Int J Comput Vis (2013) 102:198–220
DOI 10.1007/s11263-012-0545-4

Performance Evaluation of 3D Keypoint Detectors

Federico Tombari · Samuele Salti · Luigi Di Stefano

Received: 5 November 2011 / Accepted: 12 June 2012 / Published online: 20 July 2012

Abstract In the past few years detection of repeatable and distinctive keypoints on 3D surfaces has been the focus of intense research activity, due on the one hand to the increasing diffusion of low-cost 3D sensors, on the other to the growing importance of applications such as 3D shape retrieval and 3D object recognition. This work aims at contributing to the maturity of this field by a thorough evaluation of several recent 3D keypoint detectors. A categorization of existing methods in two classes, that allows for highlighting their common traits, is proposed, so as to abstract all algorithms to two general structures. Moreover, a comprehensive experimental evaluation is carried out in terms of repeatability, distinctiveness and computational efficiency, based on a vast data corpus characterized by nuisances such as noise, clutter, occlusions and viewpoint changes.

Keywords 3D detectors · Performance evaluation · 3D object recognition · 3D shape retrieval

1 Introduction

Recognition of similarities among 3D shapes represents one of the most challenging tasks in surface analysis applications. Research efforts aimed at improving the capabilities of 3D shape recognition algorithms are nowadays gaining momentum thanks to the increasing availability of low-cost, dense 3D sensors, such as stereo and Time-of-Flight cameras, as well as structured light sensors (e.g. the recent *Kinect* device by Microsoft).

The main trend for establishing similarities among surfaces relies on computing correspondences between 3D features (Chua and Jarvis 1997; Johnson and Hebert 1999; Frome et al. 2004; Novatnack and Nishino 2008; Zaharescu et al. 2009; Tombari et al. 2010; Bronstein and Kokkinos 2010). Feature-based surface matching is now one of the standard paradigm in applications such as 3D object recognition (Johnson and Hebert 1999; Mian et al. 2010; Zhong 2009), 3D reconstruction (Novatnack and Nishino 2008), 3D shape retrieval (Boyer et al. 2011), and 3D object categorization (Salti et al. 2010; Knopp et al. 2010). An alternative approach, generally adopted in the absence of clutter and occlusions, relies instead on describing the whole surface by means of a single descriptor (Shang and Greenspan 2010; Akgül et al. 1992; Shilane et al. 2004).

Feature-based matching relies on two steps: feature detection and feature description. The first step identifies *3D keypoints*, i.e. points of the shape that are prominent according to a particular definition of interestingness or saliency. 3D keypoints are extracted from surfaces by a 3D *detector*, which analyses local neighborhoods around the elements of the given surface in order to identify such interest points. Then, the neighborhood of a keypoint is described by a 3D *descriptor*, which projects the neighborhood into a proper feature space. Finally, descriptors computed on different surfaces are matched together, the most reliable matches yielding point-to-point 3D correspondences.

A detector should extract repeatable keypoints under a number of nuisances that can affect the input data, e.g. viewpoint changes, missing parts, point density or topology variations, clutter, sensor noise. A 3D detector may provide the ability of associating to each extracted keypoint a character-

F. Tombari (✉) · S. Salti · L. Di Stefano
Viale Risorgimento, 2, 40135 Bologna, Italy
e-mail: federico.tombari@unibo.it

S. Salti
e-mail: samuele.salti@unibo.it

L. Di Stefano
e-mail: luigi.distefano@unibo.it

istic *scale*, or support size, which should be repeatable under the aforementioned nuisances, and that can be provided to the following description stage to identify the neighborhood of points on which the local feature ought to be computed. It is worth pointing out that the definition of a proper and repeatable scale in 3D data differs from the scale-invariance notion of 2D features. In the 2D domain, scale invariance is motivated by the requirement to detect image structures invariantly to the possible size changes that occur when the 3D space is projected onto a 2D plane. However, 3D sensors typically provide metric data, so that the association of a characteristic scale to a keypoint serves primarily to obtain highly distinctive features, in particular by allowing for the description of the most salient neighborhood around a candidate keypoint, as well as to prevent missing important features of an object by measuring saliency based on a given support size only. Accordingly, we prefer to denote 3D detectors that associate a characteristic scale to a keypoint as *adaptive-scale* rather than scale invariant detectors.

Traditionally, distinctiveness and repeatability have been regarded as the main traits of a 3D detector: the former is the ability to detect keypoints that can be effectively described and matched, so as to possibly prevent wrong point-to-point correspondences. The latter, instead, deals with the capability to detect the same keypoints accurately under various nuisances. In our analysis and evaluation we deliberately focus on repeatability, the reason being twofold: as the same 3D structure can be represented more or less effectively by different descriptors, distinctiveness can be measured only in combination with a specific description algorithm, which renders the results of such experiments and their analysis of limited generality; distinctiveness of a local shape element is, indeed, a rather global property of a scene or model, which is therefore hard to capture by a local algorithm, such as a feature detector. To help visualize the latter issue, think of an image of a chessboard or a surface made out of identical, equally spaced bumps: clearly a lot of repeatable keypoints can be identified in both examples, but none of them can be told easily apart by looking at a local area only.

A notable scientific fervor has recently characterized the field of 3D detectors, leading to several relevant proposals (Mian et al. 2010; Chen and Bhanu 2007; Zhong 2009; Novatnack and Nishino 2008; Unnikrishnan and Hebert 2008; Akagunduz and Ulusoy 2007; Zaharescu et al. 2009; Fadaifard and Wolberg 2011; Knopp et al. 2010; Castellani et al. 2008; Sun et al. 2009). Compared to its 2D counterpart, and to 3D descriptors as well, research on 3D detectors has started much more recently (mainly in the past 3 or 4 years) and, to date, only limited taxonomic and evaluation work has been carried out, with experimental comparison usually performed separately within (and relatively to) each specific proposal. The only relevant works in this respect are described in Bronstein et al. (2010), Boyer et al. (2011), which propose an experimental evaluation of 3D detectors and descriptors focused on the 3D shape retrieval scenario, and a preliminary version of this paper, which was presented in Salti et al. (2011). A parallel line of research has targeted the evaluation of 2D detectors and descriptors on 3D objects (Moreels and Perona 2007).

Given the wealth of recent proposals concerning 3D detectors, we believe there is a lack in the literature of a survey aimed at reviewing the state of the art and comparing quantitatively the different approaches within a common and well-defined experimental framework. Hence, this paper proposes a comparison of state-of-the-art 3D detectors, which is grounded on the established methodology adopted in the related field of 2D detectors (Schmid et al. 2000; Mikolajczyk et al. 2005) and mainly focused on the 3D object recognition scenario, which is peculiarly characterized by the presence of occlusions and clutter. Such a scenario differs from that addressed by Bronstein et al. (2010), Boyer et al. (2011), as 3D shape retrieval is not required to deal with occlusion, clutter and viewpoint changes, large intraclass shape variations being instead the main nuisance to be dealt with. Moreover, the proposed framework evaluates the robustness of state-of-the-art methods with respect to noise (both real and synthetic) and addresses computational efficiency. Nonetheless, we also propose some basic retrieval experiments, to highlight how the absolute performance of detectors, and the ranking of their performance, are influenced by the specific application scenario. Therefore, this paper and (Bronstein et al. 2010; Boyer et al. 2011) usefully provide complementary perspectives on the topic of quantitative evaluation of 3D local feature detectors. All datasets and experimental results related to this evaluation have been made publicly available through the web page of this project.

The paper is structured as follows. Section 2 presents the state of the art by dividing proposals into two categories (*fixed-scale*, *adaptive-scale*) and identifying the main computational steps within each category, so as to describe then how these steps are carried out by each method. Section 3 discusses the evaluation methodology, which comprises the datasets, the performance metrics and the selected algorithms. Finally, Sect. 4 reports and discusses experimental results, while Sect. 5 draws conclusions.

2 3D Detectors

This section briefly reviews state-of-the-art methods for detection of 3D keypoints, with Table 1 reporting the adopted notation. 3D detectors are divided into two categories, namely *fixed-scale* and *adaptive-scale* detectors. Their general structure is depicted in Figs. 1 and 4, respectively.

Table 1 Notation

Generic mesh point	$\mathbf{p}$
Set of points in the support of $\mathbf{p}$	$\mathcal{N}(\mathbf{p})$
Number of points in the support of $\mathbf{p}$	$N = \lvert\mathcal{N}(\mathbf{p})\rvert$
Set of points in the Non-Maxima-Suppresion support of $\mathbf{p}$	$\mathcal{N}_{NMS}(\mathbf{p})$
Normal to the mesh in $\mathbf{p}$	$\mathbf{n}(\mathbf{p})$
Maximum curvature of the mesh in $\mathbf{p}$	$C_M(\mathbf{p})$
Minimum curvature of the mesh in $\mathbf{p}$	$C_m(\mathbf{p})$
Gaussian curvature of the mesh in $\mathbf{p}$	$C_K(\mathbf{p})$
Mean curvature of the mesh in $\mathbf{p}$	$C_H(\mathbf{p})$
Saliency of point $\mathbf{p}$	$\rho(\mathbf{p})$
Saliency of point $\mathbf{p}$ at scale t	$\rho(\mathbf{p}, t)$
Shape index of point $\mathbf{p}$	$SI(\mathbf{p})$
Scatter matrix of the support of point $\mathbf{p}$	$\Sigma(\mathbf{p})$
Set of keypoints of a mesh	$\mathcal{K}$
Gaussian kernel with standard deviation equal to σ	$G(\sigma)$
Minimum number of edges between two points of a mesh	$e(\mathbf{p}, \mathbf{q})$
Geodesic distance between two points of a mesh	$\gamma(\mathbf{p}, \mathbf{q})$

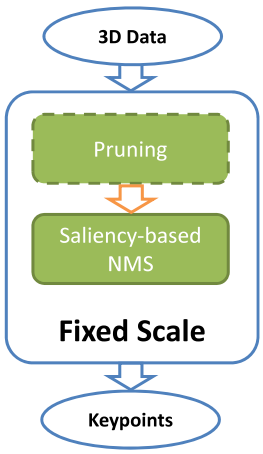


Fig. 1 General structure of a fixed-scale 3D keypoint detector (the *dashed contour line* indicates an optional block)

A fundamental step in common to both categories is the selection of keypoints as local extrema of a *saliency* measurement, whose definition determines the robustness and repeatability of the detector, for it identifies the kind of 3D structures selected by the detector.

2.1 Fixed-Scale Detectors

Fixed-scale detectors find distinctive keypoints at a specific, constant scale, which is provided as a parameter to the algorithm. As sketched in Fig. 1, these detectors can be abstracted into two main steps. The main purpose of the initial, optional, step is pruning the input data by thresholding a quality measure computed at each point. This is done, on the one hand, to reinforce keypoint selection by an additional criterion with respect to the main saliency measure deployed in the subsequent step, and on the other hand to improve the efficiency of the algorithm by reducing the number of points provided as inputs to the next step. The second step consists in a Non-Maxima Suppression (NMS) procedure based upon a saliency measure computed at each point not discarded by the initial pruning. The saliency measurement associated with each point can be either point-wise (i.e. a property of a vertex of the mesh) or region-wise (i.e. a property of a region around each vertex). In case of point-wise saliency, the input scale is used to define the size of the NMS support. With region-wise saliency, the scale usually defines the support on which the saliency measurement relies upon, whereas the NMS support is defined through an additional parameter.

2.1.1 *Local Surface Patches (LSP)*

One example of the approach relying on point-wise saliency measurements is *Local Surface Patches* (LSP), introduced in Chen and Bhanu (2007). It measures the saliency of a vertex according to its Shape Index (*SI*), as defined by Dorai and Jain (1997), which, in turn, is based on the maximum and minimum principal curvatures at the vertex,

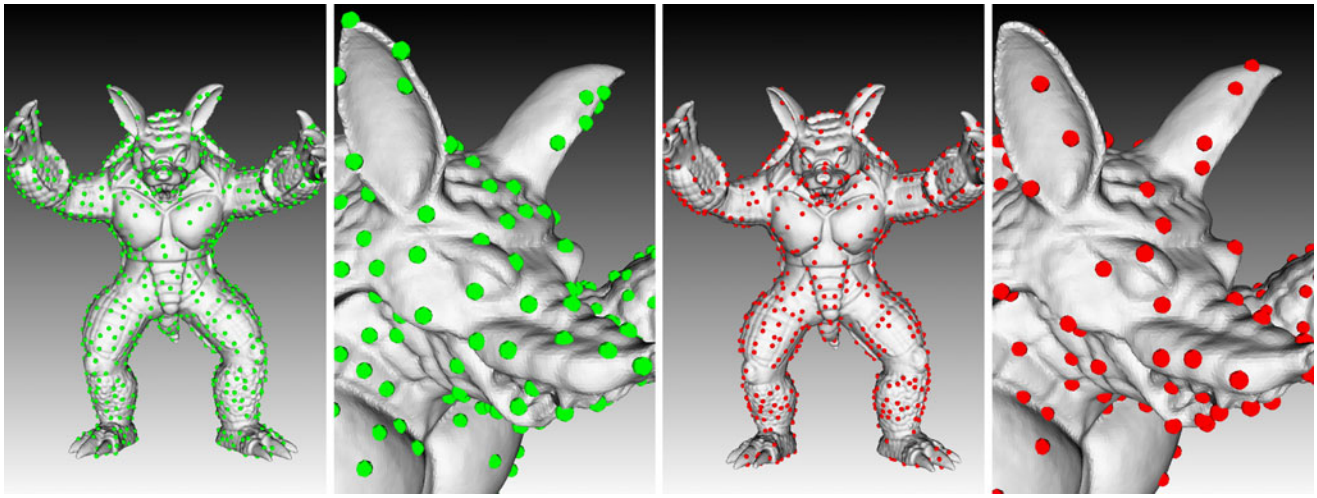
$$SI(\mathbf{p}) = \frac{1}{2} - \frac{1}{\pi} \tan^{-1} \frac{C_M(\mathbf{p}) + C_m(\mathbf{p})}{C_M(\mathbf{p}) - C_m(\mathbf{p})}. \tag{1}$$

Let μ_{SI} be the mean Shape Index within the given support,

$$\mu_{SI}(\mathbf{p}) = \frac{1}{N} \sum_{\mathbf{q} \in \mathcal{N}(\mathbf{p})} SI(\mathbf{q}), \tag{2}$$

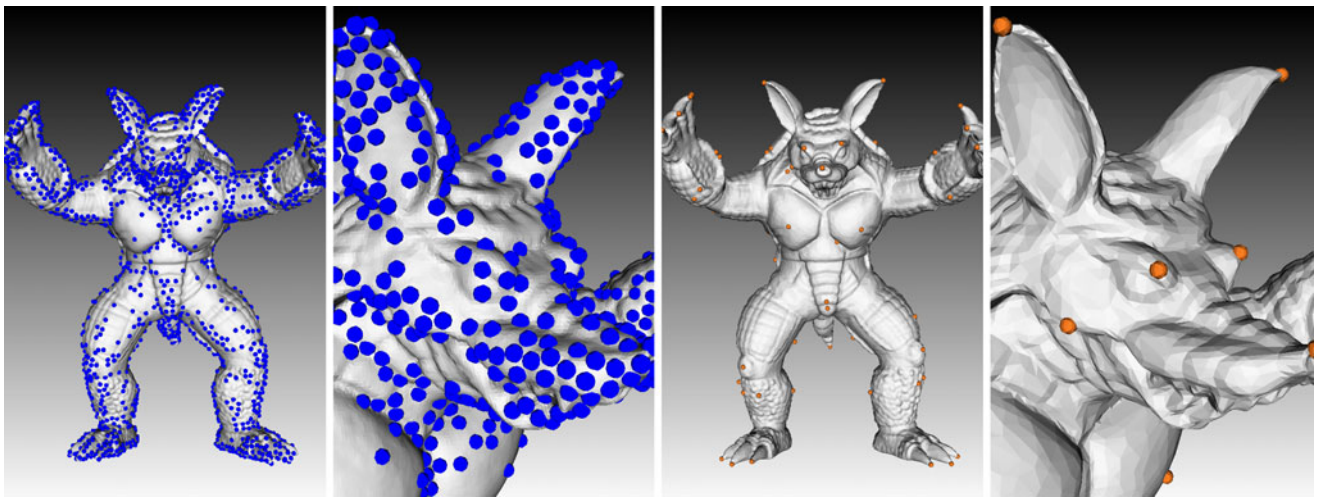
then a vertex survives the pruning step if its Shape Index is significantly greater or smaller than μ_{SI}, i.e.

$$SI(\mathbf{p}) \geq (1 + \alpha)\mu_{SI}(\mathbf{p}) \vee SI(\mathbf{p}) \leq (1 - \beta)\mu_{SI}(\mathbf{p}) \tag{3}$$



(a) LSP (b) LSP (c) ISS (d) ISS

Fig. 2 Example of keypoints detected by LSP and ISS on the Armadillo model of the *Retrieval* dataset



(a) KPQ (b) KPQ (c) HKS (d) HKS

Fig. 3 Example of keypoints detected by KPQ and HKS on the Armadillo model of the *Retrieval* dataset

where α and β are two scalar parameters defining the magnitude of the differences from the mean that are considered significant.

Non maxima suppression is then performed on the remaining points. As mentioned, the saliency of a vertex is its Shape Index,

$$\rho(\mathbf{p}) \doteq SI(\mathbf{p}), \tag{4}$$

both local minima and maxima are retained

$$\begin{aligned} &\rho(\mathbf{p}) > \rho(\mathbf{q}), \quad \forall \mathbf{q} \in \mathcal{N}_{NMS}(\mathbf{p}) \\ &\qquad \vee \\ &\rho(\mathbf{p}) < \rho(\mathbf{q}), \quad \forall \mathbf{q} \in \mathcal{N}_{NMS}(\mathbf{p}) \end{aligned} \tag{5}$$

and, in the original proposal and our implementation, $\mathcal{N}_{NMS}(\mathbf{p})$ coincides with $\mathcal{N}(\mathbf{p})$. Exemplar keypoints detected by LSP on a model of the Stanford dataset are shown in Figs. 2(a) and 2(b). The adopted saliency measure can detect keypoints spread quite uniformly over the surface, avoiding only highly planar areas such as those near to the armadillo's chest and inner ear pads. On the other hand, it does not seem to focus only on highly protruded or convex areas, selecting also weakly characterized zones. We expect this to negatively influence the repeatability of the detector.

2.1.2 *Intrinsic Shape Signatures (ISS)*

Intrinsic Shape Signatures were introduced in Zhong (2009). ISS saliency measure is based on the Eigenvalue Decompo-

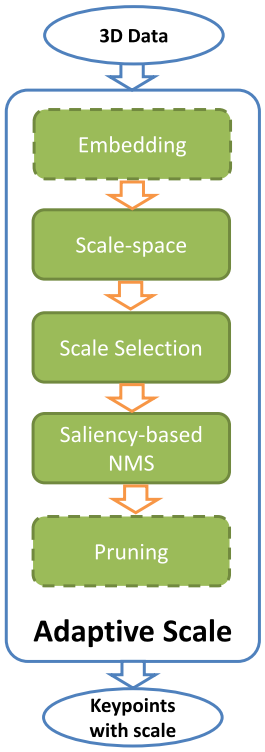


Fig. 4 General structure of an Adaptive-scale 3D keypoint detector (*dashed contour lines* indicate optional blocks)

sition (EVD) of the scatter matrix $\mathbf{\Sigma}(\mathbf{p})$ of the points belonging to the support of p, i.e.

$$\mathbf{\Sigma}(\mathbf{p}) = \frac{1}{N} \sum_{\mathbf{q} \in \mathcal{N}(\mathbf{p})} (\mathbf{q} - \mu_{\mathbf{p}})(\mathbf{q} - \mu_{\mathbf{p}})^T \quad \text{with}$$
$$\mu_{\mathbf{p}} = \frac{1}{N} \sum_{\mathbf{q} \in \mathcal{N}(\mathbf{p})} \mathbf{q}. \tag{6}$$

Given $\mathbf{\Sigma}(\mathbf{p})$, its eigenvalues in decreasing magnitude order are denoted here as $\lambda_1, \lambda_2, \lambda_3$. During the pruning stage, points whose ratio between two successive eigenvalues is below a threshold are retained,

$$\frac{\lambda_2(\mathbf{p})}{\lambda_1(\mathbf{p})} < Th_{12} \wedge \frac{\lambda_3(\mathbf{p})}{\lambda_2(\mathbf{p})} < Th_{23}. \tag{7}$$

The rationale is to avoid detecting keypoints at points exhibiting a similar spread along the principal directions, where a repeatable canonical reference frame cannot be established and, therefore, the subsequent description stage can hardly turn out effective. Among remaining points, the saliency is determined by the magnitude of the smallest eigenvalue

$$\rho(\mathbf{p}) \doteq \lambda_3(\mathbf{p}) \tag{8}$$

so as to include only points with large variations along each principal direction.

Examples of keypoints detected by ISS are depicted in Fig. 2(a) and 2(b). Although the absolute number of keypoints is similar to that of LSP, some of the points seem to focus on well identifiable, and likely repeatable, zones, such as the teeth tips or the center of the eyelids.

2.1.3 KeyPoint Quality (KPQ)

The 3D detector presented in Mian et al. (2010) is referred to here as KeyPoint Quality (KPQ). Analogously to ISS, saliency is based on the scatter matrix $\mathbf{\Sigma}(\mathbf{p})$. Pruning of non-distinctive points, however, is achieved by thresholding the ratio between the maximum lengths along the first two principal axes, after the support has been aligned to the canonical reference frame given by principal directions. This is similar to thresholding the eigenvalues ratio but, unlike ISS, KPQ considers only the first two principal directions, so that less points are pruned. As for saliency, it is determined by means of an empirical combination of curvatures within the support of $\mathbf{p}$,

$$\begin{aligned}\rho(\mathbf{p}) \doteq \frac{1000}{N^2} &\sum_{q \in \mathcal{N}(\mathbf{p})} \left|C_K(\mathbf{q})\right| \\ &+ \max_{q \in \mathcal{N}(\mathbf{p})} 100C_K(\mathbf{q}) + \min q \in \mathcal{N}(\mathbf{p}) \left|100C_K(\mathbf{q})\right| \\ &+ \max_{q \in \mathcal{N}(\mathbf{p})} 10C_M(\mathbf{q}) + \min_{q \in \mathcal{N}(\mathbf{p})} \left|10C_m(\mathbf{q})\right|. \end{aligned} \tag{9}$$

To limit the sensitivity of the estimation to noise and sampling density, curvatures are computed over a smoothed and re-sampled surface fitted to the aligned data by means of a surface fitting algorithm (D'Errico 2010).

Examples of the keypoints detected by KPQ in an exemplar model of the Stanford dataset are reported in Fig. 3(a) and 3(b). The detector finds a relevant amount of keypoints. It avoids uniform areas such as the armadillo's chest, but not the inner ear pads.

2.1.4 Heat Kernel Signature (HKS)

Heat Kernel Signature (Sun et al. 2009) employs as saliency measurement the restriction of the heat kernel to the temporal domain computed over the mesh. The heat diffusion on the generic manifold M is governed by the heat equation

$$\triangle_M u(\mathbf{x}, t) = -\frac{\partial u(\mathbf{x}, t)}{\partial t}, \tag{10}$$

where $\triangle_M u(\mathbf{x}, t)$ is the Laplace-Beltrami operator defined on M. For any M, there exists a function $k_t(\mathbf{p}, \mathbf{q})$ that can be thought of as the amount of heat that is transferred from $\mathbf{p}$ to $\mathbf{q}$ in time t given a unit heat source at $\mathbf{p}$. Such a function is called the heat kernel, and is uniquely determined by the manifold M, regardless of the quantity being diffused. As such, it provides a compact characterization of the underlying manifold. In Sun et al. (2009) the authors show that

the restriction of the heat kernel to the temporal domain, i.e. $k_t(\mathbf{p},\mathbf{p})$, does not result in a loss of information in the description. Hence they propose to use the temporal evolution of $k_t(\mathbf{p},\mathbf{p})$, sampled at logarithmically spaced time intervals, as an informative descriptor of the points of M.

To use the HKS as a keypoint detector, the saliency has been defined by the authors as the value of $k_t(\mathbf{p},\mathbf{p})$ for some large, fixed t'. Keypoints are selected as the local maxima of the saliency over a 2-rings neighborhood, i.e. as those points where

$$k_{t'}(\mathbf{p},\mathbf{p}) > k_{t'}(\mathbf{q},\mathbf{q}) \quad \forall \mathbf{q} \in \left\{\mathbf{q} : e(\mathbf{p},\mathbf{q}) \leq 2\right\} \tag{11}$$

with the 2-rings neighborhood of a vertex $\mathbf{p}$ defined as the set of vertices whose minimum number of edges from $\mathbf{p}$ is at most 2. With this saliency measure, the detector focuses on the extremities of long protrusions of the surface, as can be noted in Figs. 3(c) and 3(d). Note that this model is a down-sampled version of that used to create the previous and subsequent screenshots. This is due to the memory complexity of this method (see Sect. 3.3). Although our datasets focus on invariance to rigid transformations, the HKS and, as a consequence, the HKS detector, are invariant to isometric deformations. Therefore, the HKS detector offers a wider degree of invariance than the other detectors, that, although beneficial in other applications, e.g. non-rigid matching, can be too broad and less effective for the scenarios considered in the present evaluation.

2.2 Adaptive-Scale Detectors

As sketched in Fig. 4, the common structure of adaptive-scale detectors includes building a scale-space defined on the surface, thus directly extending to the case of 3D data the well-known concept defined for 2D images (Lindeberg 1998). Alternatively, instead of extending the scale-space theory to 3D data, an embedding of the data onto a 2D plane can be computed, so as to carry out then traditional scale-space analysis.

Successively, a characteristic scale is associated to each point, which is selected as the maximum along the scale dimension of a suitable function, generally coinciding with the saliency. As previously pointed out, such a characteristic scale is used to define the support for the subsequent description stage. Often, in the presence of local maxima, multiple keypoints with different scales are detected at the same location. If the function exhibits a monotonic trend, then no characteristic scale can be defined and the point is discarded from the set of candidate keypoints.

Next, keypoints are picked up by means of a NMS of the saliency at the characteristic scale of each point. Generally, this NMS stage is performed both spatially and along the scale dimension, although it might be done only spatially (Mian et al. 2010), using the characteristic scale to define the support on which the saliency is computed. Finally, with similar purposes as in fixed-scale methods, an optional pruning stage whereby additional points are dismissed based on different constraints may be executed.

2.2.1 Laplace-Beltrami Scale-space (LBSS)

In the proposal presented in Unnikrishnan and Hebert (2008), hereinafter referred to as Laplace Beltrami Scale Space (LBSS), the scale-space is built by computing an invariant, derived from the Laplace-Beltrami operator $\triangle_M$, on increasing supports around each point of the 3D mesh. This invariant, which provides the saliency, is defined as follows:

$$\rho(\mathbf{p},t) = \frac{2\|\mathbf{p} - A(\mathbf{p},t)\|}{t} e^{-\frac{2\|\mathbf{p}-A(\mathbf{p},t)\|}{t}}, \tag{12}$$

where $A(\mathbf{p},t)$ is an operator which can be interpreted as the displacement of a point along its normal by a quantity proportional to the mean curvature C_H:

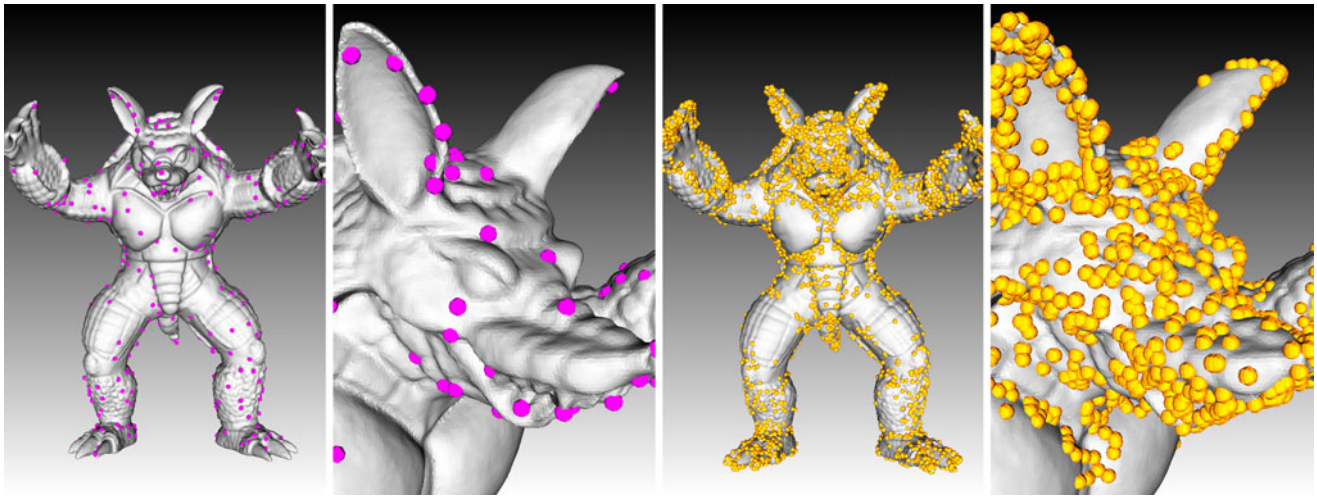
$$A(\mathbf{p},t) \approx \mathbf{p} + C_H(\mathbf{p})\mathbf{n}(\mathbf{p})t^2 = \mathbf{p} + \frac{t^2}{2}\triangle_M \mathbf{p}. \tag{13}$$

Hence, for simple shapes such as perfect spheres or planes, the saliency employed by LBSS is proportional to the mean curvature. Moreover, both in the original formulation and in our implementation, the effects of point density variations are taken into account by defining the saliency in terms of a density normalized operator $\tilde{A}(\mathbf{p},t)$. It is worth noting that the geometry of the mesh is not modified during the creation of the scale-space. Both scale selection and NMS are carried out based on $\rho(\mathbf{p},t)$ and there is no additional pruning stage.

As shown in Figs. 5(a), 5(b), LBSS is characterized by a very small number of detected keypoints, mainly residents around bumps and arched surfaces.

2.2.2 MeshDoG

Likewise (Unnikrishnan and Hebert 2008), also MeshDoG (Zaharescu et al. 2009) deploys the 3D mesh as the representation adopted to build the scale-space, which, in turn, is created by applying different normalized Gaussian derivatives through the Difference-of-Gaussians (DoG) operator, a well-known approximation of the normalized Laplacian (Lowe 2004). The operator, though, is not computed directly on the geometry of the mesh, but on a scalar function defined on the manifold, which in the original paper is either the mean curvature, the Gaussian curvature or the photometric appearance of a vertex (the mean of the RGB channels). The output of the DoG operator represents the saliency used to detect keypoints. In our evaluation we selected the mean curvature as scalar function because on our datasets it yields better results than the Gaussian curvature and RGB values



(a) LBSS (b) LBSS (c) MeshDoG (d) MeshDoG

Fig. 5 Example of keypoints detected by LBSS and MeshDoG on the Armadillo model of the *Retrieval* dataset

are not always available. With this choice, the saliency is then defined as follows:

$$\rho(\mathbf{p}, t) = C_H^{(t)}(\mathbf{p}) - C_H^{(t-1)}(\mathbf{p}), \tag{14}$$

where

$$C_H^{(t)} = C_H^{(t-1)} * G(\sigma) \tag{15}$$

$$= \frac{\sum_{\{\mathbf{q}:e(\mathbf{p},\mathbf{q})\leq 1\}} C_H^{(t-1)}(\mathbf{q}) \exp(-\frac{(\mathbf{p}-\mathbf{q})^2}{2t^2})}{\sum_{\{\mathbf{q}:e(\mathbf{p},\mathbf{q})\leq 1\}} \exp(-\frac{(\mathbf{p}-\mathbf{q})^2}{2t^2})} \tag{16}$$

i.e. $C_H^{(t)}$ is the t-th convolution of the mean curvature map with the Gaussian kernel. Similarly to LBSS (Unnikrishnan and Hebert 2008), MeshDoG does not modify the surface geometry during construction of the scale-space.

Scale selection is based on the saliency, and NMS is carried out on the scale-space using a fixed-size support determined by the 1-ring neighborhood at the current and adjacent scales. Additionally, a pruning stage is applied based on two steps. First, all keypoints are ranked according to their saliency and thresholded, so as to limit their number to a maximum value corresponding to a fixed percentage (5 %) of the number of vertices of the mesh. Then, as proposed in Lowe (2004), non-corner responses are pruned by thresholding the ratio between the largest and smallest eigenvalues of the Hessian matrix at each candidate keypoint.

As vouched by Figs. 5(c), 5(d), and unlike other approaches such as LBSS, this detector tends to extract a high number of keypoints, that accumulate around areas characterized by high local curvature.

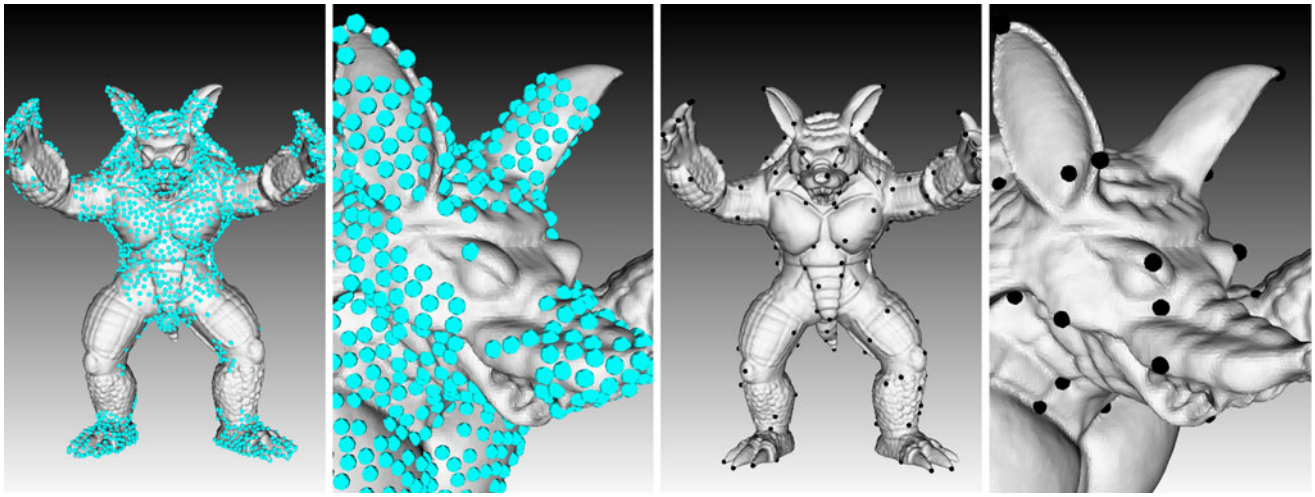
2.2.3 KeyPoint Quality—Adaptive-Scale (KPQ-AS)

In addition to the fixed-scale method, in Mian et al. (2010) an adaptive-scale detector method is also proposed, which will be referred to as KPQ-AS. The scale-space is built by increasing the size of the support over which the pruning term used by KPQ, i.e. the ratio between the maximum lengths along the first two principal axes, is computed. Then, automatic scale selection at each keypoint is carried out by means of non-maxima suppression of this term along the scale. Successively, spatial NMS is performed at the selected scale based on the same saliency, (9), as in KPQ. No further pruning step is deployed.

Figures 6(a), 6(b) depict the keypoints extracted by KPQ-AS on one model of our dataset. It is quite evident how the keypoint distribution is notably non-uniform along the global 3D surface, with keypoints extracted also within flat surfaces such as the Armadillo's chest and inner ear pads.

2.2.4 Salient Points (SP)

Similarly to Unnikrishnan and Hebert (2008) and MeshDoG Zaharescu et al. (2009), the proposal in Castellani et al. (2008), referred to in Bronstein et al. (2010) and here as Salient Points (SP), builds its scale-space directly on the 3D mesh. The saliency measure is represented by the response to the Difference-of-Gaussians (DoG) operator. However, unlike in Zaharescu et al. (2009), the DoG operator is applied directly to the 3D coordinates of the vertices. Hence, this approach modifies the geometrical structure of the surface during the construction of the scale-space. This detector looks for points that exhibit a significant displacement after filtering with different scales. To obtain a scalar quantity out



(a) KPQ-AS (b) KPQ-AS (c) SP (d) SP

Fig. 6 Example of keypoints detected by KPQ-AS and SP on the Armadillo model of the *Retrieval* dataset

of the 3D translation, the authors consider as the most relevant displacement that along the normal. Hence, saliency is defined as the projection of the translation onto the normal direction, i.e.

$$\rho(\mathbf{p}, t) = \left\| \mathbf{n}(\mathbf{p}) \cdot \mathrm{DoG}(t, \mathbf{p}) \right\| \tag{17}$$

($\cdot$ indicating the dot product), where

$$\mathrm{DoG}(t, \mathbf{p}) = g_t(\mathbf{p}) - g_{2t}(\mathbf{p}) \tag{18}$$

and

$$g_t(\mathbf{p}) = \frac{\sum_{\mathbf{q}:\gamma(\mathbf{p},\mathbf{q})\leq 2t} \mathbf{q} \exp(-\frac{(\mathbf{p}-\mathbf{q})^2}{2t^2})}{\sum_{\mathbf{q}:\gamma(\mathbf{p},\mathbf{q})\leq 2t} \exp(-\frac{(\mathbf{p}-\mathbf{q})^2}{2t^2})}. \tag{19}$$

After a normalization step, performed to enhance the highest peaks, only those points characterized by a saliency value higher than a percentage of the saliency values in their neighborhood are retained. Scale selection is performed by choosing, at each retained vertex, the scale with the highest saliency value. Then, in order to perform saliency-based NMS, all the saliency values obtained at the various scales are summed up. The final pruning step discards all local maxima whose saliency is lower than a percentage of the global maximum.

Figures 6(c), 6(d) depict the keypoints extracted by SP on one model of our dataset. As it can be seen, this method detects a limited number of keypoints, but well-localized and often at the extremities of long protrusions of the surface, such as the fingertips of hands and feet.

2.2.5 *Other Methods*

As for other adaptive-scale approaches, the proposals by Akagunduz and Ulusoy (2007), Novatnack and Nishino (2007) and Novatnack and Nishino (2008) can be grouped together based on the peculiarity that they rely on a parametrization mapping the 3D mesh onto a 2D plane, so as to exploit the lattice structure of the 2D image to build the scale-space. In Novatnack and Nishino (2007), the parametrization is computed by mapping the border of the mesh (that must be already present or manually created by cutting a watertight mesh) to the border of a 2D image and then using the parametrization algorithm proposed by Yoshizawa et al. (2004). In Novatnack and Nishino (2008) and Akagunduz and Ulusoy (2007) the parametrization is already available in the input data, since these methods work on range images.

Given the parametrization, both Novatnack and Nishino (2007) and Novatnack and Nishino (2008) create a scale-space representation of the *normal map* of the mesh, i.e. an image whose channels represent normal components. The saliency measure is the cornerness defined by the eigenvalues of the Gram matrix of the support. The algorithm flow is similar in Akagunduz and Ulusoy (2007) but the saliency measure is given by the mean (C_H) and Gaussian (C_K) curvatures (denoted as *HK maps*) and, instead of corners, connected regions of similar curvature are sought for.

Unlike previous proposals, 3D SURF (Knopp et al. 2010) builds a scale-space out of a voxelized version of the original mesh. The use of the voxelized representation allows for deployment of efficient box-filtering schemes to compute the saliency, which is given by the Hessian of second-order Gaussian derivatives and computed for each grid bin and for each octave.

Very recently, another technique which builds the scale-space directly on the 3D mesh has been proposed in Fadaifard and Wolberg (2011). This method is inspired by Sun et al. (2009) in deriving the scale-space formulation by means of the diffusion of a signal on the surface based on

the Laplace-Beltrami operator, the chosen signal being the surface curvature. Unlike HKS though, which selects all keypoints at the same, fixed scale, this proposal can associate a characteristic scale to each keypoint by exploiting the scale-space representation. More pertinently, the scale at each level of the scale-space is defined as the scale of the Gaussian that fits the transfer function of the smoothing filter for that level. Keypoints are selected as those vertices being local extrema among their spatial neighbors both on the current level and on the two adjacent levels along the scale-space. Additionally to scale selection, another advantage of this proposal with respect to HKS is the reduced computational complexity, as vouched by the reported significantly faster running times.

3 Methodology

3.1 Datasets

In our experiments we use five datasets. Two of them are synthetic, in the sense that they have been created applying known artificial deformation to 3D meshes in order to simulate two different application scenarios. The synthetic datasets have been built using models taken from the Stanford Repository.[1] The other three datasets are: the dataset[2] used for the experimental validation in Mian et al. (2010), acquired with a laser scanner; the dataset used for the experimental validation in Tombari et al. (2010), obtained by means of the SpaceTime Stereo acquisition technique; a novel dataset, acquired in our laboratory by a Microsoft Kinect device. The last two datasets are available through the SHOT website.[3] In the following we will refer to the synthetic datasets as *Retrieval* and *Random Views*, to the dataset of Mian et al. (2010) as *Laser Scanner* and to the two SHOT datasets as *Space Time* and *Kinect*, respectively.

Each dataset comprises a set of models, $\mathcal{M} = \{\mathbf{M}_h\}_{h=1}^{N}$ and a set of scenes, $\mathcal{S} = \{\mathbf{S}_l\}_{l=1}^{M}$. Each scene contains a subset of the models. Only in the SHOT datasets objects not present in the model library have been additionally used to create the scenes (clutter). The ground-truth rotations and translations, R_{hl} and t_{hl}, to align each model $\mathbf{M}_h$ with its instance in the scene $\mathbf{S}_l$ are known. In the case of the synthetic datasets, ground-truth is known by construction. For details on the way it was estimated in the other datasets the reader is referred to Tombari et al. (2010) and Mian et al. (2010). Figure 7 shows examples of models and scenes taken from four of the datasets (all except *Retrieval*, whose scenes and models coincide with the models of *Random Views*). The synthetic 2.5D views were created by applying from a random point of view the algorithm described in Katz et al. (2007) on the 3D scene built by randomly rotating and translating the selected 3D models. Both synthetic datasets have been made publicly available.

Datasets can also be categorized according to the application scenario they address. In one of the synthetic dataset, *Random Views*, as well as in *Laser Scanner*, each scene is a 2.5D mesh, i.e. a view of the spatial arrangement of the models from a specific vantage point, whereas the models are full 3D meshes. Therefore, these datasets are suitable for comparing the performance of the detectors in an object recognition scenario wherein a full 3D model is matched against a 2.5D view of the scene to detect its presence. The *Space Time* and *Kinect* datasets represent a different object recognition scenario as 2.5D models are sought for in cluttered 2.5D views. Moreover, the latter datasets are acquired with less accurate and cheaper sensing devices, delivering noisier and significantly less detailed meshes than laser scanners.

The second synthetic dataset, *Retrieval*, addresses a 3D shape retrieval scenario and is similar in spirit to the dataset used in Bronstein et al. (2010), Boyer et al. (2011): only one full 3D model is used to create each scene and there are no occlusions and clutter. On the other hand, this dataset is much simpler than that used in Bronstein et al. (2010), Boyer et al. (2011), due to the only nuisances present in scenes being rigid transformations and synthetic noise. The main purpose of this dataset is to address a retrieval scenario using the same data as the *Random Views* dataset, so as to highlight the impact of the application context on the performance of detectors.

3.2 Metrics

Absolute/relative repeatability As mentioned in Sect. 1, the most important trait of a keypoint detector is its *repeatability*. This characteristic accounts for the ability of the detector to find the same set of keypoints on different instances of a given model, where the differences may be due to noise corruption, view point change, partial occlusions/missing parts or a combination of the previous nuisances.

Similarly to the work by Schmid et al. (2000) on evaluation of 2D keypoint detectors, a keypoint extracted from the model $\mathbf{M}_h$, k_h^i and transformed according to the ground-truth rotation and translation, (R_{hl}, t_{hl}), is said to be repeatable if the distance from its nearest neighbor, k_l^j, in the set of keypoints extracted from the scene $\mathbf{S}_l$ is less than a threshold ϵ:

$$\left\| R_{hl} k_h^i + t_{hl} - k_l^j \right\| < \epsilon. \tag{20}$$

We evaluate the overall repeatability of a detector both in relative and absolute terms. Given the set RK_{hl} of repeatable keypoints for an experiment involving the model-scene pair

[1] www.graphics.stanford.edu/data/3Dscanrep.

[2] www.csse.uwa.edu.au/~ajmal.

[3] vision.deis.unibo.it/SHOT.

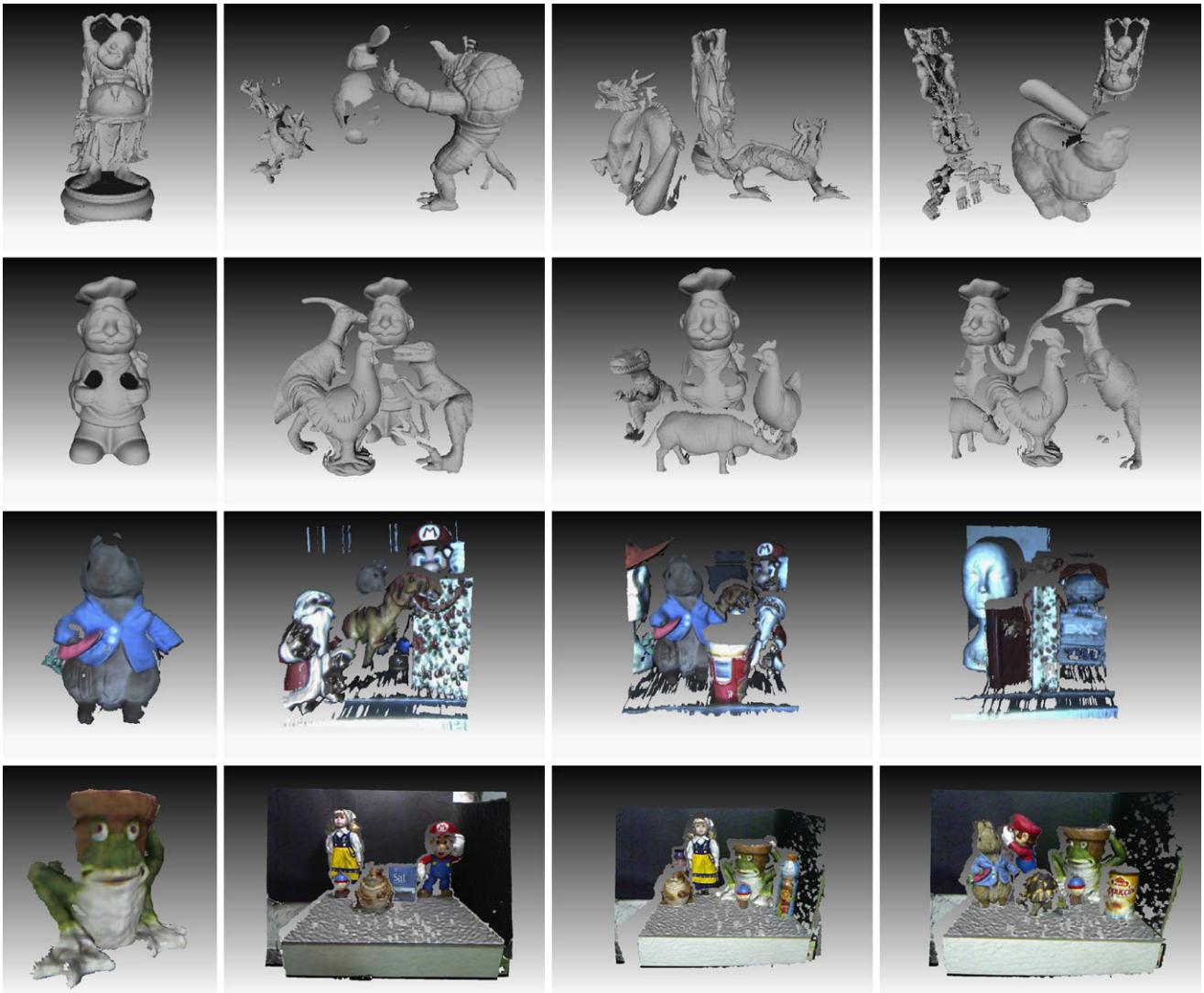


Fig. 7 One model and three scenes from the datasets. From top to bottom row: *Random Views*, *Laser Scanner*, *Space Time*, *Kinect*

$(\mathbf{M}_h, \mathbf{S}_l)$, the absolute repeatability is defined as

$$r_{abs} = |RK_{hl}| \tag{21}$$

whereas the relative repeatability is given by

$$r = \frac{|RK_{hl}|}{|K_{hl}|} . \tag{22}$$

The set K_{hl} is the set of all the keypoints extracted from the model $\mathbf{M}_h$ that are not occluded in the scene $\mathbf{S}_l$. This set is estimated by aligning the keypoints extracted on $\mathbf{M}_h$ according to the ground-truth rotation and translation and then checking for the presence of vertices in $\mathbf{S}_l$ in a small neighborhood of the transformed keypoints. This neighborhood is defined by a sphere centered at the transformed keypoint and with a radius set to 2 *mesh resolution* (*mr*) (Johnson and Hebert 1999), the mesh resolution given by the mean length of the edges in the mesh. If at least a vertex is present in the scene in such a neighborhood, the keypoint is added to K_{hl}.

We consider the absolute repeatability, as in Mikolajczyk et al. (2005), because another important trait of a detector is the amount of repeatable keypoints it can provide to the subsequent modules of an application. Detecting a small number of keypoints can not be enough to apply geometrical verification or outliers removal steps, whereas too many may waste computational resources.

To show aggregated results, we plot the average of these repeatability measures over the number of model-scene pairs of each dataset.

Scale Repeatability As discussed in the previous section, two classes of detectors are considered. In the case of adaptive-scale detectors, an additional repeatability score is introduced, i.e. scale repeatability. Given the scales σ_h^i, σ_l^j

of a pair of repeatable keypoints, (k_h^i, k_l^j), the scale repeatability of the pair is defined as:

$$r_{scale}^{ij} = \frac{V(\text{Sphere}(\sigma_h^i) \cap \text{Sphere}(\sigma_l^j))}{V(\text{Sphere}(\sigma_h^i) \cup \text{Sphere}(\sigma_l^j))} \tag{23}$$

with $\text{Sphere}(\sigma)$ indicating the sphere of radius σ and V(Sp) the volume of the 3D region Sp. The overall scale repeatability for one model versus one scene is given by

$$r_{scale} = \frac{\sum_{(k_h^i, k_l^j) \in RK_{hl}} r_{scale}^{ij}}{|RK_{hl}|}. \tag{24}$$

As noted in Unnikrishnan and Hebert (2008), the difference in dimensionality with 2D images makes this overlapping measure drop faster than for 2D detectors. Citing Unnikrishnan and Hebert (2008),

> in the 2D image domain [...] an overlap error of 50 % can be handled with a sufficiently robust descriptor. This is equivalent to an error of 65 % in 3D, or an overlap score of just 35 %.

Hence, care should be taken when interpreting the results of this measure. Similarly to the previous case, aggregated results in terms of scale repeatability are plotted by averaging this measure over the number of model-scene pairs of each dataset.

Quantity Bias Experiment The considered detectors generate different numbers of keypoints, due to several inherent factors, such as the specific design of each step composing the whole algorithm, the presence of additional pruning steps after the detection of salient structures, etc., as well as to extrinsic factors, such as the abundance of the regions considered salient by a detector in the test data. As discussed in Mikolajczyk et al. (2005), these differences may have an undesired impact on the repeatability scores: if the number of keypoints is large, many of them may be considered repeatable by accident and not because of the design of the detector. In particular, this is going to occur if the number of extracted keypoints approaches the number of points on the surface: a detector extracting as keypoints all surface points would automatically qualify, in terms of both absolute and relative repeatability, as the ideal detector. This is generally not the case of the considered detectors, as in practice the presence of a NMS step forces the extraction of a limited number of keypoints compared to the total number of surface points. On the other hand, though, the aforementioned use of a threshold ϵ to determine whether a keypoint is repeatable, instead of the theoretically possible but practically infeasible requirement of exact coordinates, increases the risk that this metric gets biased due to excessive extraction of keypoints. Intuitively, by comparing the addressed problem to a binary classification task, the repeatability metrics can be interpreted as a sort of Recall, or True Positive Rate: if not counterbalanced by a complementary measure, i.e. Precision, even a naive classifier is able to yield an optimum score for these metrics.

Understanding when this bias occurs for the various methods is in general hard a task, as it inherently depends on the local keypoint density extracted on the surface, which is usually not uniform (see, e.g., Figs. 2–6). Nevertheless, in order to better analyze how this phenomenon affects our results, we have included an additional experiment, denoted hereinafter as *Quantity Bias*. In this experiment, all keypoints extracted by a method are ranked according to the saliency specifically defined by that method: this allows limiting the maximum number of keypoints detected by each approach to those having the highest saliency according to each detector. Thus, relative repeatability can be plotted over the number of extracted keypoints. We can therefore understand how the relative performance of each detector changes from a situation where only a few keypoints are extracted, warding this bias off and exposing the real effectiveness of the proposed saliency, to a situation where we potentially reach a biased repeatability metric. Moreover, this experiment explicitly highlights the maximum quantity of detected keypoints for each method. To reduce the number of charts and improve readability we only present the results of this experiment on a subset of our test data, i.e. datasets *Retrieval* and *Kinect*, as representatives of the two addressed application scenarios.

The biasing effect of quantity on repeatability brings in also the undesirable consequence that tuning the parameters so as to maximize repeatability risks to excessively increase the number of extracted keypoints, since this tend to raise repeatability scores. For this reason, and as done in Mikolajczyk et al. (2005), we chose to use the default parameters supplied by the authors rather than tuning them. Another solution would have been to tune the detectors so as to make them extract the same number of keypoints, but this is not feasible for all the considered detectors and, in any case, the influence of the different number of regions in the data considered salient by the various detectors cannot be eliminated.

Descriptor Matching Experiment The proposed methodology aims at comparing the evaluated detectors in terms of repeatability which, as mentioned in Sect. 1, is the characteristic of the detectors on which this evaluation mainly focuses on. Although in our opinion less prominent and less fairly evaluable, in this work we also aim at assessing the detectors' performance with regards to the other main trait, i.e. distinctiveness. Hence, similarly to Mikolajczyk et al. (2005), we have included an additional experiment, referred to as *descriptor matching*, where the distinctiveness of each

detector is implicitly evaluated by plotting the *False Positives vs. True Positives* curves yielded by the correspondences established by matching descriptors computed at the extracted keypoints. For this experiment we have employed two different 3D descriptors, namely SHOT (Tombari et al. 2010) and Spin Images (Johnson and Hebert 1999). Thus, the outcome of this experiment measures how much the keypoints extracted by each evaluated detector turn out distinctive according to the *chosen* descriptors: an unavoidable bias of this kind of experiments. As for the implementation, matching is then performed by means of efficient—but approximated—indexing schemes, i.e. the *best bin first kd-tree* algorithm proposed in Beis and Lowe (1997). Analogously to the *Quantity Bias* experiment, also in this case we reduce the numbers of presented charts for the sake of readability, and hence consider only *Retrieval* and *Space Time*.

Efficiency Finally, we provide a thorough evaluation of the considered detectors in terms of computational efficiency. Specifically, we compute the time required by each method to extract a set of keypoints on several scenes whose point density was iteratively decimated in order to generate surfaces with various number of points (i.e. from $\sim 10^2$ to $\sim 10^6$ points). We plot the average measured execution time over the number of points. This has been done separately for scenes belonging to two different datasets, *Retrieval* and *Kinect*, so as to usefully provide also realistic absolute execution times for typical surface sizes used in the application scenarios addressed throughout this work.

3.3 Selected Methods

The set of 3D detectors evaluated in our experiments includes: all the fixed-scale proposals introduced in Sect. 2, namely LSP, ISS, KPQ and HKS; among adaptive-scale methods, the MeshDoG, Laplace-Beltrami Scale-Space, KPQ-AS and Salient Points detectors. As for MeshDoG, we used the publicly available original C++ implementation.[4] For SP, we obtained a binary version of the detector from the authors. All other methods have been implemented in C++ as well. In particular, unlike the previous version of this evaluation (Salti et al. 2011), for KPQ detectors we re-implemented in C++ the surface smoothing and fitting routine used internally by the detectors and available only as a MATLAB script (D'Errico 2010). Moreover, although the original source code for HKS is available at the authors' website,[5] we completely re-implemented it in C++ and Fortran, by using the same linear algebra libraries called internally by MATLAB `eigs` command, i.e. ARPACK[6] to solve the sparse generalized eigenvalues problem and UMFPACK[7] to perform the LU factorization of the mesh Laplacian operator L. This allows for fairly comparing all the proposals also in term of computational efficiency (see Sect. 4.4).

Nevertheless, HKS could not be compared to the other methods on exactly the same data. Its memory requirements force the user to apply it only on meshes with a limited number of vertices. The exact number depends on the amount of RAM available on the user machine. The authors report in (Sun et al. 2009) that with 16 GB of RAM they were able to extract HKSs on a mesh with about 100 K vertices.[8] On our 6 GB machine, using the default parameters, the upper bound to the surface size turned out to be 30 K vertices. Hence, results concerning HKS have been obtained on decimated models and scenes. More precisely, we did not decimate every mesh in a dataset with more than 30 K vertices, leaving the others unchanged, as this would modify the point density between meshes of the same dataset. We, instead, decimated every mesh in a dataset with a constant decimation ratio, such as to rescale the mesh with the greatest number of vertices to 30 K and all the others accordingly.

Some of the methods presented in Sect. 2 have specific requirements on the input data that made their inclusion in this comparison unfeasible. Specifically, Novatnack and Nishino (2007) require that one and exactly one border is present in the input mesh. While this may be reasonable for partial views registration, it is definitely not straightforward in the case of retrieval of full 3D meshes, where a cut should be introduced manually to use this method, nor within an object recognition scenario, where many separated objects can be present in one scene, each one defining a separated mesh border, so that a previous segmentation of objects would be required to apply this method.

As for the works by Akagunduz and Ulusoy (2007) and Novatnack and Nishino (2008), these methods have been designed to work with range images. They both exploit the lattice structure provided by the range image in order to build a scale-space representation of the input data. Although the meshes of the scenes we use are obtained from range images (laser scans or disparity maps), and in principle the transformation is invertible, there is no way to obtain a single range image for the 3D models. Due to this reason, they are not suited to an object recognition scenario wherein full 3D models are sought for in 2.5D views, as defined in this comparison.[9]

[4] svn://scm.gforge.inria.fr/svn/mvviewer.

[5] http://geomtop.org/software/hks.html.

[6] http://www.caam.rice.edu/software/ARPACK/.

[7] http://www.cise.ufl.edu/research/sparse/umfpack/.

[8] The requirement of 16 GB of RAM for 100 K vertices was confirmed by personal communication with the authors.

[9] Recently, the detector proposed in Novatnack and Nishino (2008) has been used (Bariya and Nishino 2010) for object recognition on the *Laser Scanner* dataset by synthesizing range images from a number

Table 2 Parameters used throughout the evaluation

LSP	α	0.35
	β	0.2
ISS	Th_{12}	0.975
	Th_{23}	0.975
KPQ	th	1.06
	n	20
MeshDoG	β	5 %
	$\lambda_{max}/\lambda_{min}$	10
HKS	h	$2mr$
	Gaussian cutoff	3

3.4 Parameters

All parameters have been fixed for the experiments on all datasets. Metric parameters, such as radii, distances, noise standard deviations, etc. are expressed throughout the paper in mesh resolution units (mr). Default parameters proposed in the original publications have been used, their values reported in Table 2. The only tuned parameter is the Non Maxima Suppression radius in ISS, the two variants of KPQ and SP. In fact, in ISS and KPQ this parameter was specified in metric units by the authors, while in SP the default value was tuned for partial views registration. Thus, we fixed the Non Maxima Suppression radius to $4mr$ after running the detectors on a tuning scene with different possible values. As anticipated in the description of the method, MeshDoG results are reported using the mean curvature as saliency measure, for we found that it yields better results than the Gaussian curvature.

In all repeatability experiments, adaptive-scale detectors have been run on the set of scales $\Sigma = \{2mr, 6mr, 10mr, 14mr, 18mr, 22mr\}$.[10] This allows the detector to look for discriminative and repeatable structures ranging from point-wise scales to local and object sub-part scales. Since the first and last scales are used by some adaptive-scale detectors only to assess the presence of local extrema in the immediately subsequent or antecedent scale, detections can happen only at scales $\tilde{\Sigma} = \{6mr, 10mr, 14mr, 18mr\}$. To compare results on the same set of structure sizes, we ran fixed-scale detectors for each scale in $\tilde{\Sigma}$. The only exception is again HKS. The already discussed limit of 30 K vertices due to memory occupancy holds only if the default parameters are used. In particular, the limit is influenced by the sparsity of the Laplace operator matrix L which in turn is controlled by parameter h. This parameter intuitively but not quantitatively corresponds to the support size analyzed by the algorithm around each point (to determine the quantitative relationship between h and the size of the support is left as a future work by the authors in Sun et al. (2009)). Therefore, given the impossibility to compute the value of h corresponding to every entry in the set $\tilde{\Sigma}$ and the need to create a decimated set of data for each value of h, we chose to evaluate it only with the default scale $h = 2mr$ used by the authors in the publicly available implementation. The distance threshold for the repeatability scores ϵ is set to $2mr$. To simulate sensor noise in the synthetic datasets we added three levels of Gaussian noise, with standard deviation equal to $0.1mr$, $0.3mr$ and $0.5mr$.

For the sake of readability, in the *Quantity bias* and *Descriptor matching* experiments only the best radius for every algorithm in every dataset, as it turned out in the repeatability experiments, was used.

The attentive reader may notice a difference between the results of KPQ on all the object recognition datasets reported in this paper and those in the preliminary version (Salti et al. 2011). In the preliminary version points too close to the border of 2.5D meshes were not discarded, as instead suggested by the authors in Mian et al. (2010). By visually inspecting the results (e.g. the screenshots reported in Fig. 8), it is evident that the exclusion from detectable points of those lying on the borders is crucial to obtain meaningful repeatability scores. Considering border points, would instead bias repeatability due to the excessive number of keypoints in an area, as discussed previously. Hence, as border points would unfairly raise the repeatability scores for this algorithm, we exclude them in this evaluation.

4 Experimental Results

4.1 Repeatability Experiments

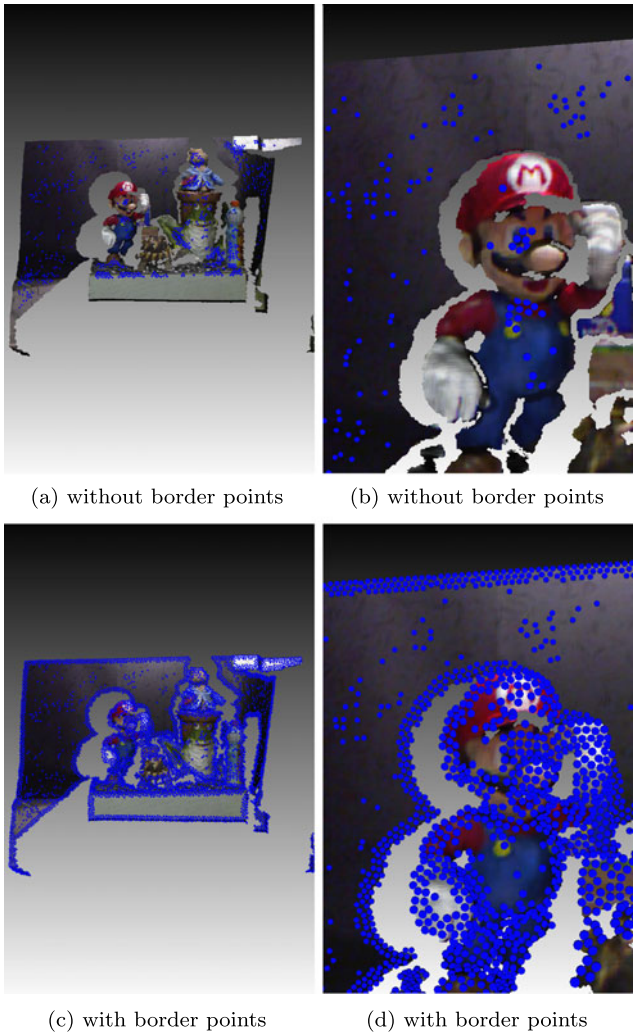
These experiments aim at comparing the selected 3D detectors in terms of relative and absolute repeatability. Scale repeatability is also assessed for those methods that extract a characteristic scale (adaptive-scale detectors).

4.1.1 Retrieval *and* Random Views *Datasets*

Between fixed-scale detectors, the highest relative repeatability on the *Retrieval* dataset (Figs. 9(a), 9(c), 9(e)) i s provided definitely by HKS. HKS, and KPQ alike, demonstrate an impressive resilience to noise, their performance in the three charts being marginally deteriorated by the increase of noise. The relative repeatability of ISS is also notably good,

of uniformly distributed overlapping views of the 3D model of the object. This technique is not suitable for our experimental comparison because the performance of detectors working on range images will be influenced by external factors such as synthetic views position and distribution.

[10] As MeshDoG implements a non-linear scale increment within its scale-space, we tuned its scale-space parameters so as to best approximate the same scale set used by the other methods.



(a) without border points (b) without border points

(c) with border points (d) with border points

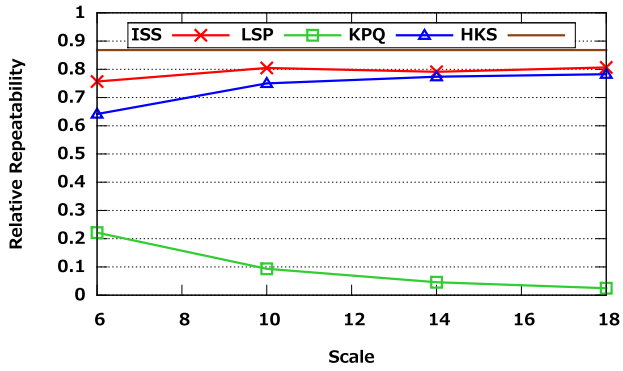
Fig. 8 Keypoints detected by KPQ with and without inclusion of border points

but compared to HKS and KPQ the method is less robust to noise. LSP, instead, performs poorly compared to other fixed scale detectors, presumably due to its saliency measure based on second-order derivatives which does not grant sufficient robustness even at the lowest noise level. Moreover, the choice of selecting the maximum SI within the local support appears to be particularly error-prone since spurious peaks in the distribution of SIs can easily occur in the presence of noise. As both ISS as well as KPQ base their saliency upon the covariance matrix Σ, the higher robustness exhibited by the latter is likely to be ascribed to the use of an effective surface resampling and smoothing procedure (D'Errico 2010).

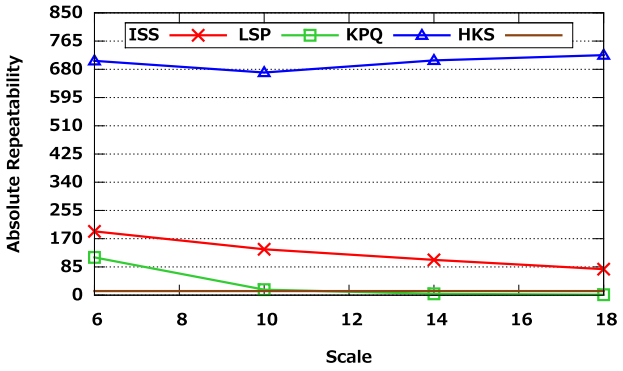
In terms of absolute repeatability (Figs. 9(b), 9(d), 9(f)), the performance of HKS turns out unsatisfactory, the mean number of repeatable keypoints being about 10. Although HKS keypoints are extremely repeatable, its effective saliency is too selective to yield enough keypoints as required in many practical applications. On the other hand, KPQ yields an impressive number of keypoints compared to other proposals. Overall ISS and LSP with radius $6mr$ provide a good quantity of keypoints ($\approx$100).

By comparing these results with those obtained on the *Random Views* dataset (Fig. 10) we can observe how algorithms performance changes significantly. Overall, fixed-scale detectors based on the scatter-matrix (i.e. ISS and KPQ) behave worse than in the retrieval scenario in the presence of partially occluded shapes, as the absence of parts of the geometric structure modifies the scatter matrix, thus reducing the repeatability of the detector. Furthermore, and conversely to the retrieval scenario, the use of large supports turns out no longer beneficial to increase repeatability due to the presence of clutter (Figs. 10(a), 10(c), 10(e)). In this dataset ISS clearly outperforms KPQ in terms of relative repeatability: the use by the latter method of the resampling and smoothing procedure turns out detrimental when matching 3D models with 2.5D scenes, as the missing parts of the mesh in the scene significantly alter the surface obtained from the resampling and therefore its curvatures, upon which saliency is based. KPQ exhibits the largest performance degradation due to the change of dataset, so that it becomes similar to LSP, which performs as poorly as in the *Retrieval* dataset due to the reasons pointed out previously. Also HKS performance degrades significantly: as a large t is used to evaluate the heat kernel in order to use the HKS as a detector, a large part of the input mesh contributes to the saliency of a point $\mathbf{p}$, this amplifying the negative impact of the missing parts of the mesh. On the other hand, the absolute number of repeatable keypoints (Figs. 10(b), 10(d), 10(f)) as well as the resilience to noise of HKS remain unchanged with respect to the Retrieval scenario. As instead ISS and KPQ decrease by one order of magnitude their absolute repeatability, all detectors end up extracting a comparable amount of repeatable keypoints in this dataset. Overall, ISS represents the best trade-off in this dataset, especially at small scales, given that detection of non-repeatable keypoints wastes computation in the description stage.

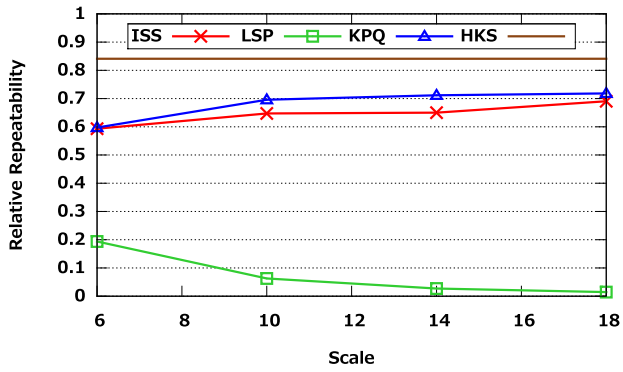
For what concerns adaptive-scale detectors, in the *Retrieval* dataset KPQ-AS reports overall the best repeatability results (Figs. 9(g), 9(h), 11). Even though at the lowest noise level SP and MeshDoG yield a higher relative repeatability and MeshDoG also a slightly higher number of repeatable keypoints, KPQ-AS neatly shows a superior robustness towards noise in terms of both absolute and relative repeatability, and a better scale repeatability (Fig. 11). This higher robustness can be motivated by the fact that the saliency of KPQ-AS averages curvatures computed at all vertices in the support smoothed with the algorithm by D'Errico (2010), while MeshDoG and SP rely on DoGs of, respectively, point-wise curvatures and point coordinates. As for



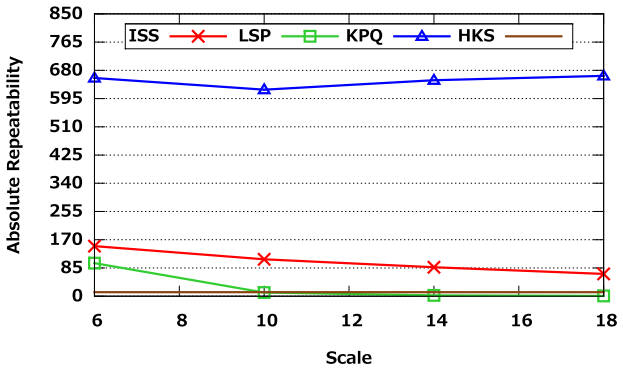
(a) σ noise $= 0.1$ mr



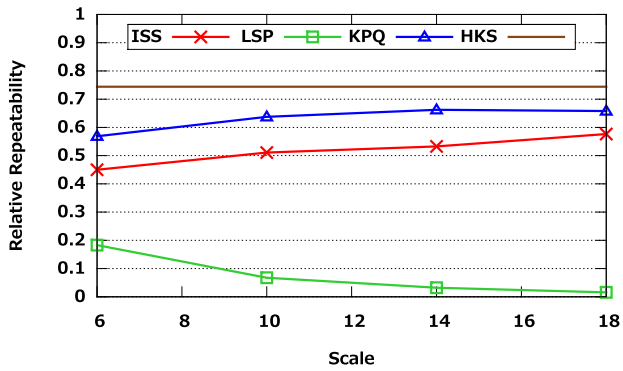
(b) σ noise $= 0.1$ mr



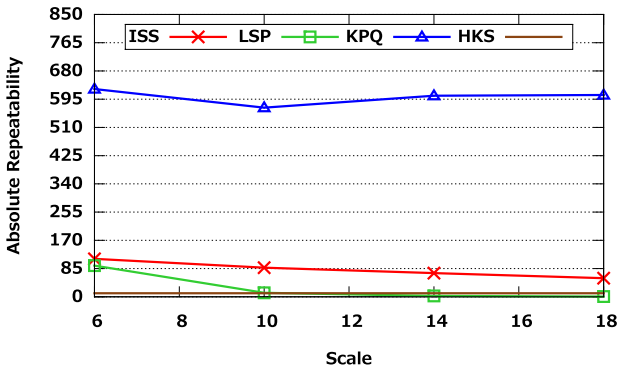
(c) σ noise $= 0.3$ mr



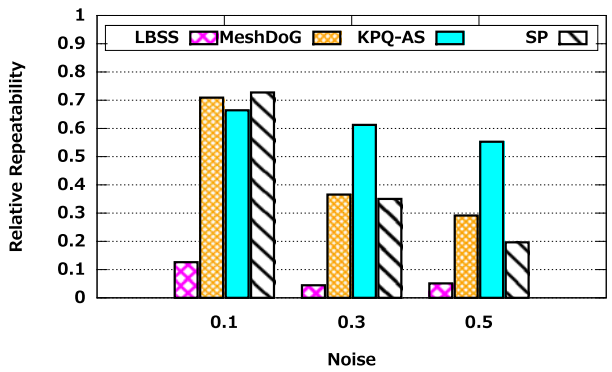
(d) σ noise $= 0.3$ mr



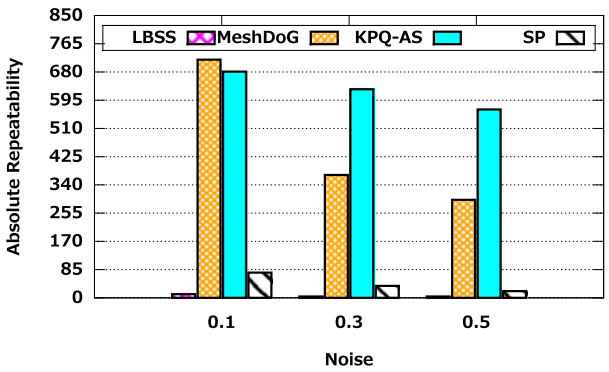
(e) σ noise $= 0.5$ mr



(f) σ noise $= 0.5$ mr

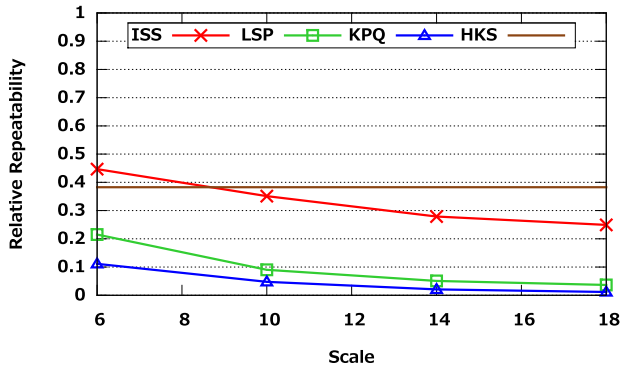


(g)

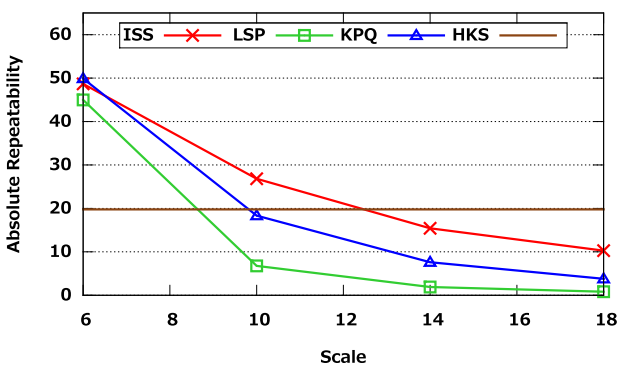


(h)

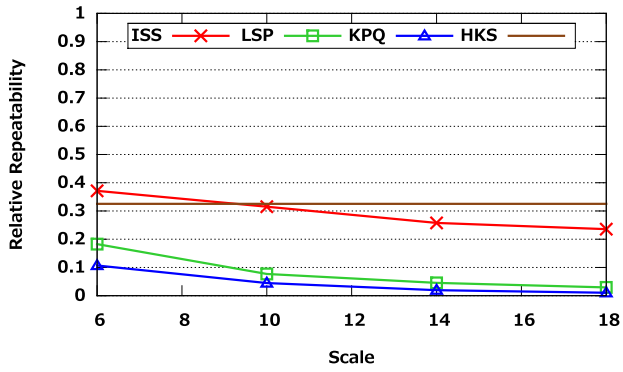
Fig. 9 Results on the *Retrieval* dataset. Fixed-scale detectors: relative (**a**, **c**, **e**) and absolute repeatability (**b**, **d**, **f**) vs. scale (support size) at different noise levels; adaptive-scale detectors: relative (**g**) and absolute (**h**) repeatability. Although a fixed-scale detector, HKS is represented by a *horizontal line* because it has been tested at one scale only (see Sect. 3.3)



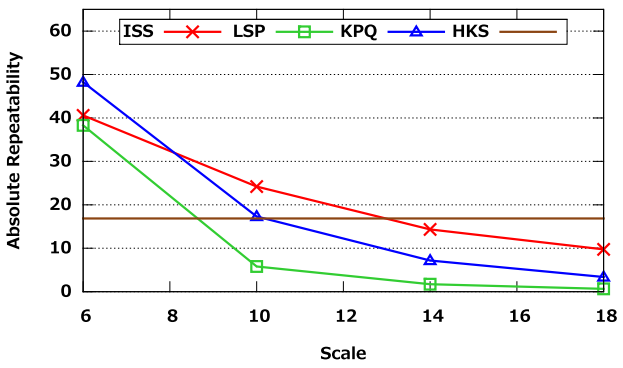
(a) σ noise $= 0.1$ mr



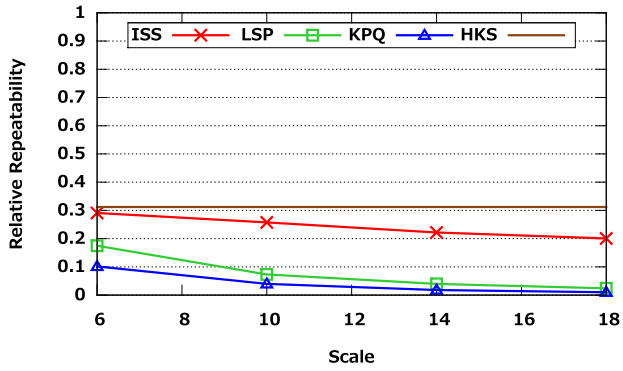
(b) σ noise $= 0.1$ mr



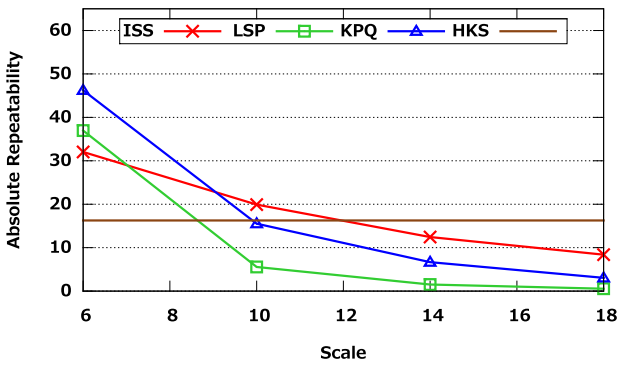
(c) σ noise $= 0.3$ mr



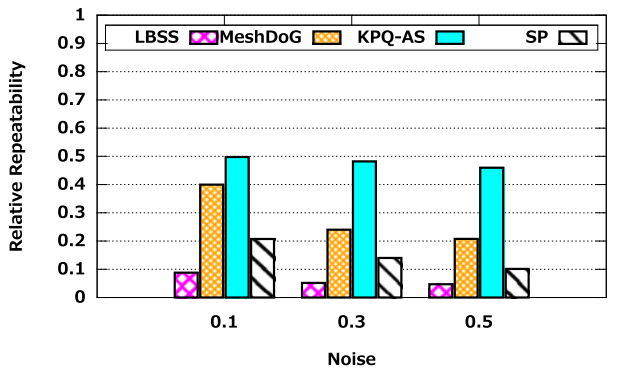
(d) σ noise $= 0.3$mr



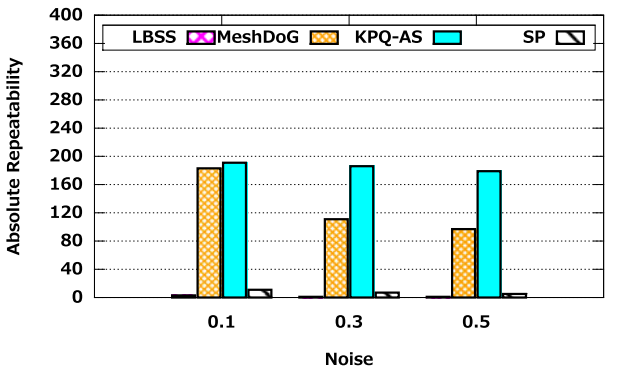
(e) σ noise $= 0.5$ mr



(f) σ noise $= 0.5$ mr



(g)



(h)

Fig. 10 Results on the *Random Views* dataset. Fixed-scale detectors: relative (**a**, **c**, **e**) and absolute (**b**, **d**, **f**) repeatability vs. scale (support size) at different noise levels; adaptive-scale detectors: relative (**g**) and absolute (**h**) repeatability. Although a fixed-scale detector, HKS is represented by a *horizontal line* because it has been tested at one scale only (see Sect. 3.3)

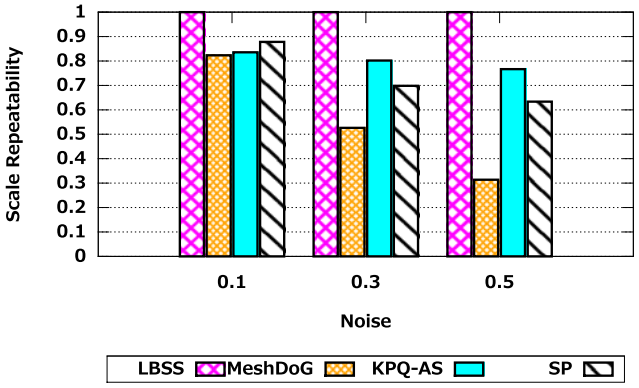


Fig. 11 Scale repeatability of adaptive-scale detectors on the *Retrieval* dataset

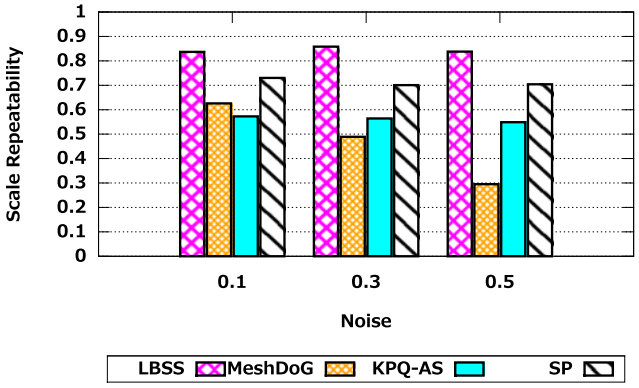


Fig. 12 Scale repeatability of adaptive-scale detectors on the *Random Views* dataset.

LBSS, the local maxima of its invariant are extremely effective in determining the characteristic scale of 3D structures even in the presence of noise (Fig. 11), which proves experimentally its theoretical characteristics. However, its performance is unsatisfactory in terms of spatial localization, yielding a low relative repeatability and a notably small number of detected keypoints.

Likewise fixed-scale detectors, the object recognition scenario turns out much more challenging than retrieval for adaptive-scale detectors, with all methods yielding decreased performance in the *Random Views* dataset. More specifically, MeshDoG, KPQ-AS and SP score lower relative as well as absolute repeatability due to missing parts of the mesh (Figs. 10(g), 10(h)). Still, KPQ-AS proves significantly more robust to noise, thus resulting the best adaptive-scale method also on this dataset. It is interesting to note that, unlike the retrieval scenario, MeshDoG and SP perform better than KPQ-AS in terms of scale repeatability at the lowest noise level (Fig. 12), due to the fact that the characteristic scale in KPQ-AS is determined by principal directions and, as aforementioned, the methods based on the scatter-matrix cannot deal effectively with partial shapes. As far as LBSS is concerned, its scale invariance is still the best one. Nevertheless, the relative and absolute repeatability are still not comparable to those achieved by the other approaches. Overall, on both datasets SP compares similarly to MeshDOG in terms of relative repeatability and scale repeatability, though yielding a notably smaller absolute repeatability.

Finally, comparing together the overall best approaches between fixed-scale and adaptive-scale detectors, HKS attains a higher relative repeatability than MeshDoG, KPQ and KPQ-AS on *Retrieval*, whilst KPQ-AS is clearly the most effective method on *RandomViews*.

4.1.2 Laser Scanner, Space Time *and* Kinect *datasets*

Results concerning these three datasets are shown for fixed-scale detectors in Figs. 13–15, for adaptive-scale detectors in Tables 3–5.

The main differences between the *Laser Scanner* dataset and the *Space Time* and *Kinect* datasets consist in the point density variation between models and scenes and the dimensionality of models. In *Space Time* and *Kinect*, models and scenes have the same dimensionality (2.5D) as well as the same point density. In *Laser Scanner* models are instead full 3D meshes and their point density is one order of magnitude higher than in scenes. The main differences between *Space Time* and *Kinect* consist in the higher noise level and amount of holes and artifacts present in the latter, which render it the most challenging dataset used throughout this evaluation.

The results obtained on these real datasets are mostly coherent with the experimental findings regarding the *Random Views* dataset. In particular, between fixed-scale detectors, ISS confirms acceptable performance in terms of both relative and absolute repeatability, with the exception of the challenging *Kinect*. The LSP detector is not robust to the sensor noise present in these datasets. As far as adaptive-scale detectors are concerned, many findings are consistent as well: LBSS provides outstanding scale overlaps among different keypoint detections but lacks spatial repeatability; the scale repeatability of MeshDoG, SP and KPQ-AS is satisfactory, although the criterion employed by the methods to determine the characteristic scale seems not particularly effective when full 3D models are compared to 2.5D scenes, as vouched by the performance drops between *Space Time* and *Laser Scanner*; the absolute repeatability of SP is usually lower than that of MeshDoG and KPQ-AS.

As for differences in the experimental findings, HKS demonstrated on the *Random Views* dataset to suffer the difference in dimensionality between scenes and models. Here its performance degrades even more, dropping at the same level as LSP. A reason for this behavior might be the absence in the real datasets of strongly protruded surfaces and details, as the adopted real sensors tend to produce smoother, less-detailed surfaces than those of Stanford models. This is especially true for *SpaceTime* and *Kinect*, whereas laser scanner data are usually richer of details. Hence, perfor-

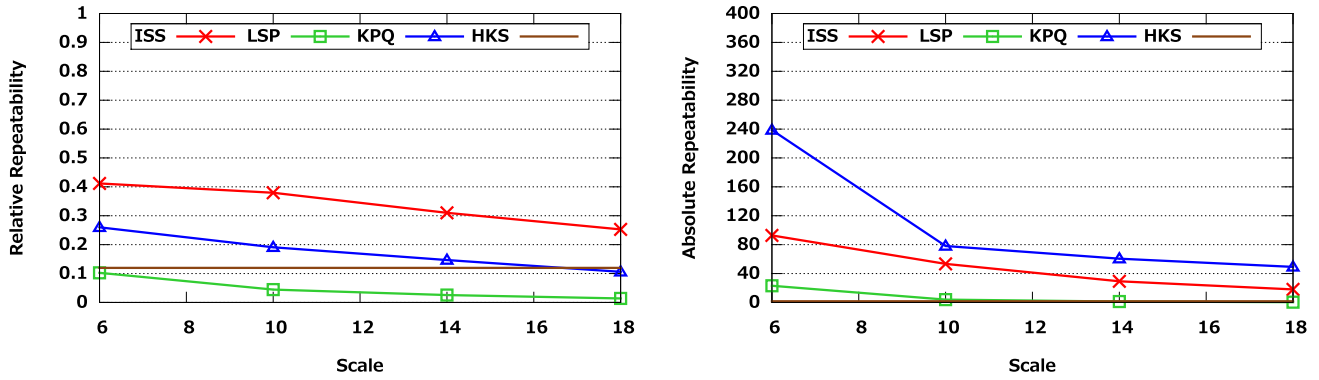


Fig. 13 Results on the *Laser Scanner* dataset for fixed-scale detectors: relative (**a**) and absolute (**b**) repeatability vs. scale (support size). Although a fixed-scale detector, HKS is represented by a *horizontal line* because it has been tested at one scale only (see Sect. 3.3)

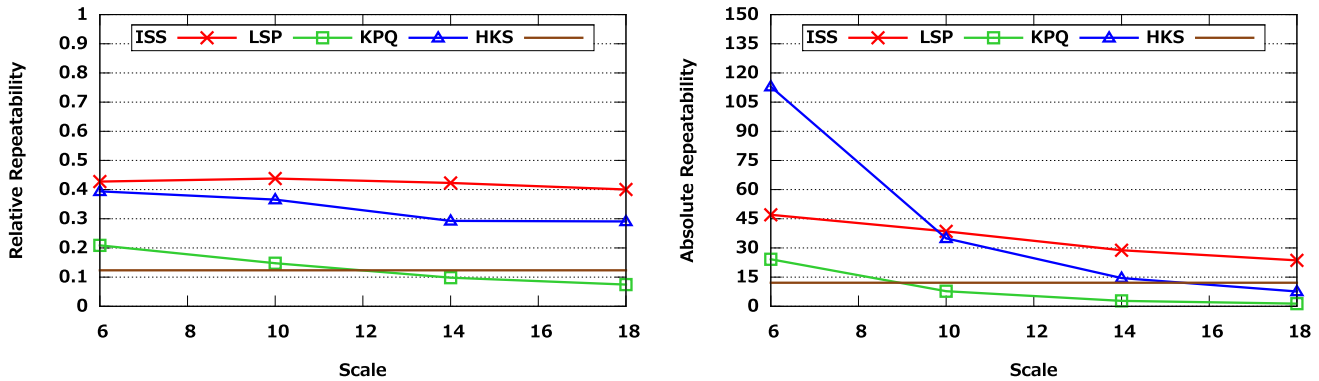


Fig. 14 Results on the *Space Time* dataset for fixed-scale detectors: relative (**a**) and absolute (**b**) repeatability vs. scale (support size). Although a fixed-scale detector, *HKS* is represented by a *horizontal line* because it has been tested at one scale only (see Sect. 3.3)

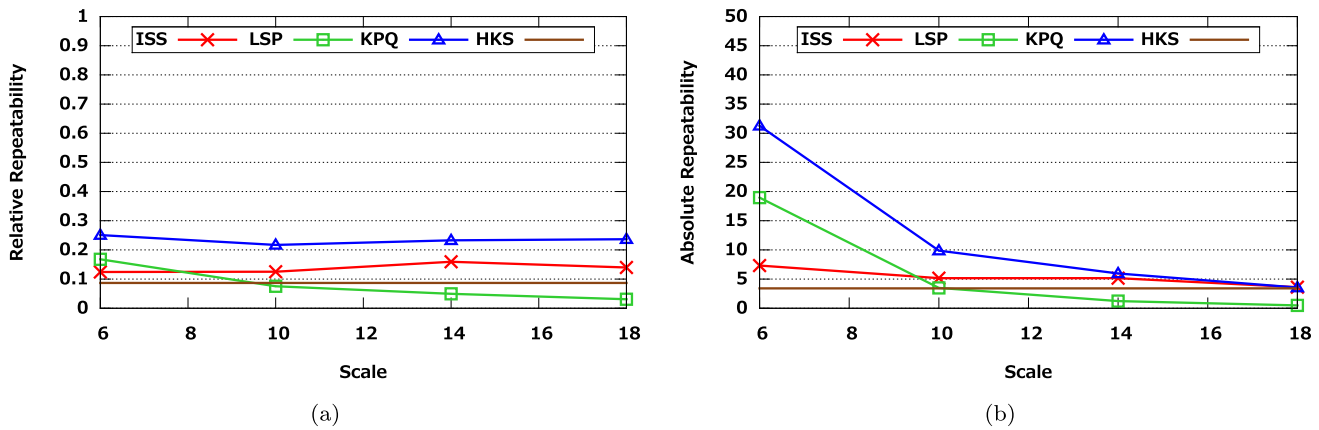


Fig. 15 Results on the *Kinect* dataset for fixed-scale detectors: relative (**a**) and absolute (**b**) repeatability vs. scale (support size). Although a fixed-scale detector, *HKS* is represented by a *horizontal line* because it has been tested at one scale only (see Sect. 3.3)

Table 3 Results on the *Laser Scanner* dataset for adaptive-scale detectors: relative, absolute and scale repeatability

	Rel. rep.	Abs. rep.	Scale rep.
LBSS	0.07	4	0.98
MeshDoG	0.29	277	0.41
KPQ-AS	0.30	233	0.51
SP	0.30	264	0.43

Table 4 Results on the *Space Time* dataset for adaptive-scale detectors: relative, absolute and scale repeatability

	Rel. rep.	Abs. rep.	Scale rep.
LBSS	0.06	2	1.00
MeshDoG	0.54	223	0.59
KPQ-AS	0.39	188	0.69
SP	0.36	96	0.55

Table 5 Results on the *Kinect* dataset for adaptive-scale detectors: relative, absolute and scale repeatability

	Rel. rep.	Abs. rep.	Scale rep.
LBSS	0.02	1	1.00
MeshDoG	0.44	94	0.63
KPQ-AS	0.39	89	0.55
SP	0.28	39	0.44

mance on the *Laser Scanner* dataset hints to the low robustness of HKS to point density variations. Unlike the results concerning the synthetic dataset, MeshDoG yields similar or higher repeatability than KPQ-AS, both in absolute and relative terms, and it is the top performer among adaptive-scale detectors on all the three datasets, tied by KPQ-AS and SP only on *Laser Scanner*.

To compare fixed- and adaptive-scales detectors, Figs. 13(a), 14(a), 15(a) and Table 3, 4, 5 indicate that while MeshDoG is overall the best detector on *Space Time* and *Kinect*, its performance deteriorates on the *Laser Scanner* dataset, so that ISS at small scales turns out the best method on the last dataset. In turn, this fact, together with the good results yielded by MeshDoG on *Random Views* (Fig. 10(g)), where one of the nuisances is the difference between model and scene dimensionality (3D versus 2.5D), indicates that MeshDoG is prone to point density variations.

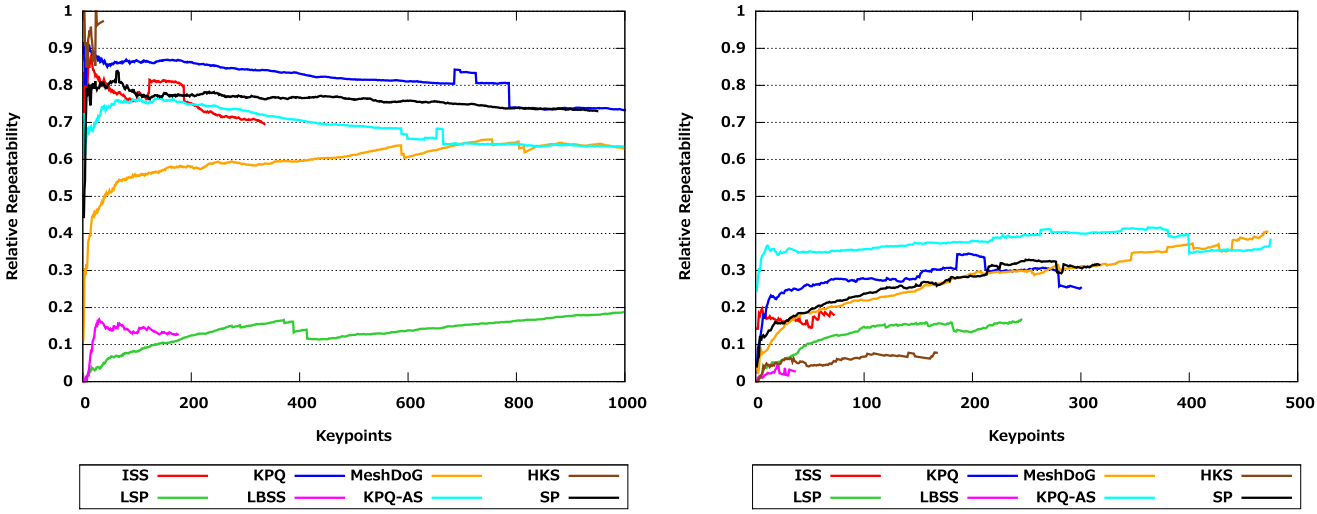
An interesting observation stems from the comparison between Figs. 10(a), 10(c), 10(e) and Fig. 14(a). In all these tests there is no difference in point density between models and scenes. The only difference is that, as with the *Random Views* dataset, models are fully 3D while scenes are 2.5D, yielding as a consequence that part of the surface included in the support of the models' keypoints will be absent in the corresponding keypoints on the scene. The fact that the performance of ISS and KPQ deteriorates at bigger scales on this dataset whereas they are constant on the *Space Time* dataset confirms that the alteration of the scatter matrix induced by the occlusion of part of the support is a severe issue for these detectors.

On *Kinect* the gap between fixed-scale and adaptive-scale detectors widens. *Kinect* is also the only dataset where KPQ performs consistently more effectively than ISS. This confirms the challenging characteristics of this dataset, and indicates that the surface smoothing step deployed by KPQ or a multi-scale analysis of the surface is a crucial factor to cope with the nuisances present in the dataset.

4.2 Quantity Bias Experiment

In this experiment, relative repeatability of all methods is plotted for increasing numbers of extracted keypoints. As previously explained, limiting the number of extracted keypoints was done by ranking them based on saliency and then selecting only the most salient ones. Figure 16(a) shows the results concerning the *Retrieval* dataset with σ noise equal to 0.1 *mr*. The overall performance trend of Fig. 9(a), 9(b) is highlighted again. HKS is the best performing method, although characterized by a severely limited number of extracted keypoints. ISS, SP, KPQ and KPQ-AS feature good repeatability scores, which decrease alongside with the increasing number of extracted keypoints, as predictable intuitively as the first keypoints extracted are those characterized by a higher saliency, hence likely the most repeatable ones. On the other hand, LBSS, LSP and MeshDoG exhibit a poor performance in the *Retrieval* scenario, especially if compared with that of the other approaches. Differently from other methods, LSP, and more evidently MeshDoG, exhibit a clear increase of repeatability when more keypoints are considered. On one side, this is a hint that the saliency used by these methods is not particularly effective, on the other this may indicate a quantity bias in the repeatability score in the previous experiment.

This phenomenon is more evident in the experiment on the *Kinect* dataset shown in Fig. 16(b). In this case, likewise the results presented in Fig. 15 and Table 5, the performance of the methods is significantly lower due to the challenging scenario. Nevertheless, their relative performance is confirmed with the exception of MeshDoG, which outperformed all methods (including KPQ-AS) in the repeatability experiment. The trend of MeshDoG in Fig. 16(b) indicates that such a good repeatability may be due to a biased score: if a limited number of keypoints is taken into account (e.g. <100), thus selecting a working point where repeatability scores are definitely not biased by quantity, we can notice that KPQ and KPQ-AS turn out the best performing methods, with their relative rank inverted compared to the *Retrieval* scenario. The SP detector exhibits a trend analogous



(a) *Retrieval* dataset (b) *Kinect* dataset

Fig. 16 Results for the *Quantity Bias* experiment

to MeshDoG in this experiment, which highlights the low suitability of this detector to the object recognition scenario. ISS yields good repeatability at small quantities, better than that reported by MeshDoG, although the maximum number of keypoints it can extract is limited compared to KPQ detectors. HKS confirms limited performance when a typical 3D object recognition scenario is taken into account, even when only the most salient keypoints are considered.

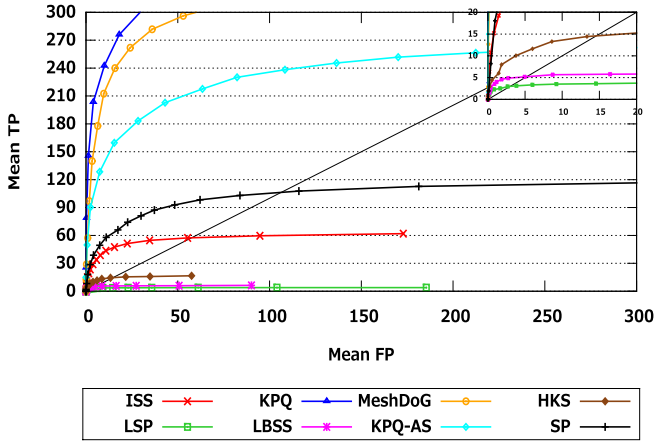
4.3 Descriptor Matching Experiment

Results for this experiment are shown in Fig. 17. On both datasets considered for this experiment and according to both the employed descriptors, it is evident that scatter matrix-based methods (ISS and the two variants of KPQ) tend to provide distinctive keypoints. This is especially evident for SHOT, which uses in a crucial step, i.e. the computation of the local Reference Frame, exactly the scatter matrix. Hence, regions where the principal directions of Σ are well defined are naturally good regions to be described by SHOT. Moreover, KPQ purposely defines its keypoints as those points where, among other characteristics, a good local Reference Frame can be defined, whereas ISS simply selects regions whose elongation along the three principal directions is strong enough. This also highlights the bias of this kind of experiments towards the specific kind of descriptor employed. In addition to ISS and the two variants of KPQ, MeshDoG and SP also perform well together with SHOT. In particular, MeshDoG turns out the best method on the *Spacetime* dataset. As for Spin Images, on the average the performance of detectors decreases, this hinting at a higher descriptive power of SHOT with regards to Spin Images. Nevertheless, also in terms of this descriptor, KPQ, KPQ-AS, MeshDoG, SP and ISS outperform all other methods in terms of distinctiveness of the extracted keypoints.

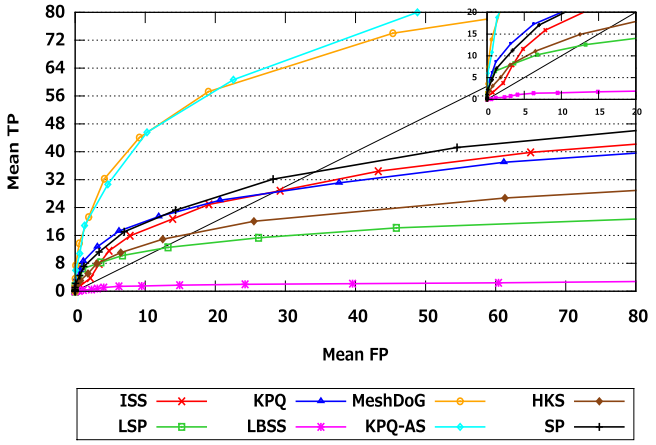
It is worth pointing out that this experiment also highlights that in a real surface matching pipeline deploying either SHOT or Spin Images the best detector choices are MeshDoG and the KPQ variants (but efficiency should also be taken into account, see next section). Moreover, from the figure it can be noted that for tight values of the matching threshold (i.e. around the origin of each chart) also HKS and LSP—!the latter only in the *Spacetime* experiment—show a trend above the bisector when paired with SHOT, thus being able to detect a set of True Positives against a small number of False Positives. This indicates that these methods can be usefully employed within their respective application scenarios and whenever such a small number of good correspondences suffices. Finally, LBSS demonstrates insufficient performance, hence being not suitable to be used as detector within a pipeline deploying either of the two descriptors.

4.4 Efficiency

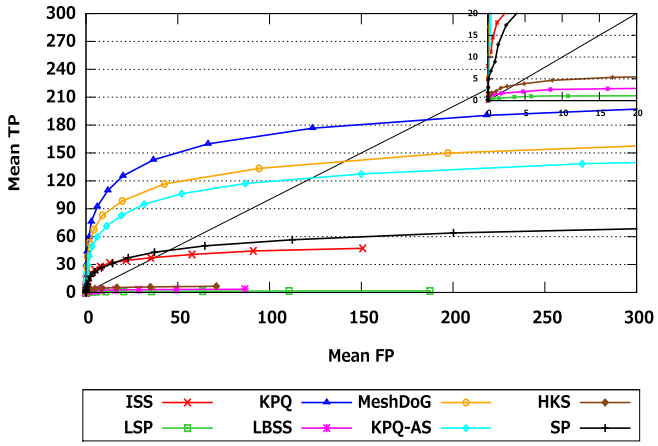
The charts in Figs. 18(a) and 18(b) report the measured detection time (in seconds) of each method computed on one scene of, respectively, the *Retrieval* and the *Kinect* dataset. As for fixed-scale detectors, the value of the radius yielding the best performance in terms of repeatability on each dataset was used in this experiment. Since all methods are implemented in the same framework, their efficiency is comparable and the charts also provide realistic absolute execution times for typical surface sizes used in the application scenarios addressed by each of the two datasets. It has to be noted that, for the aforementioned computational limits



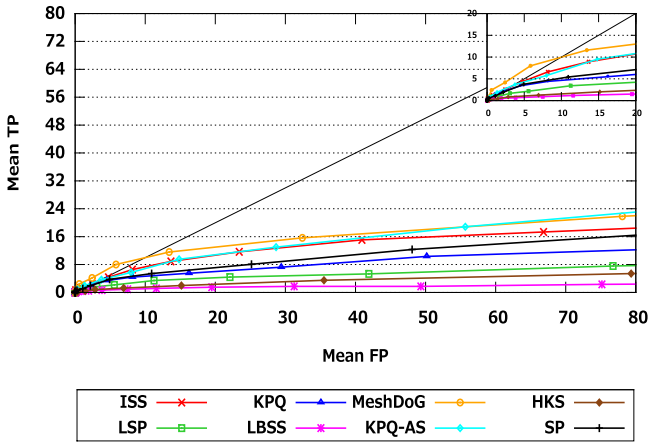
(a) *Retrieval* dataset, SHOT descriptor



(b) *SpaceTime* dataset, SHOT descriptor



(c) *Retrieval* dataset, Spin Image descriptor



(d) *SpaceTime* dataset, Spin Image descriptor

Fig. 17 Results for the *descriptor matching* experiment using SHOT (*top*) and Spin Images (*bottom*). Mean False Positive (FP) and True Positive (TP) are reported on the x and y axis, respectively. The *black dashed line* represents the line $TP = FP$. Curves are drawn by varying the threshold used to consider the nearest neighbor of a descriptor as a match to establish a point-to-point correspondence between surfaces. At the top-right corner of all figures, a zoomed view of the charts near the origin of the axis is shown

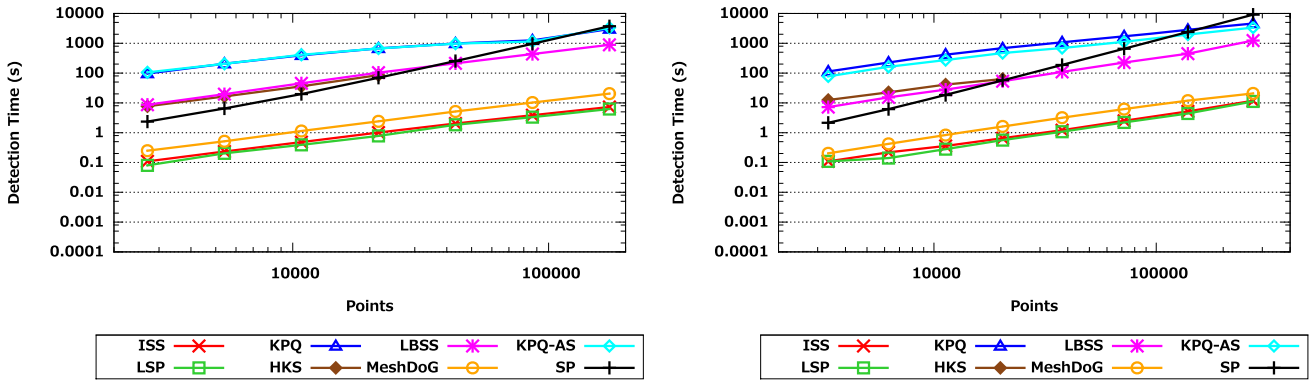
Table 6 Best overall detectors according to the different aspects evaluated throughout the comparison

	Rel. repeat.	Abs. repeat.	Scale repeat.	Robustness to noise	Distinctiveness	Efficiency
Retrieval	HKS	KPQ, KPQ-AS	LBSS	KPQ, HKS, KPQ-AS	KPQ	LSP, ISS
Recognition	KPQ-AS, ISS, MeshDoG	MeshDoG, KPQ-AS	LBSS	KPQ, HKS, KPQ-AS	MeshDoG, KPQ-AS	LSP, ISS

due to the memory footprint of HKS, its efficiency could not be measured with surfaces containing more than 30 K points.

The relative ranking is overall the same for both datasets. The most efficient detectors are ISS and LSP, yielding comparable efficiency and being slightly faster than MeshDoG and two orders of magnitude faster than LBSS and HKS. Conversely, KPQ and KPQ-AS are the most computationally intensive approaches, being one order of magnitude slower than LBSS and HKS. Finally, SP exhibits a non-linear trend, thus resulting slightly slower than the most efficient approaches with small surfaces, while being as slow as the least efficient approaches on scenes characterized by a high number of points.

It is worth pointing out that with the typical number of points computed by the Kinect sensor (of the order of 10^6), none of the detectors is able to run below one second per scene.



(a) *Retrieval* dataset (b) *Kinect* dataset

Fig. 18 Measured detection time (in seconds) on one scene of the *Retrieval* (*left*) and *Kinect* (*right*) dataset. As for fixed-scale detectors, the best radius was selected for each dataset. The scenes were repeatedly decimated to reduce the amount of vertices, thus enabling the analysis of the detectors' efficiency for varying numbers of surface points

5 Discussion and Conclusions

This paper has proposed an in-depth survey of the state of the art of 3D keypoint detectors and a thorough comparison of their performance. Beside reviewing and presenting to the reader in a comfortable common place the theory behind current methods, the survey introduces a classification between fixed-scale and adaptive-scale detectors that allows for defining the common structure of detectors belonging to each class. Abstracting all detectors to a set of common stages was useful to better highlight the peculiarities of the various approaches and understand their differences in performance throughout the analysis of the reported experimental results. We believe that this classification and abstraction can help researchers to highlight the blind spots of the path followed so far to devise repeatable detectors, either to improve the current methodology or to propose a completely different, ground-breaking approach. Another contribution of the paper is the carefully designed evaluation methodology and the definition of a varied and rich data corpus, that is made publicly available. This would allow researchers to evaluate their proposals on challenging data and compare their results to the state of the art without the need to re-implement existing methods.

The proposed experimental assessment allows for drawing interesting conclusions. In Table 6 we have highlighted the algorithms that, for the two addressed scenarios, provided the best overall performance according to the traits considered throughout the evaluation. As for the shape retrieval scenario, KPQ can be recommended as the best technique in terms of repeatability, distinctiveness (according to the considered 3D descriptors) and robustness to noise. ISS, on the other hand, features a good trade-off between absolute and relative repeatability, together with the important advantage of being highly efficient. As far as object recognition is concerned, the most repeatable methods turn out to be adaptive scale detectors, in particular MeshDoG and KPQ-AS, the latter excelling in terms of robustness to noise, as well as distinctiveness. Still, ISS can deliver reasonably good results in terms of repeatability while being particularly efficient, so that it represents a viable choice as long as data are not too noisy.

As for other detectors, LBSS's main positive trait is scale repeatability, while LSP's is efficiency. HKS is especially suited to object retrieval scenarios, where it yields excellent relative repeatability and robustness to noise. SP can offer interesting performance for shape retrieval provided that sensor noise is limited.

Our performance evaluation clearly highlights two main open issues. Firstly, the temporal and/or spatial efficiency of all existing methods is unsatisfactory. This severely limits their deployment in real applications and represents a major bottleneck for reaching the same maturity and spread as 2D applications based on local image features. The efficiency problem may become especially important with the likely spreading of portable devices able to capture and process 3D data. Secondly, there are nuisances peculiar to 3D data, i.e. point density variations among data representing the same object and the difference in dimensionality between models and scenes, that affect the performance of the considered methods. Both these nuisances can severely decrease the performance of every detector. We believe that future research efforts should be aimed at devising well grounded and efficient solutions to these open problems.

Acknowledgements The authors would like to thank Dr. Andrei Zaharescu, Dr. Jian Sun and Dr. Umberto Castellani for sharing the code of their proposals and being always kindly ready to answer our doubts and recall some details that were essential for carrying out this evaluation. The authors would also like to thank Maurizio Galassi, who assisted them in adapting the SP detector to their evaluation framework, Filippo Pulga, who helped providing the C++ implementation of several detectors, and Thomas Ricci, who worked hard to port to C++ the gridfit script.

References

Akagunduz, E., & Ulusoy, I. (2007). 3D object representation using transform and scale invariant 3D features. In *Proc. int. conf. on computer vision (ICCV)* (pp. 1–8).

Akgül, C., Sankur, B., Yemez, Y., & Schmitt, F. (1992). 3D model retrieval using probability density-based shape descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*(6), 1117–1133.

Bariya, P., & Nishino, K. (2010). Scale-hierarchical 3D object recognition in cluttered scenes. In *Proc. int. conf. on computer vision and pattern recognition (CVPR)* (pp. 1657–1664).

Beis, J., & Lowe, D. (1997). Shape indexing using approximate nearest-neighbour search in high dimensional spaces. In *Proc. int. conf. on computer vision and pattern recognition (CVPR)* (pp. 1000–1006).

Boyer, E., Bronstein, A. M., Bronstein, M. M., Bustos, B., Darom, T., Horaud, R., Hotz, I., Keller, Y., Keustermans, J., Kovnatsky, A., et al. (2011). SHREC 2011: robust feature detection and description benchmark. In *Eurographics workshop on shape retrieval* (vol. *2*, pp. 79–86).

Bronstein, M., & Kokkinos, I. (2010). Scale-invariant heat kernel signatures for non-rigid shape recognition. In *Proc. int. conf. on computer vision and pattern recognition* (pp. 1704–1711).

Bronstein, A. M., Bronstein, M. M., Bustos, B., Castellani, U., Crisani, M., Falcidieno, B., Guibas, L. J., Kokkinos, I., Murino, V., Ovsjanikov, M., Patané, G., Sipiran, I., Spagnuolo, M., & Sun, J. (2010). SHREC 2010: robust feature detection and description benchmark. In *EUROGRAPHICS workshop on 3D object retrieval (3DOR)*.

Castellani, U., Cristani, M., & Fantoni, S. (2008). Sparse points matching by combining 3D mesh saliency with statistical descriptors. In *Proc computer graphics forum* (pp. 643–652).

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

Chua, C. S., & Jarvis, R. (1997). Point signatures: a new representation for 3D object recognition. *International Journal of Computer Vision*, *25*(1), 63–85.

D'Errico, J. (2010). *Surface fitting using gridfit*. MATLAB Central File Exchange.

Dorai, C., & Jain, A. (1997). Cosmos-a representation scheme for 3D free-form objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(10), 1115–1130.

Fadaifard, H., & Wolberg, G. (2011). Multiscale 3D feature extraction and matching. In *Proc. int. conf. on 3D imaging, modeling, processing, visualization and transmission (3DIMPVT)* (pp. 228–235).

Frome, A., Huber, D., Kolluri, R., Bülow, T., & Malik, J. (2004). Recognizing objects in range data using regional point descriptors. In *Proc. Europ. conf. on computer vision (ECCV)* (vol. *3*, pp. 224–237).

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

Katz, S., Tal, A., & Basri, R. (2007). Direct visibility of point sets. *ACM Trans on Graphics*, *26*(3) 24.

Knopp, J., Prasad, M., Willems, G., Timofte, R., & Van Gool, L. (2010). Hough transform and 3D SURF for robust three dimensional classification. In *Proc. Europ. conf. on computer vision (ECCV)*.

Lindeberg, T. (1998). Feature detection with automatic scale selection. *International Journal of Computer Vision*, *30*(2), 79–116.

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

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

Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., & Van Gool, L. (2005). A comparison of affine region detectors. *International Journal of Computer Vision*, *65*(1–2), 43–72.

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

Novatnack, J., & Nishino, K. (2007). Scale-dependent 3D geometric features. In *Proc. int. conf. on computer vision (ICCV)* (pp. 1–8).

Novatnack, J., & Nishino, K. (2008). Scale-dependent/invariant local 3D shape descriptors for fully automatic registration of multiple sets of range images. In *Proc. Europ. conf. on computer vision (ECCV)* (pp. 440–453).

Salti, S., Tombari, F., & Di Stefano, L. (2010). On the use of implicit shape models for recognition of object categories in 3D data. In *Proc. Asian conf. on computer vision (ACCV)* (pp. 653–666).

Salti, S., Tombari, F., & Di Stefano, L. (2011). A performance evaluation of 3D keypoint detectors. In *Proc. int. conf. on 3D imaging, modeling, processing, visualization and transmission (3DIMPVT)* (pp. 236–243).

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*(2–3), 211–228.

Shilane, P., Min, P., Kazhdan, M., & Funkhouser, T. (2004). The Princeton shape benchmark. In *Proc. shape modeling international* (pp. 167–178).

Sun, J., Ovsjanikov, M., & Guibas, L. (2009). A concise and provably informative multi-scale signature based on heat diffusion. In *Proc. Eurographics symposium on geometry processing (SGP)* (pp. 1383–1392).

Tombari, F., Salti, S., & Di Stefano, L. (2010). Unique signatures of histograms for local surface description. In *Proc. Europ. conf. on computer vision (ECCV)* (pp. 356–369). Berlin: Springer.

Unnikrishnan, R., & Hebert, M. (2008). Multi-scale interest regions from unorganized point clouds. In *Proc. workshop on search in 3D (S3D)* (pp. 1–8).

Yoshizawa, S., Belyaev, A., & Seidel, H. P. (2004). A fast and simple stretch-minimizing mesh parameterization. In *Proc. shape modeling and applications* (pp. 200–208).

Zaharescu, A., Boyer, E., Varanasi, K., & Horaud, R. (2009). Surface feature detection and description with applications to mesh matching. In *Proc int conf on computer vision and pattern recognition (CVPR)* (pp. 373–380).

Zhong, Y. (2009). Intrinsic shape signatures: a shape descriptor for 3D object recognition. In *Proc. int. conf. on computer vision workshops* (pp. 1–8).