A Comparative Evaluation of 3D Keypoint Detectors

Sílvio Filipe, Luís A. Alexandre

IT - Instituto de Telecomunicações
Department of Computer Science, University of Beira Interior, 6200-001 Covilhã, Portugal.
E-mail:{sfilipe,lfbaa}@ubi.pt

***Abstract*—When processing 3D point cloud data, features must be extracted from a small set of points, usually called keypoints or points of interest. This is done to avoid the computational complexity required to extract features from all points in a point cloud. There are many keypoint detectors, and this suggests the need for a comparative evaluation. In this paper we propose to make a description and evaluation of the keypoint detectors most often cited in the literature and available in a public point cloud library. We make a comparative assessment to verify the invariance of the 3D keypoint detectors according to different rotations, scales changes and translations. To do this evaluation, we use absolute and relative repeatability rates. These measures assess the geometric stability of the methods under various transformations. Using these criteria, we evaluate the robustness of the keypoint detectors with respect to changes in point-of-view. The method that presents the best repeatability rate is the ISS3D.**

***Index Terms*—3D Keypoints, 3D Interest Points, 3D Object Recognition, Performance Evaluation**

I. Introduction

The computational cost of descriptors is generally high, so it does not make sense to extract descriptors in all points of a cloud. Thus, keypoint detectors are used to select interesting points in the cloud on which descriptors are then found. The purpose of the keypoint detectors is to determine the points that are different in order to allow an efficient object description and correspondence with respect to point-of-view variations [1].

This work is motivated by the need to quantitatively compare different keypoint detector approaches, in a common and well established experimentally framework, given the large number of available keypoints detectors. Inspired by the work on 2D features [2], [3] and 3D [4], and by a similar work on descriptor evaluation [5], a comparison of several 3D keypoint detectors is made in this work.

To evaluate the invariance of keypoints detection methods, we extract the keypoints directly from the original cloud. Moreover, we apply a transformation to the original 3D point cloud before extracting another set of keypoints. Once we get these keypoints from the transformed cloud, we apply an inverse transformation, so that we can compare these with the keypoints extracted from the original cloud. If a particular method is invariant to the applied transformation, the keypoints extracted directly from the original cloud should correspond to the keypoints extracted from the cloud where the transformation was applied.

The correspondence between the keypoints extracted directly from the original cloud and the ones extracted from transformed cloud is done using the 3D point-line distance [6]. A line in three dimensions can be specified by two points $\mathbf{x_1} = (x_1, y_1, z_1)$ and $\mathbf{x_2} = (x_2, y_2, z_2)$ lying on it, then a vector line is produced. The squared distance between a point on the line with parameter t and a point $\mathbf{x_0} = (x_0, y_0, z_0)$ is therefore

$$d^2 = \begin{array}{l}[(x_1 - x_0) + (x_2 - x_1)t]^2 + [(y_1 - y_0) + \\ (y_2 - y_1)t]^2 + [(z_1 - z_0) + (z_2 - z_1)t]^2\end{array}. \quad (1)$$

To minimize the distance, set $\partial(d^2)/\partial t = 0$ and solve for t to obtain

$$t = -\frac{(\mathbf{x_1} - \mathbf{x_0}) \cdot (\mathbf{x_2} - \mathbf{x_1})}{|\mathbf{x_2} - \mathbf{x_1}|^2}, \quad (2)$$

where $\cdot$ denotes the dot product. The minimum distance can then be found by plugging t back into 1. Using the vector quadruple product ($(\mathbf{A} \times \mathbf{B})^2 = \mathbf{A}^2\mathbf{B}^2 - (\mathbf{A} \cdot \mathbf{B})^2$) and taking the square root results, we can obtain:

$$d = \frac{|(\mathbf{x_0} - \mathbf{x_1}) \times (\mathbf{x_0} - \mathbf{x_2})|}{|\mathbf{x_2} - \mathbf{x_1}|}, \quad (3)$$

where $\times$ denotes the cross product. Here, the numerator is simply twice the area of the triangle formed by points $\mathbf{x_0}$, $\mathbf{x_1}$, and $\mathbf{x_2}$, and the denominator is the length of one of the bases of the triangle.

The low price of 3D cameras has increased exponentially the interest in using depth information for solving vision tasks. A useful resource for users of this type of sensors is the Point Cloud Library (PCL) library [7] which contains many algorithms that deal with point cloud data, from segmentation to recognition, from search to input/output. This library is used to deal with real 3D data and we used it to evaluate the robustness of the detectors with variations of the point-of-view in real 3D data.

The organization of this paper is as follows: the next section presents a detailed description of the methods that we evaluate; the results and the discussion appear in section III; and finally, we end the paper in section IV with the conclusions.

II. Evaluated 3D Keypoint Detectors

A. Harris3D

The Harris method [8] is a corner and edge based method and these types of methods are characterized by their high-intensity changes in the horizontal and vertical directions. These features can be used in shape and motion analysis, they can be detected directly from the grayscale images. For the 3D case, the adjustment made in PCL for the Harris3D detector replaces the image gradients in the covariance matrix by surface normals and uses the same responses. To find the keypoints, they use a Hessian matrix of the intensity C around each point. This matrix is smoothed by an isotropic Gaussian filter $w_G(\sigma)$. That is $C_{Harris} = w_G(\sigma) * C$, where σ is the standard deviation of the filter and the operation $*$ denotes convolution. A measure of the *keypoints response* at each pixel coordinates (x, y, z) is then defined by

$$r(x,y) = det(C_{Harris}(x,y,z)) - k\left(trace(C_{Harris}(x,y,z))\right)^2, \quad (4)$$

where k is a positive real valued parameter. This parameter serves roughly as a lower bound for the ratio between the magnitude of the weaker edge and that of the stronger edge.

B. SIFT3D

The Scale Invariant Feature Transform (SIFT) keypoint detector was proposed by Lowe [9]. The SIFT features are represented by vectors that represent local cloud measurements. The main steps used by the SIFT detector when locating keypoints are presented below.

The original algorithm for 3D data was presented by Flint *et al.* [11], which uses a 3D version of the Hessian to select such interest points. A density function $f(x, y, z)$ is approximated by sampling the data regularly in space. A scale space is built over the density function, and a search is made for local maxima of the Hessian determinant.

The input cloud, $I(x, y, z)$ is convolved with a number of Gaussian filters whose standard deviations $\{\sigma_1, \sigma_2, \dots\}$ differ by a fixed scale factor. That is, $\sigma_{j+1} = k\sigma_j$ where k is a constant scalar that should be set to $\sqrt{2}$. The convolutions yield smoothed images, denoted by

$$G(x, y, z, \sigma_j), i = 1, \dots, n. \quad (5)$$

The adjacent smoothed images are then subtracted to yield a small number (3 or 4) of Difference-of-Gaussian (DoG) clouds, by

$$D(x, y, z, \sigma_j) = G(x, y, z, \sigma_{j+1}) - G(x, y, z, \sigma_j). \quad (6)$$

C. SUSAN

The Smallest Univalue Segment Assimilating Nucleus (SUSAN) corner detector has been introduced by Smith and Brady [12] and relies on a different technique. Rather than evaluating local gradients, which might be noise-sensitive and computationally expensive, a morphological approach is used. SUSAN is a generic low-level image processing technique, which apart from corner detection has also been used for edge detection and noise suppression. For each pixel in the image, we consider a circular neighborhood of fixed radius around it. The center pixel is referred to as the nucleus, and its intensity value is used as reference. Then, all other pixels within this circular neighborhood are partitioned into two categories: similarity or differentiation, depending on whether they have "similar" intensity values as the nucleus or "different" intensity values. This way, each cloud point has associated with it a local area of similar brightness, whose relative size contains important information about the structure of the cloud at that point. In more or less homogeneous parts of the cloud, the local area of similar brightness covers almost the entire circular neighborhood. Hence, corners can be detected as locations in the cloud where the number of points with similar intensity value in a local neighborhood reaches a local minimum and is below a predefined threshold.

D. ISS3D

Intrinsic Shape Signatures (ISS) [13] is a method relying on region-wise quality measurements. This method uses the magnitude of the smallest eigenvalue (to include only points with large variations along each principal direction) and the ratio between two successive eigenvalues (to exclude points having similar spread along principal directions).

The ISS $S_i = \{F_i, f_i\}$ at a point p_i consists of: the intrinsic reference frame $F_i = \{p_i, \{e_i^x, e_i^y, e_i^z\}\}$ where p_i is the origin, and $\{e_i^x, e_i^y, e_i^z\}$ is the set of basis vectors. The intrinsic frame is a characteristic of the local object shape and independent of viewpoint. Therefore, the view independent shape features can be computed using the frame as a reference. However, its basis $\{e_i^x, e_i^y, e_i^z\}$, which specifies the vectors of its axes in the sensor coordinate system, are view dependent and directly encode the pose transform between the sensor coordinate system and the local object-oriented intrinsic frame, thus enabling fast pose calculation and view registration.

III. Experimental Evaluation and Discussion

A. Repeatability measure

The most important feature of a keypoint detector is its *repeatability*. This feature takes into account the capacity of the detector to find the same set of keypoints in different instances of a particular model. The differences may be due to noise, view-point change, occlusion or by a combination of the above.

The repeatability measure used in this paper is based on the measure used in [2] for 2D keypoints and in [4] for 3D Keypoints. A keypoint extracted from the model M_h, k_h^i transformed according to the rotation, translation and scale, (R_{hl}, t_{hl}), is said to be repeatable if the distance from its nearest neighbor, k_l^j, in the set of keypoints extracted from the scene S_l is less than a threshold ϵ, $||R_{hl}k_h^i + t_{hl} - k_l^j|| < \epsilon$.

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

(M_h,S_l), the absolute repeatability is defined as $r_{abs} = |RK_{hl}|$ and the relative repeatability is given by $r = \frac{|RK_{hl}|}{|K_{hl}|}$. The set K_{hl} is the set of all the keypoints extracted on the model M_h that are not occluded in the scene S_l. This set is estimated by aligning the keypoints extracted on M_h according to the rotation, translation and scale and then checking for the presence of vertices in S_l in a small neighborhood of the transformed keypoints. If at least a vertex is present in the scene in such a neighborhood, the keypoint is added to K_{hl}.

B. Results and Discussion

To perform the evaluation of keypoint detectors, we use a subset of the large dataset of 3D point clouds from [14]. This dataset is a hierarchical multi-view object dataset collected using an RGB-D camera. The RGB-D Object Dataset[1] [14] contains clouds of 300 physically distinct objects taken from multiple views, organized into 51 categories, containing a total of 207621 segmented clouds. The chosen objects are commonly found in home and office environments, where personal robots are expected to operate.

In this article, we intend to evaluate the invariance of the methods presented, in relation to rotation, translation and scale changes. For this, we vary the rotation according to the three axes (X, Y and Z). The rotations applied ranged from 5^o to 35^o, with 10^o step. The translation is performed simultaneously in the three axes and the image displacement applied on each axis is obtained randomly. Finally, we apply random variations (between $]1\times, 5\times]$) to the scale.

Figure 1 shows the results of the evaluation of the different methods with various applied transformations. The threshold distances analyzed vary between $[0, 2]$ cm, with small jumps in a total of 33 distances calculated equally spaced. As we see in section II, the methods have a relatively large set of parameters to be adjusted: the values used were the ones set by default in PCL.

Regarding the relative repeatability (shown in figures 1(a), 1(b), 1(c), 1(g), 1(h) and 1(i)) the methods presented have a fairly good performance in general. In relation to the rotation (see figures 1(a), 1(b), 1(c) and 1(g)), increasing the rotation angle of the methods tends to worsen the results. Ideally, the method results should not change independently of the transformations applied. Regarding the applied rotation, the method ISS3D is the one that provides the best results. In this transformation (rotation), the biggest difference that appears between the various methods is in the 5 degrees rotation. In this case, the method ISS3D achieves almost total correspondence keypoints with a distance between them of 0.25 cm. Whereas for example the SIFT3D only achieves this performance for keypoints at a distance of 1 cm. In both the scaling and translation (shown in figures 1(h) and 1(i)), the methods exhibit very similar results to those obtained for small rotations (5^o rotation in figure 1(a)) with the exception of the SUSAN method, that has a relatively higher invariance to scale changes.

[1]The dataset is publicly available at http://www.cs.washington.edu/rgbd-dataset.

Figures 1(d), 1(e), 1(f), 1(j), 1(k) and 1(l) show the absolute repeatability, that present the number of keypoints obtained by the methods. With these results we can see that the method that has higher absolute repeatability (SUSAN) is not the one that shows the best performance in terms of relative repeatability. In terms of the absolute repeatability the ISS3D and SIFT3D are dramatically more efficient than the SUSAN method regarding the invariance transformations evaluated in this work.

IV. Conclusions

In this paper we focused on the available keypoint detectors on the PCL library, explaining how they work, and made a comparative evaluation on public available data with real 3D objects. The experimental comparison proposed in this work has outlined aspects of state-of-the-art methods for 3D keypoint detectors. This work allowed us to evaluate the best performance in terms of various transformations (rotation, scaling and translation). Overall, SIFT3D and ISS3D yielded the best scores in terms of repeatability and ISS3D demonstrated to be the most efficient. Future work includes extension of some methodologies proposed for the keypoint detectors work with large rotations and occlusions, and the evaluation of the best combination of keypoint detectors/descriptors.

Acknowledgment

This work is supported by '*FCT - Fundação para a Ciência e Tecnologia*' (Portugal) through the research grant 'SFRH/BD/72575/2010', and the funding from 'FEDER - QREN - Type 4.1 - *Formação Avançada*', subsidized by the European Social Fund and by Portuguese funds through 'MCTES'.

References

[1] A. Mian, M. Bennamoun, and R. Owens, "On the Repeatability and Quality of Keypoints for Local Feature-based 3D Object Retrieval from Cluttered Scenes," *International Journal of Computer Vision*, vol. 89, no. 2-3, pp. 348–361, 2010.

[2] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of Interest Point Detectors," *International Journal of Computer Vision*, vol. 37, no. 2, pp. 151–172, 2000.

[3] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. V. Gool, "A Comparison of Affine Region Detectors," *International Journal of Computer Vision*, vol. 65, no. 1-2, pp. 43–72, Oct. 2005.

[4] S. Salti, F. Tombari, and L. D. Stefano, "A Performance Evaluation of 3D Keypoint Detectors," in *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, 2011, pp. 236–243.

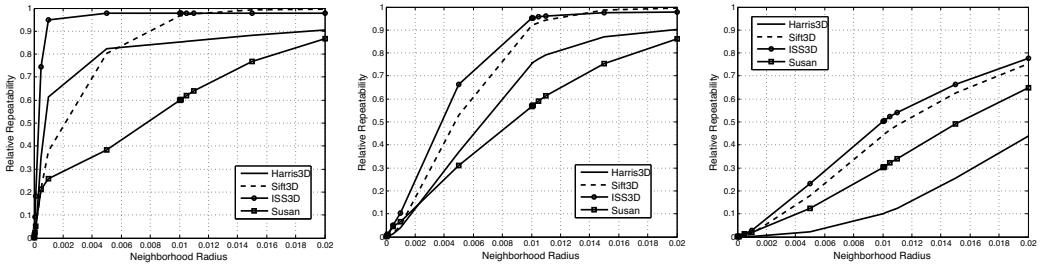
[5] L. A. Alexandre, "3D descriptors for object and category recognition: a comparative evaluation," in *Workshop on Color-Depth Camera Fusion in Robotics at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vilamoura, Portugal, October 2012.

[6] E. W. Weisstein, "Point-line distance–3-dimensional," http://mathworld.wolfram.com/Point-LineDistance3-Dimensional.html, 2012.

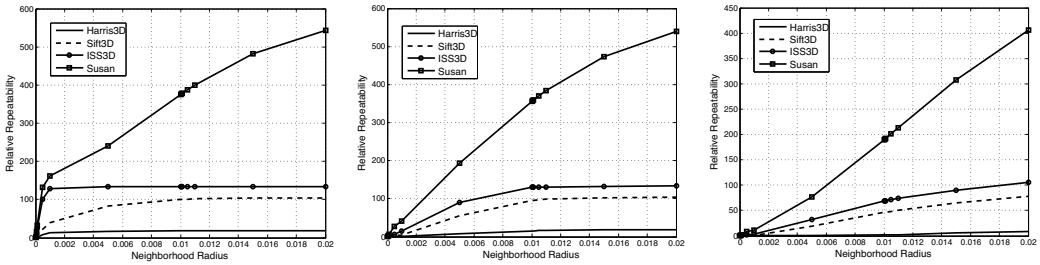
[7] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *International Conference on Robotics and Automation*, Shanghai, China, May 9-13 2011.

[8] C. Harris and M. Stephens, "A combined corner and edge detector," in *Alvey Vision Conference*, Manchester, 1988, pp. 147–152.

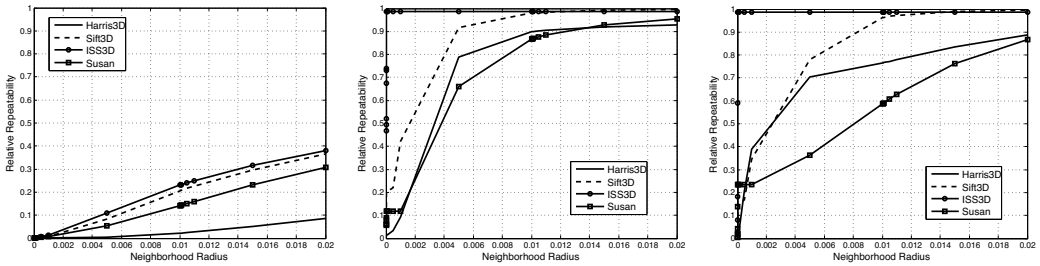
[9] D. Lowe, "Local feature view clustering for 3D object recognition," *Computer Vision and Pattern Recognition*, vol. 1, pp. I–682–I–688, 2001.



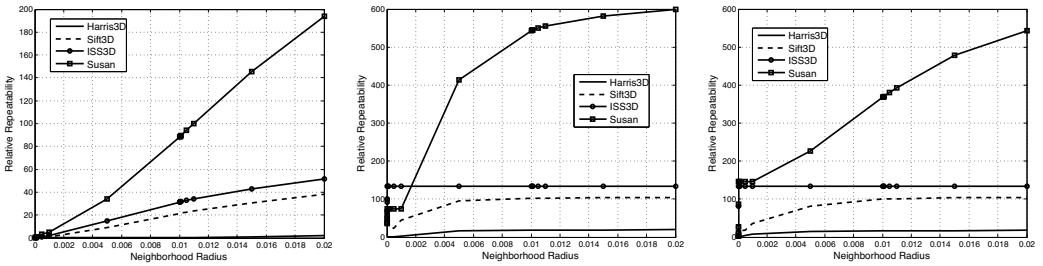
(a) Relative repeatability for 5^o rotation. (b) Relative repeatability for 15^o rotation. (c) Relative repeatability for 25^o rotation.



(d) Absolute repeatability for 5^o rotation. (e) Absolute repeatability for 15^o rotation. (f) Absolute repeatability for 25^o rotation.



(g) Relative repeatability for 35^o rotation. (h) Relative repeatability for scale change. (i) Relative repeatability for translation cloud.



(j) Absolute repeatability for 35^o rotation. (k) Absolute repeatability for scale change. (l) Absolute repeatability for translation cloud.

Fig. 1. Results for the relative and absolute repeatability measures (best viewed in color). The relative repeatability is presented in figures (a), (b), (c), (g), (h) and (i), and the absolute repeatability in figures (d), (e), (f), (j), (k) and (l). The presented neighborhood radius is in meters.

[10] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[11] A. Flint, A. Dick, and A. Hengel, "Thrift: Local 3D Structure Recognition," in *9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications*, Dec. 2007, pp. 182–188.

[12] S. M. Smith and J. M. Brady, "SUSAN – A new approach to low level image processing," *International Journal of Computer Vision*, vol. 23, no. 1, pp. 45–78, 1997.

[13] Y. Zhong, "Intrinsic shape signatures: A shape descriptor for 3D object recognition," *International Conference on Computer Vision Workshops*, pp. 689–696, Sep. 2009.

[14] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view RGB-D object dataset," in *International Conference on Robotics and Automation*, May 2011, pp. 1817–1824.