We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

3D Object Registration and Recognition using Range Images

Erdem Akagündüz and İlkay Ulusoy
Middle East Technical University
Turkey

1. Introduction

In recent years, retrieving semantic information from digital cameras, for instance object recognition, has become one of the hottest topics of computer vision. Since the boundaries of this problem range from recognizing objects in a range image to estimating the pose of an object from an image sequence, a variety of studies exist in the literature. Before remembering the previous approaches on the subject, it is better to refer to the definitions of the elementary attributes of the fundamental step in object recognition, feature extraction, which are repeatability under orientation, scaling, sampling and noise.

Orientation invariance in computer vision is the ability, which enables a method to extract the same features from the original and the rotated (oriented) version of an image (2D or range image etc.). Since any movement in the scene or camera introduces considerable rotation in the signal, this is a basic ability which is usually satisfied in recent object recognition approaches. The critical setback for orientation invariance is the self occlusion caused by the orientation of the object with respect to the camera. Many recent studies solve this problem by partial matching algorithms, which also enable pose estimation in scenes with diverse rotation.

The ability to extract features independent of their scale, namely scale invariance, is another important ability for a method. For 2D images, it is both related to object's size and pixel resolution. Thus scale invariance in 2D also correlates with invariance under sampling, which, as it name implies, is the ability to extract similar features from similar signals with different sampling rates. However, since range images encapsulate metric information independent of the resolution they have, the notion of scale and sampling invariance is interpreted in a different manner for them. This different notion is discussed in the succeeding sections.

Finally, the very basic ability of a computer vision system is its robustness to noise. Noise exists in various forms depending on the scene and sensor attributes. Since a very basic operation in image processing literature, unlimited number of approaches exits. However, methods using scale-space of signals have proven to be most robust under different types of noise.

1.1 Previous work

The state of the art challenges in 3D object recognition systems are scale invariance and robustness to occlusion. These two issues are usually the main reasons from which the real world recognition problems inherit their complexity. For object recognition from range

images, the literature neither satisfactorily discusses these issues, nor has yet proposed a sufficient solution.

Range images, which have been usually processed as 2D images, carry both the 3D metric and geometric information of objects in the scene. Several local or global 3D point features and 3D descriptors were derived from these sampled surfaces and used for 3D object recognition, 3D object category recognition, 3D surface matching and 3D registration. Some of these features are SIFT (Lowe, 2004) (as directly applied to 2D rendered range images), 2.5D SIFT (Lo & Siebert, 2009), multi-scale features (Li & Guskov, 2007; Pauly et al., 2003), spin images (Johnson & Hebert, 1999), 3D point signatures (Chua & Jarvis, 1997), 3D shape context (Frome et al, 2004), surface depth, normal and curvature histograms (Hetzel et al., 2001), 3D point fingerprints (Sun & Abidi, 2001), or extended Gaussian images (Horn, 1984).

3D descriptors or histograms generally define the whole or a part of an object, using different properties of the surfaces such as curvatures, normal directions, distances to a base point etc. They are very powerful in representing a surface patch for recognition purpose. However when they are globally defined, they are brittle against occlusions. On the other hand, local descriptors are defined around feature points. However, detecting feature points and estimating the effect region of the local descriptor around a feature point are serious problems. Using fixed sized local descriptors obtained from random points on the surface (Johnson & Hebert, 1999) is one of the earlier approaches.

3D feature points are salient points that are extracted from the range image surfaces. If they are sufficiently repeatable, stable and invariant to scale and orientation, a sparse and robust representation of the sampled surface can be obtained. In addition, if the scales of the features are known, then effect regions can be defined around their center. Thus, multi-scale features obtained using the scale-space of the input surface are very advantageous for scale invariance and robustness to occlusions.

Until recently, very few studies have been reported on the invariance limits of scale in 3D feature extraction. Reference (Li & Guskov, 2007) extracts multi-scale salient features using only two scale levels of the surface normals and analyzes its performance on object recognition for the Stuttgart range image database. Reference (Pauly et al., 2003) extracts multi-scale features which are classified based on surface variation estimation using covariance analysis of local neighborhoods, in order to construct line features. Reference (Lo & Siebert, 2009) define the 2.5D SIFT, the direct implementation of SIFT (Lowe, 2004) framework on range images, however they present their comparison with simple match matrices and avoid giving a comparison of recognition capabilities. All of these methods (Lowe, 2004; Li & Guskov, 2007; Pauly et al., 2003) construct a scale space of the surface using difference of Gaussians (DoG) and seek for the maxima within this scale space, while neither of them attempts to test the scale invariance limits of their methods on scale varying database.

1.2 The notion of scale for range images

Traditionally scale concept in computer vision relates to object's pixel dimension in the image, which correlates to both sensor resolution and object's actual size. Accordingly the scale invariance definition in 2D image processing is the ability to extract features independent of both size and sampling. For this reason solely object size is usually unknown or obtained only proportional to scene or other objects. However for range images, the metric size of the object is acquired independent of sensor resolution. Thus, the range image encapsulates both the object's metric size and its sampling information. This is

the reason why the concept of scale invariance for range images is different than its conventional definition in 2D.

In this approach we define the scale invariance concept for range images as extracting the features independent of the scale or sampling of the object together with a metric size parameter. This enables us to match similar objects of different size and also indicate the scale ratio between them. For this reason we resample the range scans such that the average of the distances between neighboring points is a constant value. When curvature values are calculated at a constant scale/sampling ratio, they become invariant of resolution. If a scale-space definition is used they become scale invariant as well. Thus, the thresholds used for classifying curvature classification become universal. In this study all range images are re-sampled to an average value of 0.5 mm/sample ratio before the features are extracted.

1.3 Scale-space approach

Scale invariance concept is strongly related to the scale-space concept. As thoroughly examined in (Lindeberg, 1994) feature's actual scale or metric size can be obtained from the scale-space of that signal. The nuance between scale invariance and scale information in feature extraction lies in the definition of scale-space. Scale invariance may be satisfied with very simple methods, however if features with scale information is required, a definition of scale-space of the signal is needed. Reader should refer to (Lindeberg, 1994) for a detailed analysis of scale-space concept in computer vision. A scale space of range images are depicted in Figure 1.
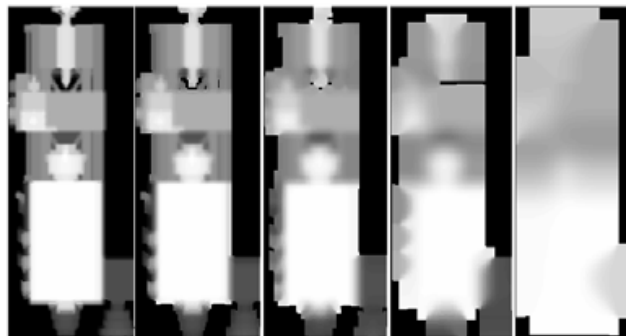

Fig. 1. The scale levels of a range image (Hetzel et al., 2001) constructed by pyramiding are depicted. The black regions are the invalid points.

Among many methods to construct a scale-space for a signal, pyramiding is one of the most preferred. While constructing a pyramid for a range image, the invalid points must be processed carefully. Invalid points are either background or simply unknown since they could not been acquired properly by the scanner. For most 3D data processing methods, these invalid points are simply ignored and calculations are carried out using only the valid points. However if a pyramid of a 3D range image, which contains a group of invalid points, is to be constructed; these invalid points should be handled properly. When constructing a pyramid for a 2D intensity image, the pyramiding operator is convolved throughout the image with no exceptions. However for a depth image, including invalid points; it is possible that one might experience difficulties in boundary regions where valid and invalid points are next to each other. In these occasions, the segments of the pyramiding filter, which corresponds to invalid points, should be omitted. This way the true shape of the object may be preserved in higher scales. In Figure 1, the scale space of a surface is

constructed using Gaussian pyramiding [Burt & Adelson, 1984]. Around the valid point boundaries the smoothing filter avoids blending with invalid points. This way, sharp boundaries of an object can be preserved.

2. 3D feature extraction

In this section, curvature types used to extract features from 3D surfaces are summarized. Different curvatures classification types are reminded.

2.1 Curvature classification

In literature, there are different types of surface curvatures which are used to classify surface patches. We commence by the very basic curvatures, the principle curvatures, from which other curvature values are obtained.

2.1.1 Principle curvatures

In differential geometry, the two principal curvatures at a given point of a surface measure how the surface bends by different amounts in different directions at that point. At each point p of a differentiable surface in 3D Euclidean space one may choose a unique unit normal vector. A normal plane at p is one that contains the normal, and will therefore also contain a unique direction tangent to the surface and cut the surface in a plane curve. This curve will in general have different curvatures for different normal planes at p. The principal curvatures at p, denoted κ_1 and κ_2, are the maximum and minimum values of this curvature. Figure 2 depicts these curvatures and their normal planes.
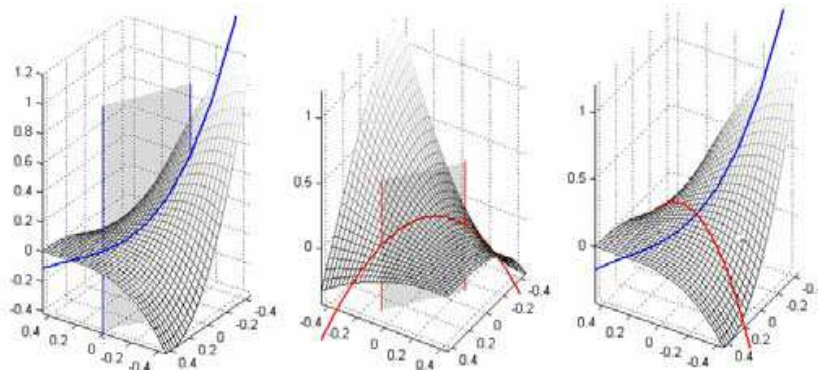
Fig. 2. The normal plane with the maximum curvature is seen. b) The normal plane with the minimum curvature is seen c) κ_1 = 1.56 and κ_2 = -2.37. The surface is a patch from a monkey saddle: $z(x,y) = x^3 - 3 \cdot x \cdot y^2$.

The principal curvature values and the principle directions of the curvatures are calculated by taking the eigenvalue decomposition of the Hessian Matrix which is defined as:

$$\mathbf{H} = \begin{bmatrix} \dfrac{\partial^2 \mathbf{X}}{\partial u^2} & \dfrac{\partial^2 \mathbf{X}}{\partial uv} \\[3mm] \dfrac{\partial^2 \mathbf{X}}{\partial uv} & \dfrac{\partial^2 \mathbf{X}}{\partial v^2} \end{bmatrix} \tag{1}$$

The eigenvalues of this symmetric matrix give the principal curvatures κ_1 and κ_2, where the eigenvectors give the principle curvature directions. Surface points can be classified according to their principal curvature values at that point. A point on a surface is classified as:

Elliptic: ($\kappa_1 \cdot \kappa_2 > 0$) if both principal curvatures have the same sign. The surface is locally convex or concave.

Umbilic: ($\kappa_1 = \kappa_2$) if both principal curvatures are equal and every tangent vector can be considered a principal direction (and *Flat-Umbilic* if $\kappa_1 = \kappa_2 = 0$).

Hyperbolic: ($\kappa_1 \cdot \kappa_2 < 0$) if the principal curvatures have opposite signs. The surface will be locally saddle shaped.

Parabolic: ($\kappa_1 = 0$, $\kappa_2 \neq 0$) if one of the principal curvatures is zero. Parabolic points generally lie in a curve separating elliptical and hyperbolic regions.

This is the basic classification for surfaces according to their principal curvatures. Mean (H) and Gaussian (K) curvatures, shape index (S) and curvedness (C) can also be calculated using the principal curvatures and more essential classifications can be performed using these values.

2.1.2 Mean and gaussian curvatures

Using principal curvatures, Mean (H) and Gaussian (K) Curvatures are calculated as:

$$H = (\kappa_1 + \kappa_2)/2 , K = \kappa_1 \cdot \kappa_2 \tag{2}$$

H is the average of the maximum and the minimum curvature at a point, thus it gives a general idea on how much the point is bent. K is the multiplication of the principal curvatures and its sign indicates whether the surface is locally elliptic or hyperbolic. Using HK values, the regions are defined as in Table 1.

	K>0	K=0	K<0
H<0	Convex (Elliptic or Umbilic)	Ridge (Convex Parabolic)	Saddle Ridge (Hyperbolic)
H=0	(Not possible)	Planar (Flat-Umbilic)	Minimal (Hyperbolic)
H>0	Concave (Elliptic or Umbilic)	Valley (Concave Parabolic)	Saddle Valley (Hyperbolic)

Table 1. Shape Classification in HK curvature space.

2.1 Shape index and curvedness

(Koenderink & Doorn, 1992) defines an alternative curvature representation using the principal curvatures. This approach defines two measures: the shape index (S) and the curvedness (C). Shape index (S) defines the shape type and curvedness (C) decides if the shape is locally planar or not.

$$S = 2/\pi \cdot \arctan\left(\kappa_1 + \kappa_2/\kappa_1 - \kappa_2\right) \ (\kappa_1 > \kappa_2), \ \ C = \sqrt{\kappa_1^2 + \kappa_2^2/2} \tag{3}$$

The shape index value of a point is independent of the scaling of that shape. However C is not scale or resolution invariant. Both S and C are orientation invariant. (Koenderink &

Doorn, 1992) uses S value in order to classify a point. S values changes between [-1,+1] where -1 defines cup shapes (convex elliptical) and + 1 defines cap shapes (concave elliptical). They define constant shape index values in order to define shape types. These values are given in Table 2 below. However their original classification does not differentiate hyperbolic regions into three different types (yellow-orange-red regions, i.e. saddle valley, hyperbola and saddle ridge). For this reason another constant shape index value may be defined (3/16) for this purpose. The Curvedness (C) values are used to understand if the region is planar or not. For planar regions C value is very close to zero (i.e. below the zero threshold C_{zero}) (Table 2).

Convex (Elliptic)	$S \in [+5/8,1] \cap C > C_{zero}$
Convex (Parabolic)	$S \in [+3/8,+5/8] \cap C > C_{zero}$
Saddle Ridge	$S \in [+3/16,+3/8] \cap C > C_{zero}$
Planar	$C < C_{zero}$
Hyperbola	$S \in [-3/16,+3/16] \cap C > C_{zero}$
Concave (Elliptic)	$S \in [-1,-5/8] \cap C > C_{zero}$
Concave (Parabolic)	$S \in [-5/8,-3/8] \cap C > C_{zero}$
Saddle Valley	$S \in [-3/16,+3/16] \cap C > C_{zero}$

Table 2. Shape Index and Curvedness Classification

2.2 Curvature scale-spaces

The curvature values are calculated using surface gradients. Thus they give basic information on surface behavior. In order to calculate surface gradients analytically, explicit surface functions may be used. However in real world applications, the 3D surfaces are digitized into sampled points and there's no global explicit function of the surface. Hence, the surface gradients are calculated within a neighborhood of sampled points. For this reason the calculated curvatures are local approximations, which are valid on a certain scale. Therefore the curvatures are calculated for each scale level, so that curvature scale-spaces are obtained.

In Figure 3, H, K, S and C curvature scale spaces of the range image given in Figure 1 are depicted. Figure 3.a shows the scale-space of H values. The concave regions which have positive H values, are painted in red, where convex regions with negative H values are painted in blue. For both regions the magnitude of the curvature is demonstrated by color intensity. As seen from the figure, the convex region denoted by number 1 is designated as a peak in higher scales since this element is relative large for the given resolution of the 3D scan. Similarly in Figure 3.b, K values obtained from different scales of the surface are depicted. The parabolic regions with positive K values are painted in red, where the hyperbolic regions with negative K values are painted in blue. In Figure 3.c, the shape index values obtained from different scale levels of the surface are examined. Since shape index value is capable of classifying the surface into the fundamental surface types (except planes), the scale-space of the shape index values clearly demonstrate the scale coordinates of the surface features (colors correspond to Table 2). As seen from this figure, the convex region denoted by number 1 is designated as a peak in higher scales and is not existent in lower scales. Finally in Figure 3.d, C values obtained from different scale levels are examined.

Curvedness values can be used to detect planar regions, since regions having sufficiently small curvedness values are defined as planes. In Figure 3.d, the gray level intensities designate C values, where zero C value corresponds to black. In this figure, the region denoted by numbers 1, 2, 3 and 4 in different scale levels corresponds to a planar region on the surface and it is designated as plane in the first four scale levels in C scale–space. However in the fifth C scale-space level is it not designated as plane, since in this scale, the region is designated as pit (Figure 3.a and 3.c).
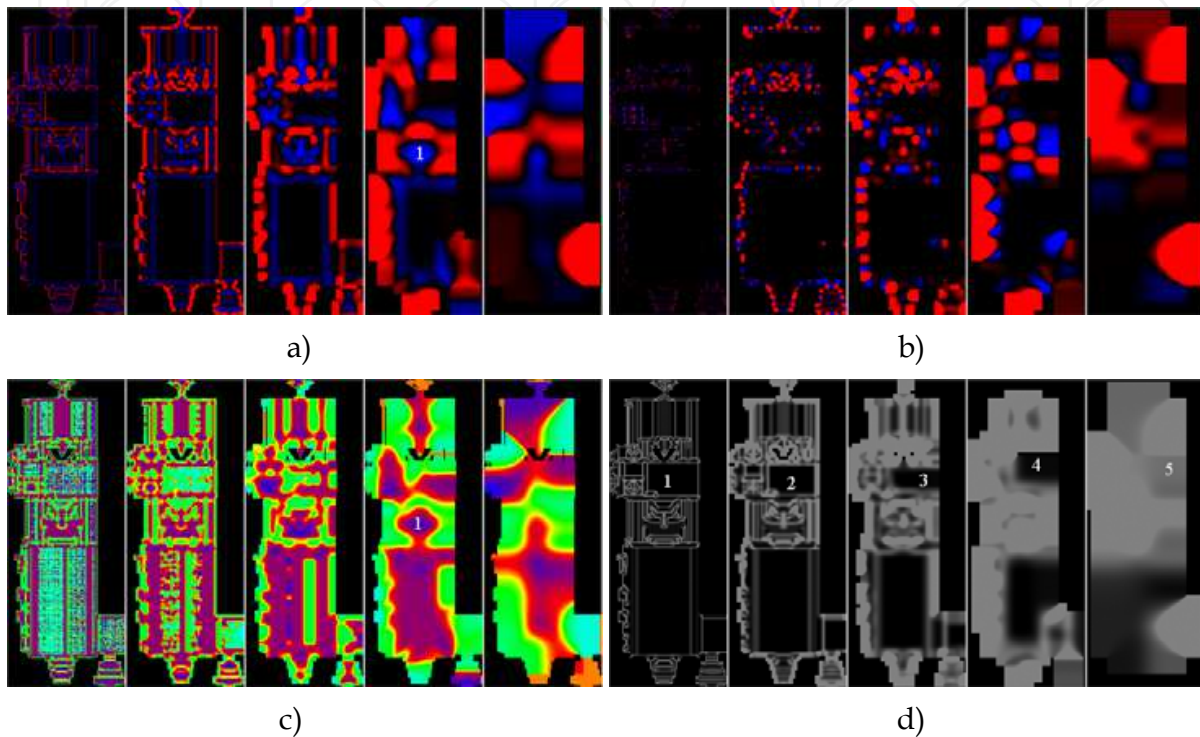
Fig. 3. Curvature scale-spaces: a) Mean Curvature (H), b) Gaussian Curvature (K), c) Shape Index (S), d) Curvedness (C).

2.3 Extraction of scale invariant features

Most curvature oriented methods find salient points on the range image which are defined around a local patch. If the size of this effective local region is kept constant, only the features which are smaller than this size of the local region can be extracted over the surface. In other words, if principal curvature values of a point are calculated using the neighboring points around a radius of d mm, only the features which have sizes smaller than $\pi \cdot d^2$ mm^2 could be extracted. Thus, if the method uses a constant radius of locality when extracting the features, there is no notion of scale invariance. Even though shape index value is invariant to scale, if the principal curvatures are calculated at a constant scale level, the feature's scale is still ambiguous, unknown and incomparable.

The only method to overcome this fact is to search for the features at different scale levels of the surface. For this purpose, a scale space of the surface should be constructed and curvature values should be calculated at all levels of this scale-space. When moved through this scale-space, smaller features vanish and larger features which were not obvious at smaller scales become visible towards the higher scales.

There are different methods to construct a scale-space of a surface. In this study, we use the Gaussian Pyramid approach [6] because image size decreases exponentially with the scale level and hence also the amount of computation required to process the data. First, HK (or SC) values are calculated and pixels are classified based on surface types separately for each pyramid layer using Table 1. The regions are classified to eight different types, namely peak, saddle ridge, convex cylinder, pit, saddle valley, concave cylinder, plane and hyperbolic, similar to [1]. Then each layer is expanded by up-sampling to a fixed size. Finally, by putting each classified curvature scale level on top of each other, a scale-space of the classified features is obtained. We call this scale space as UVS space where U and V are used for surface dimensions and S is used for scale (Figure 4). The method for constructing a UVS space is detailed in [7].
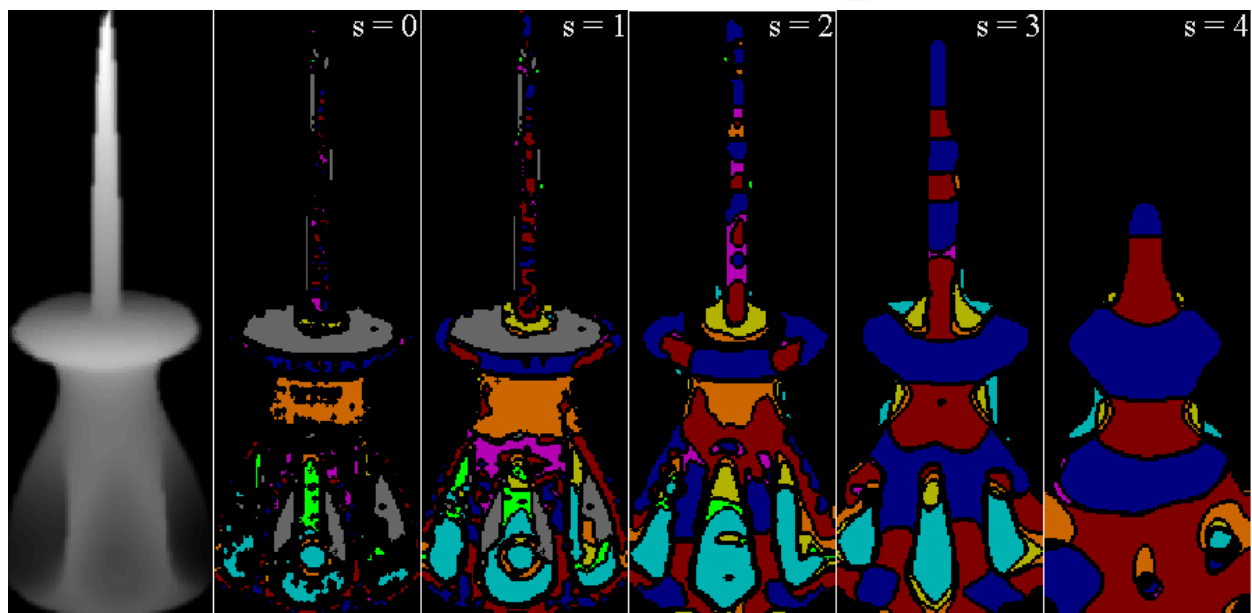
Fig. 4. Labeled layers of UVS space constructed by HK (or SC) values where S is increasing from left to right. The original surface level is indicated by "S=0". Labels are given by colors. (*peak: blue, saddle ridge: red, convex cylinder: purple, pit: cyan, saddle valley: yellow, concave cylinder: green, hyperbolic: orange, plane: gray.*)

In order to extract the features from curvature scale spaces, the following procedure is applied. Inside the classified scale-space, each connected component of the same type of voxels is found and considered as a feature element on the surface. The total number of voxels inside the connected component represents the feature's volume (v_i) and the centre of mass of the connected component is the positional centre of the feature ($\mathbf{x}_i$). Since a connected component may have different number of elements in different scales, a weighted average of the scale value is calculated for each connected component as the actual scale of that feature (s_i). Although each connected component has different numbers of elements in each different scale level, it has the biggest number of elements on the scale which is closest to its actual scale. The area (A) of the connected component at this layer is used to calculate the radius (r_i) (4), which also defines the size of that fundamental element.

$$r = \sqrt{A \Big/ 4 \cdot \pi} \tag{4}$$

Finally, for each feature element extracted from the surface, the following attributes are obtained: the type (t_i), the positional centre of mass ($\mathbf{x}_i$), the scale (s_i), the size (r_i) and the volume (v_i) (Figure 5).
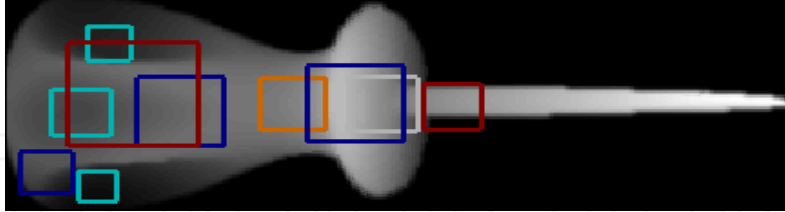
Fig. 5. Ten largest extracted features are shown as squares where the feature center ($\mathbf{x}_i$) is given by the square center, the feature size (r_i) is given by the square size and the feature type is given by its color.

2.3.1 Scale-space localization

As mentioned in the previous sections, the location and the scale of a feature are estimated by computing the weighted average of the curvature values of the elements (voxels) covered by the connected component defining that feature in the UVS volume. In this section, we would like to show that this kind of localization is very precise.

A weight value is assigned to each voxel inside the connected component using the second norm of the absolute differences of the curvature values from the applied threshold values (if HK curvature scale-space is to be constructed):

$$w_{i,j} = \left(\left(\mathbf{H_{i,j}} - \in_\mathbf{H} \right)^2 + \left(\mathbf{K_{i,j}} - \in_\mathbf{K} \right)^2 \right)^{\frac{1}{2}} \tag{5}$$

The localization of the features in the UVS volume and on the surface is crucial. This localization should be robust to noise and any type of transformations. For example in (Lowe, 2004), the SIFT descriptor is sought in a scale space of different octaves, where all local maxima (or minima) are selected as features. Instead of using the weighted averages, a similar approach to (Lowe, 2004) may be applied to our method, where a single maximum for each connected component is found in the UVS space. However, as it is seen in Figure 6,
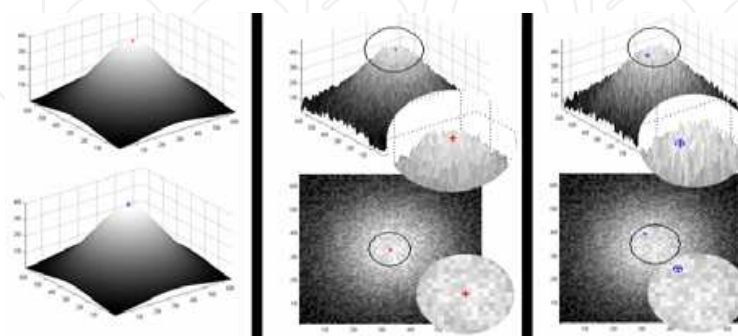
Fig. 6. Localization of the peak feature using both methods a) For the ideal surface both methods localize the peak feature correctly (top: weighted average, bottom: single maximum). b) For the noisy surface, the feature is still correctly localized using weighted averages. c) The localization fails under noise when the single maximum method is used.

this approach will fail under noise or in complex scenes. Imagine that we have a simple surface with a single peak and its noisy version. When the center of the peaky feature is sought over the surface with noise, it is seen that a local maximum value inside a connected component may divert the center from its original position (Figure 6.c) although the surface is smoothed in higher scales. In return for this, our method localizes the features correctly (Figure 6.b) and is robust to noise.

2.3.2 Importance of scale-space search

As mentioned before, the main motivation and contribution in this study is the scale-space search of curvature values. In order to show the significance of scale space search, the feature extraction results with and without scale-space search are depicted in this subsection.

When the scale-space search is omitted, the curvatures are found only at the given resolution. Even though scale/sampling ratio is controlled (usually it is not controlled since this ratio changes even when the distance between the 3D scanner and the object changes), the types and the sizes of the features are detected wrong when only the given resolution is considered. In Figure 7, the extracted features of the original screwdriver object and its scaled version (by 0.8), both with and without scale-space search, are depicted. The features extracted from the original and the scaled versions of the screwdriver object using only the given resolution are usually mislabeled (Figure 7.a,b). For example, only some planar features are located on the handle of the object and the actual shape of the handle could not be extracted. Thus, most surface structures are generally labeled as planes when only the given resolution is considered. Since the original resolutions of 3D images are very high, even inside a peaky region, a point may be considered as a plane because the neighboring points are very close to each other. As a result, many small planar regions are detected in large concave areas. Thus, only when a scale space search is performed, the real types and sizes of the surface features could be extracted (Figure 7.c,d).
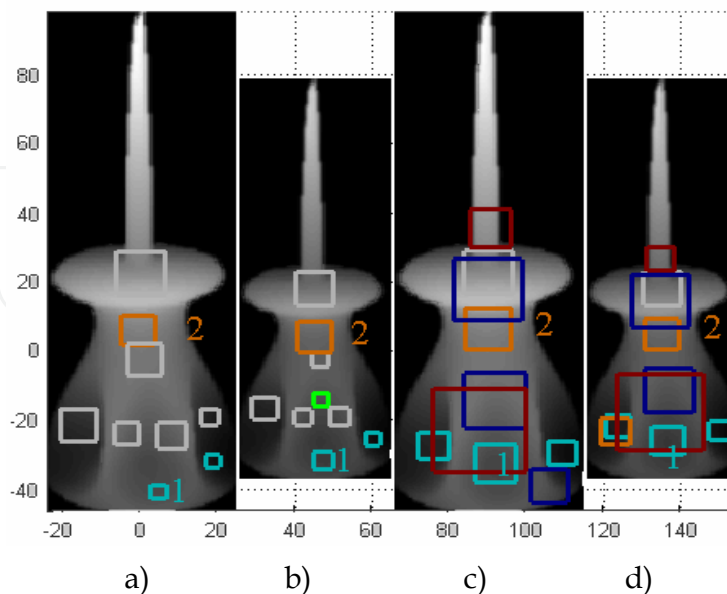
Fig. 7. Ten largest features extracted using a) original image without scale-space search, b) scaled image (by 0.8) without scale-space search, c) original image with scale-space search, d) scaled image (by 0.8) with scale-space search.

Second of all, the feature sizes can only be correctly extracted if scale-space search is used. Since Figure 7.d is 0.8 times resized version of Figure 7.c, the radius of a feature found in Figure 7.d is also 0.8 times smaller than the radius of the corresponding feature found in Figure 7.c. However, this property of robustness under scaling can not be observed when only the given resolution is used for feature detection. Although most features are correspondent in Figures 7.a and 7.b, their radii are faulty. The shape in Figure 7.b is smaller than the one in Figure 7.a by a ratio of 0.8, however the radii of the features numbered as 1 and 2 in Figure 7.b are larger than the radii of the corresponding features in Figure 7.a. Thus it is clearly seen that in order to extract features with their size properties, scale-space search is a must.

3. Topological model

In this section, using the feature elements extracted from the UVS space, a global 3D surface representation is constructed. A scale and orientation invariant representation is proposed, where the spatial topology of the object is given as a graph structure which carries the relative information among the features over the 3D surface. The relativity is not only in terms of spatial information but in terms of orientation and scaling as well.

As explained in the previous subsection, for each feature element the following attributes are obtained: the type (t_i), the volume (v_i), the positional centre of mass ($\mathbf{x}_i$), the orientation vector ($\mathbf{n}_i$), the scale (s_i) and the size (r_i). If each feature is referred as a node in a topology graph where the nodes carry the feature element's attributes and the links between the nodes carry some relative information; a topological representation may be obtained.

In order to make this representation orientation and scale invariant, the links between the nodes must carry "relative" or in other words "normalized" information. An example of this type of relation could be the length between two nodes normalized using a scale invariant measure specific for that topology. The relative 3D direction between two nodes might also be used. Furthermore scale difference between the nodes would carry scale invariant information. These relative link attributes can be listed such as:

Normalized distance from Node A to B ($|\mathbf{x}_B - \mathbf{x}_A|/r_A$ or $|\mathbf{x}_B - \mathbf{x}_A|/2^s$): The distance between two nodes can be normalized using the scale or the size of a base node (A) in the topology. Thus this relation stays invariant under scaling and orientation of the source signal.

Link Vectors or Link Angles $(\mathbf{x}_B - \mathbf{x}_A)/|\mathbf{x}_B - \mathbf{x}_A|$: The unit vector from a node (B) to a specific base node (A) in the topology will also remain invariant under scale. However this link vector will be variant under orientation. In order to make this information both scale and orientation invariant, the angles between these unit vectors might be used. For an n-node topology there would be n-1 unit vectors. For any two these unit vectors, an angle can be calculated. This angle will be invariant of both scale and orientation. For n-1 number of unit vectors, we would obtain C(n-1,2) number of angles, which is also equal to (n-1)·(n-2)/2. The angle can be calculated as:

$$\alpha_{BAC} = \alpha_{CAB} = \cos^{-1}\left(\left[(\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A)/|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A|\right]^{\mathbf{T}} \cdot \left[(\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A)/|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A|\right]\right) \tag{6}$$

Normal Vector Difference ($\mathbf{n}_B - \mathbf{n}_A$): The difference vector between the unit normal vector of node B ($\mathbf{n}_B$) and unit normal vector of the specific base node A ($\mathbf{n}_A$) will stay invariant of orientation and scale.

Feature Scale Difference and Size Ratio (s_A-s_B or r_A/r_B): As explained in (Lindeberg, 1994), the scale difference between two nodes is invariant to scaling. The ratio of the size of a node (B) (which is strongly related to scale of that feature) to the size a specific base node (A) will stay invariant of scaling as well.

Imagine we have a four-node topology with nodes A, B, C and D. Assume that node A is defined as the base node of the topology. Then the following vector will be scale and orientation invariant:

$$\lambda_i = \left[\ t_A, t_B, t_C, t_D, \frac{|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A|}{r_{A(or\, 2^{s_A})}}, \frac{|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A|}{r_A}, \frac{|\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A|}{r_A}, \alpha_{BAC}, \alpha_{BAD}, \alpha_{DAC} \cdots \right. \tag{7}$$
$$\left. \cdots \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_C - \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_D - \vec{\mathbf{n}}_A, r_B/r_A\, (or\, s_B - s_A), r_C/r_A, r_D/r_A\ \right]$$

This feature vector λ_i has 16 elements (as scalars or vectors). For an n-node topology, the number of elements in this vector will be n+3·(n-1)+(n-1)·(n-2)/2. This four-node relation may also be shown on a topological chart as shown in Figure 8. Node A is called the base node because the link relations are calculated relative to this node.
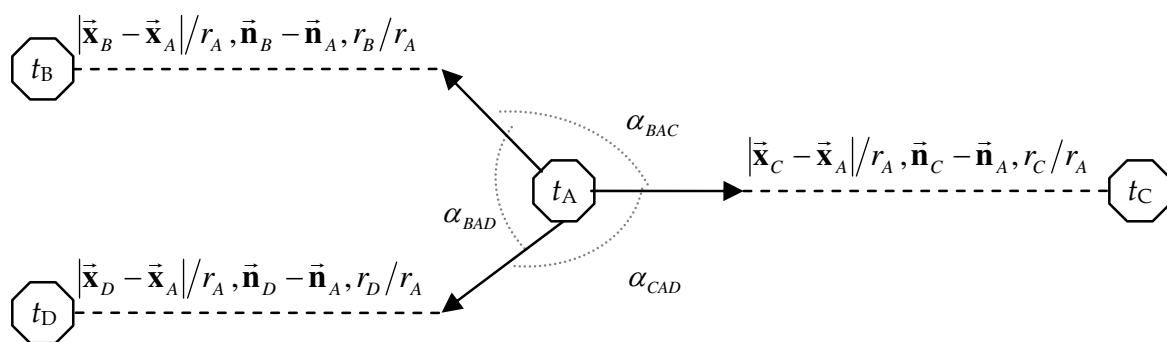
Fig. 8. Four-node, scale and orientation invariant feature vector is shown in a topological chart.

This vector is orientation and scale invariant since all relations are defined relative to node A. However for some applications, orientation and/or scale invariance may not be desired. For example, if the metric size of the object to be recognized is known, then scale invariance is unnecessary. Similarly if the orientation of the object relative to the sensor device is fixed, then orientation invariance capability of a recognition system will be redundant. For this reason orientation and/or scale dependent versions of this vector may be defined.

$$\lambda_i = \left[\ t_A, t_B, t_C, t_D, |\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A|, |\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A|, |\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A|, \alpha_{BAC}, \alpha_{BAD}, \alpha_{DAC} \cdots \right. \tag{8}$$
$$\left. \cdots \vec{\mathbf{n}}_B - \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_C - \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_D - \vec{\mathbf{n}}_A, r_A, r_B, r_C, r_D, \cdots\ \right]$$

Equation (8) is an example of an orientation invariant but not scale invariant feature vector representing a four-node topology, since the link lengths and feature sizes are not normalized according to the base node A. On the other hand the feature vector in Equation (9) is scale invariant but not orientation invariant because the feature normal vectors and link vectors are not normalized according to node A.

Using one of (7), (8) or (9), a feature vector λ_i among n-nodes may be obtained. However excessive number of nodes might be (actually are usually) extracted from a range image.

$$\lambda_i = \Big[\ t_A, t_B, t_C, t_D, \frac{|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A|}{r_A}, \frac{|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A|}{r_A}, \frac{|\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A|}{r_A}, \ldots$$
$$\ldots \frac{(\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A)}{|\vec{\mathbf{x}}_B - \vec{\mathbf{x}}_A|}, \frac{(\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A)}{|\vec{\mathbf{x}}_C - \vec{\mathbf{x}}_A|}, \frac{(\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A)}{|\vec{\mathbf{x}}_D - \vec{\mathbf{x}}_A|}, \ \vec{\mathbf{n}}_A, \vec{\mathbf{n}}_B, \vec{\mathbf{n}}_C, \vec{\mathbf{n}}_D, . r_B/r_A, r_C/r_A, r_D/r_A \ \Big] \tag{9}$$

Therefore, many n-node combinations may be obtained from a range image and these different n-node feature vectors represents different parts of the 3D surface. The algorithm to obtain a complete set of feature vectors which represents the whole 3D surface is given below:

- Extract all feature elements from the surface:

Node1: $(t_1), (v_1), (x_1), (n_1), (s_1), (r_1)$
Node2: $(t_2), (v_2), (x_2), (n_2), (s_2), (r_2)$
Node3: $(t_3), (v_3), (x_3), (n_3), (s_3), (r_3)$
Node4:   …

- Select M nodes with M largest radii (r_i) OR volume (v_i).
- Order these selected M largest nodes according to the criteria below:
- Order them according to their type.
- If there is more than one occurrence of a type, order them according to their radius (r_i) OR volume (v_i).
- For the ordered M nodes do the following:
- Select node group size n: (2<n<M)
- Obtain all possible (k number of) combinations of node groups of n among the M ordered nodes: k=C(M,n)
- Among each n-node combination, keep the order of the nodes according to the previous step.
- For all k combinations,
- Select the first feature as the base node.
- Extract the feature vectors λ_i using one of (7), (8) or (9).
- Then stack these row vectors in a feature matrix $\Lambda = [\lambda^T \ \lambda_2^T \ \lambda_3^T \ \ldots \ \lambda_k^T]T$.

This feature matrix Λ will be a scale and/or orientation representation depending on the equation used to calculate the feature vectors.

4. Experimental work

In this section, some experiments using proposed method is demonstrated. The transform invariant n-node topology is used to register and recognize range images.

4.1 3D object registration

Surface registration is an intermediate but crucial step within the computer vision systems workflow. The goal of registration is to find the Euclidian motion between a set of range images of a given object taken from different positions in order to represent them all with respect to a reference frame. Registration in general can be divided into two: coarse registration and fine registration (Salvi et al, 2006). In coarse registration, the main goal is to compute an initial estimation of the grid motion between two clouds of 3D points using correspondences between both surfaces. In fine registration, the goal is to obtain the most accurate solution as possible. Needless to say that the latter method usually uses the output

of the former one as an initial estimate so as to represent all range image points with respect to a common reference system. Then it refines the transformation matrix by minimizing the distance between the temporal correspondences, known as closest points. For a wide literature survey on registration of range images reader may refer to (Salvi et al, 2006).

In this study, we perform coarse registration using the proposed scale invariant features. The homogenous transformation, which includes 3D rotation, translation and scaling between two range images, is estimated. However different from previous approaches not single features are matched as correspondences, instead the triples that were used to recognize object categories are used.

4.1.1 Triple correspondences

In the previous sections, n-node topologies of scale invariant features were constructed. Using these n-node topologies, transform invariant object recognition was performed and analyzed. The topology set included n=3 number of nodes, namely triplets. Only a general consensus between all matched triplets would give the true transformation and prove that the matched triplets are actually true features that represent the objects. For this purpose using the extracted triplets in the previous subsection, a coarse registration is carried out using "random sample consensus" (RANSAC) method.

4.1.2 RANSAC using triplets

RANSAC is an abbreviation for "RANdom SAmple Consensus". It is an iterative method to estimate parameters of a mathematical model from a set of observed data which contains outliers. It is a non-deterministic algorithm in the sense that it produces a reasonable result only with a certain probability, with this probability increasing as more iterations are allowed. The algorithm was first published by (Fischler & Bolles, 1981).

A basic assumption is that the data consists of "inliers", i.e., data whose distribution can be explained by some set of model parameters, and "outliers" which are data that do not fit the model. In addition to this, the data can be subject to noise. The outliers can come, e.g., from extreme values of the noise or from erroneous measurements or incorrect hypotheses about the interpretation of data. RANSAC also assumes that, given a (usually small) set of inliers, there exists a procedure which can estimate the parameters of a model that optimally explains or fits this data.

The RANSAC algorithm is often used in computer vision, e.g., to simultaneously solve the correspondence problem and estimate the fundamental matrix related to a pair of stereo cameras.

For our case, very similar to solving the correspondence problem in stereo images, the method is used to estimate the homogeneous transformation between to range images. Instead of using candidate point matches, candidate triplet matches are used. As explained in the previous section, the triplets are matched by some well-defined similarity measures. In order to eliminate the false matches, and obtained the transformation only between the true triplet matches, RANSAC is run.

RANSAC also requires a similarity measure to test for the homogenous transformation between two triplet matches. However for RANSAC, different from the similarity measures used to find candidate matches, only the spatial information is used. In other words, at any iteration of RANSAC, if there is candidate transformation, the absolute Euclidian difference between the first triplet and the transformed second triplet is used. The output of the

RANSAC is a homogenous transformation vector, which defines the transformation between any two corresponding points on the registered range images, such that:

$$\overline{\mathbf{P}} = \mathbf{A} \cdot \mathbf{P} = \begin{bmatrix} \overline{P}_x \\ \overline{P}_x \\ \overline{P}_x \\ \overline{s}_p \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{3x3} & \mathbf{T}_{3x1} \\ \hline \mathbf{0}_{1x3} & S \end{bmatrix} \cdot \begin{bmatrix} P_x \\ P_y \\ P_z \\ s_p \end{bmatrix} = \begin{bmatrix} \mathbf{R} \cdot \mathbf{P} + s\mathbf{T} \\ \\ S \cdot s_p \end{bmatrix} \tag{10}$$

4.1.3 Results

In this subsection some experiments on registration using the proposed feature are presented. The first experiment is the simplest case, where there is only in-plane rotation between two artificial surfaces. There's 135° in-plane rotation between the surfaces. The result of RANSAC is given below together with the ideal transformation matrix. The results prove a quite successful coarse registration. In Figure 9, the matched features can also be seen. Colors designate different feature types.

RANSAC result is:
$$\begin{bmatrix} -0.6627 & 0.6479 & -0.1149 & 43.1966 \\ -0.6896 & -0.6807 & 0.0394 & 156.433 \\ 0.0342 & 0.1096 & 0.9344 & -5.0095 \\ \hline 0 & 0 & 0 & 1.0772 \end{bmatrix},$$

and the ideal result is:
$$\begin{bmatrix} -0.7071 & 0.7071 & 0 & 45 \\ -0.7071 & -0.7071 & 0 & 155 \\ 0 & 0 & 1 & -5 \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}$$
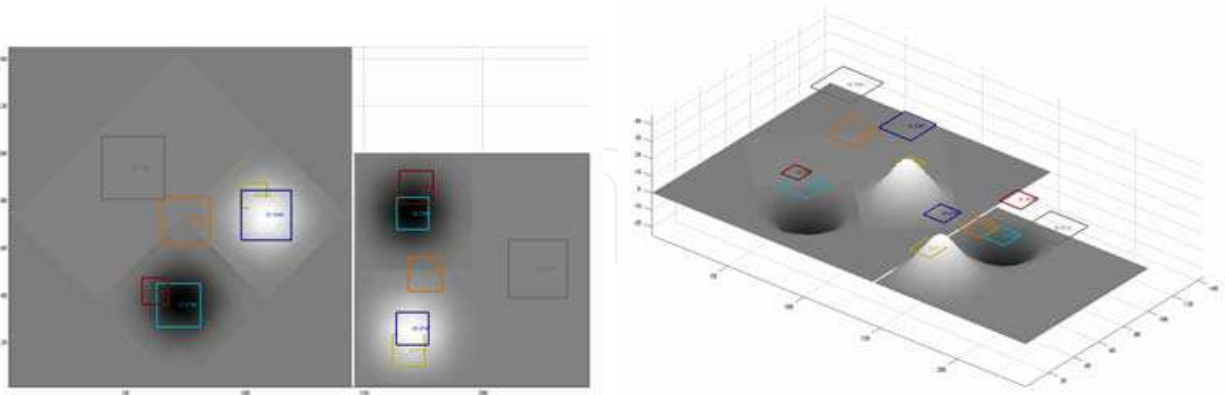
Fig. 9. Matched features from the registered artificial range images.

The same experiment was also performed on range images from the Stuttgart database. Since these images are not captured with controlled rotation, we have chosen two images with rotation on a single axis. The result is given below together with the ideal transformation matrix. In Figure 10, the matched features can also be seen. Colors designate different feature types.

RANSAC result is:
$$\begin{bmatrix} 0.8616 & -0.0265 & 0.2065 & 0.06336 \\ -0.0032 & 0.9691 & -0.0105 & -0.4065 \\ -0.1914 & 0.0068 & 0.8159 & 8.8006 \\ \hline 0 & 0 & 0 & 1.0042 \end{bmatrix},$$

and the ideal result is ($\alpha \approx 30^o$):
$$\begin{bmatrix} \cos(\sim \alpha) & 0 & \sin(\sim \alpha) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\sim \alpha) & 0 & \cos(\sim \alpha) & T_z \\ \hline 0 & 0 & 0 & 1 \end{bmatrix}.$$
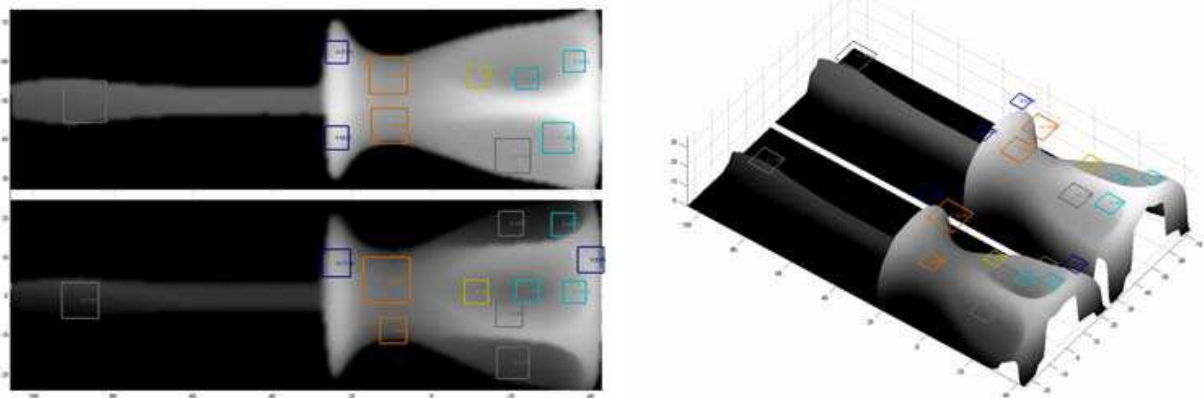
Fig. 10. Matched features from the registered range images from Stuttgart Database

Since the extracted features are scale invariant (and so as the triplets), it is possible to register scaled versions of range images and calculated the scaling ratio between to objects. For this purpose using the %25 scaled versions of the Stuttgart database objects (vs. the originals), registration is performed. The result of RANSAC is given below together with the ideal transformation matrix. The results on scaled range images demonstrate successful scale invariant coarse registration. In Figure 11, the matched features can also be seen. Colors, as usual, designate different feature types.

RANSAC result is:
$$\begin{bmatrix} 0.9442 & -0.0143 & -0.0623 & 0.6810 \\ 0.0349 & 0.9899 & -0.0204 & 0.4130 \\ 0.0519 & -0.0067 & 0.9449 & 0.6304 \\ \hline 0 & 0 & 0 & 0.4955 \end{bmatrix},$$

and the ideal result is:
$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & \mathbf{0.5} \end{bmatrix}.$$
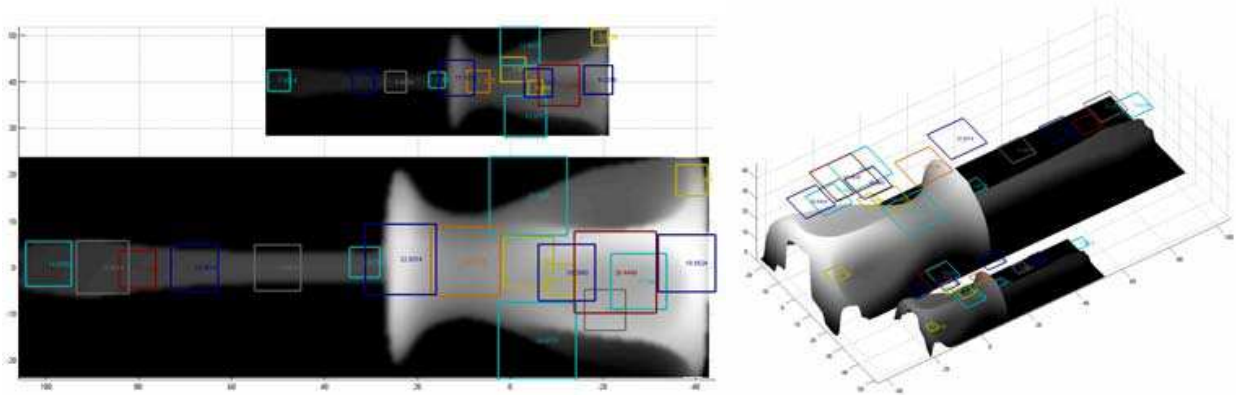
Fig. 11. Matched features from the registered range images from Stuttgart Database. The surfaces are scaled versions of the same range image.

4.1 3D object recognition

In order to prove the importance of scale-space search for scale-invariant object recognition, we have created 0.5 times scaled versions (i.e. 0.25 times scaled in area) of eight objects from Stuttgart range image database. We have used original 258 images of each object as training images and 258 scaled images for testing. 3D geometric hashing is used as the recognition method. A feature vector is constructed using the ten largest feature elements as in Figure 5. By obtaining all possible triplets among these 10 features (totally C(10,3) = 120 triplets for each range image) the feature vector is built. In order to make this feature vector scale and orientation invariant; equation (9) is used.

In this study, these feature vectors are used in a geometric hashing method for object recognition purpose. The main reason behind using geometric hashing is that it allows partial matching of small topologies in a general topology so that the recognition process becomes robust to noise, rotation and even occlusion. In our hashing method, indexing is done by the types of the triplets and each entry includes the feature vector of the triplet besides the code of the pose and the object. At the preprocessing stage of hashing, for each range image in the training set of category of object (for example range images of bunny taken from various angles), the feature vectors are calculated. This database construction stage can be computed offline. For various range image training sets belonging to different objects, this operation is completed. Consequently at the recognition stage, features of the test model (i.e. the model to be recognized) are extracted and then related hash table indexes are obtained. By comparing the hash table entries using a similarity measure between the feature vectors, matching features are found. Corresponding to the indexes in the training sets, matched model's vote is incremented by one for a particular training set. Finally, the database model which receives the greatest number of votes is taken as the match of the test object.

Using the scale-space search, %96.56 average recognition success is obtained. When the scale space search is omitted, the success falls dramatically to %59.54. It is clearly seen that scale-space search is a must, if scaled versions of the objects are to be recognized. In Figure 6, the results are depicted as confusion matrices. The left-most column indicates the scaled test images, and the upper-most row indicates the original training images. Each value in the diagonal cells indicates the percentage success of recognizing a scaled version of an object using the features obtained from the originally sized range image (Figure 12).

a)

b)

Fig. 12. a) Result for the test, in which the scale-space search is omitted, is shown. The average success dramatically falls to %59.54 when the scale-space search is omitted. b) Result for the test with scale-space search is shown. The average recognition success is %96.56. It is clearly seen that scale-space search is a must, if scaled versions of objects are to be recognized.

5. Conclusions

Understanding the notion of scale invariance for 3D surfaces is crucial to extract scale invariant features with their scale information. For this reason, a scale (size and resolution) and orientation invariant 3D feature detection method is proposed in this study. The method extracts the size of the feature independent of the sampling, scale or the orientation of the objects. For this purpose a scale-space search of HK curvatures is developed.

It is clearly seen in various experiments that scale-space search is a must if utter scale and orientation invariance is intended. With only the given resolution of the surface, the feature extraction method is always dependent on the sampling rate or the thresholds used. For a feature detection system to be independent of any property of the source signal, the features must be sought within a scale-space.

In addition, when scale-space is used, the size of a feature, which defines the effective extent of the detected interest region, can be obtained. By this way, scale and orientation invariant points and their effective sizes can be provided for local region based descriptors such as spin image or splash representation.

The use of scale-space also brings robustness to noise. Since scale-space levels are created using Gaussian filters, within the higher levels of the scale-space, any level of noise disappears and large features can be extracted very clearly although the surfaces are noisy.

When the scale invariant feature extraction is used for 3D object recognition, the resulting recognition performance is much better than when feature extraction is done only for the given scale.

Results show that, the extracted features can be used for coarse registration of range images. The registration inherits the scale, sampling and orientation invariant nature of the features, and rotated and scaled versions of objects can be successfully matched. The scale ratio between the matched objects can be obtained.

In addition recognition in a scale varying database can be performed if scale invariant features are used. Objects of different size from a huge database, are efficiently recognized using sparse feature vectors.

As a feature study, the features may be applied to time-of-flight (TOF) camera images and robust implementations may be developed for real-time robotics applications.

6. References

Burt, P. & Adelson, E. (1983) The Laplacian Pyramid as a Compact Image Code. *IEEE Transactions on Communications*, Vol. 31, No. 4, Apr 1983, pp. 532-540.

Chua, C. S. & Jarvis, R. (1997) Point Signatures: a new representation for 3D object recognition. *Int. J. of Computer Vision*, Vol.25 , No.1, October 1997, pages: 63 – 85

Fischer, M.A. & Bolles, R. (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, Vol.24, No.6, June 1981, pp.381-395.

Frome, A.; Huber D.; Kolluri, R.; Bülow, T.; Malik, T. (2004) Recognizing objects in range data using regional point descriptors. *Proceeedings of European Conf. on Computer Vision ECCV 2004*, vol 3, pp. 224-237

Hetzel, G.; Leibe, B.; Levi, P. & Schiele, B.(2001) 3D Object Recognition from Range Images using Local Feature Histograms. *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR'01)*, vol.2, pp. 394-399

Horn, B.K.P. (1984) Extended Gaussian Images. *Proceedings of the IEEE*, Vol.72, No.2, December 1984, pp. 1671-1686,

Johnson, E. & Hebert, M. (1999) Using Spin Images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 5: 433-449.

Koenderink J. & Doorn, A. J. (1992). Surface shape and curvature scale, *Image Vision and Computing*, Vol. 10, No. 8, October 1992, pp. 557–565.

Li, X. J. & Guskov, I. (2007) 3D Object recognition from range images using pyramid matching", *Proceedings of ICCV'07 Workshop on 3D Representation for Recognition (3dRR-07)*, pp. 1-6

Lindeberg, T. (1994). *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers,Norwell, MA, USA

Lo , T.W.R. & Siebert, J.P. (2009). Local feature extraction and matching on range images 2.5D SIFT. *J. Computer Vision and Image Understanding*, Vol. 113, No. 12, December 2009, pages 1235-1250.

Lowe, D. G. (2004). Distinctive Image Features from Scale-invariant Keypoints. *Int. J. Computer Vision*, vol. 60, no. 2, pp. 91-110

Pauly, M.; Keiser,R. & Gross M. (2003). Multi-scale Feature Extraction on Point-Sampled Surfaces. *Proceedings of EUROGRAPHICS'03*, Vol. 22, No. 3.

Salvi, J; Matabosch, C.; Fofi, D & Forest, J. (2006) A review of recent range image registration methods with accuracy evaluation. *Image Vision and Computing*, Vol. 25, No.5, May 2007, pp. 578-596.

Sun, Y. & Abidi, M. A. (2001) Surface Matching by 3D Points Fingerprint. *Proceeding of Eighth International Conference on Computer Vision (ICCV'01)*, Vol. 2, pp.263.
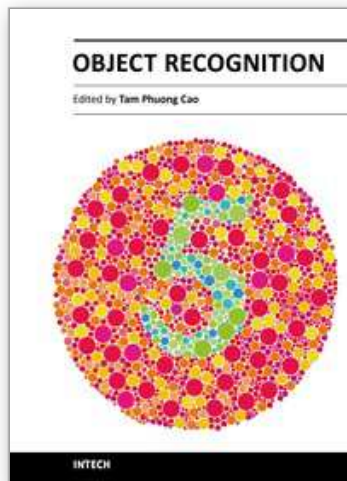
Object Recognition

Edited by Dr. Tam Phuong Cao

Vision-based object recognition tasks are very familiar in our everyday activities, such as driving our car in the correct lane. We do these tasks effortlessly in real-time. In the last decades, with the advancement of computer technology, researchers and application developers are trying to mimic the human's capability of visually recognising. Such capability will allow machine to free human from boring or dangerous jobs.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Erdem Akagündüz and Ilkay Ulusoy (2011). 3D Object Registration and Recognition using Range Images, Object Recognition, Dr. Tam Phuong Cao (Ed.), ISBN: 978-953-307-222-7, InTech, Available from: http://www.intechopen.com/books/object-recognition/3d-object-registration-and-recognition-using-range-images

INTECH
open science | open minds