3D Object Recognition in Cluttered Scenes with Local Surface Features: A Survey

Yulan Guo, Mohammed Bennamoun, Ferdous Sohel, Min Lu, and Jianwei Wan

Abstract—3D object recognition in cluttered scenes is a rapidly growing research area. Based on the used types of features, 3D object recognition methods can broadly be divided into two categories—global or local feature based methods. Intensive research has been done on local surface feature based methods as they are more robust to occlusion and clutter which are frequently present in a real-world scene. This paper presents a comprehensive survey of existing local surface feature based 3D object recognition methods. These methods generally comprise three phases: 3D keypoint detection, local surface feature description, and surface matching. This paper covers an extensive literature survey of each phase of the process. It also enlists a number of popular and contemporary databases together with their relevant attributes.

Index Terms—3D object recognition, keypoint detection, feature description, range image, local feature

1 Introduction

Object recognition in cluttered scenes is a fundamental research area in computer vision. It has numerous applications, such as intelligent surveillance, automatic assembly, remote sensing, mobile manipulation, robotics, biometric analysis and medical treatment [1], [2], [3], [4], [5]. In the last few decades, 2D object recognition has been extensively investigated and is currently a relatively mature research area [6]. Compared to 2D images, range images have shown to exhibit several advantages for object recognition. For instance, (i) range images provide more geometrical (depth) information compared to 2D images. Range images also encode surface metric dimensions unambiguously. (ii) Features extracted from range images are commonly not affected by scale, rotation and illumination [7]. (iii) An estimated 3D pose of an object from range images is more accurate compared to an estimated pose from 2D images. Accordingly, range images have the potential to overcome many of the difficulties faced by 2D images in the context of object recognition [8]. These advantages make 3D object recognition an active research topic [9]. Moreover, the rapid technological development of low cost 3D acquisition systems (e.g., Microsoft Kinect) make range images more accessible [10], [11], [12]. Furthermore, advances in computing devices enable the processing of any computationally intensive 3D object recognition algorithm to run in a fairly acceptable manner. All these combined factors have contributed to the flourishment of research towards the development of 3D object recognition systems.

Existing 3D object recognition methods can be divided into two broad categories: global feature based methods and local feature based methods [13], [14]. The global feature based methods process the object as a whole for recognition. They define a set of global features which effectively and concisely describe the entire 3D object (or model) [15]. These methods have been widely used in the context of 3D shape retrieval and classification [16], [17]. Examples in this category include geometric 3D moments [18], shape distribution [19], viewpoint feature histogram [20], and potential well space embedding [21]. They, however, ignore the shape details and require a priori segmentation of the object from the scene [11]. They are therefore not suitable for the recognition of a partially visible object from cluttered scenes [14]. On the other hand, the local feature based methods extract only local surfaces around specific keypoints. They generally handle occlusion and clutter better compared to the global feature based methods [14]. This type has also proven to perform notably better in the area of 2D object recognition [22]. This conclusion has also been extended to the area of 3D object recognition [23]. On that basis, the focus of this paper is on 3D object recognition in cluttered scenes with local surface features.

Several survey papers were published on 3D object recognition and its related fields, such as 3D shape matching and modeling. Among these, several survey papers on 3D object recognition are also available. For instance, the articles in [24], [25], [26], [27], [28] and [29]. Two reviews on 3D modeling and range image registration by [30] and [31] are also worth mentioning. However, none of these papers specifically focuses on local feature based 3D object recognition.

Bronstein et al. [32] presented a review on keypoint detection and local surface feature description methods. They however, covered only four keypoint detectors and seven feature descriptors. An article on the evaluation of

- *Y. Guo is with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha, Hunan 410073, China, and with the School of Computer Science and Software Engineering, The University of Western Australia, Perth, WA 6009, Australia. E-mail: yulan.guo@nudt.edu.cn.*
- *M. Bennamoun and F. Sohel are with the School of Computer Science and Software Engineering, The University of Western Australia, Perth, WA 6009, Australia.*
- *M. Lu and J. Wan are with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha, Hunan 410073, China.*

Manuscript received 6 May 2013; revised 24 Nov. 2013; accepted 3 Apr. 2014. Date of publication 10 Apr. 2014; date of current version 9 Oct. 2014. Recommended for acceptance by A. Fitzgibbon. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TPAMI.2014.2316828

TABLE 1
Popular Range Image Databases

No.	Name and Reference	Data Type	Acquisition	#Objects	#Images	Var.	Color
1	OSU [49]	Point Cloud	Minolta Vivid	-	> 1000	o	Not Available
2	Stuttgart [50]	Mesh	Synthetic	45	11610	o	Not Available
3	UWA [38]	Mesh	Minolta Vivid	5	50	o, c	Not Available
4	Queen's LIDAR [51]	Point Cloud	NextEngine	5	80	o, c	Not Available
5	Queen's Stereo [51]	Point Cloud	Stereo	5	365	o, c	Not Available
6	UBC VRS [52]	Point Cloud	-	603	60	o, c	Available
7	Bologna 1&2 [3]	Mesh	Synthetic	6	45	c	Not Available
8	Bologna 3 [3]	Mesh	Spacetime	8	15	o, c	Available
9	Bologna 4 [53]	Mesh	Spacetime	8	16	o, c	Available
10	Bologna 5 [33]	Mesh	Spacetime	7	17	o, c	Available
11	RGB-D Object [8]	Point Cloud, Depth Image	Kinect Style	300	900 videos	o	Available
12	RGB-D Scenes [8]	Point Cloud, Depth Image	Kinect Style	-	8 videos	o, c	Available
13	B3DO [54]	Depth Image	Kinect	50	850	o, c	Available
14	RGB-D SLAM [55]	Depth Image	Kinect	-	videos	o, c	Available
15	Ca' Foscari Venezia [56]	Mesh	Synthetic	20	150	o, c	Not Available
16	TUWIEN Kinect [11]	Point Cloud	Kinect	35	50	o, c	Not Available

keypoint detection methods has just been published [33]. However, this article only covers eight keypoint detection methods. A large number of significant research contributions on local feature based methods is available and there is no review article which comprehensively analyzes these methods.

Compared with the existing literature, the main contributions of this paper are as follows:

i) To the best of our knowledge, this is the first survey paper in the literature that focuses on *3D object recognition based on local surface features.*
ii) As opposed to previous reviews, e.g., in [32] and [33], we adequately cover the contemporary literature of keypoint detection and local surface description methods. We present a *comprehensive review* of 29 keypoint detection and 38 feature description methods.
iii) This paper also provides an insightful analysis on all aspects of surface matching, including feature matching, hypothesis generation and verification.
iv) Compared to the earlier surveys, this paper covers the *most recent and advanced* work. It therefore, provides the reader with the state-of-the-art methods.
v) A comparative summary of attributes is reported in tabular forms (e.g., Tables 3, 4 and 5).

The rest of this paper is organized as follows. Section 2 describes the background concepts and terminology of 3D object recognition based on local surface features. Sections 3 and 4 provide a comprehensive survey of the existing methods for 3D keypoint detection and local surface feature description, respectively. Section 5 presents a review of 3D object recognition methods. Section 6 presents a brief discussion on potential future research directions. Finally, Section 7concludes this paper.

2 Background Concepts and Terminology

2.1 Background Concepts

A *range image* can be represented in three types, namely a depth image, a point cloud or a polygonal mesh. Given a range image, the goal of *3D object recognition* is to correctly identify objects present in the range image, and determine their poses (i.e. positions and orientations) [34].

At a conceptual level, a typical local feature based 3D object recognition system consists of three main phases: 3D keypoint detection, local surface feature description and surface matching. In the *3D keypoint detection* phase, the 3D points with rich information content are identified as keypoints. The inherent scale of each keypoint is also detected. Both the location and scale (i.e., neighborhood size) of a keypoint define a local surface which is used in the subsequent feature description phase [35]. In the *local surface feature description* phase, the geometric information of the neighborhood surface of the keypoint is encoded into a representative feature descriptor. During the *surface matching* phase, the scene features are matched against all model features in the library, resulting in a set of feature correspondences and hypotheses. These hypotheses are finally verified to infer the identity and pose of the object.

2.2 Databases and Evaluation

Many databases have been built to test various algorithms. A set of popular 2.5D range image and 3D model databases are enlisted together with their major attributes in Tables 1 and 2, respectively. The variations (var.), including occlusion (o), clutter (c) and deformation (d), in each database are also provided. The symbol '-' denotes that the corresponding item is not reported. In addition to range images/models, registered 2D color (usually RGB) images are also simultaneously provided in several databases (shown in Tables 1 and 2).

There are series of evaluation criteria which are frequently used to assess the performance of each phase of a 3D object recognition system [3], [36], [37], [38]. *Repeatability score* is a frequently used criteria for 3D keypoint detector evaluation. It is computed as the ratio of the number of corresponding keypoints to the minimum number of keypoints in the two images [33], [39], [40], [41], [42]. *Recall vs* 1 *-Precision* is a frequently used criteria for local surface feature descriptor evaluation. It is generated by varying the thresholds for feature matching and calculating the feature recall and precision under each threshold [3], [43], [44], [45], [46]. *Recognition rate* is commonly used to evaluate the overall performance of a recognition system [38], [44], [47], [48].

TABLE 2
Popular 3D Model Databases

No.	Name and Reference	Data Type	Data Size	#Objects	Var.	Color
1	Stanford 3D Scanning Repository [57]	Mesh	881-14M vertices	9		Not Available
2	PSB [58]	Mesh	-	1814		Not Available
3	Nonrigid World [59]	Mesh	~3k vertices	148	d	Not Available
4	TOSCA [17]	Mesh	~50K vertices	80	d	Not Available
5	SHREC 2011 [60]	Mesh	~50K vertices	1237	d	Not Available
6	SHREC Photometric [61]	Mesh	-	270	d	Available
7	PhotoMesh [62]	Mesh	-	3	d	Available

3 3D Keypoint Detection

Keypoint detection is the first major phase of a local surface feature based 3D object recognition system. The simplest keypoint detection methods are surface sparse sampling and mesh decimation [38], [63], [64]. However, these methods do not result in qualified keypoints in terms of repeatability and informativeness. That is because they give no or little consideration to the richness of discriminative information of these detected keypoints [4]. Therefore, it is necessary to detect keypoints according to their distinctiveness.

Based on whether the scale is predetermined or adaptively detected, keypoint detection methods can be classified into two categories: fixed-scale keypoint detection methods and adaptive-scale keypoint detection methods. We adopt the same classification as [33] in this paper.

3.1 Fixed-Scale Keypoint Detection

Fixed-scale keypoint detection methods define a point, which is distinctive within a predetermined neighborhood, as a keypoint. The neighborhood size is determined by the scale, which is an input parameter to the algorithm [35]. As described in the following subsections, distinctiveness measures can either be curvatures or other surface variation (OSV) measures.

3.1.1 Curvature Based Methods

These methods use different curvatures as distinctiveness measures to detect keypoints.

Mokhtarian et al. [65] detected keypoints using the Gaussian and mean curvatures. They declared a point $\boldsymbol{p}$ as a keypoint if its curvature value was larger than the curvature values of its 1-ring neighbors (k-ring neighbors are defined as the points which are distant from $\boldsymbol{p}$ by k edges). Yamany and Farag [66] used simplex angles to detect keypoints. A simplex angle φ is related to the mean curvature. The keypoints are detected at the locations where the simplex angles satisfy the constraint $|\sin(\varphi)| \geq \tau$. Their threshold τ is crucial for the performance of their keypoint detection, and choosing an appropriate threshold is still an unresolved issue [66]. Gal and Cohen-Or [67] proposed a saliency grade for keypoint detection. A saliency grade for a point $\boldsymbol{p}$ is a linear combination of two terms. The first term is the sum of the curvatures of its neighboring points, and the second term is the variance of the curvature values in the neighborhood. The points with high saliency grades are selected as keypoints. Chen and Bhanu [68] detected keypoints based on shape index values. That is, within a neighborhood, the point $\boldsymbol{p}$ is marked as a keypoint only if its shape index value is a local optimum (maximum/minimum). Experimental results showed that the detected keypoints were quite uniformly distributed over the surface [33]. However, this method is sensitive to noise [33].

3.1.2 Other Surface Variation Measure Based Methods

These methods use other surface variation measures rather than just curvatures to detect keypoints.

Matei et al. [69] used the smallest eigenvalue λ_3 of the covariance matrix of the neighboring points to measure the surface variation around a point $\boldsymbol{p}$. The points were sorted according to their surface variations. Zhong [70] employed the ratio of two successive eigenvalues to prune the points. Only the points which satisfy the constraints $\frac{\lambda_2}{\lambda_1} < \tau_{21}$ and $\frac{\lambda_3}{\lambda_2} < \tau_{32}$ are retained and further detected based on the smallest eigenvalue λ_1. Guo et al. [44] first decimated a range image and then chose the points, which satisfied the constraint $\frac{\lambda_1}{\lambda_2} > \tau$ from the decimated image, as keypoints. These methods achieve good results in terms of repeatability. Moreover, they are particularly computationally efficient [33].

Glomb [71] introduced four propositions to extend the popular Harris detector [72] from 2D images to 3D meshes. They found that the Harris detector, which used the derivative of a fitted quadratic surface, achieved the best results. Following this proposition, Sipiran and Bustos [40] proposed a "Harris 3D" detector. Given a point $\boldsymbol{p}$, the neighboring points were first translated to the centroid and then rotated to align the normal at $\boldsymbol{p}$ with the z axis. Next, these transformed points were fitted into a quadratic surface, described mathematically by $f(u,v) = a^{\mathrm{T}}(u^2, uv, v^2, u, v, 1)$. A symmetric matrix $\mathbf{E}$ was then defined using the derivatives of this function. The Harris 3D operator value at the point $\boldsymbol{p}$ was calculated as $V(\boldsymbol{p}) = \det(\mathbf{E}) - \alpha(\mathrm{tr}(\mathbf{E}))^2$, where $\det(\mathbf{E})$ and $\mathrm{tr}(\mathbf{E})$ represented the determinant and trace of the matrix $\mathbf{E}$, respectively. α was a parameter which needs to be tuned experimentally. Finally, a fixed percentage of points with the largest values of $V(\boldsymbol{p})$ were selected as keypoints. Experimental results showed that it was robust to several transformations including noise, change of tessellations, local scaling, shot noise and presence of holes. It also outperformed [73] (described in Section 3.2.4) and [15] (described in Section 3.2.1) in many aspects, especially in the cases of high levels of local scaling and shot noise.

Since only a fixed-scale is used to find the keypoints, their implementation is straightforward. However, these methods have several major drawbacks. First, it is possible that they may detect too few keypoints, particularly on the less curved parts of the 3D object [4]. This would be a problem for object recognition. Second, fixed-scale methods determine the scale empirically. They do not

fully exploit the scale information encoded in the local geometric structures to detect the inherent scale of a keypoint. Therefore, the neighborhood size cannot be adaptively determined [48].

3.2 Adaptive-Scale Keypoint Detection

Adaptive-scale keypoint detection methods first build a scale-space for a given range image. They then pick the points with extreme distinctiveness measures in both the spatial and scale neighborhoods as keypoints. As a result, both the locations and scales of the keypoints are detected. According to the scale-space construction technique, these methods can be divided into four categories: coordinate smoothing (CS), geometric attribute smoothing (GAS), surface variation and transform based methods.

3.2.1 Coordinate Smoothing Based Methods

These methods construct a scale-space by successively smoothing the 3D coordinates of a range image. They are broadly based on the 2D scale-space theory first introduced in [74].

Akagündüz and Ulusoy [75] first obtained the scale-space of a 3D surface by constructing a Gaussian pyramid of the surface. They then computed the mean and Gaussian curvature values at all points for all scales, and classified each point as one of the eight surface types based on its Gaussian and mean curvatures [24]. Within the classified scale-space, each connected volume that had the same surface type was detected. The center of the connected volume was chosen as the location of a keypoint. The weighted average of the scale values within each connected volume was selected as the scale of that keypoint. This algorithm is invariant to scale and rotation. The results showed that for scale varying databases, the adaptive-scale features ensured a superior recognition performance to the fixed-scale features [13].

Castellani et al. [15] resampled the mesh $\mathcal{M}$ at N_O levels to yield a set of octave meshes $\mathcal{M}^j(j = 1, 2, \ldots, N_O)$. They then applied N_S Gaussian filters on each octave mesh $\mathcal{M}^j$ to obtain a set of filtering maps $\mathcal{F}_i^j(i = 1, 2, \ldots, N_S)$. Next, they projected $\mathcal{F}_i^j(\boldsymbol{p})$ to the normal of $\boldsymbol{p}$ to get a scalar map $\mathcal{M}_i^j(\boldsymbol{p})$, which was further normalized to an inhibited saliency map $\hat{\mathcal{M}}_i^j$. Finally, keypoints were detected as the local maxima within the inhibited saliency map $\hat{\mathcal{M}}_i^j$. Only the potential keypoints which appeared in at least three octaves were considered as validate keypoints. Experimental results showed that this method detected only a limited number of keypoints, which were well-localized and often at the extremities of long protrusions of the surface [15], [33]. Darom and Keller [76] followed the work of [15]. They however used density-invariant Gaussian filters to obtain octave meshes, which was more robust to varying mesh resolutions and nonuniform sampling compared to [15].

Li and Guskov [77] projected the 3D points onto a series of increasingly smoothed versions of the shape to build a scale-space. They then computed the normal differences between neighboring scales at each point $\boldsymbol{p}$. The points whose normal differences were larger (or smaller) than all of their spatial and scale neighbors were detected as keypoints. Experimental results showed that the keypoints at coarse scales were more repeatable compared to their counterparts at finer scales.

Lo and Siebert [78] detected keypoints on a depth image using an enhanced version of the Scale Invariant Feature Transform (SIFT) algorithm [22], namely 2.5D SIFT. They created a discrete scale-space representation of the depth image by using Gaussian smoothing and Difference Of Gaussian (DOG) pyramid techniques. The signal maxima and minima were detected within the DOG scale-space. The keypoints were finally validated and located by comparing the ratio of the principal curvatures (i.e., $\frac{k_1}{k_2}$) with a predefined threshold. The 2.5D SIFT achieved a superior matching performance compared to the 2D SIFT [78].

Knopp et al. [23] first voxelized a mesh to a 3D voxel image. Next, they calculated the second-order derivatives $\mathcal{L}(\boldsymbol{v}, \sigma)$ at each voxel $\boldsymbol{v}$ using box filters with increasing standard deviations σ. They defined a saliency measure s for each voxel $\boldsymbol{v}$ and scale σ based on the Hessian matrix. Local extrema over the space $s(\boldsymbol{v}, \sigma)$ were used to detect the "3D SURF" keypoints and their corresponding scales.

Coordinate smoothing based methods directly apply the 2D scale-space theory to 3D geometric data by replacing pixel intensities with 3D point coordinates. Consequently, the extrinsic geometry and topology of a 3D shape are altered, and the causality property of a scale-space is violated [79]. The causality property is an important axiom for a scale-space representation. It implies that any structure (feature) in a coarse scale should be able to find its cause in the finer scales [80].

3.2.2 Geometric Attribute Smoothing Based Methods

These methods construct a scale-space by successively smoothing the geometric attributes of a range image. Since the filtering is applied to the geometric attributes rather than the range image itself, no modification is made to the extrinsic geometry of the 3D shape. Therefore, the causality property of the scale-space is preserved.

Novatnack and Nishino [81] represented a surface using its normals and parameterized the surface on a 2D plane to obtain a dense and regular 2D representation. They then constructed a scale-space of the normal field by successively convolving the 2D normal map with geodesic Gaussian kernels of increasing standard deviations. Keypoints and their corresponding scales were detected by identifying the points in the scale-space where the corner responses were maxima along both the spatial and scale axes. This work was one of the first to consider a geometric scale space that can directly be used on range images, preceding several methods including [82] and [83]. Experimental results showed that the keypoints could be detected and localized robustly, and the number of detected keypoints was sufficiently large [48], [84]. One major limitation of this method is that it requires accurate estimations of the surface normals to construct the 2D scale-space [41], [85].

Flint et al. [86] convolved the 3D voxel image of a range image with a set of Gaussian kernels to build a density scale-space. The keypoints were detected over the scale-space using the determinant of the Hessian matrix. Tests on

a number of scenes showed that, this THRIFT method can repeatably extract the same keypoints under a range of transformations. However, it is sensitive to the variations of point density [87]. Moreover, regular resampling in a 3D space throughout the data is very time-consuming [88].

Hua et al. [89] first mapped a 3D surface to a canonical 2D domain by using a non-linear optimization method. They then encoded the surface curvatures and conformal factors into the rectangular 2D domain, resulting in a shape vector image. Next, they built a vector-valued scale-space on the shape vector image through a geodesic distance-weighted inhomogeneous linear diffusion and a Difference of Diffusion (DoD) computation. They detected keypoints as the points that had maxima/minima DoD values across the scales in both the curvature and conformal factor channels. Experiments showed that this method achieved a very high repeatability. It was also superior to the regular anisotropic diffusion and isotropic diffusion methods [89]. However, it is difficult to apply this method to large and topologically complicated surfaces [83], [90] and certain high-genus surfaces (i.e., surfaces which have a large number of holes.) [89].

Zou et al. [90] proposed a Geodesic Scale-Space (GSS) based on the convolution of a variable-scale geodesic Gaussian kernel with the surface geometric attributes. Keypoints were detected by searching the local extrema in both the spatial and scale domains in the GSS. These keypoints were further pruned based on their contrasts and anisotropies. Experiments demonstrated that this method was robust to noise and varying mesh resolutions. However, the cost of computing geodesic distances is extremely high when the scale increases [83].

Zaharescu et al. [91] first defined a scalar field (photometric or geometric attribute) for each point $\boldsymbol{p}$. They then convolved the scalar field with a set of geodesic Gaussian kernels and performed DOG calculations to obtain their scale-space. MeshDOG keypoints were claimed as the maxima in the DOG scale-space. These keypoints were finally pruned by a set of operations including non-maximum suppression, thresholding and corner detection. This method offers a canonical formula to detect both photometric and geometric keypoints on a mesh surface. It is capable to detect a sufficient number of repeatable keypoints. It is however, sensitive to varying mesh resolutions [33], [83].

Zou et al. [82] formalized an Intrinsic Geometric Scale-Space (IGSS) of a 3D surface by gradually smoothing the Gaussian curvatures via shape diffusion. Keypoints were identified as extrema in the normalized Laplacian of the IGSS with respect to both the spatial and scale domains. The IGSS representation is invariant to conformal deformations (i.e., transformations which preserve both the size and the sign of angles). Experimental results showed that the detected keypoints spread across various scales and were robust to noise.

Hou and Qin [83] downsampled the surface as the scale increased, and built a scale-space of the scalar field (e.g., curvature and texture) on the surface by using a Gaussian kernel. Keypoints were finally detected as local extrema in the scale-space. The capability of simultaneous sampling in both the spatial and scale domains makes this method efficient and stable. Experimental results showed that this method significantly reduced the processing time compared to GSS.

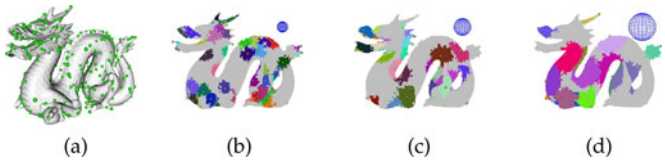


Fig. 1. Keypoints detected on the Dragon model. (a) Keypoints detected by [88]. Different sizes of the spheres correspond to different scales of the keypoints. (b), (c), (d) Keypoints and their neighborhoods detected at three scales by [41]. Each colored patch corresponds to the neighborhood of a keypoint, the sizes of blue spheres correspond to the scales.

It is also more stable under different mesh resolutions than MeshDOG. Another merit is its invariance to isometric deformations (i.e., transformations which preserve distance).

3.2.3 Surface Variation Based Methods

These methods first calculate surface variations at a set of varying neighborhood sizes. They then detect keypoints by finding the maxima of surface variations in the local neighborhood with different neighborhood sizes. They are based on the assumption that the neighborhood size can be regarded as a discrete scale parameter, and increasing the local neighborhood size is similar to applying a smoothing filter [92]. These methods avoid making direct changes to the 3D surfaces, and they are straightforward to implement.

Pauly et al. [92] measured the surface variation δ by using the eigenvalues λ_1, λ_2 and λ_3 of the covariance matrix of the neighboring points, that is $\delta = \frac{\lambda_3}{(\lambda_1 + \lambda_2 + \lambda_3)}$. Local maxima in the surface variation space were determined as keypoints. Experiments demonstrated that the surface variation corresponded well to the smoothed surface using the standard Gaussian filtering. Two major drawbacks of this method are that, surface variation is sensitive to noise, and a heuristic pre-smoothing procedure is required to detect the maxima in the surface variation space [41], [88].

Ho and Gibbins [88] used the standard deviation of the shape index values of the neighboring points to measure the surface variation. The detected keypoints on the Dragon model are illustrated in Fig. 1a. Experimental results showed that this method was effective and robust to minor noise. It achieved high repeatability results even with noisy surfaces. Later, Ho and Gibbins [85] estimated the curvedness at different scales, and picked the points that had extreme values in the scale-space as keypoints. Similarly, Ioanou et al. [93] proposed a multi-scale Difference of Normals (DoN) operator for the segmentation of large unorganized 3D point clouds. The DoN operator provides a substantial reduction in points, which reduces the computational cost of any subsequent processing phase of the scene (when processing is performed on the segmented parts).

Unnikrishnan and Hebert [41] proposed an integral operator $B(\boldsymbol{p}, \rho)$ to capture the surface variation at a point $\boldsymbol{p}$ with a neighborhood size ρ. In fact, $B(\boldsymbol{p}, \rho)$ displaces a point $\boldsymbol{p}$ along its normal direction $\boldsymbol{n}$, and the magnitude of displacement is proportional to the mean curvature. They then defined the surface variation $\delta(\boldsymbol{p}, \rho)$ as:

$$\delta(\boldsymbol{p}, \rho) = \frac{2\|\boldsymbol{p} - B(\boldsymbol{p}, \rho)\|}{\rho} \exp\left(-\frac{2\|\boldsymbol{p} - B(\boldsymbol{p}, \rho)\|}{\rho}\right). \quad (1)$$

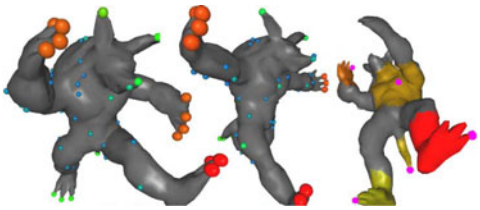


Fig. 2. Keypoints detected on the Armadillo models with different poses, originally shown in [94].

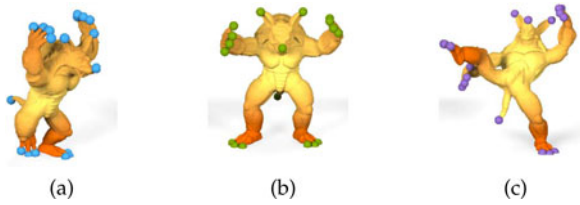


Fig. 3. Keypoints detected on the Armadillo model, originally shown in [73].

An illustration of the keypoints detected at three scales is given in Figs. 1b, 1c, and 1d. This method is very effective in determining the characteristic scales of 3D structures even in the presence of noise. However, its repeatability is relatively low and the number of detected keypoints is small [33]. Stuckler and Behnke [87] used Euclidean distances instead of geodesic distances to calculate the surface variation.

Mian et al. [4] proposed a keypoint detection method together with a keypoint quality measurement technique. They first rotated the neighboring points so that the normal at $\boldsymbol{p}$ was aligned with the positive z axis. Next, they performed principal component analysis (PCA) on the covariance matrix of the neighboring points, and used the ratio between the length along the first two principal axes (i.e., x and y) to measure the surface variation δ. They detected the keypoints by comparing their surface variations with a threshold. The scale for a keypoint was determined as the neighborhood size ρ for which the surface variation δ reached the local maximum. This method can detect sufficient keypoints. These keypoints are repeatable and are also robust to noise. However, one major limitation of this method is its computational inefficiency [33].

3.2.4 Transform Based Methods

These methods consider a surface $\mathcal{M}$ as a Riemannian manifold, and transform this manifold from the spatial domain to another domain (e.g., spectral domain). Subsequently, they detect keypoints in the transformed domain rather than the original spatial domain.

Hu and Hua [94] extracted keypoints in the Laplace-Beltrami spectral domain. Let $f \in \mathbb{C}^2$ be a real function defined on a Riemannian manifold $\mathcal{M}$. Using the Laplacian eigenvalue equation, f can be written as $f = \sum_{i=1}^{N} c_i \Phi_i$, where Φ_i is the ith eigenvector and c_i is related to the ith eigenvalue. The geometry energy of a point $\boldsymbol{p}$ corresponding to the ith eigenvalue is defined as $E_i(\boldsymbol{p}) = \|c_i \times \Phi_i(\boldsymbol{p})\|_2$. The point whose geometry energy is larger than these of its neighboring points within several neighboring frequencies, is picked up as a keypoint. Meanwhile, the spatial scale of a keypoint is provided by the "frequency" information in the spectral domain. An illustration of the detected keypoint on the Armadillo models is given in Fig. 2. Experimental results showed that the keypoints were very stable and invariant to rigid transformations, isometric deformations and different mesh triangulations [94].

Sun et al. [73] restricted the heat kernel to the temporal domain to detect keypoints. Let $\mathcal{M}$ be a compact Riemannian manifold, the heat diffusion process over $\mathcal{M}$ is governed by the heat equation. They restricted the heat kernel $K_t(\boldsymbol{p}, \boldsymbol{q})$ to the temporal domain $K_t(\boldsymbol{p}, \boldsymbol{p})$, and used the local maxima of the function $K_t(\boldsymbol{p}, \boldsymbol{p})$ to find keypoints. Here, the time parameter t is related to the neighborhood size of $\boldsymbol{p}$, and therefore the time parameter provides the information of its inherent scale. This method is invariant to isometric deformations. It usually captures the extremities of long protrusions on the surface [84], as shown in Fig. 3. It is able to detect highly repeatable keypoints and also robust to noise [95]. However, it is sensitive to varying mesh resolutions, and the number of detected keypoints is very small. Besides, it also requires a large computer memory [33].

3.3 Summary

Table 3 gives a summary of the keypoint detection methods. These methods are listed chronologically by year of publication and alphabetically by first author within a given year.

- Most of the recent papers are on adaptive-scale keypoint detection methods. Note again that an adaptive-scale method is able to detect the associated inherent scale of a keypoint. This capability improves the performance of both feature description and object recognition methods.
- Existing methods use different measures to define the neighborhood of a point. These measures include the Euclidean distance, the geodesic distance and the k-rings. The methods based on geodesic distances (e.g., [81], [83], [89], [90]) are invariant to isometric deformations. However, the calculation of a geodesic distance is very time consuming. The Euclidean distance is computationally efficient, but it is sensitive to deformations. In contrast, k-rings (as in [71], [76], [88], [91], [96]) provide a good approximation of the geodesic distance between two points on a uniformly sampled mesh. They are also computationally efficient.
- Many 3D keypoint detection methods are inspired by their successful 2D ancestors. For example, the Harris 3D [40], 2.5D SIFT [78], and 3D SURF [23] detectors are respectively extensions of the 2D Harris [72], SIFT [22], SURF [97] detectors. The successful use of scale space with the SIFT detector [22] also motivated the progress of scale space construction in the case of range images. Analysing the basic ideas behind the successful 2D keypoint detectors may also give us some hints for the development of future 3D keypoints.

4 Local Surface Feature Description

Once a keypoint has been detected, geometric information of the local surface around that keypoint can be extracted

TABLE 3
Methods for 3D Keypoint Detection

No.	Reference	Data Type	Scale	Category (Section)	Performance
1	Mokhtarian et al. 2001 [65]	Mesh	Fixed	Curvature (3.1.1)	Robust on smoothed surfaces
2	Yamany & Farag 2002 [66]	Mesh	Fixed	Curvature (3.1.1)	Can capture most of the object's landmarks
3	Pauly et al. 2003 [92]	Point Cloud	Adaptive	SV (3.2.3)	Sensitive to noise, need a pre-smoothing procedure
4	Li & Guskov 2005 [77]	Mesh	Adaptive	CS (3.2.1)	Keypoints at coarse scales are more repeatable
5	Gal & Cohen-Or 2006 [67]	Mesh	Fixed	Curvature (3.1.1)	Effectively characterizes a local partial shape
6	Matei et al. 2006 [69]	Point Cloud	Fixed	OSV (3.1.2)	Repeatable, computationally efficient
7	Akagunduz & Ulusoy 2007 [75]	Mesh	Adaptive	CS (3.2.1)	Invariant to scale and rotation
8	Chen & Bhanu 2007 [68]	Depth Image	Fixed	Curvature (3.1.1)	Uniformly distributed over a surface, sensitive to noise
9	Flint et al. 2007 [86]	Point Cloud	Adaptive	GAS (3.2.2)	Sensitive to point density variations, time-consuming
10	Novatnack & Nishino 2007 [81]	Mesh	Adaptive	GAS (3.2.2)	Robust to noise and varying mesh resolutions
11	Castellani et al. 2008 [15]	Mesh	Adaptive	CS (3.2.1)	Detects a limited number of but well-localized keypoints
12	Ho & Gibbins 2008 [88]	Mesh	Adaptive	SV (3.2.3)	Repeatable, robust to minor noise
13	Hua et al. 2008 [89]	Mesh	Adaptive	GAS (3.2.2)	Repeatable, difficult to apply to complicated surfaces
14	Unnikrishnan & Hebert 2008 [41]	Point Cloud	Adaptive	SV (3.2.3)	Effective in determining the scale, low repeatability with few detected keypoints
15	Zou et al. 2008 [90]	Mesh	Adaptive	GAS (3.2.2)	Robust to noise and varying mesh resolutions
16	Glomb 2009 [71]	Mesh	Fixed	OSV (3.1.2)	Four propositions for extending 2D Harris to 3D meshes
17	Hu & Hua 2009 [94]	Mesh	Adaptive	Transform (4.3)	Invariant to isometric deformations, robust to different mesh triangulations
18	Lo & Siebert 2009 [78]	Depth Image	Adaptive	CS (3.2.1)	Outperforms the standard 2D SIFT
19	Sun et al. 2009 [73]	Mesh	Adaptive	Transform (4.3)	Invariant to isometric deformations, detects a limited number of highly repeatable keypoints
20	Zaharescu et al. 2009 [91]	Mesh	Adaptive	GAS (3.2.2)	Detects sufficient number of both photometric and geometric keypoints
21	Zhong 2009 [70]	Point Cloud	Fixed	OSV (3.1.2)	Repeatable, computationally efficient
22	Zou et al. 2009 [82]	Mesh	Adaptive	GAS (3.2.2)	Keypoints spread across various scales, robust to noise
23	Hou & Qin 2010 [83]	Mesh	Adaptive	GAS (3.2.2)	Invariant to isometric deformations, computationally efficient
24	Knopp et al. 2010 [23]	Mesh	Adaptive	CS (3.2.1)	Extension of SURF to 3D
25	Mian et al. 2010 [4]	Point Cloud	Adaptive	SV (3.2.3)	Detects sufficient keypoints with high repeatability, time-consuming
26	Sipiran & Bustos 2011 [40]	Mesh	Fixed	OSV (3.1.2)	Robust to noise, tessellation, local scaling, shot noise, holes
27	Stuckler & Behnke 2011 [87]	Depth Image	Adaptive	SV (3.2.3)	Copes with irregular sampling density and occlusion
28	Darom & Keller 2012 [76]	Mesh	Adaptive	CS (3.2.1)	Robust to varying mesh resolutions and nonuniform sampling
29	Guo et al. 2013 [44]	Mesh	Fixed	OSV (3.1.2)	Repeatable, computationally efficient

and encoded into a feature descriptor. According to the approaches employed to construct the feature descriptors, we classify the existing methods into three broad categories: signature based, histogram based, and transform based methods.

4.1 Signature Based Methods

These methods describe the local neighborhood of a keypoint by encoding one or more geometric measures computed individually at each point of a subset of the neighborhood [3].

Stein and Medioni [98] first obtained a circular slice around the keypoint $\boldsymbol{p}$ using the geodesic radius r. They then constructed a local reference frame (LRF) by using the normal $\boldsymbol{n}$ and the tangent plane at the point $\boldsymbol{p}$. Using this frame, they encoded the relationship (angular distance) between the normal at the point $\boldsymbol{p}$ and the normal at each point on the circular slice into a 3D vector (ϕ, ψ, θ). Next, a straight line segment was fitted to this 3D curve, and the curvatures and torsion angles of the 3D segment were encoded as the "splash" descriptor. An illustration of the method is given in Fig. 4a. Experimental results showed that this method is robust to noise.

Chua and Jarvis [99] obtained a contour ζ on the surface by intersecting the surface with a sphere of radius r centered at the keypoint $\boldsymbol{p}$. Then, they fitted a plane to these contour points and translated the plane to $\boldsymbol{p}$. They projected all points on ζ onto this fitted plane to obtain a curve ζ'. Thus, they characterized each point on ζ by two parameters: the signed distance d between the point and its correspondence on ζ', and the clockwise rotation angle θ from a reference direction $\boldsymbol{n}_2$. $\boldsymbol{n}_2$ was defined as the unit vector from $\boldsymbol{p}$ to the projected point on ζ' that had the largest positive distance, as shown in Fig. 4b. The "point signature" descriptor was expressed by a discrete set of values $d(\theta)$. This method does not require any surface derivative and is therefore robust to noise. However, it also has several limitations. First, the reference direction $\boldsymbol{n}_2$ may not be unique. In such a case, multiple signatures could be obtained from the same point $\boldsymbol{p}$ [4]. Second, the point signature is sensitive to varying mesh resolutions [31]. Moreover, computing the intersection of a sphere with the surface is not easy, especially when the surface is represented as a point cloud or a mesh [30].

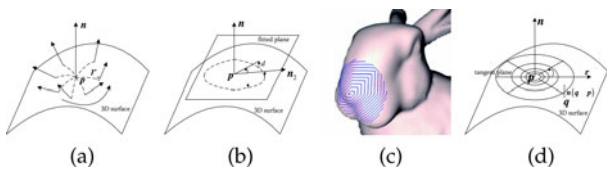


Fig. 4. Illustration of signature based methods. (a) Splash. (b) Point signature. (c) HMM, originally shown in [15]. (d) LPHM.

Sun and Abidi [100] generated geodesic circles around the keypoint $\boldsymbol{p}$ using the points that had the same geodesic distances to $\boldsymbol{p}$. They also constructed an LRF using the normal $\boldsymbol{n}$ and the tangent plane at $\boldsymbol{p}$. They then projected these geodesic circles onto the tangent plane to get a set of 2D contours. These 2D contours together with the normal and radius variations of the points along the geodesic circles formed the "point's fingerprint" descriptor. The advantages of this method are: it carries more discriminative information than the methods which only use one contour (e.g., point signature) or a 2D histogram (e.g., spin image which is described in Section 4.2.1); and the computational cost is cheaper than the methods which use a 2D image representation [100].

Li and Guskov [77] defined $N_1 \times N_2$ grids which sampled a disc around a keypoint $\boldsymbol{p}$ on its tangent plane. They then projected the normals at the neighboring points onto the tangent plane and computed the weighted sum of the surface normals in each grid, resulting in an $N_1 \times N_2$ matrix $\mathbf{A}$. Next, they applied a discrete cosine transform and a discrete Fourier transform to $\mathbf{A}$, resulting in a new matrix $\widetilde{\mathbf{A}}$. They used the elements from the upper left corner (most significant Fourier coefficients) of $\widetilde{\mathbf{A}}$ to form a "Normal Based Signature (NBS)". Object recognition performance of the NBS descriptor outperformed the spin image on the Stuttgart database [101].

Malassiotis and Strintzis [102] first constructed an LRF by performing an eigenvalue decomposition on the covariance matrix of the neighboring points of a keypoint $\boldsymbol{p}$. They then placed a virtual pin-hole camera at a distance d on the z axis and looking toward $\boldsymbol{p}$. The x and y axes of the camera coordinate frame were also aligned with the x and y axis of the LRF at $\boldsymbol{p}$. They projected the local surface points onto the image plane of the virtual camera and recorded the distance of these points from the image plane as a "snapshot" descriptor. The snapshot descriptor is robust to self-occlusion and very efficient to compute. Snapshot achieved better pairwise range image alignment results compared to spin image. Mian et al. [4] also defined an LRF for a local surface and then fitted the local surface with a uniform lattice. They used depth values of the local surface to form a feature descriptor, which was further compressed using a PCA technique.

Castellani et al. [15] first built a clockwise spiral pathway around the keypoint $\boldsymbol{p}$. The pathway is illustrated in Fig. 4c. Along this pathway, they extracted a set of attributes including the saliency level, the maximal curvature, the minimal curvature and the surface normal deviation. They then used a discrete time Hidden Markov Model (HMM) to encode the information of the local surface. This HMM descriptor is robust to rotation, nonuniform sampling and varying mesh resolutions. Experimental results showed that its matching performance was better than the spin image and the 3D shape context (described in Section 4.2.1).

Novatnack and Nishino [103] first mapped each neighboring point $\boldsymbol{q}$ into a 2D domain using a geodesic polar coordinate frame $[d(\boldsymbol{q}), \theta(\boldsymbol{q})]$. Here, $d(\boldsymbol{q})$ is the geodesic distance between $\boldsymbol{q}$ and $\boldsymbol{p}$, and $\theta(\boldsymbol{q})$ is the polar angle of the tangent of the geodesic between $\boldsymbol{q}$ and $\boldsymbol{p}$. After this mapping, they constructed the "Exponential Map (EM)" descriptor by encoding the surface normals of the neighboring points into this 2D domain.

Masuda [104] defined a local log-polar coordinate frame (r, θ) on the tangent plane of the keypoint $\boldsymbol{p}$. They then projected the neighboring points $\boldsymbol{q}$ onto the tangent plane, and stored the depth $\boldsymbol{n} \cdot (\boldsymbol{q} - \boldsymbol{p})$ in the log-polar coordinate frame, resulting in a "Log-Polar Height Map (LPHM)" descriptor. This method is illustrated in Fig. 4d. Since an LPHM feature is not invariant to the rotation around the surface normal $\boldsymbol{n}$, they applied a Fourier transform to the LPHM in the θ axis, and used the Fourier Power Spectrum to form a new descriptor (namely, FPS). Experimental results showed that this method performed well for range image registration. In comparison with the spin image, this method does not depend on the uniform sampling of the mesh [104].

Steder et al. [105], [106] first aligned the local patch around the keypoint $\boldsymbol{p}$ with the normal of $\boldsymbol{p}$. They then overlaid a star pattern onto this aligned patch, where each beam corresponded to a value in the "Normal Aligned Radial Feature (NARF)" descriptor. The NARF descriptor captures the variation of pixels under each beam. In order to make the descriptor invariant to rotation, they extracted a unique orientation from the descriptor and shifted the descriptor according to this orientation. The NARF descriptor outperformed spin image on feature matching.

do Nascimento et al. [107] extracted a local patch around the keypoint $\boldsymbol{p}$ from an RGB-D image. They aligned the local patch with a dominant direction and scaled it using the depth information. They fused the geometrical and intensity information of the local patch by encoding the intensity variations and surface normal displacements into a "Binary Robust Appearance and Normal Descriptor (BRAND)". The BRAND outperformed SIFT, SURF, spin image, and CSHOT in terms of matching precision and robustness.

4.2 Histogram Based Methods

These methods describe the local neighborhood of a keypoint by accumulating geometric or topological measurements (e.g., point numbers, mesh areas) into histograms according to a specific domain (e.g., point coordinates, geometric attributes) [3]. These methods can further be divided into spatial distribution histogram (SDH), geometric attribute histogram (GAH) and oriented gradient histogram (OGH) based methods.

4.2.1 Spatial Distribution Histogram

These methods describe the local neighborhood of a keypoint by generating histograms according to the spatial distributions (e.g., point coordinates) of the local surface. They first define an LRF/axis for the keypoint, and partition the 3D local neighborhood into bins according to the LRF. They then count the spatial distribution measurements (e.g., point numbers, mesh areas) up in each bin to form the feature descriptor.

Johnson and Hebert [64] used the normal $\boldsymbol{n}$ of a keypoint $\boldsymbol{p}$ as the local reference axis and expressed each neighboring point $\boldsymbol{q}$ with two parameters: the radial distance α and the signed distance β. That is, $\alpha = \sqrt{\|\boldsymbol{q}-\boldsymbol{p}\|^2 - (\boldsymbol{n}\cdot(\boldsymbol{q}-\boldsymbol{p}))^2}$ and $\beta = \boldsymbol{n} \cdot (\boldsymbol{q} - \boldsymbol{p})$. They then discretized

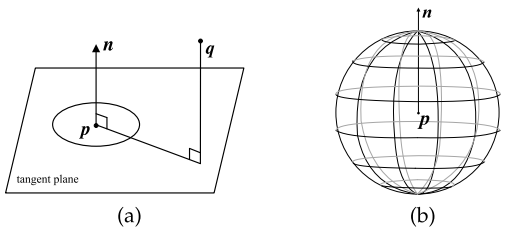


Fig. 5. Illustration of spatial distribution histogram based methods. (a) Spin image. (b) 3D shape context.

the $\alpha - \beta$ space into a 2D array accumulator, and counted up the number of points that fell into the bin indexed by (α, β). The 2D array was further bilinearly interpolated to obtain a "spin image" descriptor. An illustration of this method is shown in Fig. 5a. The spin image is invariant to rigid transformations and is robust to clutter and occlusion [64]. It has been employed in many applications, and has been considered to be the de facto benchmark for the evaluation of local surface features [3], [44]. However, the spin image has several drawbacks, e.g., (i) It is sensitive to varying mesh resolutions and nonuniform sampling [4]; (ii) Its descriptive power is limited since the cylindrical angular coordinate is omitted. Several variants of the spin image can also be found in the literature, including a face-based spin image [108], a spin image signature [109], a multi-resolution spin image [110], a spherical spin image [111], a scale invariant spin image [76], a Tri-Spin-Image (TriSI) descriptor [46] and a color spin image [112].

Frome et al. [63] extended the 2D shape context method [113] to a 3D surface, namely "3D Shape Context (3DSC)". They used the normal $\boldsymbol{n}$ at a keypoint $\boldsymbol{p}$ as the local reference axis. The spherical neighborhood was then divided equally along both the azimuth and elevation dimensions, but logarithmically along the radial dimension, as illustrated in Fig. 5b. The 3DSC descriptor was generated by counting up the weighted number of points falling into each bin. They also applied a spherical harmonic transform to the 3DSC to generate a "Harmonic Shape Context (HSC)" descriptor. They tested the two descriptors in vehicle recognition experiments. It was reported that both the 3DSC and HSC achieved higher recognition rates in noisy scenes compared to the spin image, while the 3DSC outperformed the HSC and spin image in cluttered scenes. In a follow up work, Tombari et al. [114] proposed a unique shape context (USC) by associating each keypoint with a repeatable and unambiguous LRF. Experimental results showed that the USC significantly decreased the memory requirement and improved the accuracy of feature matching compared to the 3DSC. Recently, Sukno et al. [115] proposed an "asymmetry patterns shape context (APSC)" to provide azimuth rotation invariance to the 3DSC. Experimental results showed that APSC achieved comparable performance to 3DSC. Another generalization of the 2D shape context is the "intrinsic shape context (ISC)" descriptor [116], which is invariant to isometric deformations.

Mian et al. [38] chose a pair of vertices which satisfied certain geometric constraints to construct an LRF. They then constructed a local 3D grid over the range image, and summed the surface areas which intersected each bin of the grid, to generate a "3D tensor" descriptor. This method is very robust to noise and varying mesh resolutions. Experimental results showed it outperformed the spin image. One limitation of this method is the combinatorial explosion of the vertex pairs for the construction of the LRF [70].

Zhong [70] first constructed an LRF by performing PCA on the covariance matrix of the neighboring points. They then divided the spherical angular space into relatively uniformly and homogeneously distributed cells using a discrete spherical grid. They also evenly divided the radial distances. The "intrinsic shape signature (ISS)" descriptor was constructed by summing the density weights of all points that fell into each cell. Experimental results showed that ISS outperformed the spin image and 3DSC in the presence of noise, occlusion and clutter.

Guo et al. [44] first constructed a unique, unambiguous and robust LRF by calculating the covariance matrix of all points lying on the local surface rather than just mesh vertices (in contrast to [3], [70]). They then extracted a "Rotational Projection Statistics (RoPS)" descriptor for the keypoint $\boldsymbol{p}$ by rotationally projecting the neighboring points onto 2D planes and calculating a set of statistics (including low-order central moments and entropy) of the distribution of these projected points. Experimental results showed that RoPS was very robust to noise, varying mesh resolutions and holes. RoPS outperformed spin image, local surface patches (LSP), THRIFT, SHOT and MeshHOG in terms of feature matching. Guo et al. [117] then extended the RoPS descriptor to encode both shape and color (e.g., RGB) information of a local surface.

4.2.2 Geometric Attribute Histogram

These methods describe the local neighborhood of a keypoint by generating histograms according to the geometric attributes (e.g., normals, curvatures) of the local surface.

Yamany and Farag [66] used simplex angles to estimate the curvature values on a free-form surface. They generated the "surface signature" by accumulating the simplex angles into a 2D histogram. One dimension of the histogram is the distance d from the keypoint $\boldsymbol{p}$ to a neighboring point $\boldsymbol{q}$. Another dimension is the angle $\arccos\left(\frac{\mathbf{n}\cdot(\mathbf{q}-\mathbf{p})}{\|\mathbf{q}-\mathbf{p}\|}\right)$, where $\boldsymbol{n}$ is the surface normal at $\boldsymbol{p}$. It was demonstrated that the surface signature was more descriptive compared to the splash, point signature and spin image [66].

Chen and Bhanu [68] proposed "local surface patches" by accumulating the number of points into a 2D histogram. One dimension of the histogram is the shape index values of the neighboring points, and another dimension is the angles between the normal of the keypoint $\boldsymbol{p}$ and these of the neighboring points. Experimental results showed that LSP was as effective as spin image for 3D object recognition but it was more computationally efficient.

Flint et al. [86] proposed a "THRIFT" descriptor by accumulating the number of points into a 1D histogram according to the surface normal angles. They calculated two normals for each point on the local surface by fitting two planes with two different windows. That is, a normal $\boldsymbol{n}_s$ was calculated from a small window and another normal $\boldsymbol{n}_l$ from a larger window. The surface normal angle of a point is calculated as the angle between its two normals. Later,

Flint et al. [39] used the angles between the normal of the keypoint and these of the neighboring points to generate a weighted histogram.

Taati et al. [51] first performed PCA on the covariance matrix of the neighboring points of each point $\boldsymbol{q}$ on the surface to obtain an LRF and three eigenvalues. They then calculated a set of position properties, direction properties and dispersion properties for each point $\boldsymbol{q}$ based on the LRF and these eigenvalues. Next, they selected a subset of these properties using a feature selection algorithm. Finally, they accumulated these selected properties of all the neighboring points of a keypoint $\boldsymbol{p}$ into a histogram (i.e., the "variable dimensional local shape descriptor (VD-LSD)"). The VD-LSD formulation offers a generalized platform that subsumes a large class of descriptors, including the spin image, 3D tensor and point signature [51]. Taati and Greenspan [47] also provided a way to choose the optimal descriptors for 3D object recognition based on the geometry of the models and the characteristics of the sensing device. Experimental results showed that VD-LSD outperformed the spin image on several data sets in terms of recognition rate.

Rusu et al. [118] proposed "point feature histograms" (PFH) to encode a local surface. For each pair of points in the neighborhood of a keypoint $\boldsymbol{p}$, a Darboux frame was first defined. They then calculated four measures using the angles between the points' normal and the distance vector between them, in the same manner as in [119]. They accumulated these measures of all pairs of points into a 16-bin histogram (i.e., PFH). In order to improve the computational efficiency of PFH, Rusu et al. [120] proposed a "simplified point feature histogram (SPFH)" by accumulating only these measures between the keypoint and its neighbors. They finally used these SPFH values of the neighboring points of $\boldsymbol{p}$ to obtain "fast point feature histograms (FPFH)". FPFH retains the majority discriminative power of the PFH with a reduced computational complexity.

Tombari et al. [3] first constructed an LRF for a keypoint $\boldsymbol{p}$, and divided the neighborhood space into 3D spherical volumes. They then generated a local histogram for each volume by accumulating the number of points according to the angles between the normal at the keypoint and these at the neighboring points. They concatenated all local histograms to form an overall "Signature of Histograms of OrienTations (SHOT)" descriptor. The SHOT descriptor is highly descriptive, computationally efficient and robust to noise. Experimental results showed that SHOT outperformed the spin image and point signature at all levels of noise. One shortcoming of this method is its sensitivity to varying point densities. Further, Tombari et al. [53] developed a "Color-SHOT (CSHOT)" by combining the histograms of shape-related measures with the histograms of texture-related measures. Experimental results showed that combining the texture information into a geometric feature descriptor would gain additional benefits to its performance.

Tang et al. [121] represented the normal vector orientation as an ordered pair of azimuthal and zenith angles. They proposed a "Histogram of Oriented Normal Vectors (HONV)" by concatenating local histograms of azimuthal angles and zenith angles. Object detection and classification results on the RGB-D database showed that HONV outperformed the Histograms of Oriented Gradients (HOG) descriptor.

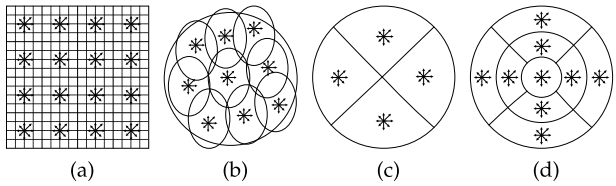


Fig. 6. Illustration of the oriented gradient histogram based methods. (a) The method in [89] and SI-SIFT. (b) 2.5D SIFT. (c) MeshHOG. (d) The method in [83].

4.2.3 Oriented Gradient Histogram

These methods describe the local neighborhood of a keypoint by generating histograms according to the oriented gradients of the local surface.

Hua et al. [89] first mapped a 3D surface to a 2D canonical domain and encoded the shape characteristics (i.e., mean curvatures and conformal factors) of the surface into a two-channel shape vector image. For each channel, a descriptor was generated using the same technique as SIFT [22]. That is, the 2D plane was divided into 16 subregions, and an eight-element histogram was generated for each subregion according to the orientations of the gradients, as shown in Fig. 6a. They concatenated all the histograms of the two channels to form the overall descriptor. Experimental results confirmed that this descriptor was very robust and suitable for surface matching purpose.

Lo and Siebert [78] first divided the local patch of a keypoint $\boldsymbol{p}$ into nine elliptical subregions, as shown in Fig. 6b. For each of the nine elliptical subregions, they generated a nine-element histogram of the surface types according to the shape index values. They also generated an eight-element histogram according to the gradient orientations. They finally concatenated all histograms from the nine subregions to form a "2.5D SIFT" descriptor. Experimental results showed that the 2.5D SIFT produced consistently more reliable feature matches compared to the SIFT algorithm. However, the 2.5D SIFT requires a complicated preprocessing step (e.g., recovering the 3D poses of all keypoints) [14]. More recently, another extension of the SIFT method was proposed by [76], namely "Local Depth SIFT (LD-SIFT)". Experimental results showed that LD-SIFT outperformed the spin image representation.

Zaharescu et al. [91] first defined a scalar function f on the mesh vertices and calculated the gradients ∇f. The function f can be the curvature, normal, heat, density or texture. They then constructed an LRF for each keypoint $\boldsymbol{p}$ and projected the gradient vectors onto the three orthonormal planes of the LRF. They divided each plane into four polar subregions, and generated an eight-element histogram for each subregion according to the orientations of the gradients ∇f. An illustration is shown in Fig. 6c. The "MeshHOG" descriptor was finally obtained by concatenating the histograms of all the subregions of the three planes. Its effectiveness has been demonstrated by feature matching on rigid and nonrigid objects. However, the MeshHOG cannot be applied to objects with large deformations.

Bayramoglu and Alatan [14] first divided the local patch of a keypoint $\boldsymbol{p}$ into 16 subregions using the same technique as the SIFT [22], as shown in Fig. 6a. They then generated an eight-element histogram for each subregion according to

the gradient orientations of the shape index values. The 16 histograms were finally concatenated to form a "SI-SIFT" descriptor. Experimental results indicated that the SI-SIFT was robust to clutter and occlusion. It improved the performance of the LSP and 2.5D SIFT.

Hou and Qin [83] first defined a scalar function f on the mesh vertices and calculated the gradients ∇f. They parameterized each neighboring point $\boldsymbol{q}$ by a polar coordinate system $[d(\boldsymbol{q}), \theta(\boldsymbol{q})]$. Here, $d(\boldsymbol{q})$ is the geodesic distance from $\boldsymbol{p}$ to $\boldsymbol{q}$, $\theta(\boldsymbol{q})$ is the projected polar angle of $\boldsymbol{q}$ from the orientation of $\boldsymbol{p}$ in the local tangent plane. They then divided the 2D parametrization plane into nine subregions, and generated an eight-element histogram for each subregion according to the orientations of the gradients ∇f, as shown in Fig. 6d. The descriptor was finally obtained by concatenating the nine histograms. Matching results showed that this method achieved higher recall and precision than the MeshHOG. It worked well even under large isometric deformations.

4.3 Transform Based Methods

These methods first transform a range image from the spatial domain to another domain (e.g., spectral domain), and then describe the 3D surface neighborhood of a given point by encoding the information in that transformed domain.

Hu and Hua [94] first performed a Laplace-Beltrami transform on the local surface to get the spectrum of the local surface. They then generated a histogram according to the spectral values of the local surface. This histogram was used as the feature descriptor. Experiments demonstrated that this descriptor was very powerful in matching similar shapes. The descriptor is also invariant to rigid transformations, isometric deformations and scaling.

Sun et al. [73] proposed a "Heat Kernel Signature (HKS)". They considered a mesh $\mathcal{M}$ as a Riemannian manifold and restricted the heat kernel $K_t(\boldsymbol{p}, \boldsymbol{q})$ to the temporal domain as $K_t(\boldsymbol{p}, \boldsymbol{p})$. The HKS descriptor $K_t(\boldsymbol{p}, \boldsymbol{p})$ can be interpreted as a multi-scale notion of the Gaussian curvature, where the time parameter t provides a natural notion of scale. The HKS is an intrinsic attribute of the shape. It is stable against perturbations of the shape and invariant to isometric deformations. In a follow up work, a scale invariant HKS [122] was proposed for nonrigid shape retrieval [123], and a photometric HKS was also introduced [61].

Knopp et al. [23] developed an extended version of the 2D Speeded Up Robust Feature (SURF) [97], namely "3D SURF". They first voxelized a mesh to a 3D voxel image, and applied a Haar wavelet transform to the voxel image. They then defined an LRF for each keypoint $\boldsymbol{p}$ by using the Haar wavelet responses in the neighborhood. Next, they divided the neighborhood volume into $N_b \times N_b \times N_b$ bins. For each bin, a vector $\boldsymbol{v} = (\sum d_x, \sum d_y, \sum d_z)$ was calculated, where d_x, d_y and d_z are respectively the Haar wavelet responses along the x, y and z axes. They finally combined all the vectors of the $N_b \times N_b \times N_b$ bins to form the 3D SURF descriptor.

4.4 Summary

Table 4 presents a summary of the local surface feature description methods. The methods are listed chronologically by year of publication and alphabetically by first author within a given year.

- Histogram based methods are the most frequently investigated methods among the three categories. Both geometric attribute histogram based methods and oriented gradient histogram based methods are dependent on the calculation of the first-order and/or second-order derivatives of the surface. Therefore, they are relatively sensitive to noise.
- The majority of these methods achieve invariance to rigid transformations by resorting to an LRF. They first define an LRF for a keypoint and then express the local neighborhood with respect to that frame. Therefore, the repeatability and robustness of the LRF play an important role in the performance of a feature descriptor [44].
- Some methods address isometric deformations by resorting to geodesic distances (or their k-ring approximations) rather than Euclidean distances. Examples include [83], [91] and [116]. Other methods however, achieve deformation invariance by converting a surface to a certain intrinsic domain, including the Laplace-Beltrami spectrum domain [94] and the temporal domain [73].
- Determining the neighborhood size is important for any local feature descriptor. A large neighborhood enables a descriptor to encode more information while being more sensitive to occlusion and clutter. Several methods proposed an adaptive-scale keypoint detector dedicated with the descriptor, such as [4], [78], [91], [103]. However, any keypoint detection method can be combined with a local feature description method to yield an optimal performance.

5 Surface Matching

Most of the existing surface matching methods contain three modules, i.e., feature matching, hypothesis generation and verification. It first establishes a set of feature correspondences between the scene and the model by matching the scene features against the model features. It then uses these feature correspondences to vote for candidate models and generate transformation hypotheses. Next, these candidates are verified to obtain the identities and poses of the objects present in the scene.

5.1 Feature Matching

The task of feature matching is to establish a set of feature correspondences. There are three popular strategies for feature matching, i.e., threshold based, nearest neighbor (NN) based and nearest neighbor distance ratio (NNDR) based strategies [6]. It is demonstrated that NN and NNDR based strategies achieve better feature matching performance than a threshold based strategy [6].

Once an appropriate matching strategy is selected, efficient search over the model feature library is another important issue. The simplest way is to perform a brute-force search, which compares a scene feature with all model features. The computational complexity of this approach is $O(N_f)$, where N_f is the number of model features. A faster alternative is to adopt an appropriate data structure or indexing method. For example, [64] used a slicing based algorithm [99], [125] used a 2D index table,

TABLE 4
Methods for Local Surface Feature Description

No.	Name	Reference	Data Type	Category (Section)	Performance
1	Splash	Stein & Medioni 1992 [98]	Mesh	Signature (4.1)	Robust to noise
2	Point Signature	Chua & Jarvis 1997 [99]	Mesh	Signature (4.1)	Not unique, sensitive to mesh resolution
3	Spin Image	Johnson & Hebert 1998 [124]	Mesh	SDH (4.2.1)	Most cited method, with a set of variants including [109], [110], [111], [76], [46], [112]
4	Point's Fingerprint	Sun & Abidi 2001 [100]	Mesh	Signature (4.1)	Outperforms point signature, spin image
5	Surface Signature	Yamany & Farag 2002 [66]	Mesh	GAH (4.2.2)	Outperforms splash, point signature, spin image
6	3DSC/HSC	Frome et al. 2004 [63]	Point Cloud	SDH (4.2.1)	Outperforms spin image
7	NBS	Li & Guskov 2005 [77]	Mesh	Signature (4.1)	Outperforms spin image
8	3D Tensor	Mian et al. 2006 [38]	Mesh	SDH (4.2.1)	Outperforms spin image
9	LSP	Chen & Bhanu 2007 [68]	Depth Image	GAH (4.2.2)	Comparable to spin image
10	THRIFT	Flint et al. 2007 [86]	Point Cloud	GAH (4.2.2)	Sensitive to noise
11	Snapshot	Malassiotis & Strintzis 2007 [102]	Mesh	Signature (4.1)	Outperforms spin image
12	VD-LSD	Taati et al. 2007 [51]	Point Cloud	GAH (4.2.2)	A generalized descriptor platform
13	HMM	Castellani et al. 2008 [15]	Mesh	Signature (4.1)	Outperforms spin image, 3DSC
14	Hua's	Hua et al. 2008 [89]	Mesh	OGH (4.2.3)	Suitable for surface matching
15	EM	Novatnack & Nishino 2008 [103]	Mesh	Signature (4.1)	Has scale dependent and invariant descriptors
16	PFH	Rusu et al. 2008 [118]	Point Cloud	GAH (4.2.2)	Time consuming
17	Spectral Feature	Hu & Hua 2009 [94]	Mesh	Transform (4.3)	Invariant to isometric deformations
18	2.5D SIFT	Lo & Siebert 2009 [78]	Depth Image	OGH (4.2.3)	Outperforms SIFT
19	LPHM/FPS	Masuda 2009 [104]	Mesh	Signature (4.1)	Robust to occlusion
20	FPFH	Rusu et al. 2009 [120]	Point Cloud	GAH (4.2.2)	More time efficient than PFH
21	HKS	Sun et al. 2009 [73]	Mesh	Transform (4.3)	Invariant to isometric deformations
22	MeshHOG	Zaharescu et al. 2009 [91]	Mesh	OGH (4.2.3)	Can capture the properties of both local geometry and scalar function
23	ISS	Zhong 2009 [70]	Point Cloud	SDH (4.2.1)	Outperforms spin image, 3DSC
24	SI-SIFT	Bayramoglu & Alatan 2010 [14]	Depth Image	OGH (4.2.3)	Outperforms LSP, 2.5D SIFT
25	Hou's	Hou & Qin 2010 [83]	Mesh	OGH (4.2.3)	Invariant to isometric deformations, outperforms MeshHOG
26	3D SURF	Knopp et al. 2010 [23]	Mesh	Transform (4.3)	Outperforms SIFT
27	Depth Values	Mian et al. 2010 [4]	Point Cloud	Signature (4.1)	Outperforms spin image, comparable to 3D tensor
28	NARF	Steder et al. 2010 [105]	Point Cloud	Signature (4.1)	Outperforms spin image
29	SHOT	Tombari et al. 2010 [3]	Mesh	GAH (4.2.2)	Outperforms point signature, spin image, EM
30	USC	Tombari et al. 2010 [114]	Mesh	SDH (4.2.1)	Outperforms 3DSC
31	CSHOT	Tombari et al. 2011 [53]	Mesh	GAH (4.2.2)	Outperforms MeshHOG, SHOT
32	LD-SIFT	Darom & Keller 2012 [76]	Mesh	OGH (4.2.3)	Outperforms spin image
33	ISC	Kokkinos et al. 2012 [116]	Mesh	SDH (4.2.1)	Invariant to isometric deformations, outperforms HKS
34	HONV	Tang et al. 2012 [121]	Depth Image	GAH (4.2.2)	Outperforms HOG
35	APSC	Sukno et al. 2013 [115]	Mesh	SDH (4.2.1)	Comparable to 3DSC
36	BRAND	do Nascimento et al. 2013 [107]	Depth Image	Signature (4.1)	Outperforms spin image and CSHOT
37	TriSI	Guo et al. 2013 [46]	Mesh	SDH (4.2.1)	Outperforms spin image, LSP, SHOT
38	RoPS	Guo et al. 2013 [44]	Mesh	SDH (4.2.1)	Outperforms spin image, LSP, THRIFT, MeshHOG, SHOT

[63], [38] and [98] used hash tables, [70] used a locality sensitive tree, [44], [46] and [56] used a k-d tree method to perform feature matching.

5.2 Hypothesis Generation

The tasks of hypothesis generation are twofold. The first task is to obtain candidate models which are potentially present in the scene. The second task is to generate transformation hypotheses for each candidate model.

If only the positions of the keypoints are used, three feature correspondences are required to deduce a transformation between a candidate model and the scene, as in [47], [48] and [98]. If both the positions and surface normals of the keypoints are used, two feature correspondences are sufficient to deduce a transformation, as in [126]. Moreover, if an LRF has been established for each keypoint, one correspondence is sufficient to derive a transformation, as in [3], [4], [38], [44] and [70]. However, feature correspondences contain both true and false matches. In order to obtain accurate transformation hypotheses, several techniques have been intensively investigated, including geometric consistency, pose clustering, constrained interpretation tree, Random Sample Consensus (RANSAC), game theory, generalized Hough transform, and geometric hashing.

Geometric consistency. This technique is based on the assumption that correspondences that are not geometrically consistent will produce transformations with large errors. Therefore, geometric consistency can reduce mismatched correspondences and improve the robustness of the hypothesized transformations.

Given a list of feature correspondences $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_{N_c}\}$, this technique first chooses a seed correspondence $\mathcal{C}_i$ from the list $\mathcal{C}$ and initializes a group $\mathcal{G}_i = \{\mathcal{C}_i\}$. It then finds the correspondence $\mathcal{C}_j$ that is geometrically consistent with the group $\mathcal{G}_i$. This procedure for group $\mathcal{G}_i$ continues until no more correspondences can be added to it. Therefore, it results in a group of geometrically consistent correspondences for each seed correspondence $\mathcal{C}_i$. The group $\mathcal{G}_i$ is used to calculate a transformation hypothesis.

Multiple hypotheses may be generated for the list $\mathcal{C}$ by selecting a set of seed correspondences. Examples of the methods based on this technique include [11], [64], [68] and [98].

Pose clustering. This technique is based on the assumption that if the scene and the model do match, these transformation hypotheses are expected to cluster in the transformation space near the ground truth transformation.

Given a list of feature correspondences $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_{N_c}\}$, this technique first calculates a transformation for every k correspondences. Here, k is the minimal number of correspondences that are required to determine a transformation. Therefore, a total of $C_k^{N_c}$ transformations can be calculated. This technique then performs clustering on the transformations. The cluster centers are considered transformation hypotheses. Examples of the methods based on this technique include [4], [9], [44], [46], [70] and [127].

Constrained interpretation Tree. This technique aims to find a tree of consistent interpretations for the transformation hypotheses. That is, as the number of nodes grows in the tree, the commitment to a particular hypothesis increases [128].

This technique creates an interpretation tree for each model. It contains no feature correspondence at the root of the tree. It then builds each successive level of the tree by picking a model feature and adding the scene features that are highly similar to that model feature to the tree as nodes. Therefore, each node in the tree contains a hypothesis. The hypothesis at a node is formed by the feature correspondences at that node and all its parent nodes. Examples of the methods based on this technique include [34] and [48].

RANSAC. This technique randomly selects a minimal set of feature correspondences to calculate a rigid transformation which aligns the model with the scene, and counts the number of inlier point pairs which are consistent with this transformation. This process repeats until the number of inliers meets a predefined threshold, or all possible sets have been tested. The transformation which results in the largest number of inliers is considered to be the transformation hypothesis. Examples of the methods based on this technique include [47], [126] and [129].

Game Theory. This technique uses a selection process derived from game theory to select a set of feature correspondences which satisfy a global rigidity constraint. All feature correspondences derived from feature matching are first let to compete in a non-cooperative game. The competition induces a selection process in which incompatible feature correspondences are driven to extinction whereas a set of reliable feature correspondences survive. These remaining feature correspondences are used to calculate several transformation hypotheses. Examples of the methods based on this technique include [130] and [56].

Generalized hough transform. This technique performs voting in a space called parametric Hough space (e.g., rotation [131], translation [131], and position [132]) using the feature correspondences. Each point of the Hough space corresponds to the existence of a transformation between the model and the input scene. The peaks in the Hough accumulator are considered to be transformation hypotheses. Examples of the methods based on this technique include [23], [131], [132] and [133].

Geometric hashing. This technique uses a hash table to record the coordinates of all model points with respect to a reference basis during preprocessing. The process is then repeated for every basis. During recognition, a basis is selected and the other points are expressed in this coordinate system and are then used to index into the hash table. Each index produces a model basis and the basis with the maximum support is used to calculate the transformation hypothesis. Examples of the methods based on this technique include [134].

5.3 Verification

The task of verification is to distinguish true hypotheses from false hypotheses. The existing verification methods can be divided into individual and global verification methods.

Individual verification methods. These methods first individually align a candidate model with the scene using a transformation hypothesis. This alignment is further refined using a surface registration method, such as the Iterative Closest Point (ICP) algorithm [135]. If the accuracy of alignment is greater than a predefined threshold, the hypothesis is accepted. Subsequently, the scene points that correspond to this model are recognized and segmented.

Several approaches have been proposed to measure the accuracy of alignment. Johnson and Hebert [64], [124] used the ratio τ_a between the number of corresponding points to the number of points on the model. One limitation of this approach is that objects with occlusions larger than 1-τ_a cannot be recognized [11]. Mian et al. [38] used the same measure in [64] and another measure which was proportional to the accuracy of alignment and the similarity between the features. Chen and Bhanu [68] used the residual error of the ICP algorithm to measure the accuracy of alignment. They also used two types of active space violations for further verification. Bariya et al. [34], [48] calculated the overlap area between the candidate model and the scene, and used the ratio between this overlap area and the total visible surface area of the model as a measure of the accuracy of alignment. Guo et al. [44] used the residual error of the ICP algorithm, together with the ratio between the number of corresponding points to the number of points on the scene, to perform verification. It is very challenging for these approaches to determine the optimal thresholds in order to improve the recognition rate while maintaining a low number of false positives [11], [44].

Global Verification Methods. Rather than considering each candidate model individually (as in the case of individual verification methods), these methods take into account the whole set of hypotheses by considering the verification process as a global (or partially global) optimization problem.

Papazov and Burschka [126] proposed an acceptance function which consists of a support term and a penalty term to define the quality of a transformation hypothesis. The support and penalty terms are respectively proportional to the number of corresponding points and the number of points on the model which occlude the scene. These hypotheses were then filtered using the two terms, and the remaining hypotheses were used to construct a conflict graph. The final hypothesis was selected by performing a non-maximum suppression on this conflict graph based on the acceptance function. This approach takes into account

TABLE 5
Major Systems for Local Surface Feature Based 3D Object Recognition

No.	Reference	Keypoint Detector	Feature Descriptor	Feature Matching	Hypothesis Generation	Verification	Performance
1	Stein & Medioni 1992 [98]	Fixed-Scale	Splash	Hash Table	Geometric Consistency	-	Tested on an in-house database (9 models + 4 scenes)
2	Chua & Jarvis 1997 [99]	Sparse Sampling	Point Signature	Index Table	Feature Correspondences	-	Tested on an in-house database (15 models + 15 scenes)
3	Johnson & Hebert 1999 [64]	Sparse Sampling	Spin Image	Slicing based	Geometric Consistency	Individual Verification	Tested on an in-house database (20 models + 8 scenes)
4	Frome et al. 2004 [63]	Sparse Sampling	3DSC	Hash Table	-	-	Tested on an in-house database
5	Mian et al. 2006 [38]	Mesh Decimation	3D Tensor	Hash Table	Feature Correspondences	Individual Verification	96.6% RR with up to 84% occlusion on the UWA database
6	Chen & Bhanu 2007 [68]	Fixed-Scale	LSP	Hash Table	Geometric Consistency	Individual Verification	Tested on the OSU database in single-object scenes and two-object scenes
7	Li and Guskov 2007 [101]	Adaptive-Scale	NBS	Hash Table	-	-	98.02% RR on the Stuttgart database
8	Zhong 2009 [70]	Fixed-Scale	ISS	Locality Sensitive Tree	Pose Clustering	Individual Verification	97% RR on a database originated from the PSB database
9	Drost et al. 2010 [9]	Sparse Sampling	Point Pair Feature	Hash Table	Pose Clustering	Individual Verification	97.0% RR with up to 84% occlusion on the UWA database
10	Mian et al. 2010 [4]	Adaptive-Scale	Depth Values	-	Pose Clustering	Individual Verification	Above 95% RR with up to 80% occlusion on the UWA database
11	Papazov & Burschka 2011 [126]	Sparse Sampling	Point Pair Feature	Hash Table	RANSAC	Global Verification	Outperforms spin image and 3D tensor based algorithms [38] on the UWA database
12	Taati & Greenspan 2011 [47]	Sparse Sampling	VD-LSD	Slicing based	RANSAC	-	98.5% RR with up to 83% occlusion on the UWA database; 80.0% RR on the Queen's LIDAR database; 74.7% RR on the Queen's Stereo database
13	Bariya et al. 2012 [48]	Adaptive-Scale	EM	Tree based	Constrained Interpretation Tree	Individual Verification	97.5% RR with up to 84% occlusion on the UWA database; 82.43% RR on the Queen's LIDAR database
14	Aldoma et al. 2012 [11]	Sparse Sampling	SHOT	Muja's Algorithm	Geometric Consistency	Global Verification	100% RR on the UWA database; 79.5% RR on the Kinect database
15	Rodolà et al. 2013 [56]	Fixed-Scale	SHOT	*k*-d Tree	Game Theory	Individual Verification	Outperforms spin image and 3D tensor [38], keypoint [4] and EM [48] based algorithms on the UWA database; Tested on the Ca' Foscari Venezia database
16	Guo et al. 2013 [46]	Sparse Sampling	TriSI	*k*-d Tree	Pose Clustering	Individual Verification	98.4% RR on the UWA database
17	Guo et al. 2013 [44]	Fixed-Scale	RoPS	*k*-d Tree	Pose Clustering	Individual Verification	100%, 98.9%, 95.4% and 96.0% RR respectively when tested on the Bologna 1, UWA, Queen's LIDAR and Ca' Foscari Venezia databases; Outperforms [38], [4], [47], [48] and [56]

the interaction between hypotheses, which makes it more possible to get a global optimal transformation. Aldoma et al. [11] determined a cost function which encompasses geometrical cues including model-to-scene/scene-to-model fitting, occlusion and model-to-model interactions. They obtained optimal hypotheses by minimizing this cost function over the solution space. One major advantage of this approach is that, it can detect significantly occluded objects without increasing the number of false positives.

5.4 Summary

Table 5 shows a list of major systems for local feature based 3D object recognition. 'RR' is the abbreviation for 'recognition rate'. It can be succinctly interpreted from the analysis of this section that:

- Recent papers usually report their performance using the criterion of recognition rate, while the earlier developed systems demonstrate the performance without any quantitative recognition results (e.g., [64], [98], [99]). Many recent systems achieve high recognition rates which are above 95 percent (e.g., [38], [44], [47] and [48]). This reveals the promising capability of 3D object recognition systems.
- The UWA database is the most frequently used database. The best recognition rate reported on this database is already 100 percent [11]. The Queen's databases is more challenging compared to the UWA database since the former is more noisy and the points are not uniformly distributed. The best recognition rate reported on the Queen's LIDAR database is 95.4 percent [44]. The Ca' Foscari Venezia database contains several objects with large flat and featureless areas. The best recognition rate reported on this database is 96.0 percent [44].

6 Future Research Directions

Based on the contemporary research, this section presents a brief discussion of future research directions.

- *Benchmarking protocol*. There is a need for a benchmarking protocol of existing 3D keypoint detection, local surface feature description and surface matching methods. This will facilitate the benchmarking of

the state-of-the-art algorithms in this area. It will also help in improving the interpretation of the research results and will offer clear comparisons between the existing algorithms.

- *Nonrigid object recognition*. One crucial scope of improvement in object recognition performance is to better handle deformations. There could be a number of directions that one might focus on. One direction would be to extract local features that are invariant to shape deformations. Another direction would be to map different shapes into a canonical form which is a deformation-invariant representation of the shape [136], and to perform object recognition in this canonical form.
- *Fusion of photometric and geometric information*. The fusion of photometric and geometric information (e.g., in an RGB-D database)is expected to achieve improved results, especially in scenarios where pure geometric or pure photometric features based methods fail to work. Different types of fusion can be adopted depending on the processing stage at which the fusion takes place.
- *Object recognition from 3D videos*. Since a 3D video contains both spatial and temporal information, and scans an objects from a set of views. It is expected that using spatiotemporal/multiview information to detect keypoints and construct local surface feature descriptors will help to improve the performance of many systems for 3D object recognition in cluttered scenes.
- *3D object recognition on kinect data*. Kinect data become increasingly popular due to many reasons including its high speed and low cost. Since the Kinect data is quite different from traditional data (e.g., the UWA database), traditional algorithms may face new challenges when tested on the Kinect database. Several methods on object classification in segmented Kinect images have been proposed [137], more results on object recognition in cluttered low depth resolution (e.g., Kinect) images are expected.

7 Conclusion

This paper has presented a unique survey of the state-of-the-art 3D object recognition methods based on local surface features. A comprehensive taxonomy of the 3D object recognition methods published since 1992 has also been presented. Merits and demerits of the various feature types and their extraction methods are also analyzed.

Acknowledgments

This research was supported by a China Scholarship Council (CSC) scholarship and Australian Research Council grants (DE120102960, DP110102166).

References

[1] Y. Lei, M. Bennamoun, M. Hayat, and Y. Guo, "An efficient 3D face recognition approach using local geometrical signatures," *Pattern Recognit.*, vol. 47, no. 2, pp. 509–524, 2014.

[2] Y. Guo, J. Wan, M. Lu, and W. Niu, "A parts-based method for articulated target recognition in laser radar data," *Optik*, vol. 124, no. 17, pp. 2727–2733, 2013.

[3] F. Tombari, S. Salti, and L.Di Stefano, "Unique signatures of histograms for local surface description," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 356–369.

[4] A. Mian, M. Bennamoun, and R. Owens, "On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes," *Int. J. Comput. Vis.*, vol. 89, no. 2, pp. 348–361, 2010.

[5] F. M. Sukno, J. L. Waddington, and P. F. Whelan, "Comparing 3D descriptors for local search of craniofacial landmarks," in *Proc. Int. Symp. Vis. Comput.*, 2012, pp. 92–103.

[6] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[7] C. Creusot, N. Pears, and J. Austin, "A machine-learning approach to keypoint detection and landmarking on 3D meshes," *Int. J. Comput. Vis.*, vol. 102, no. 1–3, pp. 146–179, 2013.

[8] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view RGB-D object dataset," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 1817–1824.

[9] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3D object recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 998–1005.

[10] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 1–4.

[11] A. Aldoma, F. Tombari, L. Di Stefano, and M. Vincze, "A global hypotheses verification method for 3D object recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 511–524.

[12] M. Kaiser, X. Xu, B. Kwolek, S. Sural, and G. Rigoll, "Towards using covariance matrix pyramids as salient point descriptors in 3D point clouds," *Neurocomputing*, vol. 120, pp. 101–112, 2013.

[13] E. Akagündüz and İ. Ulusoy, "3D object recognition from range images using transform invariant object representation," *Electron. Lett.*, vol. 46, no. 22, pp. 1499–1500, 2010.

[14] N. Bayramoglu and A. Alatan, "Shape index SIFT: Range image recognition using local features," in *Proc. 20th Int. Conf. Pattern Recognit.*, 2010, pp. 352–355.

[15] U. Castellani, M. Cristani, S. Fantoni, and V. Murino, "Sparse points matching by combining 3D mesh saliency with statistical descriptors," *Comput. Graph. Forum*, vol. 27, no. 2, pp. 643–652, 2008.

[16] B. Bustos, D. Keim, D. Saupe, T. Schreck, and D. Vranić, "Feature-based similarity search in 3D object databases," *ACM Comput. Surv.*, vol. 37, no. 4, pp. 345–387, 2005.

[17] A. Bronstein, M. Bronstein, and R. Kimmel, *Numerical Geometry of Non-Rigid Shapes*. New York, NY, USA: Springer-Verlag, 2008.

[18] E. Paquet, M. Rioux, A. Murching, T. Naveen, and A. Tabatabai, "Description of shape information for 2-D and 3-D objects," *Signal Process.: Image Commun.*, vol. 16, no. 1, pp. 103–122, 2000.

[19] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions," *ACM Trans. Graphics*, vol. 21, no. 4, pp. 807–832, 2002.

[20] R. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3D recognition and pose using the viewpoint feature histogram," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 2155–2162.

[21] L. Shang and M. Greenspan, "Real-time object recognition in sparse range images using error surface embedding," *Int. J. Comput. Vis.*, vol. 89, no. 2, pp. 211–228, 2010.

[22] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[23] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. Van Gool, "Hough transform and 3D SURF for robust three dimensional classification," in *Proc. 11th Eur. Conf. Comput. Vis.*, 2010, pp. 589–602.

[24] P. Besl and R. Jain, "Three-dimensional object recognition," *ACM Comput. Surv.*, vol. 17, no. 1, pp. 75–145, 1985.

[25] J. Brady, N. Nandhakumar, and J. Aggarwal, "Recent progress in the recognition of objects from range data," in *Proc. 9th Int. Conf. Pattern Recognit.*, 1988, pp. 85–92.

[26] F. Arman and J. Aggarwal, "Model-based object recognition in dense-range images–A review," *ACM Comput. Surv.*, vol. 25, no. 1, pp. 5–43, 1993.

[27] R. Campbell and P. Flynn, "A survey of free-form object representation and recognition techniques," *Comput. Vis. Image Understanding*, vol. 81, no. 2, pp. 166–210, 2001.

[28] G. Mamic and M. Bennamoun, "Representation and recognition of 3D free-form objects," *Digit. Signal Process.*, vol. 12, no. 1, pp. 47–76, 2002.

[29] A. Mian, M. Bennamoun, and R. Owens, "Automated 3D model-based free-form object recognition," *Sens. Rev.*, vol. 24, no. 2, pp. 206–215, 2004.

[30] J. Salvi, C. Matabosch, D. Fofi, and J. Forest, "A review of recent range image registration methods with accuracy evaluation," *Image Vis. Comput.*, vol. 25, no. 5, pp. 578–596, 2007.

[31] A. Mian, M. Bennamoun, and R. Owens, "Automatic correspondence for 3D modeling: An extensive review," *Int. J. Shape Model.*, vol. 11, no. 2, pp. 253–291, 2005.

[32] A. Bronstein, M. Bronstein, and M. Ovsjanikov, "3D features, surface descriptors, and object descriptors," http://www.inf.usi.ch/phd/sarimbekov/courses/ti2010/index.php?action=topic&id=14, 2010.

[33] F. Tombari, S. Salti, and L. Di Stefano, "Performance evaluation of 3D keypoint detectors," *Int. J. Comput. Vis.*, vol. 102, no. 1, pp. 198–220, 2013.

[34] P. Bariya and K. Nishino, "Scale-hierarchical 3D object recognition in cluttered scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1657–1664.

[35] S. Salti, F. Tombari, and L. Stefano, "A performance evaluation of 3D keypoint detectors," in *Proc. Int. Conf. 3D Imaging, Model., Process., Vis. Transmiss.*, 2011, pp. 236–243.

[36] K. Bowyer, K. Chang, and P. Flynn, "A survey of approaches and challenges in 3D and multi-modal 3D+2D face recognition," *Comput. Vis. Image Understanding*, vol. 101, no. 1, pp. 1–15, 2006.

[37] H. Chen and B. Bhanu, "Human ear recognition in 3D," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 718–737, May 2007.

[38] A. Mian, M. Bennamoun, and R. Owens, "Three-dimensional model-based object recognition and segmentation in cluttered scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 10, pp. 1584–1601, Oct. 2006.

[39] A. Flint, A. Dick, and A. Van den Hengel, "Local 3D structure recognition in range images," *IET Comput. Vis.*, vol. 2, no. 4, pp. 208–217, 2008.

[40] I. Sipiran and B. Bustos, "Harris 3D: A robust extension of the Harris operator for interest point detection on 3D meshes," *The Vis. Comput.: Int. J. Comput. Graph.*, vol. 27, pp. 963–976, 2011.

[41] R. Unnikrishnan and M. Hebert, "Multi-scale interest regions from unorganized point clouds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2008, pp. 1–8.

[42] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Gool, "A comparison of affine region detectors," *Int. J. Comput. Vis.*, vol. 65, no. 1, pp. 43–72, 2005.

[43] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 2, 2004, pp. 498–506.

[44] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, "Rotational projection statistics for 3D local surface description and object recognition," *Int. J. Comput. Vis.*, vol. 105, no. 1, pp. 63–86, 2013.

[45] Y. Guo, F. Sohel, M. Bennamoun, J. Wan, and M. Lu, "RoPS: A local feature descriptor for 3D rigid objects based on rotational projection statistics," in *Proc. 1st Int. Conf. Commun., Signal Process., Appl.*, 2013, pp. 1–6.

[46] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, "TriSI: A distinctive local surface descriptor for 3D modeling and object recognition," in *Proc. 8th Int. Conf. Comput. Graph. Theory Appl.*, 2013, pp. 86–93.

[47] B. Taati and M. Greenspan, "Local shape descriptor selection for object recognition in range data," *Comput. Vis. Image Understanding*, vol. 115, no. 5, pp. 681–694, 2011.

[48] P. Bariya, J. Novatnack, G. Schwartz, and K. Nishino, "3D geometric scale variability in range images: Features and descriptors," *Int. J. Comput. Vis.*, vol. 99, no. 2, pp. 232–255, 2012.

[49] P. Flynn and R. Campbell, "A WWW-accessible database for 3D vision research," in *Proc. IEEE Workshop Empir. Eval. Methods Comput. Vis.*, 1998, pp. 148–154.

[50] G. Hetzel, B. Leibe, P. Levi, and B. Schiele, "3D object recognition from range images using local feature histograms," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 2, 2001, pp. II-394–II-399.

[51] B. Taati, M. Bondy, P. Jasiobedzki, and M. Greenspan, "Variable dimensional local shape descriptors for object recognition in range data," in *Proc. 11th IEEE Int. Conf. Comput. Vis.*, 2007, pp. 1–8.

[52] S. Helmer, D. Meger, M. Muja, J. Little, and D. Lowe, "Multiple viewpoint recognition and localization," in *Proc. 10th Asian Conf. Comput. Vis.*, 2010, pp. 464–477.

[53] F. Tombari, S. Salti, and L. Di Stefano, "A combined texture-shape descriptor for enhanced 3D feature matching," in *18th IEEE Int. Conf. Image Process.*, 2011, pp. 809–812.

[54] A. Janoch, S. Karayev, Y. Jia, J. Barron, M. Fritz, K. Saenko, and T. Darrell, "A category-level 3-D object dataset: Putting the kinect to work," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2011, pp. 1168–1174.

[55] J. Sturm, S. Magnenat, N. Engelhard, F. Pomerleau, F. Colas, W. Burgard, D. Cremers, and R. Siegwart, "Towards a benchmark for RGB-D SLAM evaluation," in *Proc. RGB-D Workshop Adv. Reason. Depth Cameras Robot.: Sci. Syst. Conf.*, 2011, pp. 1–3.

[56] E. Rodolà, A. Albarelli, F. Bergamasco, and A. Torsello, "A scale independent selection process for 3D object recognition in cluttered scenes," *Int. J. Comput. Vis.*, vol. 102, pp. 129–145, 2013.

[57] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Tech.*, 1996, pp. 303–312.

[58] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The Princeton shape benchmark," in *Proc. Int. Conf. Shape Model. Appl.*, 2004, pp. 167–178.

[59] A. Bronstein, M. Bronstein, and R. Kimmel, "Efficient computation of isometry-invariant distances between surfaces," *SIAM J. Sci. Comput.*, vol. 28, no. 5, pp. 1812–1836, 2006.

[60] E. Boyer, A. Bronstein, M. Bronstein, B. Bustos, T. Darom, R. Horaud, I. Hotz, Y. Keller, J. Keustermans, A. Kovnatsky, R. Litman, J. Reininghaus, I. Sipiran, D. Smeets, P. Suetens, D. Vandermeulen, A. Zaharescu, and V. Zobel, "SHREC 2011: Robust feature detection and description benchmark," in *Proc. Eurographics Workshop Shape Retrieval*, 2011, pp. 79–86.

[61] A. Kovnatsky, M. Bronstein, A. Bronstein, and R. Kimmel, "Photometric heat kernel signatures," in *Proc. 3rd Int. Conf. Scale Space Variational Methods Comput. Vis.*, pp. 616–627, 2012.

[62] A. Zaharescu, E. Boyer, and R. Horaud, "Keypoints and local descriptors of scalar functions on 2D manifolds," *Int. J. Comput. Vis.*, vol. 100, pp. 78–98, 2012.

[63] A. Frome, D. Huber, R. Kolluri, T. Bülow, and J. Malik, "Recognizing objects in range data using regional point descriptors," in *Proc. 8th Eur. Conf. Comput. Vis.*, 2004, pp. 224–237.

[64] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.

[65] F. Mokhtarian, N. Khalili, and P. Yuen, "Multi-scale free-form 3D object recognition using 3D models," *Image Vis. Comput.*, vol. 19, no. 5, pp. 271–281, 2001.

[66] S. M. Yamany and A. A. Farag, "Surface signatures: An orientation independent free-form surface representation scheme for the purpose of objects registration and matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 8, pp. 1105–1120, Aug. 2002.

[67] R. Gal and D. Cohen-Or, "Salient geometric features for partial shape matching and similarity," *ACM Trans. Graph.*, vol. 25, no. 1, pp. 130–150, 2006.

[68] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches," *Pattern Recognit. Lett.*, vol. 28, no. 10, pp. 1252–1262, 2007.

[69] B. Matei, Y. Shan, H. Sawhney, Y. Tan, R. Kumar, D. Huber, and M. Hebert, "Rapid object indexing using locality sensitive hashing and joint 3D-signature space estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 7, pp. 1111–1126, Jul. 2006.

[70] Y. Zhong, "Intrinsic shape signatures: A shape descriptor for 3D object recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2009, pp. 689–696.

[71] P. Glomb, "Detection of interest points on 3D data: Extending the Harris operator," *Comput. Recognit. Syst.*, vol. 7, pp. 103–111, 2009.

[72] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. Alvey Vis. Conf.*, 1988, vol. 15, p. 50.

[73] J. Sun, M. Ovsjanikov, and L. Guibas, "A concise and provably informative multi-scale signature based on heat diffusion," *Comput. Graph. Forum*, vol. 28, no. 5, pp. 1383–1392, 2009.

[74] T. Lindeberg, "Scale-space theory: A basic tool for analyzing structures at different scales," *J. Appl. Stat.*, vol. 21, nos. 1-2, pp. 225–270, 1994.

[75] E. Akagündüz and İ. Ulusoy, "3D object representation using transform and scale invariant 3D features," in *Proc. 11th IEEE Int. Conf. Comput. Vis.*, 2007, pp. 1–8.

[76] T. Darom and Y. Keller, "Scale invariant features for 3D mesh models," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2758–2769, May 2012.

[77] X. Li and I. Guskov, "Multi-scale features for approximate alignment of point-based surfaces," in *Proc. 3rd Eurographics Symp. Geometry Process.*, 2005, p. 217.

[78] T. Lo and J. Siebert, "Local feature extraction and matching on range images: 2.5D SIFT," *Comput. Vis. and Image Understanding*, vol. 113, no. 12, pp. 1235–1250, 2009.

[79] J. Novatnack, K. Nishino, and A. Shokoufandeh, "Extracting 3D shape features in discrete scale-space," in *Proc. 3rd Int. Symp. 3D Data Process., Visualization, and Transmiss.*, 2006, pp. 946–953.

[80] J. Koenderink, "The structure of images," *Biol. Cybern.*, vol. 50, no. 5, pp. 363–370, 1984.

[81] J. Novatnack and K. Nishino, "Scale-dependent 3D geometric features," in *Proc. 11th IEEE Int. Conf. Comput. Vis.*, 2007, pp. 1–8.

[82] G. Zou, J. Hua, Z. Lai, X. Gu, and M. Dong, "Intrinsic geometric scale space by shape diffusion," *IEEE Trans. Visualization and Comput. Graph.*, vol. 15, no. 6, pp. 1193–1200, Nov./Dec. 2009.

[83] T. Hou and H. Qin, "Efficient computation of scale-space features for deformable shape correspondences," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 384–397.

[84] H. Dutagaci, C. Cheung, and A. Godil, "Evaluation of 3D interest point detection techniques via human-generated ground truth," *Vis. Comput.*, vol. 28, no. 9, pp. 901–917, 2012.

[85] H. Ho and D. Gibbins, "Curvature-based approach for multi-scale feature extraction from 3D meshes and unstructured point clouds," *IET Comput. Vis.*, vol. 3, no. 4, pp. 201–212, 2009.

[86] A. Flint, A. Dick, and A. Hengel, "THRIFT: Local 3D structure recognition," in *Proc. 9th Conf. Digital Image Comput. Tech. and Appl.*, 2007, pp. 182–188.

[87] J. Stuckler and S. Behnke, "Interest point detection in depth images through scale-space surface analysis," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 3568–3574.

[88] H. Ho and D. Gibbins, "Multi-scale feature extraction for 3D surface registration using local shape variation," in *Proc. 23rd Int. Conf. Image Vis. Comput.*, 2008, pp. 1–6.

[89] J. Hua, Z. Lai, M. Dong, X. Gu, and H. Qin, "Geodesic distance-weighted shape vector image diffusion," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 6, pp. 1643–1650, Nov./Dec. 2008.

[90] G. Zou, J. Hua, M. Dong, and H. Qin, "Surface matching with salient keypoints in geodesic scale space," *Comput. Anim. Virtual Worlds*, vol. 19, no. 3/4, pp. 399–410, 2008.

[91] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, "Surface feature detection and description with applications to mesh matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 373–380.

[92] M. Pauly, R. Keiser, and M. Gross, "Multi-scale feature extraction on point-sampled surfaces," *Comput. Graph. Forum*, vol. 22, no. 3, 2003, pp. 281–289.

[93] Y. Ioannou, B. Taati, R. Harrap, and M. Greenspan, "Difference of normals as a multi-scale operator in unorganized point clouds," in *Proc. 2nd Int. Conf. 3D Imaging, Model., Process., Vis. Transmiss.*, 2012, pp. 501–508.

[94] J. Hu and J. Hua, "Salient spectral geometric features for shape matching and retrieval," *Vis. Comput.*, vol. 25, no. 5, pp. 667–675, 2009.

[95] A. M. Bronstein, M. M. Bronstein, B. Bustos, U. Castellaniy, M. Crisani, B. Falcidieno, L. J. Guibasy, I. Kokkinos, V. Murino, I. Isipiran, M. Ovsjanikovy, G. Patané, M. Spagnuolo, and J. Sun, "SHREC 2010: Robust feature detection and description benchmark," *Proc. Eurographics Workshop 3D Object Retrieval*, vol. 2, no. 5, 2010, p. 6.

[96] Y. Caspi and M. Irani, "Spatio-temporal alignment of sequences," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 11, pp. 1409–1424, Nov. 2002.

[97] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Proc. 9th Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.

[98] F. Stein and G. Medioni, "Structural indexing: Efficient 3D object recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 125–145, Feb. 1992.

[99] C. Chua and R. Jarvis, "Point signatures: A new representation for 3D object recognition," *Int. J. Comput. Vis.*, vol. 25, no. 1, pp. 63–85, 1997.

[100] Y. Sun and M. Abidi, "Surface matching by 3D point's fingerprint," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, 2001, vol. 2, pp. 263–269.

[101] X. Li and I. Guskov, "3D object recognition from range images using pyramid matching," in *Proc. 11th IEEE Int. Conf. Comput. Vis.*, 2007, pp. 1–6.

[102] S. Malassiotis and M. Strintzis, "Snapshots: A novel local surface descriptor and matching algorithm for robust 3D surface alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 7, pp. 1285–1290, Jul. 2007.

[103] J. Novatnack and K. Nishino, "Scale-dependent/ invariant local 3D shape descriptors for fully automatic registration of multiple sets of range images," in *Proc. 10th Eur. Conf. Comput. Vis.*, 2008, pp. 440–453.

[104] T. Masuda, "Log-polar height maps for multiple range image registration," *Comput. Vis. Image Understanding*, vol. 113, no. 11, pp. 1158–1169, 2009.

[105] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard, "NARF: 3D range image features for object recognition," in *Proc. Workshop Defining Solving Realistic Percept. Problems Personal Robot. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, vol. 44.

[106] B. Steder, R. Rusu, K. Konolige, and W. Burgard, "Point feature extraction on 3D range scans taking into account object boundaries," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 2601–2608.

[107] E. R. do Nascimento, G. Leivas, A. Wilson, and M. F. Campos, "On the development of a robust, fast and lightweight keypoint descriptor," *Neurocomputing*, vol. 120, pp. 141–155, 2013.

[108] O. Carmichael, D. Huber, and M. Hebert, "Large data sets and confusing scenes in 3-D surface matching and recognition," in *Proc. 2nd Int. Conf. 3-D Digital Imaging Model.*, 1999, pp. 358–367.

[109] J. Assfalg, M. Bertini, A. Bimbo, and P. Pala, "Content-based retrieval of 3-D objects using spin image signatures," *IEEE Trans. Multimedia*, vol. 9, no. 3, pp. 589–599, Apr. 2007.

[110] H. Dinh and S. Kropac, "Multi-resolution spin-images," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2006, vol. 1, pp. 863–870.

[111] S. Ruiz-Correa, L. Shapiro, and M. Melia, "A new signature-based method for efficient 3-D object recognition ," *in IEEE Conf. Comput. Vis. Pattern Recognit.*, 2001, vol. 1, pp. I-769–I-776.

[112] G. Pasqualotto, P. Zanuttigh, and G. Cortelazzo, "Combining color and shape descriptors for 3D model retrieval," *Signal Process.: Image Commun.*, vol. 28, pp. 608–623, 2013.

[113] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. IntelL.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[114] F. Tombari, S. Salti, and L.Di Stefano, "Unique shape context for 3D data description," in *Proc. ACM Workshop 3D Object Retrieval*, 2010, pp. 57–62.

[115] F. M. Sukno, J. L. Waddington, and P. F. Whelan, "Rotationally invariant 3D shape contexts using asymmetry patterns," in *Proc. 8th Int. Conf. Comput. Graph. Theory Appl.*, 2013.

[116] I. Kokkinos, M. Bronstein, R. Litman, and A. Bronstein, "Intrinsic shape context descriptors for deformable shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 159–166.

[117] Y. Guo, F. Sohel, M. Bennamoun, J. Wan, and M. Lu, "Integrating shape and color cues for textured 3D object recognition," in *Proc. 8th IEEE Conf. Ind. Electron. Appl.*, 2013, pp. 1614–1619.

[118] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 3384–3391.

[119] E. Wahl, U. Hillenbrand, and G. Hirzinger, "Surflet-pair-relation histograms: a statistical 3D-shape representation for rapid classification," in *Proc. 4th Int. Conf. 3-D Digit. Imaging and Model.*, 2003, pp. 474–481.

[120] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 3212–3217.

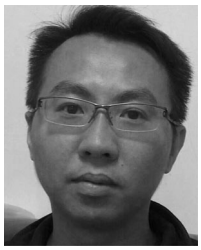
[121] S. Tang, X. Wang, X. Lv, T. X. Han, J. Keller, Z. He, M. Skubic, and S. Lao, "Histogram of oriented normal vectors for object recognition with a depth sensor," in *Proc. 11th Asian Conf. Comput. Vis.*, 2012, pp. 525–538.

[122] M. Bronstein and I. Kokkinos, "Scale-invariant heat kernel signatures for non-rigid shape recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1704–1711.
[123] M. M. Bronstein and A. M. Bronstein, "Shape recognition with spectral distances," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 1065–1071, May 2011.
[124] A. E. Johnson and M. Hebert, "Surface matching for object recognition in complex three-dimensional scenes," *Image Vis. Comput.*, vol. 16, no. 9-10, pp. 635–651, 1998.
[125] S. Nene and S. Nayar, "Closest point search in high dimensions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1996, pp. 859–865.
[126] C. Papazov and D. Burschka, "An efficient RANSAC for 3D object recognition in noisy and occluded scenes," in *Proc. 10th Asian Conf. Comput. Vis.*, 2011, pp. 135–148.
[127] Y. Guo, M. Bennamoun, F. Sohel, J. Wan, and M. Lu, "3D free form object recognition using rotational projection statistics," in *Proc. IEEE 14th Workshop Appl. Comput. Vis.*, 2013, pp. 1–8.
[128] P. J. Flynn and A. K. Jain, "BONSAI: 3D object recognition using constrained search," in *Proc. 3rd Int. Conf. Comput. Vis.*, 1990, pp. 263–267.
[129] C. Papazov, S. Haddadin, S. Parusel, K. Krieger, and D. Burschka, "Rigid 3D geometry matching for grasping of known objects in cluttered scenes," *Int. J. Robot. Res.*, vol. 31, no. 4, pp. 538–553, 2012.
[130] A. Albarelli, E. Rodola, F. Bergamasco, and A. Torsello, "A non-cooperative game for 3D object recognition in cluttered scenes," in *Proc. Int. Conf. 3D Imag. Modeling Process. Vis. Transmiss.*, 2011, pp. 252–259.
[131] A. P. Ashbrook, R. B. Fisher, C. Robertson, and N. Werghi, "Finding surface correspondence for object recognition and registration using pairwise geometric histograms," in *Proc. 5th Eur. Conf. Comput. Vis.*, 1998, pp. 674–686.
[132] F. Tombari and L. D. Stefano, "Hough voting for 3d object recognition under occlusion and clutter," *IPSJ Trans. Comput. Vis. Appl.*, vol. 4, pp. 20–29, 2012.
[133] J. Knopp, M. Prasad, and L. Van Gool, "Orientation invariant 3D object classification using hough transform based methods," in *Proc. ACM Workshop 3D Object Retrieval*, 2010, pp. 15–20.
[134] Y. Lamdan and H. J. Wolfson, "Geometric hashing: A general and efficient model-based recognition scheme," in *Proc. Int. Conf. Comput. Vision*, 1988, vol. 88, pp. 238–249.
[135] P. Besl and N. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
[136] S. Wuhrer, Z. Ben Azouz, and C. Shu, "Posture invariant surface description and feature extraction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 374–381.
[137] K. Lai, L. Bo, X. Ren, and D. Fox, "A scalable tree-based approach for joint object and pose recognition," in *Proc. 25th Conf. Artif. Intell.*, 2011, pp. 1474–1480.

Yulan Guo received the BEng degree in communication engineering from the National University of Defense Technology in 2008, where he is currently working toward the PhD degree. He is a visiting (joint) Ph.D. student at the University of Western Australia since November 2011. His research interests include 3D object recognition, 3D face recognition, 3D modeling, pattern recognition, and signal processing.

Mohammed Bennamoun received the MSc degree in control theory from Queens University, Kingston, ON, Canada, and the Ph.D. degree in computer vision from Queens/QUT in Brisbane, Australia. He is currently a Winthrop Professor at the University of Western Australia, Australia. He served as a guest editor for a couple of special issues in International journals such as the *International Journal of Pattern Recognition and Artificial Intelligence*. His research interests include control theory, robotics, obstacle avoidance, object recognition, artificial neural networks, signal/image processing, and computer vision. He published more than 200 journal and conference publications. He was selected to give conference tutorials at the European Conference on Computer Vision (ECCV) and the International Conference on Acoustics Speech and Signal Processing (ICASSP). He organized several special sessions for conferences, e.g., the IEEE International Conference in Image Processing (ICIP). He also contributed in the organization of many local and international conferences.

Ferdous Sohel received the BSc degree in computer science and engineering from the Bangladesh University of Engineering and Technology, Dhaka, Bangladesh, in 2002, and the PhD degree from Monash University, Australia, in 2008. He is currently a Research Assistant Professor at the University of Western Australia. His research interests include computer vision, image processing, pattern recognition, and shape coding. He has published more than 40 scientific articles. He has served on the program committee of a number of international conferences including an ICCV workshop. He is a recipient of prestigious Discovery Early Career Research Award (DECRA) funded by the Australian Research Council. He is a recipient of Mollie Holman Doctoral Medal and Faculty of Information Technology Doctoral Medal form Monash University.



Min Lu received the BEng and PhD degrees from the National University of Defense Technology (NUDT) in 1997 and 2003, respectively. He is currently an associate professor with the College of Electronic Science and Engineering at NUDT. He is also a visiting scholar at the University of Waterloo, Canada. His research interests include computer vision, pattern recognition, remote sensing, and signal and image processing.

Jianwei Wan received the PhD degree from the National University of Defense Technology (NUDT) in 1999. He is currently a professor with the College of Electronic Science and Engineering at NUDT. He has been a senior visiting scholar at the University of Windsor, Canada, from May 2008 to May 2009. He published more than 100 journal and conference publications. His research interests include signal processing, remote sensing, computer vision, pattern recognition, and hyperspectral image processing.