3D Feature Detector-Descriptor Pair Evaluation on Point Clouds

Paula Stancelova
Faculty of Mathematics, Physics and Informatics
Comenius University in Bratislava
Slovakia
paula.budzakova@fmph.uniba.sk

Elena Sikudova
Faculty of Mathematics and Physics
Charles University
Prague, Czech Republic
sikudova@cgg.mff.cuni.cz

Zuzana Cernekova
Faculty of Mathematics, Physics and Informatics
Comenius University in Bratislava
Slovakia
zuzana.cernekova@uniba.sk

***Abstract*—In recent years, computer vision research has focused on extracting features from 3D data. In this work, we reviewed methods of extracting local features from objects represented in the form of point clouds. The goal of the work was to make theoretical overview and evaluation of selected point cloud detectors and descriptors. We performed an experimental assessment of the repeatability and computational efficiency of individual methods using the well known Stanford 3D Scanning Repository database with the aim of identifying a method which is computationally-efficient in finding good corresponding points between two point clouds. We also compared the efficiency of detector-descriptor pairing showing that the choice of a descriptor affects the performance of the object recognition based on the descriptor matching. We summarized the results into graphs and described them with respect to the individual tested properties of the methods.**

***Index Terms*—3D detector, 3D descriptor, point cloud, feature extraction**

I. Introduction

With the availability of the low-cost 3D sensors, it is natural that using three dimensional point clouds has gained a lot of attention in the past decade. 3D local features provide similar benefits in point clouds as two dimensional local features in images. The number of research works proposing 3D detectors increased especially over the last years. 3D local features are useful to deal with many computer vision tasks working with 3D point clouds such as point cloud registration (pose estimation) [1], automatic object localization and recognition in 3D point cloud scenes [2], model retrieval [3] or face recognition [4].

The procedure of solving these tasks usually starts with detecting keypoints using detectors designed by studying the geometric properties of local structures [1]. Then, the description of the keypoints is performed and the final step is the classification or matching of the keypoints. The latest deep learning approach combines all three steps into one. The features and their descriptions are learned during the training of the classifier [5, 6].

When using the hand crafted features we need to ensure that the detected keypoints are representative and the descriptors are robust to clutter and occlusion in order to get reliable results (classification, localization,...).

The goal of the paper is to provide primary performance evaluation of hand crafted feature detectors for point clouds, as they have a significant effect on the overall performance of aforementioned computer vision tasks. We inspect the ability of the detectors to find repeatable points and their suitability for an object recognition task in cluttered scenes when combined with different feature descriptors.

II. Related Work

Many works exist that test and compare methods to extract 3D local features. A quantitative comparison of the performance of the three most general detectors and two descriptors for monitoring plant health and growth is given in [7]. The choice of test methods was based on their most common use in various applications. Filipe and Alexandre [8] test 3D detectors in the Point Cloud Library (PCL) [9] that work directly on RGB-D data. They evaluate the performance of the detectors in terms of various transformations. Hänsch, Weber, and Hellwich [10] focus on the benefits of 3D detectors and descriptors in the specific context of point cloud fusion that can be used, for example, to reconstruct a surface. Tombari, Salti, and di Stefano [11] test the state-of-the-art detectors, which they divide into two new categories (fixed-scale, adaptive-scale). The testing scenes are built by randomly rotating and translating the selected 3D models of synthetic data. Mian, Bennamoun, and Owens [12] propose a multi-scale keypoint detection algorithm for extracting scale invariant local features with automatic selection of the appropriate scale at each keypoint. They test the methods for 3D object retrieval from complex scenes containing clutter and occlusion.

Inspired by [11] we propose a comparison of 3D detectors that work primarily on the point clouds without the use of the color information. We focus on the object recognition scenario, which is characterized mainly by a heavy occlusion. Subsequently, we assess the performance of the selected 3D descriptors in combination with the tested detectors. Also, the purpose is to show the performance of selected combinations in recognizing objects in a scene, as we have not found these combinations in the existing papers.

III. Methodology

In our approach we focused on detectors and descriptors that are available in the PCL. For the implementation details, consult the PCL webpage[1].

A. 3D Detectors

1) Harris3D: The Harris3D detector [13] is the extension of the 2D corner detection method of Harris and Stephens [14]. Harris3D is using first order derivatives along two orthogonal directions on the surface. The derivatives are obtained so that the neighboring area is approximated by quadratic surface. Afterwards, similarly to 2D detector, the covariance matrix is calculated, but the image gradients are replaced by the surface normals. For final detection of keypoints, the Hessian matrix of intensity is used for every point, smoothed by an isotropic Gaussian filter.

2) Normal Aligned Radial Feature (NARF): Detector NARF [15] is designed to choose the keypoints, so that they are able to repair the information about the object borders and its surface to obtained highest robustness of the method. The selected points are supposed to be in positions where the surface is stable and where there are sufficient changes in the local neighborhood to be robustly detected in the same place even if observed from different perspectives. At first, the detector determines the object border, for which the change of distances between two adjacent points is used. To achieve the keypoint stability, for every image point and its local neighborhood, the amount and dominant direction of the surface changes at this position are determined. The interest value is calculated as the difference between orientations in area and changes of the surface. Afterwards, the smoothing of the interest values and non-maximum suppression is performed to determine the final keypoints.

3) SIFT3D: SIFT 3D is an extension of the original 2D SIFT detector, where for finding the keypoints the scale pyramid of difference of Gaussians (DoG) is used [16]. SIFT 3D is using 3D version of Hessian matrix, where the density function is approximated by sampling the data regularly in space. Over the density function a scale space is built, where search for local maxima of the Hessian determinant is performed. The input cloud is convolved with a number of Gaussian filters whose standard deviations differ by a fixed scale factor. Using the convolution the point clouds are smoothed and the difference is calculated to obtained three to four DoG point clouds. These two steps are repeated to get higher number of DoG in the scale-space. The 3D SIFT keypoints are identified as the scale-space extrema of the DoG function similarly to 2D SIFT.

4) Intrinsic Shape Signatures 3D (ISS3D): The ISS3D keypoint detector [17] relies on measuring the local distances or the points density. It uses the eigenvalues and eigenvectors of the scatter matrix, created from supporting region points. The eigenvector corresponding to the smallest eigenvalue is used as a surface normal and the ratio between two successive eigenvalues is used to reject similar points.

[1]https://pointclouds.org/

B. 3D Descriptors

Many existing 3D local features descriptors use histograms to represent different characteristics of the local surface. Principally, they describe the local surface by combining geometric or topological measurements into histograms according to point coordinates or geometric attributes. We classify these descriptors into two groups: *spatial distribution histogram* and *geometric attribute histogram* based descriptors.

Spatial distribution histogram based descriptors represent local surface by generating histograms according to spatial coordinates. They mostly start with the construction of a Local Reference Frame (LRF) or Local Reference Axis (LRA) for keypoint, and divide 3D support region to the LRF/LRA. Afterwards, they generate a histogram for local surface by accumulating the spatial distribution measurements for each spatial bin. *3D Shape Context (3DSC)* and *Unique Shape Context (USC)* are spatially based.

Geometric attribute histogram based descriptors represent local surface by generating histograms based on geometric attributes, as normals or curvatures, of the points on the surface. These descriptors also mostly construct the LRF/LRA. *Point Feature Histogram (PFH)*, *Fast Point Feature Histogram (FPFH)* and *Signature of Histograms of OrienTations (SHOT)* are geometrically based.

1) 3D Shape Context: 3DSC descriptor [18] is extension of the 2D Shape Contexts descriptor [19]. The algorithm use the surface normal at the keypoint as its LRA. First, a spherical grid is placed at the keypoint, which is the center of the grid, with the north pole of the grid being aligned with the surface normal. The support region is divided into several bins, logarithmic along the radial dimension and linear along the azimuth and the elevator dimensions of the spherical grid. The 3DSC descriptor is generated by counting the weighted number of points falling into each bin of the 3D grid. The final descriptor is chaining the three partial descriptors, that represent the numbers of bins along the radial, azimuth and elevation axes.

2) Unique Shape Context: USC descriptor [20] is extension of 3DSC, which bypasses computing multiple descriptors at a given keypoint. LRF is constructed for each keypoint and aligned with the local surface in order to provide invariance to rotations and translations. The support region is divided into several bins. Final USC descriptor is generated analogous to the approach used in 3DSC.

3) Point Feature Histogram: The PFH descriptor [21] works with point pairs in the support region. First, the Darboux frame is defined by using the surface normals and point positions, for each pair of the points in the support region. Then, four features for each point pair using the Darboux frame, the surface normals, and their point positions are calculated. The PFH descriptor is generated by accumulating points in particular bins along the four dimensions. The length

of the final PFH is based on the number of histogram bins along each dimension.

4) Fast Point Feature Histogram: The FPFH descriptor [22] consists of two steps. In the first step, a Simplified Point Feature Histogram (SPFH) is generated for each point by calculating the relationships between the point and its neighbors. In SPFH, the descriptor is generated by chaining three separate histograms along each dimension. In the second step, FPFH descriptor is constructed as the weighted sum of the SPFH of the keypoint and the SPFHs of the points in the support region. The length of the FPFH descriptor is based on the number of histogram bins along each dimension.

5) Signature of Histograms of OrienTations: The SHOT descriptor [23] originated as an inspiration by SIFT descriptor. The descriptor encodes the histograms of the surface normals in different spatial locations. First, the LRF is constructed for each keypoint and its neighboring points in the support region are aligned with the LRF. The support region is divided into several units along the radial, azimuth and elevation axes. Local histogram for each unit is generated by accumulating point counts into bins according to the angles between the normals at the neighboring points within the volume and the normal at the keypoint. The final descriptor is chaining all the local histograms.

C. Descriptors matching

The objects in the test scenes are detected using the nearest neighbor distance ratio matching method. In the nearest neighbor matching method, two descriptors match, if they are the closest and their distance is below a threshold. With this strategy, there is at most one match but it might not be correct, especially when there are few keypoints with similar geometric properties in the point cloud. In this case the distances from the nearest and the second nearest descriptors are similar. To overcome this property we use the nearest neighbor distance ratio (NNDR) matching, where the descriptors D_A and D_B are matched if

$$||D_A - D_B||/||D_A - D_C|| < t, \tag{1}$$

where D_B is the first and D_C is the second nearest neighbor to D_A and t is a specified threshold.

IV. Experiments

A. Dataset

Four point cloud models from the publicly available 3D model database the Stanford 3D Scanning Repository [24] were used to perform our experiments (the Stanford Bunny, the Happy Budha, the Dragon, the Armadillo). All the selected models were scanned by a Cyber-ware 3030 MS scanner. We used scaled versions of the models with the fixed-scale of 6, 9, 12, 15 times the cloud resolution. We also combined several transformed models to create 15 testing scenes having 265-290 thousands of points. The scenes were created by models combination such way that the models were close to each other and their different parts were occluded as seen in Figure 1.

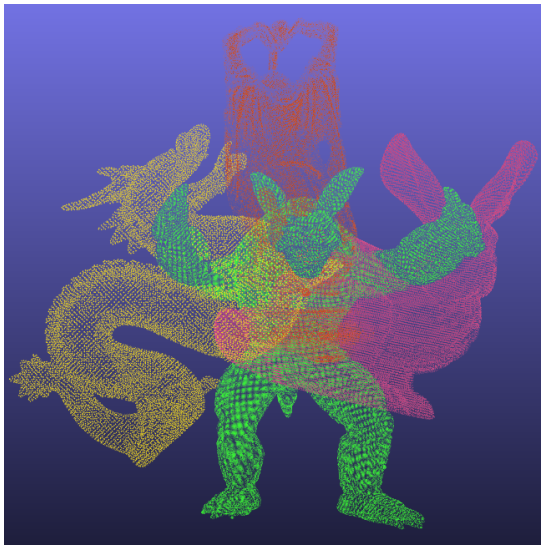

Fig. 1. Example of the scene which we used.

B. Evaluation Criteria

If a keypoint is detectable under changing conditions including viewpoint change, noise, partial neighborhood occlusion, it can be useful for identifying an object. This property is called repeatability. A keypoint is repeatable [11], if it is found in a small neighborhood of the expected location in the scene. Let k_M be a keypoint found on the original model. The expected location in the scene is then $T(k_M)$, where $T()$ is the transformation of the original model (rotation and translation) when inserted into the scene. Let k_S be the keypoint in the scene, which is the closest to $T(k_M)$. The keypoint k_M is repeatable, when

$$\|T(k_M) - k_S\| < \varepsilon. \tag{2}$$

Tombari et al. [11] defined also the *absolute* and *relative repeatability* of the keypoints. When R is the set of repeatable keypoints found on a given model in a given scene, the absolute repeatability is the number of the repeatable keypoints

$$r_a = |R| \tag{3}$$

and the relative repeatability is measured relatively to the number of keypoints found on the model in the scene having a corresponding keypoint on the original model (K_S)

$$r_r = \frac{|R|}{|K_S|}. \tag{4}$$

The *descriptiveness* of selected descriptors is evaluated by combining them with the tested detectors. The purpose is to demonstrate the effect of different detectors on the performance of feature descriptors. The performance of each detector-descriptor combination is measured by the Precision and Recall generated as follows. First, the keypoints are detected from all the scenes and all the models. A feature descriptor is then computed for each keypoint using the selected descriptors. Next, we use NNDR to perform feature matching. If the distance between the pair of keypoint descriptors is less

than half of the support radius, which is the threshold, the match is assumed correct. Otherwise, it is a false match. The *precision* is calculated as

$$\text{precision} = \frac{\text{Nr. of correct matches}}{\text{Nr. of identified matches}}. \quad (5)$$

The *recall* is calculated as

$$\text{recall} = \frac{\text{Nr. of correct matches}}{\text{Nr. of corresponding features}}. \quad (6)$$

We also evaluate the *computational effectiveness* of the detectors and descriptors. We calculate the time needed for detecting the keypoints on several objects or scenes with various point density to determine the effectiveness of detector. Also, we calculate the time needed to describe the extracted keypoints. All methods are used as implemented in PCL. For all the selected detectors, we use the default parameters proposed in the original publications.

C. Results

1) Detectors: Based on the result of the experiments we can draw interesting conclusions. In Fig.2, we present the results of absolute and relative repeatability of HARRIS3D, NARF, SIFT3D and ISS3D detectors. To test the effect of occlusion in scenes on the detector performance, we investigate scenes where up to 30% of each tested object is occluded.

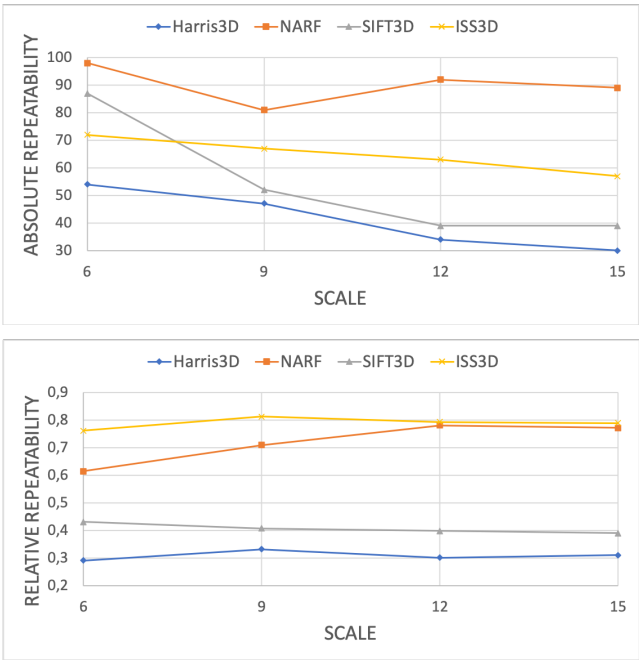

Fig. 2. The absolute (up) and the relative (down) repeatability of tested detectors.

In the model-scene scenario, the results show that NARF achieves the highest absolute repeatability for all scales, followed by ISS3D, but the relative repeatability of the ISS3D points was higher. In both tests, the SIFT3D and HARRIS3D detectors reached lower numbers of repeatable points. The only exception is the scale 6, where the absolute number of repeatable points detected by SIFT3D was the second highest.

The measured detection time is almost constant over scenes with varying number of points. In the scale fixed to 6 the scenes have 265-290 thousands of points. ISS3D is the exception where the detection time rises with the number of points in the scene. The average time $\pm$ standard deviation is 24.24 ± 2.58 seconds. NARF and SIFT3D are the fastest with 3.25 ± 0.09 and 9.94 ± 0.5 seconds respectively. The slowest is Harris3D with 28.7 ± 0.78seconds. All methods are used as implemented in the PCL.

2) Descriptors: In Figure 3 we present the precision and recall measures of the descriptor matching task on scale 6 objects. Precision and recall are averaged over all models and all scenes. All detector-descriptor combinations achieve more than 80% recall, and 70% precision. We can draw several conclusions from the results.

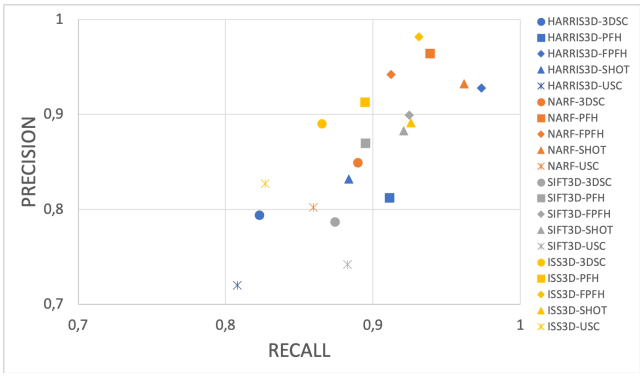

Fig. 3. Performance of the selected descriptors combined with different 3D keypoint detectors.

First, the spatial distribution histogram based descriptors (USC and 3DSC) achieve worse results than the geometric attribute histogram based descriptors (SHOT, PFH and FPFH) regardless of the detector used.

Second, the best performance was achieved by the FPFH descriptor for all the detectors, except the NARF detector, where SHOT was the best, but the FPFH combined with any detector reached over 90% precision and recall, which does not hold for any other descriptor.

Third, the results reveal that the best choice of descriptor for HARRIS3D detector is FPFH, since all other descriptors did not reach 85% precision. For ISS3DD detector the best choice is also the FPFH, but the other geometrically based descriptors are not that far behind, achieving about 90% precision. NARF detector combined with any geometry based descriptor surpassed the 90% precision and recall. SIFT3D detector combined with any descriptor did not show satisfactory results.

Some other experiments we performed with different amount of occlusion are described in [25].

As with the detectors, the descriptors are implemented in the same PCL framework, so their performance can be compared by taking the average time needed to generate the descriptor of the same model represented in different scales. The measured average time of keypoint description is highly dependent on the number of points in the local region for PFH a FPFH.

For small-sized support regions (100–1000 points) the PFH a FPFH descriptors are orders of magnitude faster than the other descriptors, taking only 0.1–8 ms against 120–180 ms. But for 10000, they are 10 times slower (9900 ms vs. 720 ms). For the 5000 points neighborhood the time were comparable (611 ms vs. 408ms). All descriptors are comparably fast for neighborhoods with around 5000 points.

V. Conclusions

This paper presents the performance evaluation of hand crafted feature detectors for point clouds. The repeatability tests of the detectors show that the data which contained occlusions have a high impact on their performance. The best repeatable detector of the set of test detectors in scale 6 is NARF for absolute and ISS3D for relative repeatability. Our tests show that descriptors based on geometric attribute histogram (SHOT, PFH and FPFH) achieve better results than the spatial distribution based descriptors (USC and 3DSC). The worst detector-descriptor pairs performance for object recognition scenario is achieved by SIFT3D and HARRIS3D detector combined with any descriptor. Our tests also show that choosing the right detector impacts the descriptor's performance in the recognition process. The FPFH and PFH descriptors are the fastest in time measurements for small number of local region points. At higher number of points, the descriptors measurements are comparable.

Acknowledgment

This work has been funded by Slovak Ministry of Education under contract VEGA 1/0796/00 and by the Charles University grant SVV-260588.

References

[1] Haowen Deng, Tolga Birdal, and Slobodan Ilic. 3D local features for direct pairwise registration. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019.

[2] J.A.C. Vargas, A.G. Garcia, S. Oprea, S.O. Escolano, and J.G. Rodriguez. *Object recognition pipeline: Grasping in domestic environments*, pages 18–33. IGI Global, 04 2018.

[3] N. Bold, C. Zhang, and T. Akashi. 3D point cloud retrieval with bidirectional feature match. *IEEE Access*, 7:164194–164202, 2019. ISSN 2169-3536.

[4] N. Markuš, I. Pandžić, and J. Ahlberg. Learning local descriptors by optimizing the keypoint-correspondence criterion: Applications to face matching, learning from unlabeled videos and 3D-shape retrieval. *IEEE Transactions on Image Processing*, 28(1):279–290, Jan 2019. ISSN 1941-0042.

[5] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5099–5108. Curran Associates, Inc., 2017.

[6] J. Li, B. M. Chen, and G. H. Lee. SO-Net: Self-organizing network for point cloud analysis. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9397–9406, June 2018.

[7] Shiva Azimi, Brejesh Lall, and Tapan K Gandhi. Performance evalution of 3D keypoint detectors and descriptors for plants health classification. In *2019 16th International Conference on Machine Vision Applications (MVA)*, pages 1–6. IEEE, 2019.

[8] Silvio Filipe and Luis A Alexandre. A comparative evaluation of 3D keypoint detectors in a RGB-D object dataset. In *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, volume 1, pages 476–483. IEEE, 2014.

[9] Radu Bogdan Rusu and Steve Cousins. 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 9-13 2011.

[10] Ronny Hänsch, Thomas Weber, and Olaf Hellwich. Comparison of 3D interest point detectors and descriptors for point cloud fusion. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2(3):57, 2014.

[11] Federico Tombari, Samuele Salti, and Luigi di Stefano. Performance evaluation of 3D keypoint detectors. *International Journal of Computer Vision*, 102(1-3):198–220, 2013.

[12] Ajmal Mian, Mohammed Bennamoun, and Robyn Owens. On the repeatability and quality of keypoints for local feature-based 3D object retrieval from cluttered scenes. *International Journal of Computer Vision*, 89(2-3):348–361, 2010. ISSN 0920-5691.

[13] Ivan Sipiran and Benjamin Bustos. Harris 3D: a robust extension of the harris operator for interest point detection on 3D meshes. *The Visual Computer*, 27(11):963, 2011.

[14] Christopher G Harris, Mike Stephens, et al. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 23.1–23.6, 1988.

[15] Bastian Steder, Radu Bogdan Rusu, Kurt Konolige, and Wolfram Burgard. Point feature extraction on 3D range scans taking into account object boundaries. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 2601–2608. IEEE, 2011.

[16] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. IEEE, 1999.

[17] Yu Zhong. Intrinsic shape signatures: A shape descriptor for 3D object recognition. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 689–696. IEEE, 2009.

[18] Andrea Frome, Daniel Huber, Ravi Kolluri, Thomas Bülow, and Jitendra Malik. Recognizing objects in range data using regional point descriptors. In *European conference on computer vision*, pages 224–237. Springer, 2004.

[19] Serge Belongie, Greg Mori, and Jitendra Malik. Matching with shape contexts. In *Statistics and Analysis of Shapes*, pages 81–105. Springer, 2006.

[20] Federico Tombari, Samuele Salti, and Luigi Di Stefano. Unique shape context for 3D data description. In *Proceedings of the ACM workshop on 3D object retrieval*, pages 57–62, 2010.

[21] Radu Bogdan Rusu, Nico Blodow, Zoltan Csaba Marton, and Michael Beetz. Aligning point cloud views using persistent feature histograms. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3384–3391. IEEE, 2008.

[22] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (FPFH) for 3D registration. In *2009 IEEE international conference on robotics and automation*, pages 3212–3217. IEEE, 2009.

[23] Samuele Salti, Federico Tombari, and Luigi Di Stefano. SHOT: Unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding*, 125:251–264, 2014.

[24] Marc Levoy, J Gerth, B Curless, and K Pull. The stanford 3D scanning repository, 2005. URL http://www-graphics.stanford.edu/data/3Dscanrep.

[25] Paula Stancelova, Elena Sikudova, and Zuzana Cernekova. Performance evaluation of selected 3D keypoint detector–descriptor combinations. Accepted for International Conference on Computer Vision and Graphics, 2020.