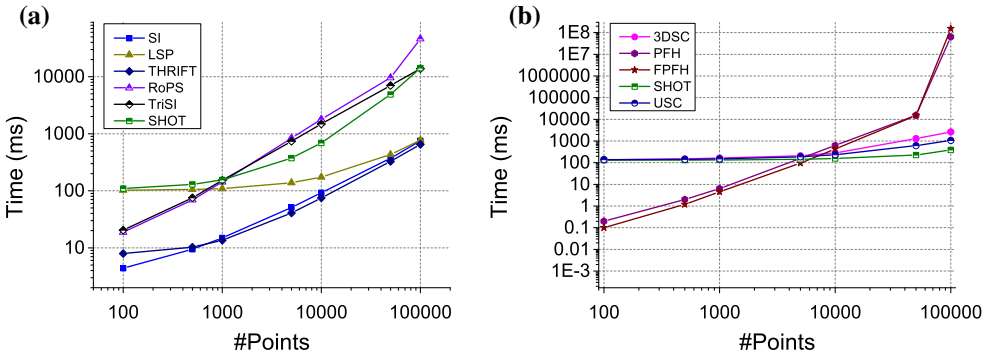

Fig. 11 Computational time required to generate a feature descriptor for a local surface with varying number of points in the support region. **a** Descriptors implemented in MATLAB **b** descriptors implemented in PCL (Color figure online)

that, the performance deterioration of TriSI, SI, and 3DSC is more significant compared to the other descriptors. This is because TriSI, SI, and 3DSC are highly sensitive to the accuracy of the keypoint localization (as shown in Sect. 5.3.6 and Table 4).

5.6 Efficiency

Figure 11a and b show the computational time required to generate various feature descriptors implemented in MATLAB 2011b and PCL Version 1.7.1, respectively. In order to make the results in Fig. 11a and b comparable, the SHOT descriptor was implemented in both MATLAB and PCL, with its corresponding computational time plotted in both Fig. 11a and b.

For the descriptors implemented in MATLAB (Fig. 11a), the most efficient descriptors are SI and THRIFT. Their computational performance are comparable. For local surfaces with less than 1000 points, SHOT and LSP are the most computationally intensive descriptors, being one order of magnitude slower compared to SI and THRIFT. As the number of points in a local surface increases, RoPS, TriSI and SHOT become the slowest methods. Their computational performance is similar, which is one order of magnitude slower compared to SI and THRIFT.

For the descriptors implemented in PCL (Fig. 11b), FPFH and PFH achieve the best computational performance when the number of points in the support region is less than 5000. Specifically, FPFH and PFH are faster than all the other descriptors by about three orders of magnitude when the number of points in the support region is 100, the computational efficiency of FPFH and PFH is still better than the others by an order of magnitude when the number of points in the support region is 1000. However, when the number of points in the support region is more than 5000, it is very time-consuming to generate the PFH and FPFH descriptors. The most efficient descriptor in this case is SHOT. Besides, USC consistently outperforms 3DSC in terms of computational efficiency in all cases with respect to different numbers of points in the support region. It is also worth noting that, FPFH outperforms PFH when the number of points in the support region is less than 50,000. When the number of points in the support region is further increased, PFH performs better than FPFH in terms of computational efficiency.

Taking the computational time of SHOT implemented in both MATLAB and PCL as a benchmark, several observations can be made for all these descriptors. *First*, FPFH, PFH, THRIFT, and SI are among the most efficient descriptors when the number of points in the support region is less than 1000. In contrast, 3DSC and USC are the most computationally expensive descriptors. *Second*, when the number of points in the support region is between 1000 and 5000, the most efficient descriptors are FPFH, PFH, THRIFT, and SI, while 3DSC, USC, RoPS, and TriSI are very time-consuming. *Third*, as the number of points in the support region is more than 5000, FPFH and PFH are inferior in terms of computational efficiency compared to all the other descriptors, with SI and THRIFT being the most efficient descriptors. Overall, SI consistently achieves a very good performance while SHOT takes an average time for all cases with respect to different numbers of points in the support region. FPFH and PFH are extremely efficient to be generated for small local surfaces.

6 Summary and Discussion

In order to further provide a guidance for the selection of an appropriate 3D feature descriptor for a specific application, several points are summarized below.

For time-crucial applications (e.g., real-time systems) on point clouds with a small number of points, FPFH is the best option. That is because the FPFH descriptor is reasonably descriptive, computationally efficient (for both feature generation and matching), and lightweight (for feature storage). It provides a good balance between feature matching accuracy and computational efficiency. For time-crucial applications on point clouds with a large number of points, SHOT achieves

a good performance in terms of both descriptiveness and computational efficiency.

For space-crucial applications (e.g., embedded devices), FPFH is the best option. That is because its memory requirement for feature storage is low. RoPS can also be considered as it achieves a better descriptiveness performance at the cost of slightly higher storage requirements. For scenarios where a high registration accuracy (or recognition rate) is required, RoPS is strongly recommended due to its higher discriminative power compared to other descriptors.

If the characteristics (e.g., noise level, resolution) of a dataset are unknown, RoPS is the best option as it consistently produces good results on all kinds of datasets (see Table 3). The RoPS descriptor achieves a very stable performance across different datasets.

The feature matching performance of the selected descriptors can significantly be improved when combined with 3D keypoint detection methods (as opposed to uniform sampling or a random selection of the keypoints). ISS-BR consistently achieved the best performance when combined with these selected descriptors.

TriSI, USC, and 3DSC have the best scalability with respect to an increasing number of models in the dataset. However, the descriptiveness of the USC and 3DSC descriptors is relatively low and their performance variations across different datasets are significant. Moreover, both the computational and storage costs of USC and 3DSC are high (with a dimensionality of the descriptor of 1980). Consequently, TriSI is the best choice for applications on large datasets.

Note that, although these descriptors perform well with high resolution datasets (collected using expensive scanners), their performance is rather weak with data from low-cost low-resolution sensors (e.g., *Kinect* and *Dense Stereo*). Research should therefore be directed towards the design of suitable descriptors for low resolution and high-level noise data, or the design of higher resolution and low-cost RGBD cameras.

7 Conclusions

This paper has presented a comprehensive evaluation of 3D local feature descriptors on a variety of datasets. The descriptiveness of these descriptors was evaluated on eight datasets for different application contexts (i.e., 3D object recognition, 3D shape retrieval, and 3D modeling). The robustness of the selected descriptors was tested with respect to a set of nuisances (including Gaussian noise, shot noise, varying mesh resolutions, mesh boundary, keypoint localization errors, occlusion and clutter). The compactness and scalability of these descriptors were also presented. Next, these descriptors were tested with the combination of different 3D keypoint detectors. Finally, the computation efficiency of these descriptors were analyzed. This paper can therefore, serve as a "User Guide" for the selection of the most appropriate feature descriptor in the area of 3D computer vision.

Acknowledgments This research is supported by a National Natural Science Foundation of China (NSFC) fund (No. 61471371), a China Scholarship Council (CSC) scholarship and Australian Research Council Grants (DE120102960, DP110102166, DP150100294).

References

Aldoma, A., Marton, Z., Tombari, F., Wohlkinger, W., Potthast, C., Zeisl, B., et al. (2012a). Tutorial: Point cloud library: Three-dimensional object recognition and 6 DOF pose estimation. *IEEE Robotics & Automation Magazine*, *19*(3), 80–91.

Aldoma, A., Tombari, F., Di Stefano, L., & Vincze, M. (2012b). A global hypotheses verification method for 3D object recognition. In *European Conference on Computer Vision*, (pp 511–524).

Alexandre, L.A. (2012). 3D descriptors for object and category recognition: A comparative evaluation. In *Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

Assfalg, J., Bertini, M., Bimbo, A., & Pala, P. (2007). Content-based retrieval of 3-D objects using spin image signatures. *IEEE Transactions on Multimedia*, *9*(3), 589–599.

Bariya, P., Novatnack, J., Schwartz, G., & Nishino, K. (2012). 3D geometric scale variability in range images: Features and descriptors. *International Journal of Computer Vision*, *99*(2), 232–255.

Bayramoglu, N., & Alatan, A. (2010). Shape index SIFT: Range image recognition using local features. In *20th International Conference on Pattern Recognition*, (pp. 352–355).

Bennamoun, M., Guo, Y., & Sohel, F. (2015). Feature selection for 2D and 3D face recognition, In *Encyclopedia of electrical and electronics engineering*. Book Chapter (pp. 1–54).

Besl, P. J., & McKay, N. D. (1992). A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *14*(2), 239–256.

Boyer, E., Bronstein, A., & Bronstein, M., et al. (2011). SHREC 2011: Robust feature detection and description benchmark. In *Eurographics Workshop on Shape Retrieval*, (pp. 79–86).

Bronstein, A., Bronstein, M., & Bustos, B., et al. (2010). SHREC 2010: Robust feature detection and description benchmark. In *Eurographics Workshop on 3D Object Retrieval*, vol 2, p 6.

Bronstein, A., Bronstein, M., Guibas, L., & Ovsjanikov, M. (2011). Shape google: Geometric words and expressions for invariant shape retrieval. *ACM Transactions on Graphics*, *30*(1), 1–20.

Burghouts, G. J., & Geusebroek, J. M. (2009). Performance evaluation of local colour invariants. *Computer Vision and Image Understanding*, *113*(1), 48–62.

Chen, H., & Bhanu, B. (2007a). 3D free-form object recognition in range images using local surface patches. *Pattern Recognition Letters*, *28*(10), 1252–1262.

Chen, H., & Bhanu, B. (2007b). Human ear recognition in 3D. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(4), 718–737.

Chen, X., & Schmitt, F. (1992). Intrinsic surface properties from surface triangulation. In *European Conference on Computer Vision*, (pp. 739–743).

Curless, B., & Levoy, M. (1996). A volumetric method for building complex models from range images. In *23rd Annual Conference on Computer Graphics and Interactive Techniques*, (pp. 303–312).

Darom, T., & Keller, Y. (2012). Scale invariant features for 3D mesh models. *IEEE Transactions on Image Processing*, *21*(5), 2758–2769.

Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and roc curves. In *23rd International Conference on Machine learning*, (pp. 233–240).

Dinh, H., & Kropac, S. (2006). Multi-resolution spin-images. *IEEE International Conference on Computer Vision and Pattern Recognition*, *1*, 863–870.

Filipe, S., & Alexandre, L.A. (2014). A comparative evaluation of 3D keypoint detectors in a RGB-D object dataset. In *9th International Conference on Computer Vision Theory and Applications*, (pp. 1–8).

Flint, A., Dick, A., & Hengel, A. (2007). THRIFT: Local 3D structure recognition. In *9th Conference on Digital Image Computing Techniques and Applications*, (pp. 182–188).

Flint, A., Dick, A., & Van den Hengel, A. (2008). Local 3D structure recognition in range images. *IET Computer Vision*, *2*(4), 208–217.

Frome, A., Huber, D., Kolluri, R., Bülow, T., & Malik, J. (2004). Recognizing objects in range data using regional point descriptors. In *8th European Conference on Computer Vision*, (pp. 224–237).

Gao, Y., & Dai, Q. (2014). View-based 3-D object retrieval: Challenges and approaches. *IEEE Multimedia*, *21*(3), 52–57.

Guo, Y., Bennamoun, M., Sohel, F., Wan, J., & Lu, M. (2013a). 3D free form object recognition using rotational projection statistics. In *IEEE 14th Workshop on the Applications of Computer Vision*, (pp. 1–8).

Guo, Y., Sohel, F., Bennamoun, M., Lu, M., & Wan, J. (2013b). Rotational projection statistics for 3D local surface description and object recognition. *International Journal of Computer Vision*, *105*(1), 63–86.

Guo, Y., Sohel, F., Bennamoun, M., Lu, M., Wan, J. (2013c). TriSI: A distinctive local surface descriptor for 3D modeling and object recognition. In *8th International Conference on Computer Graphics Theory and Applications*, (pp. 86–93).

Guo, Y., Bennamoun, M., Sohel, F., Lu, M., & Wan, J. (2014a). 3D object recognition in cluttered scenes with local surface features: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(11), 2270–2287.

Guo, Y., Bennamoun, M., Sohel, F., Lu, M., Wan, J., & Zhang, J. (2014b). Performance evaluation of 3D local feature descriptors. In *12th Asian Conference on Computer Vision*, (pp. 1–17).

Guo, Y., Sohel, F., Bennamoun, M., Wan, J., & Lu, M. (2014c). An accurate and robust range image registration algorithm for 3D object modeling. *IEEE Transactions on Multimedia*, *16*(5), 1377–1390.

Guo, Y., Zhang, J., Lu, M., Wan, J., & Ma, Y. (2014d). Benchmark datasets for 3D computer vision. In *The 9th IEEE Conference on Industrial Electronics and Applications*.

Guo, Y., Sohel, F., Bennamoun, M., Wan, J., & Lu, M. (2015). A novel local surface feature for 3D object recognition under clutter and occlusion. *Information Sciences*, *293*(2), 196–213.

Johnson, A. E., & Hebert, M. (1998). Surface matching for object recognition in complex three-dimensional scenes. *Image and Vision Computing*, *16*(9–10), 635–651.

Johnson, A. E., & Hebert, M. (1999). Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *21*(5), 433–449.

Ke, Y., & Sukthankar, R. (2004). PCA-SIFT: A more distinctive representation for local image descriptors. *IEEE Conference on Computer Vision and Pattern Recognition*, *2*, 498–506.

Kim, H., & Hilton, A. (2013). Evaluation of 3D feature descriptors for multi-modal data registration. In *International Conference on 3D Vision*, (pp. 119–126).

Koenderink, J., & van Doorn, A. (1992). Surface shape and curvature scales. *Image and Vision Computing*, *10*(8), 557–564.

Lai, K., Bo, L., Ren, X., & Fox, D. (2011). A scalable tree-based approach for joint object and pose recognition. In *25th Conference on Artificial Intelligence*.

Lei, Y., Bennamoun, M., Hayat, M., & Guo, Y. (2014). An efficient 3D face recognition approach using local geometrical signatures. *Pattern Recognition*, *47*(2), 509–524.

Lo, T., & Siebert, J. (2009). Local feature extraction and matching on range images: 2.5D SIFT. *Computer Vision and Image Understanding*, *113*(12), 1235–1250.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.

Matei, B., Shan, Y., Sawhney, H., Tan, Y., Kumar, R., Huber, D., et al. (2006). Rapid object indexing using locality sensitive hashing and joint 3D-signature space estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(7), 1111–1126.

Meek, D. S., & Walton, D. J. (2000). On surface normal and gaussian curvature approximations given data sampled from a smooth surface. *Computer Aided Geometric Design*, *17*(6), 521–543.

Mian, A., Bennamoun, M., & Owens, R. (2006a). Three-dimensional model-based object recognition and segmentation in cluttered scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(10), 1584–1601.

Mian, A., Bennamoun, M., & Owens, R. A. (2006b). A novel representation and feature matching algorithm for automatic pairwise registration of range images. *International Journal of Computer Vision*, *66*(1), 19–40.

Mian, A., Bennamoun, M., & Owens, R. (2010). On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. *International Journal of Computer Vision*, *89*(2), 348–361.

Mikolajczyk, K., & Schmid, C. (2003). A performance evaluation of local descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition*, vol 2, (pp. II-257).

Mikolajczyk, K., & Schmid, C. (2004). Scale & affine invariant interest point detectors. *International Journal of Computer Vision*, *60*(1), 63–86.

Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(10), 1615–1630.

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., et al. (2005). A comparison of affine region detectors. *International Journal of Computer Vision*, *65*(1), 43–72.

Moreels, P., & Perona, P. (2005). Evaluation of features detectors and descriptors based on 3D objects. In *10th IEEE International Conference on Computer Vision*, vol 1, (pp. 800–807).

Moreels, P., & Perona, P. (2007). Evaluation of features detectors and descriptors based on 3D objects. *International Journal of Computer Vision*, *73*(3), 263–284.

Restrepo, M.I., & Mundy, J.L. (2012). An evaluation of local shape descriptors in probabilistic volumetric scenes. In *British Machine Vision Conference*, (pp. 1–11).

Rodolà, E., Albarelli, A., Bergamasco, F., & Torsello, A. (2013). A scale independent selection process for 3D object recognition in cluttered scenes. In *International Journal of Computer Vision* pp 1–17.

Ruiz-Correa, S., Shapiro, L., & Melia, M. (2001). A new signature-based method for efficient 3-D object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, vol 1, (pp. I-769).

Rusu, R.B., & Cousins, S. (2011). 3D is here: Point cloud library (PCL). In *IEEE International Conference on Robotics and Automation*, pp 1–4.

Rusu, R.B., Blodow, N., Marton, Z.C., & Beetz, M. (2008). Aligning point cloud views using persistent feature histograms. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (pp. 3384–3391).

Rusu, R.B., Blodow, N., & Beetz, M. (2009). Fast point feature histograms (FPFH) for 3D registration. In *IEEE International Conference on Robotics and Automation*, (pp. 3212–3217).

Salti, S., Tombari, F., & Stefano, L. (2011). A performance evaluation of 3D keypoint detectors. In *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission* (pp. 236–243).

Salti, S., Petrelli, A., Tombari, F., & Di Stefano, L. (2012). On the affinity between 3D detectors and descriptors. In *2nd International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, (pp. 424–431).

Salti, S., Tombari, F., & Stefano, L. D. (2014). SHOT: Unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding*, *125*(8), 251–264.

Schmid, C., Mohr, R., & Bauckhage, C. (2000). Evaluation of interest point detectors. *International Journal of Computer Vision*, *37*(2), 151–172.

Shang, L., & Greenspan, M. (2010). Real-time object recognition in sparse range images using error surface embedding. *International Journal of Computer Vision*, *89*(2), 211–228.

Sipiran, I., & Bustos, B. (2011). Harris 3D: a robust extension of the harris operator for interest point detection on 3D meshes. *The Visual Computer* pp. 1–14.

Sukno, F.M., Waddington, J.L., & Whelan, P.F. (2013). Rotationally invariant 3D shape contexts using asymmetry patterns. In *8th International Conference on Computer Graphics Theory and Applications*.

Sun, J., Ovsjanikov, M., & Guibas, L. (2009). A concise and provably informative multi-scale signature based on heat diffusion. *Computer Graphics Forum*, *28*, 1383–1392.

Taati, B., & Greenspan, M. (2011). Local shape descriptor selection for object recognition in range data. *Computer Vision and Image Understanding*, *115*(5), 681–694.

Tangelder, J., Veltkamp, R. (2004). A survey of content based 3D shape retrieval methods. In *IEEE International Conference on Shape Modeling and Applications*, (pp. 145–156).

Tombari, F., Salti, S., & Di Stefano, L. (2010a), Unique shape context for 3D data description. In *ACM Workshop on 3D Object Retrieval*, (pp. 57–62).

Tombari, F., Salti, S., & Di Stefano, L. (2010b). Unique signatures of histograms for local surface description. In *European Conference on Computer Vision*, Springer, New York, (pp. 356–369).

Tombari, F., Salti, S., & Di Stefano, L. (2013). Performance evaluation of 3D keypoint detectors. *International Journal of Computer Vision*, *102*(1), 198–220.

Zaharescu, A., Boyer, E., Varanasi, K., & Horaud, R. (2009). Surface feature detection and description with applications to mesh matching. In *IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 373–380).

Zaharescu, A., Boyer, E., & Horaud, R. (2012). Keypoints and local descriptors of scalar functions on 2D manifolds. *International Journal of Computer Vision*, *100*, 78–98.

Zhong, Y. (2009). Intrinsic shape signatures: A shape descriptor for 3D object recognition. In *IEEE International Conference on Computer Vision Workshops*, (pp. 689–696).