Published in IET Computer Vision
Received on 26th June 2008
Revised on 29th October 2008
doi: 10.1049/iet-cvi:20080037

Special Issue - selected papers from DICTA 2007

ISSN 1751-9632

Local 3D structure recognition in range images

A. Flint A. Dick A. van den Hengel

School of Computer Science, University of Adelaide, Australia
E-mail: Anton.vandenhengel@adelaide.edu.au

Abstract: A feature detector and a feature descriptor are presented, which are applicable to 3D range data. The feature detector is used to identify locations in the range data at which the feature descriptor is applied. The feature descriptor, or feature transform, calculates a signature for each identified location on the basis of local shape information. The approach used in both the feature detector and the descriptor is motivated by the success of the scale invariant feature transform and speeded up robust features approaches in the 2D case. Using synthetic data, the authors evaluate the repeatability of the detector and robustness of the descriptor to global transformations and image noise. The complete system is then applied to the problem of automatic detection of repeated structure in real range images.

1 Introduction

The identification of image points, which correspond to the projections of the same scene point, is one of the fundamental problems in computer vision. This has been labelled the correspondence problem in stereo, but is also a critical component of object tracking, object recognition and image-based classification among a host of other problems.

Significant gains have been made recently in the use of local feature descriptors that, given a keypoint in an image, calculate a signature describing an image region centred at that point. Using a collection of such local descriptors to describe an object visible in an image set provides robustness to partial occlusion, and depending on the design of the descriptor, can also provide robustness to changes in illumination and viewpoint.

For example, the scale invariant feature transform (SIFT) [1] calculates a signature that characterises the image in the neighbourhood of a keypoint in a way that is robust to changes in global illumination, object rotation and scale. The signature is based on histograms of image grey-level gradients, which are normalised with respect to a locally dominant orientation and scale.

The idea of this work is to build a local 3D feature descriptor for range images with comparable robustness to missing data and changes in viewpoint. Although this has many applications, it was initially motivated by work in image-based modelling. In this domain, it is common to have a 3D data set – whether captured from a range finder, or the output of structure and motion estimation, or modelled manually – that is incomplete. Often it is the case that this 3D data will contain repeated structure, some instances of which are captured or modelled with higher fidelity than others. If such repetition can be recognised automatically, information from instances that are well modelled can be propagated to those that are poorly modelled, resulting in a more accurate overall model.

Previous 3D feature descriptors include shape contexts [2] and spin images [3]. Both of these describe local shape by partitioning the volume around a keypoint into spatial bins and then counting the number of 3D points in each bin. Neither transform is invariant to scale, and, although they exhibit some robustness to rotation (histograms are calculated relative to estimated surface normal), they are sensitive to small changes in the computed surface normal.

The success of local feature descriptors depends on the choice of keypoint locations. In 2D images, good keypoints are those that can be well localised, such as corner points where the intensity gradient is high in all directions. Several techniques, such as the Harris corner detector [4] and more recently SURF [5], have been developed to identify these points. In 3D images, we also require keypoints that can be well localised, but in 3D this requires

that the spatial gradient of the surface about a keypoint be high in all three directions.

This paper proposes a new 3D feature detector and descriptor that extend the successful SIFT and SURF algorithms to keypoint selection, identification and matching in range data. It brings many of the advantages of these 2D algorithms to bear on the problem of 3D structure recognition. We show how this 3D keypoint detector and descriptor can be combined to detect repeated 3D structure in range data of building facades.

The remainder of this paper is organised as follows. Section 2 describes our interest point detector, whereas Section 3 describes our 3D descriptor. In Sections 4 and 5, we present empirical results on synthetic and real data, and Section 6 concludes the paper.

2 Feature detection in 3D

Our detector accepts a range image as input, and outputs a set of 3D points with corresponding characteristic scales. We adopt Lindeberg's principle for scale selection, which suggests looking for the scale space maxima of normalised derivatives [6]. However, the notion of a derivative is not well defined in a range image, and so we must first apply a suitable transformation. We use a density sampling operation that produces a regular density map. The basic operation of our detector can be summarised as follows:

1. Sample the density function at regularly spaced locations to create a density map

2. Construct a scale space for the density map

3. Find local scale space maxima within the density map.

In the following sections, we describe each step in more detail.

2.1 Generating the density map

Let the range data be given as a set of points

$$\mathcal{X} = \{x_i \in \mathbb{R}^3\}$$

Let $n(B)$ be the number of data points in the region $B \subseteq \mathbb{R}^3$. We define a set of equal-sized boxes $\mathcal{B} = \{B_{ijk}\}_{(i,j,k)\in I\subset\mathbb{Z}^3}$ distributed regularly in each spatial dimension, where I is the domain of the input range image.

We approximate the density in each box B_{ijk} by the number of range points in it, normalised by the maximum number of points in any box

$$D(i,j,k) = \frac{n(B_{ijk})}{\underset{(i,j,k)\in I}{\operatorname{argmax}}\{n(B_{ijk})\}}$$

where D is the normalised density map.

2.2 Scale space construction

We have no a priori knowledge of the scale at which interesting features might occur, and so it is natural that we consider several different scales. To do this, we construct a scale space [6], following the standard approach in the 2D setting.

Given a continuous signal $f:\mathbb{R}^n \to \mathbb{R}$ (in our case, f is the normalised density map D), the scale-space representation $L:\mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ of f is defined as the family of functions resulting from convolution with Gaussian kernels of increasing scale

$$L(\cdot\,;\sigma) = f(\cdot) \otimes g(\cdot\,;\sigma)$$

In general, it will not be possible to obtain an analytic form for L, and so we instead sample L at a set of discrete scales $\Sigma = \{\sigma_1, \ldots, \sigma_N\}$. Following [6], we sample the scale such that each σ_i is separated by a constant factor k

$$\begin{aligned}\sigma_1 &= 1\\ \sigma_{i+1} &= k\sigma_i \quad i \geq 2\end{aligned}$$

For simplicity, we choose k such that there are an integral number of scales between σ and 2σ; each such doubling of σ is an octave, and the number of scales sampled per octave will be denoted n_L. The scale separation factor k is related to n_L by $k = 2^{1/n_L}$.

2.3 Identifying interest points

The detector must be able to extract the same (or similar) interest points under a range of transformations, such as changes of viewpoint or scale. This requires that interest points be well localised in all three spatial directions.

We use local maxima of a function of spatial derivatives as the criteria for interest point selection. Our approach is partially motivated by the extensive 2D detector literature indicating that the locations of maxima of spatial derivatives are robust to a range of transformations [4, 6, 7].

Our detector operates as follows. First, we generate a scale space for the input density map using the process described in the previous section. Next, we use finite differences to compute the Hessian

$$\mathcal{H}(\mathbf{x};\sigma) = \begin{bmatrix} L_{xx}(\mathbf{x};\sigma) & L_{xy}(\mathbf{x};\sigma) & L_{xz}(\mathbf{x};\sigma) \\ L_{yx}(\mathbf{x};\sigma) & L_{yy}(\mathbf{x};\sigma) & L_{yz}(\mathbf{x};\sigma) \\ L_{zx}(\mathbf{x};\sigma) & L_{zy}(\mathbf{x};\sigma) & L_{zz}(\mathbf{x};\sigma) \end{bmatrix}$$

of the scale space at each sampled location. We apply a user-defined threshold T_D to the determinant of the Hessian to eliminate weak responses, and finally identify all remaining local extrema by comparing each voxel with its scale space neighbours. Only voxels that are greater than all neighbours will be included in the final set of interest points.

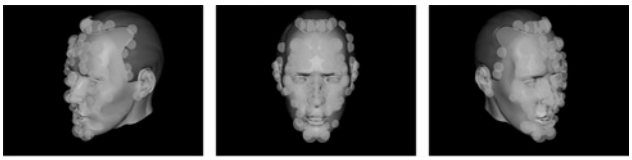

Figure 1 *Interest points detected on a synthetic face model*
Notice how interest points are concentrated at distinctive features (nose, eyes, ears and so on) and not on smooth surfaces (such as the hair, cheek or neck)
Notice also the strong similarity between the features detected on the left and right sides of the face

When a feature is detected, we can automatically assign a characteristic scale, which is the scale space level σ at which the feature was detected. This is useful for recognising features appearing at different scales, as shown in the following section. Fig. 1 shows some detected feature locations on a 3D face model.

3 Feature description in 3D

3.1 Using surface normals

As previously mentioned, the SIFT descriptor [1] and its variants have been shown to consistently outperform other descriptors for the purpose of object recognition from 2D images. All these descriptors use image gradient orientations as the basic descriptive element, and a number of authors have conjectured that this is the reason for their success [5, 8, 9]. The image gradient orientation at a pixel is the direction in which the image intensity changes fastest. The obvious 3D generalisation is the principal direction or surface normal [10].

We therefore use surface normals as the basic geometric element by which to characterise local surface shape. In addition to their intuitive appeal, surface normals have a number of desirable properties for local surface description, including robustness to sampling density and some types of noise [11].

As surface normals are not typically provided with range data, we use local plane fitting to calculate them from nearby 3D points. To determine the surface normal vector $\boldsymbol{n}$ at some point $\boldsymbol{x}$ in the range image, we fit a least squares plane to the points within distance r of $\boldsymbol{x}$ and then take the normal to this plane. We determine r dynamically: when estimating the surface normal at a range point $\boldsymbol{x}_i$ we set r to the distance to the nth closest neighbour of $\boldsymbol{x}_i$, where n is a user-defined parameter. In practice, we found $n = 10$ a suitable choice.

3.2 Generating the feature descriptor vector

In this section, we detail how a feature description is obtained for an interest point $\boldsymbol{y}$ detected at the characteristic scale σ on the basis of nearby surface normals. The feature vectors are generated independently for each detected point, and so we describe this process in terms of a single point.

3.2.1 Algorithm overview: We first define the support region S containing all points within a distance R of the interest point $\boldsymbol{y}$

$$S = \{\boldsymbol{x}_i \in \mathcal{X} : \|\boldsymbol{y} - \boldsymbol{x}_i\| \leq R\}$$

Based on [6], we choose R proportional to the characteristic scale σ of the feature. Experimentally, we have found that setting $R = 2\sigma$ gives good performance. Note that R is much larger than the radius r used in plane fitting, and, hence, although r encompasses an area small enough to be approximately planar, R is intended to encompass whatever scene feature was detected at $\boldsymbol{y}$ (which will certainly not be planar since the Hessian determinant on a plane is zero). Points in S are called the support points for $\boldsymbol{y}$.

Next, we compute the surface normals $\boldsymbol{n}_i$ for each support point $\boldsymbol{x}_i$ as described earlier. For invariance to global object or viewpoint transformations, we compute the surface normal $\boldsymbol{n}_y$ at the interest point $\boldsymbol{y}$ and then measure all other surface normals by the angle θ_i that they form with $\boldsymbol{n}_y$

$$\theta_i = \cos^{-1}(\boldsymbol{n}_i \cdot \boldsymbol{n}_y)$$

We call θ_i the deviation angle for the support point $\boldsymbol{x}_i$.

Next, we form a histogram $\boldsymbol{v}$ over deviation angles

$$v(i) = \sum_{\boldsymbol{v}_{i \in bin(i)}} k_i$$

where k_i is the contribution of the support point $\boldsymbol{x}_i$. k_i is determined by two factors

- Contributions are normalised for density such that the contribution from each part of the support region is based on its surface area rather than the density of point samples within it. We find the distance d from $\boldsymbol{x}_i$ to the n^{th} closest point (n is a user-specified parameter; we use $n = 10$) and then approximate the density by $\rho_i = 1/d^2$.

- Contributions are convolved with a Gaussian centred at the interest point with width $\sigma = R/2$. This ensures that the feature vector changes smoothly as the interest point location changes, which is important since the detector may not localise all interest points exactly.

Hence, the contribution of each point is given by

$$k_i = \frac{g(\boldsymbol{x}_i - \boldsymbol{y}; R/2)}{\rho_i}$$

The histogram bins are spaced linearly between 0° and 90°, with deviation angles $\theta_i > 90°$ mapped to $180 - \theta_i$. The final feature vector $\boldsymbol{v}$ is formed simply by taking the values

doi: 10.1049/iet-cvi:20080037

in each histogram bin. For invariance to global sampling density, the feature vector is normalised such that the sum of its entries is 1.

3.2.2 Deviation angles: The deviation angles θ_i do not uniquely describe the surface normals $\boldsymbol{n}_i$ because there are many surface normals that would be assigned the same deviation angle: these are the vectors obtained by sweeping out a cone by rotating $\boldsymbol{n}_i$ around the reference vector $\boldsymbol{n}_y$. This may lead to a false match.

To remove this ambiguity, Frome *et al.* [2] introduce another variable ϕ describing the rotation of each $\boldsymbol{n}_i$ radially about the reference vector. However, this requires an object-centric 'reference direction' against which to measure the rotation of each $\boldsymbol{n}_i$. Obtaining such a reference direction robustly is difficult and computationally expensive. Instead, we use just the deviation angle and leave the radial ambiguity in the descriptor. We found this to be a good trade-off between robustness, discrimination and computational expense.

3.3 Matching descriptor vectors

Thus far, we have attempted to design a descriptor that will generate similar feature vectors for similar surface shapes and different feature vectors for different surface shapes. In this section, we investigate how to determine whether two descriptor vectors represent the same shape or not.

Our feature vector is derived directly from a histogram, and it is normalised such that its sum is 1, and thus, we may regard it as a frequency distribution over deviation angles. We use the earth movers distance (EMD) [12], a metric designed specifically for frequency distributions, which explicitly uses the spatial relations between bins.

To decide whether a feature has a match at all, we use a neighbour ratio test [1] as follows. If $\boldsymbol{v}_B$ is the nearest neighbour of $\boldsymbol{v}_A$, then we find the second nearest neighbour $\boldsymbol{v}_C$ and compute the ratio

$$dr = \frac{d(\boldsymbol{v}_A, \boldsymbol{v}_B)}{d(\boldsymbol{v}_A, \boldsymbol{v}_C)} \tag{1}$$

where $d(\cdot, \cdot)$ is the EMD. If dr is below a user-defined threshold, then the feature vectors are declared a match; otherwise, the feature vectors have no match.

4 Tests on synthetic data

In order to empirically determine settings for the constants used in the detection and description computation, and to obtain some quantitative error analysis, initial tests were run on synthetically generated range data.

To generate this data, a simulator was implemented to convert a polygonal 3D mesh to a 3D point cloud. The simulator accurately mimics the operation of a laser range finder, including spacing point samples according to the obliqueness of the surface to the scanner, as shown in Fig. 2*b*. Our synthetic data set consisted of 81 models composed of between 12 and 20 000 vertices. Some examples of polygonal meshes that were used as the basis for these models are shown in Fig. 2*a*.

All experiments are conducted using four octaves of scale, four levels per octave ($n_L = 4$) and a detection threshold

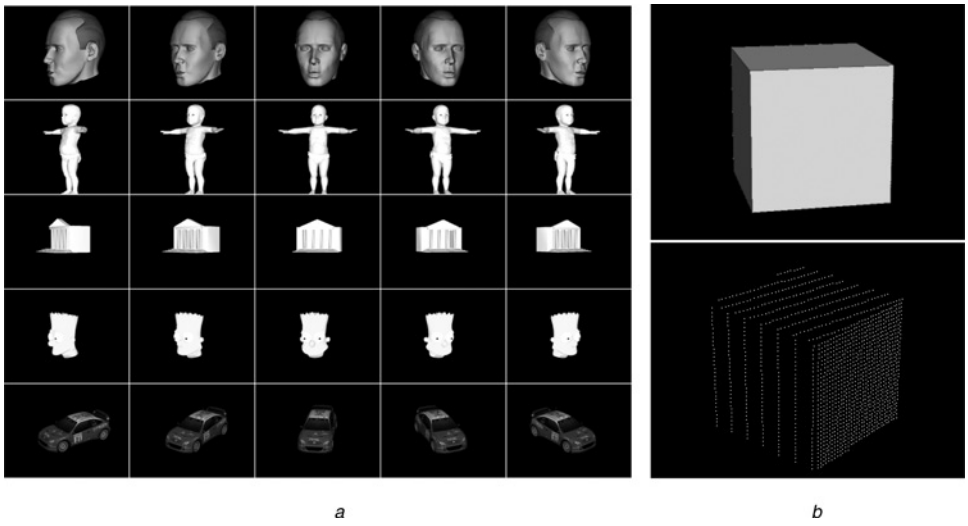

Figure 2 *Sample from the synthetic test set, and generated range data*

a Sample from the test set for viewpoint changes
b Range data generated by simulator for cube model
The sampling density depends on the angle between the scanner and the surface normal
The range image is rotated for clearer visualisation

$T_D = 10^{-5}$. These values were based on preliminary tests, and not changed subsequently.

4.1 Detector repeatability tests

Given a pair of range images A and B separated by some transformation T, we compute the repeatability of the detector as follows. We first detect all interest points in A and B. Next, we transform each interest point from B into A and look for a match among the interest points detected for A. Point $\boldsymbol{y}_A$ is defined to match $\boldsymbol{y}_B$ if the distance between $\boldsymbol{y}_A$ and $\boldsymbol{y}_B$ is less than the smaller of the characteristic scales of $\boldsymbol{y}_A$ and $\boldsymbol{y}_B$

$$\begin{aligned} m(\boldsymbol{y}_A, \boldsymbol{y}_B) &= \text{true}, \quad \text{if } \|\boldsymbol{y}_B - \boldsymbol{y}_A\| < \min(\sigma_A, \sigma_B) \\ &= \text{false}, \quad \text{otherwise} \end{aligned} \tag{2}$$

This is similar to the test used by Mikolajczyk in his evaluation of 2D detectors [7].

If we find a match, then the interest point is a repeat. Then, we perform the same operation in reverse, mapping interest points from A into B. The final repeatability is the number of repeats divided by the total number of interest points. A perfect detector would always attain a repeatability of 1.

4.1.1 Repeatability under viewpoint changes: In this experiment, we measured the repeatability of the detector under viewpoint changes. We simulate range scans at 13 viewpoints along a 120° arc and compute the repeatability between the central viewpoint and each other viewpoint.

Because of our simulation method, a change of viewpoint results in a complete re-sampling of the model. This experiment is difficult because the appearance of the object changes drastically through a 60° rotation. For example, half of the 'face' model is occluded after just a 30° change in viewpoint, and so for this model, our detector could not possibly achieve repeatability above 50%.

Fig. 3 shows the repeatability of our detector as well as the number of corresponding interest points as a function of viewpoint. As there are no true 3D feature detectors with which to compare our results, we instead test against 3D recognition systems that use 2D detectors on projections of range data. In [5], an evaluation of four 2D detectors on 3D scenes comparable to our own test set showed that at a viewpoint separation of 20°, the detectors achieved a repeatability of between 60 and 80%, whereas, this dropped to less than 5% for viewpoint changes of 60°. Our own results show an average repeatability above 70% for a viewpoint change of 20°, but at 60° the average repeatability of our detector remains near 40%, compared with the near zero repeatability attained by 2D detectors. In general, we have found that our detector is comparable to 2D detectors for small viewpoint changes but significantly outperforms them for large viewpoint changes.

Fig. 3*b* shows that our detector was able to find a significant number of corresponding interest points, even at viewpoint changes above 60°. High-level recognition systems often require as few as four matching interest points [1], yet our detector generates on average more than 15 matches in every experiment, and more than 50 for viewpoint changes less than 20°. In comparison, the evaluation in [5] showed that 2D detectors were able to generate just one or zero matching interest points at a viewpoint change of 60°.

4.1.2 Repeatability under scale changes: In this experiment, we measured the repeatability of the detector's under scale changes. We obtained the test set by simulating range scans at different distances from the model.

Fig. 4 shows a consistent repeatability above 50% for all decreases in scale (scale factor <1), with the average repeatability consistently above 75%. This is a significant

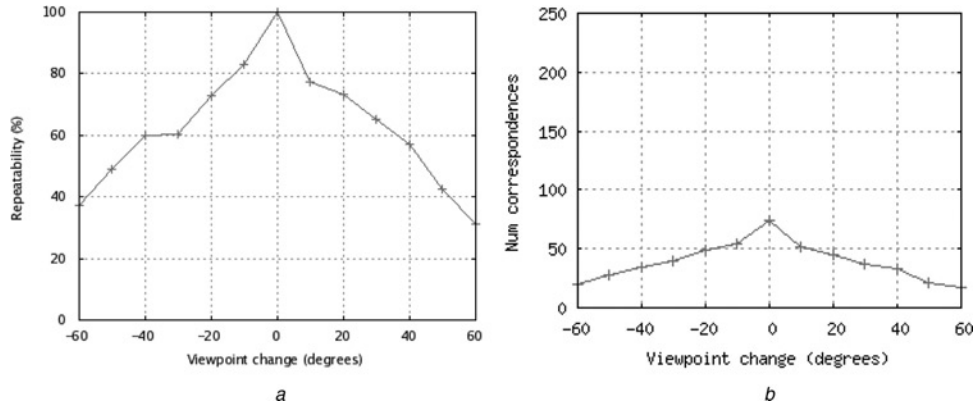

Figure 3 *Detector stability with respect to changes in viewpoint*

a Repeatability against viewpoint change, average for all models
b Total correspondences against viewpoint change, average for all models

doi: 10.1049/iet-cvi:20080037

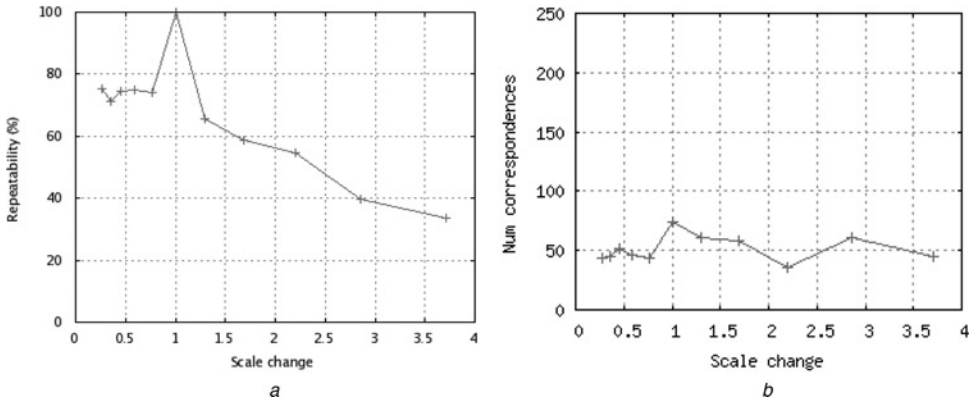

Figure 4 *Detector stability with respect to change of scale*

a Repeatability against scale change, average for all models
b Total correspondences against scale, average for all models

improvement over the results reported in [7] showing that 2D detectors achieved repeatability consistently below 50% (even though that evaluation only uses planar scenes). Our detector has more difficulty with scale increases. This is because of large increase in total interest points caused by the high sampling density, which decreases the overall repeatability even though the number of correct correspondences remains fairly static (Fig. 4*b*).

4.1.3 Repeatability under noise: To test the repeatability of the detector in the presence of range finder measurement noise, we added i.i.d Gaussian noise of varying variance to the synthetic 3D range points.

Fig. 5 shows a sharp decline in both repeatability and the number of correspondences with increasing noise variance. However, both metrics reach a lower threshold around a repeatability of 10%/ eight correspondences and do not decrease further, which indicates that the detector continues to identify the most salient scene features even after the addition of 10% noise.

4.2 Descriptor tests

We now turn to an empirical evaluation of our descriptor using the same simulation system as described previously.

Each experiment in this section measures the descriptor's precision/recall (PR) curve [8, 13] between a range image pair (A, B), that are related by a global transform, with a varying neighbour ratio (1) threshold. To ensure the same interest points appear in both images, each point detected in image A is duplicated in B, and vice versa.

Each interest point $\boldsymbol{y}$ that has a match (i.e. has enough points in its support region to compute a descriptor, and the neighbour ratio test passes) is denoted a true positive if $m(\boldsymbol{y}, \boldsymbol{y}')$ is true, and false otherwise.

Having classified each match, we compute the precision and recall for this test

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{recall} = \frac{\text{TP}}{N}$$

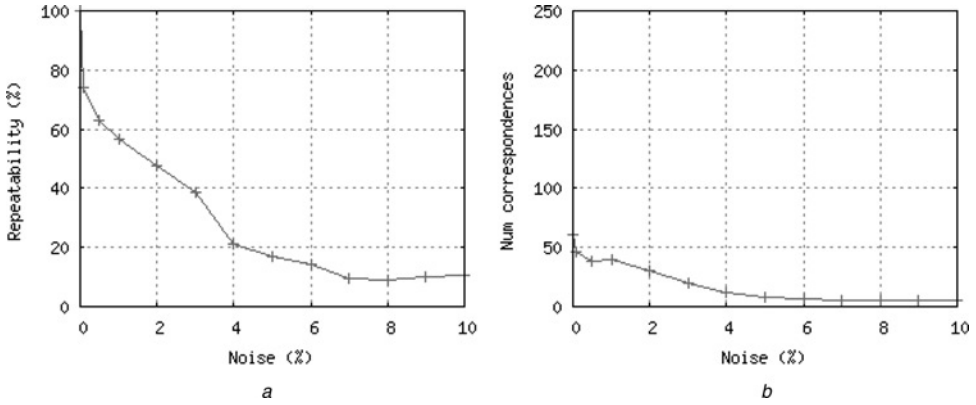

Figure 5 *Results for noise added to the range image*

The noise magnitude was proportional to the radius of a bounding sphere around each model, and hence, 1% noise indicates we used a Gaussian with variance equal to one hundredth the model radius
a Repeatability against range image noise, average for all models
b Total correspondences against range image noise, average for all models

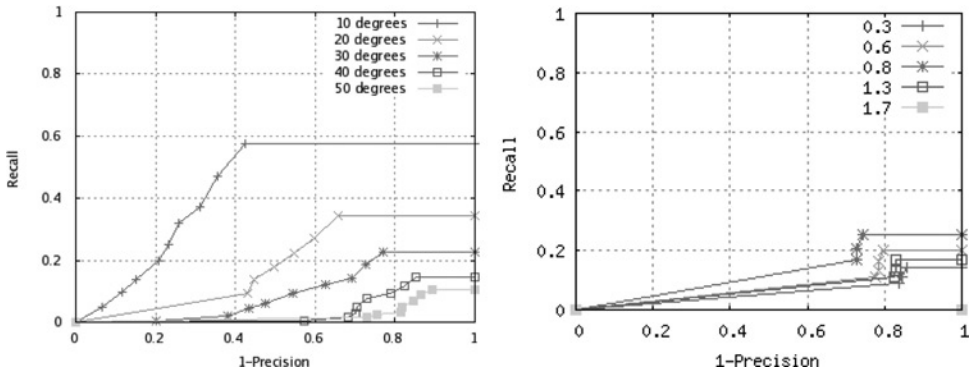

Figure 6 *PR curves for our descriptor under varying viewpoint and scale*

a Viewpoint

b Scale changes. Each data point represents an average over our complete test set for pairs separated by a particular rotation angle

where TP is the number of true positives, FP the number of false positives and N the total number of interest points that have matches. Note that because we only ever allow a feature to match with its best-matching counterpart, our experiments do not attain 100% recall even for 0% precision.

4.2.1 Recognition under viewpoint changes: We tested our descriptor's performance under viewpoint changes using the same data set used for the detector viewpoint evaluation. Fig. 6 shows PR curves for our descriptor for viewpoint changes between 10° and 50°. The red line shows that at a viewpoint change of 10°, the descriptor identifies 60% of all the matches present in the data. As the viewpoint change increases, the descriptor's recognition rate falls. However, at 40°, our descriptor still achieves a recall of 10% with false positives of 80%. This indicates that our descriptor is able to recognise some features even after a large viewpoint change. Furthermore, high-level recognition systems have been shown to work with comparably noisy input data [14].

The comparison with 2D descriptors is not completely fair because 2D and 3D descriptors use different input data. We note, however, that 2D descriptors were able to achieve a detection rate of only 10% at a viewpoint change of just 20° in [15], and that in that study viewpoints beyond 40° were not even tested.

4.2.2 Recognition under scale changes: We tested the performance of our descriptor under scale changes using the same data set as that for the detector scale evaluation. Fig. 6 (right) shows aggregate PR curves over our entire data set. The descriptor struggles with scale changes that cause significant numbers of features to either appear (with increasing scale) or disappear (descreasing scale). as this can significantly alter calculated surface normals. For example, the front of 'face' model becomes almost flat when the scale is very small, in contrast to the many variations (eyes, nose, mouth) present at high scales.

5 Results for range finder data

Although we cannot perform a quantitative error analysis on real range data, this section shows that the performance of the detector is similar to that in the case of synthetic data, and that the descriptor is able to match similar local structure.

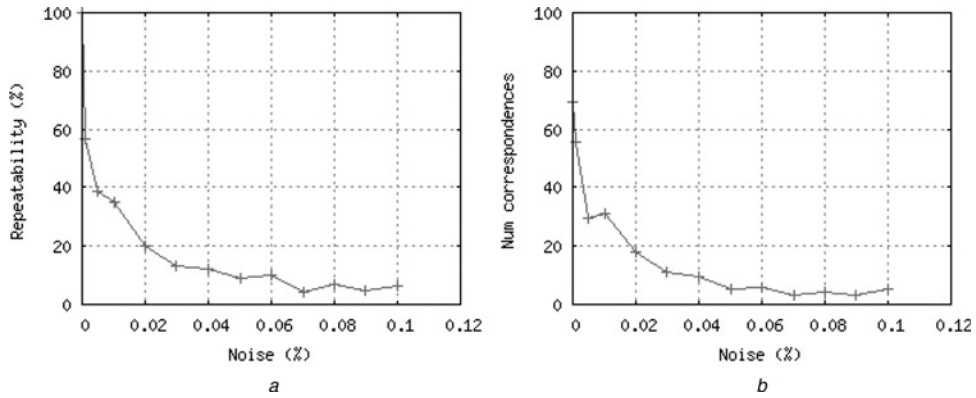

Figure 7 *Effect of range image noise on repeatability and total correspondences, average for all models*

Viewpoint changes from 10 to 50 degrees

a Repeatability against range image noise, average for all models

b Total correspondences against range image noise, average for all models

doi: 10.1049/iet-cvi:20080037

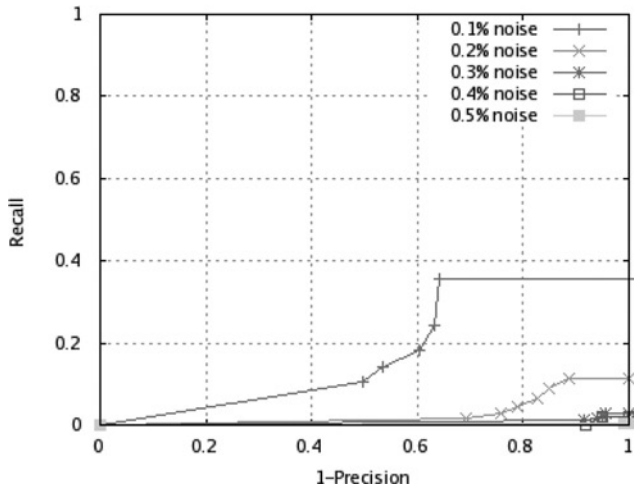

Figure 8 *PR curves for our descriptor under addition of noise below 1%*

Each data point is an average over our test set for a particular neighbour ratio threshold

5.1 Detector repeatability

5.1.1 Repeatability with noise: Our first set of experiments measured the repeatability of the detector under the addition of noise into the range image. These experiments are identical to those performed for synthetic data, except that here we use real range data.

Similar to the synthetic experiment, the addition of noise to the range image (Fig. 7) produces decreasing repeatability with a lower threshold below which the repeatability does not drop. The threshold for range data appears to be ~5%; slightly lower than that for synthetic data, which reflects the greater average complexity of the range data scenes. For the magnitude of noise expected in practice (below 1%), our detector achieves repeatability above 95%.

5.2 Descriptor tests

In this section, we evaluate our descriptor using data captured with a range finder. We use the same data set as that for the detector experiments. We begin with an evaluation of our descriptor under the addition of noise and then proceed to full recognition tasks. Our final experiment uses the detector and descriptor together to automatically identify repeated structure in architectural scenes.

5.2.1 Recognition under noise: We test the performance of our descriptor under the addition of noise by

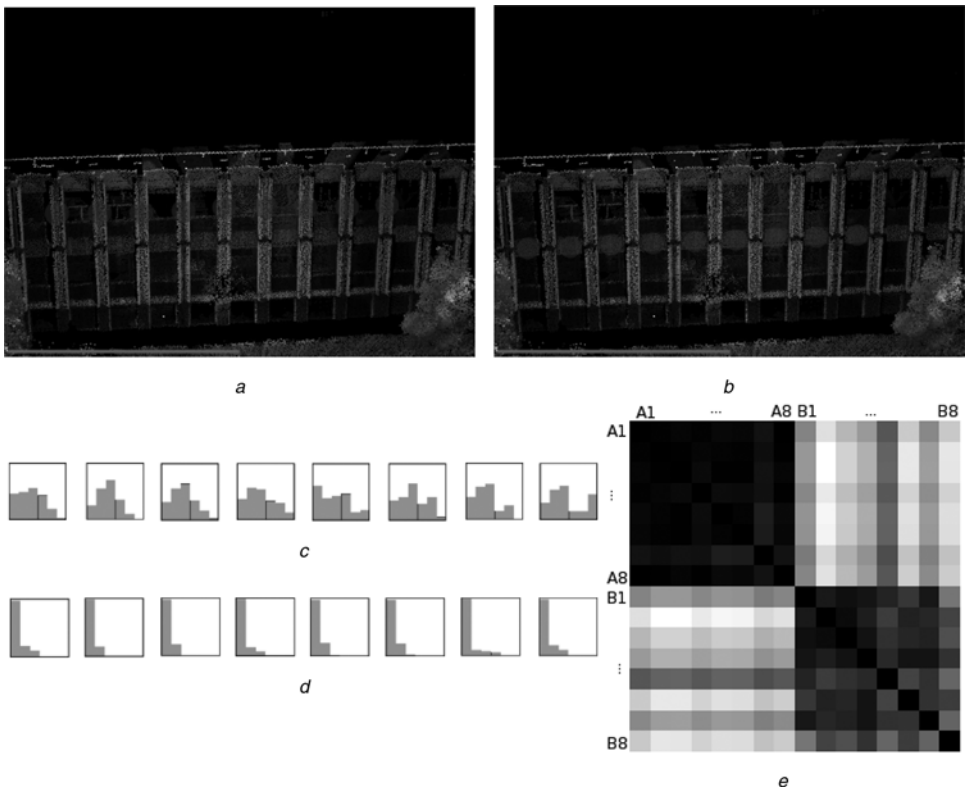

Figure 9 *Results for the two-category classification problem*

a Category A interest points
b Category B interest points
c Category A features
d Category B features
e Pairwise distance matrix (darker is nearer)

adding independent Gaussian noise with several variances to the points comprising each range image and measuring precision and recall with respect to the original data.

Using the same noise variances as in the synthetic experiments destroys all salient features in the real range data. Instead, we modified our experiments to use noise between 0.1% and 0.5%. In Fig. 8, we observe a reasonable recognition rate at noise of 0.1%, which deceases significantly after noise of 0.2%. Note that these noise levels are still much larger than those which real range finders produce [16].

5.2.2 Two-category classification: One potential application for our system is detecting repeated structure within a scene. This problem can be considered as recognition of a single object from different viewpoints and hence, our previous results are applicable to this problem. However, to test our descriptor for this specific problem, we manually selected eight instances of two different repeated features (Fig. 9). Then, we generated feature vectors for each selected region and computed all pairwise distances using the EMD. The pairwise distance matrix is shown in Fig. 9*e*. The figure has a 2×2 checker board appearance which shows that features in the same category are very similar to one another, but features in different categories are dissimilar.

5.2.3 Repeated structure detection: The above experiment shows that our descriptor can recognise repeated structure, when its search region is hand selected. In the next experiment, we use the detector and descriptor together to automatically detect repeated structure.

We manually select a reference region, and then use the detector and descriptor to find the 50 best-matching regions. This was performed by detecting all interest points and generating the corresponding features, and then choosing the 50 closest features according to the EMD. Then, we manually count the number of matches that represent true repeated structure against false positives. Table 1 shows the results for these experiments, and Fig. 10 shows the specific points our system identified.

These results show that our detector/descriptor pair can accurately identify repeated scene features. The first two test scenes, 'library-sparse' and 'windows-sparse', contain facades that are oblique to the range finder, resulting in sparsely sampled surfaces. For these scenes, our descriptor is still able to achieve a recognition accuracy above 80%. The last scene, 'windows-front' contains a more densely sampled facade, leading to a corresponding increase in accuracy (94%).

Table 1 Out of the strongest 50 matches, the percentages of those that were correct

Scene	% correct matches, %
library-sparse	86
windows-sparse	84
windows-front	94

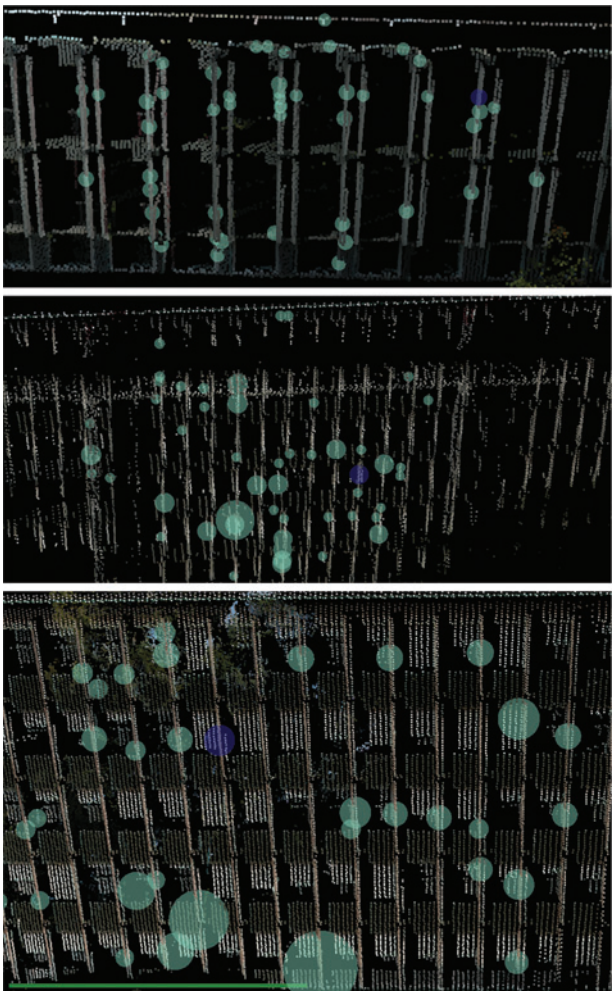

Figure 10 *Repeated structure detection for the 'library-sparse' (top), 'windows-sparse' (middle), 'windows-front' (bottom) scenes*

Blue sphere shows the region we picked and the green regions show the 50 best matches found by the detector/descriptor system

A higher-level recognition system would integrate the output of our descriptor and extract an overall scene model. For example, a simple approach might be to take the Hough transform over feature offsets to determine the spacing between feature repetitions.

6 Conclusion

In this article, we have presented a 3D detector and descriptor for the purpose of recognition in range images. Our detector builds on proven 2D detection techniques that have not previously been applied in the 3D setting.

doi: 10.1049/iet-cvi:20080037

We have shown that our detector can repeatably extract the same scene features under a range of transformations, and in particular, that our detector exhibits high repeatability at viewpoint changes of up to 60°. This is a significant improvement over 2D detectors such as those used in previous 3D recognition systems.

Our local 3D descriptor uses surface normals as its basic descriptive element, and is invariant to rigid transformations. The use of deviation angles and histograms gives robustness to noise and sampling density. We have presented empirical results that show good recognition performance under viewpoint changes and addition of noise. We have also successfully applied our descriptor to two real-world recognition tasks involving repeated structure detection.

7 Acknowledgments

The authors thank Prof. David Suter for providing the range data, all of which depict buildings on the Monash University campus.

This work was supported by a Google Research Award.

8 References

[1] LOWE D.: 'Distinctive image features from scale-invariant keypoints', *Int. J. Comput. Vis*, 2004, **60**, (2), pp. 91–110

[2] FROME A., HUBER D., KOLLURI R., BULOW T., MALIK J.: 'Recognizing objects in range data using regional point descriptors'. Proc. European Conf. Computer Vision, 2004

[3] JOHNSON A.E., HEBERT M.: 'Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1999, **21**, (5), pp. 433–449

[4] HARRIS C., STEPHENS M.: 'A combined corner and edge detector'. Proc. 4-Alvey Vision Conf., 1988, pp. 147–151

[5] BAY H., TUYTELAARS T., VAN GOOL L.: 'SURF: speeded up robust features'. 9-European Conf. Computer Vision, May 2006

[6] LINDEBERG T.: 'Feature detection with automatic scale selection', *Int. J. Comput. Vis.*, 1998, **30**, (2), pp. 77–116

[7] MIKOLAJCZYK K., TUYTELAARS T., SCHMID C., *ET AL.*: 'A comparison of affine region detectors', *Int. J. Comput. Vision*, 2005, **65**, (1–2), pp. 43–72

[8] MIKOLAJCZYK K., SCHMID C.: 'A performance evaluation of local descriptors', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2005, **27**, (10), pp. 1615–1630

[9] KE Y., SUKTHANKAR R.: 'PCA-SIFT: a more distinctive representation for local image descriptors', *cvpr*, 2004, **2**, pp. 506–513

[10] KOBAYASHI S., NOMIZU K.: 'Foundations of differential geometry' vol. 1 (Interscience Publishers, London, NY, 1963)

[11] FLINT A.: 'Thrift: a local detector and descriptor for 3D object recognition'. Honours Computer Science Thesis, University of Adelaide, 2007

[12] RUBNER Y., TOMASI T., GUIBAS L.J.: 'The earth mover's distance as a metric for image retrieval'. Technical report, Stanford University, Stanford, CA, USA, 1998

[13] FORSYTH D.A., PONCE J.: 'Computer vision: a modern approach' (Prentice Hall Professional Technical Reference, 2002)

[14] ARMAN F., AGGARWAL J.K.: 'Model-based object recognition in dense-range images: a review', *ACM Comput. Surv.*, 1993, **25**, (1), pp. 5–43

[15] MOREELS P., PERONA P.: 'Evaluation of features detectors and descriptors based on 3D objects'. ICCV '05: Proc. 10th IEEE Int. Conf. Computer Vision (ICCV'05), Washington, DC, USA, 2005, vol. 1, pp. 800–807

[16] YAMANY S.M., FARAG A.A.: 'Surfacing signatures: an orientation independent free-form surface representation scheme for the purpose of objects registration and matching', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2002, **24**, (8), pp. 1105–1120