

M2DP: A Novel 3D Point Cloud Descriptor and Its Application in Loop Closure Detection*

Li He¹, Xiaolong Wang² and Hong Zhang¹

Abstract—In this paper, we present a novel global descriptor M2DP for 3D point clouds, and apply it to the problem of loop closure detection. In M2DP, we project a 3D point cloud to multiple 2D planes and generate a density signature for points for each of the planes. We then use the left and right singular vectors of these signatures as the descriptor of the 3D point cloud. Our experimental results show that the proposed algorithm outperforms state-of-the-art global 3D descriptors in both accuracy and efficiency.

I. INTRODUCTION

Over the past decade, the Simultaneous Localisation and Mapping (SLAM) problem has received considerable attention. An important issue in SLAM is loop closure detection, i.e., whether a robot has returned to a previously visited location. Compared with visual loop closure detection, lidar-based loop closure detection has not received significant attention in spite of its robustness to illumination changes and high accuracy in localization. For lidar-based loop closure detection, it is essential to be able to match two point clouds generated from a lidar sensor.

In general, there are two categories of methods for point cloud matching. The first category is the point-to-point matching, which operates directly on the point clouds. Popular methods in this category include the iterative closest points (ICP) [1] and its variants [2] [3], in the case when two point clouds are already roughly aligned. For recovering larger motion between two point clouds, one can resort to the 4-Points Congruent Sets (4PCS) [4], the Super-4PCS [5], the Congruent Pyramids method [7] and the 2D-3PCS [6]. In contrast, the second category of point cloud matching methods generates one global descriptor for a cloud, and estimates the similarity between clouds by comparing their descriptors. Compared with point-to-point methods, matching methods in this latter category have the important advantages in storage and descriptor matching efficiency. Motivated by its success in 2D image matching, the bags-of-words (BoW) [20] method is a possible way in describing a 3D cloud. BoW first calculates local descriptors of key points, and vector-quantize the local descriptors into words of a dictionary that

has been constructed off line. The histogram of the words is used as the descriptor.

One problem for bags-of-words is the complexity of vector quantization, or the clustering of local descriptors. Secondly, the local descriptors are typically constructed from surface normals in a support neighborhood to capture the local geometry when much of a robot environment often shares similar local geometric characteristics. Furthermore, to handle the large number of points in one cloud, local descriptors are always calculated only on selected local key points. However, the detection of distinctive key points with high repeatability is still a challenging problem in 3D point cloud analysis.

In addition to bags-of-words, other global descriptors exist such as Viewpoint Feature Histogram (VFH) [12], Clustered Viewpoint Feature Histogram (CVFH) [13], Ensemble of Shape Functions (ESF) [14] and Small-Sized Signatures (Z-projection) [15]. Compared with the bags-of-words method, these global descriptors that do not require local key point detection have a lower complexity in descriptor computation.

In this paper, we propose a novel global point cloud descriptor, named multiview 2D projection or M2DP, for describing a 3D point cloud and we apply it to the problem of lidar-based loop closure detection. Unlike most existing methods that perform analysis on a point cloud in the 3D space, we first project the 3D points to a series of 2D planes representing different viewpoints of the cloud. By characterizing the projection of the points in terms of its spatial density distribution on a plane, we obtain many such density distributions or signatures of a single cloud. We then use the first left and right singular vectors of these 2D signatures as the descriptor.

The rest of this paper is organized as follows. Section II briefly reviews some related works. Section III introduces the proposed multiview 2D projection method. Section IV presents experimental results of the proposed descriptor in the loop closure detection task. Conclusion is drawn in Section V.

II. RELATED WORKS

Point cloud descriptors, describing either local or global shape of a point cloud, can be categorized into two classes, *signatures* and *histograms* [8]. Signatures split the support region of a point into indexed bins by defining an invariant local reference axis or frame. Inside each bin a signature method computes one or more geometric measurements, such as the number of points, the normals, and encode the information from that bin. Histograms generate counts of

*This work was supported by NSERC through the NSERC Canadian Field Robotics Network (NCFRN), and by the Frontier and Key Technology Innovation Special Funds of Guangdong Province (Grant No. 2016B090910003).

¹Li He and Hong Zhang are with Department of Computing Science, University of Alberta, Edmonton, AB, Canada lhe2@ualberta.ca, hzhang@ualberta.ca

²Xiaolong Wang is with the Department of Applied Mathematics, School of Science, Northwestern Polytechnical University, Xian 710072, PR China xwangnwpu@mail.nwpu.edu.cn

the trait values on each point or a subset of points, and concatenate these counts the descriptor.

One of the first algorithms among signature methods is Structural Indexing (SI) [18]. SI constructs a representation from either a 3D curve or a splash according to the 3D support. The former generates the curve from points and encodes the angles between consecutive segments of the edges of the curve, while the latter takes into account the local distribution of surface orientations along a circle. The 3D SURF [19] extends the popular 2D SURF descriptor to 3D data by voxelizing the 3D mesh and defining the saliency of each voxel by the Haar wavelet response.

More recently introduced descriptors mostly fall under the histogram class. Spin Image [9] employs a cylinder support around a keypoint. This cylinder is then divided radially and vertically into volumes, and the number of points lying inside each volume is counted. VFH [12] consists of two steps in its construction. First VFH finds the viewpoint direction to ensure rotation invariance. Then, VFH counts the angles between points normal to this direction and bins angles into an histogram. To improve the robustness of VFH against occlusion, CVFH [13] first divides the whole cloud into smooth regions. Then, CVFH uses the region's average normal and centroid to compute the VFH descriptor. Small-Sized Signatures [15] is a simple method focusing specially on the problem of loop closure detection with lidar data. It first calculates the normals of all points, and then bins the components along the z-axis of normals into a histogram as the descriptor. VFH, CVFH and Small-Sized Signatures all require a pre-processing step to compute the normals of all points. The ESF [14] method avoids the normal calculation by using only shape properties - distances, angles and areas - in defining a descriptor. ESF uses a voxel grid to approximate the real surface, iteratively samples three points and computes the shape properties. SHOT [8] can be seen as a hybrid of signature and histogram methods. SHOT splits a sphere that is centered at a keypoint into several bins, and collects the histogram of normal angles in each bin to build the descriptor. Finally, 3D-Normal Distributions Transform (NDT) [24][25] divides the 3D space into a regular grid of cells. Each cell is classified into one of the three classes, linear, planar or spherical, according to the covariance matrix of that cell. By counting the number of cells that belong to one class, NDT takes the histogram of classes as the descriptor of one point cloud.

Most existing methods utilize normals of points in constructing their 3D descriptors. For point clouds with noisy data, it is often difficult to obtain an accurate normal for one point. For normal-free methods, such as Spine Image or ESF, the lack of spatial information in these descriptors prevents them from capturing intricate details in different clouds. In this paper, our goal is to build an efficient and accurate descriptor for 3D point clouds that is easy to compute and able to capture local geometric details of a point cloud, two properties that are important in loop closure detection.

III. MULTIVIEW 2D PROJECTION ALGORITHM

Although there are many local/global descriptors which have been successfully employed in cloud-based object recognition/detection tasks, it is still a challenge to strike a good balance between accuracy and efficiency for solving loop closure detection task by a 3D descriptor.

A. Algorithm Overview

Our point cloud descriptor is of the signature type. To gain a brief understanding of the proposed method, we start from a two-view example to explain M2DP and then extend the idea to more than two views. Consider a 3D point cloud P and two projection 2D planes X and Y and denote the projected points of P onto X and Y as P_x and P_y , respectively. With the assumptions that X and Y are not parallel and that no occlusion occurs in projection, we are able to reconstruct P from P_x and P_y and the angle between X and Y . By deriving a signature of P_x as v_x and one of P_y as v_y , we can represent P by a signature matrix $A = [v_x^T \ v_y^T]^T$. Thus, we can match two point clouds by comparing their signature matrices. By increasing the number of projection planes, we obtain an enlarged signature matrix A . To achieve a compact signature, we can employ SVD for dimensionality reduction. In our current implementation, we use the first left and right singular vectors of the signature matrix A as a concise description of a point cloud. Details of the M2DP algorithm are described in the next three subsections, and the M2DP algorithm is summarized in Fig. 1.

B. Pre-processing of a Point Cloud

In loop closure detection, it is essential for the descriptor to be invariant to shift and rotation in 3D space. To ensure shift-invariance, we first use the centroid of the input point cloud as the origin of the descriptor reference frame. This approach can be found in many global descriptors, such as VFH and CVFH.

Once the point cloud is shifted with a zero mean, we further perform PCA on the points, and define both x-axis and y-axis of the descriptor reference frame by the first and second PCs. This step assumes that each point cloud has two dominant directions, and it allows us to align coarsely the two point clouds being matched along these dominant directions. This assumption is easily satisfied in our application of loop closure detection. Thus, we use PCA to align input clouds to achieve rotation invariance.

C. 2D Signature of a Single View

After pre-processing, we move on to generate several 2D signatures, each from the projection of the 3D points to a plane. We first define a 2D plane X with normal vector m and require X to pass through the origin. Since the normal vector m is characterized by its azimuth angle θ and elevation angle ϕ , the pair of parameters $[\theta, \phi]$ will lead to a unique X . We project the cloud, as well as the centroid and the x-axis in 3D space, onto X with parameters $[\theta, \phi]$.

To capture the structure of points on X , we divide the 2D plane into bins. Centered at the projected centroid, we first

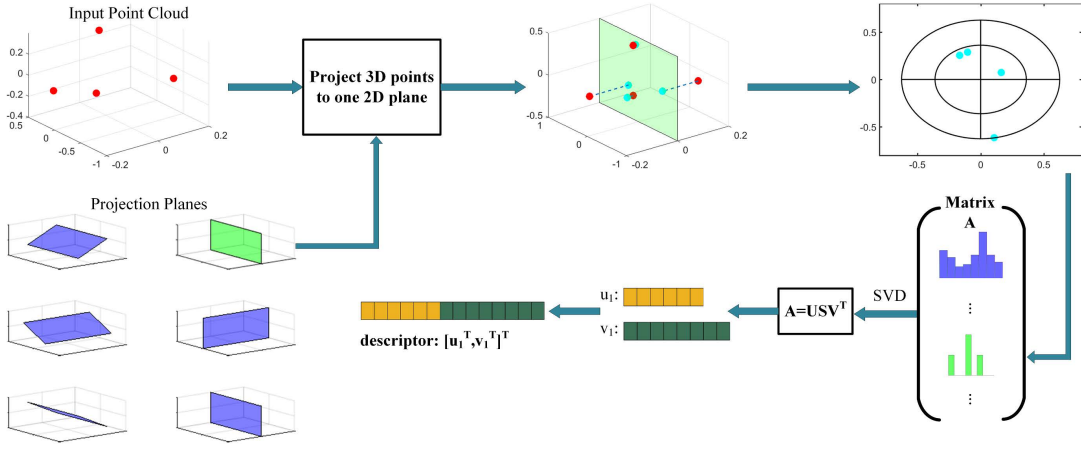


Fig. 1: Framework of the proposed M2DP method

Algorithm 1 M2DP Algorithm

INPUT: Point cloud P , number of concentric circles l , number of bins for each circle t , number of 2D planes along the azimuth p , and number of 2D planes along the elevation q .

OUTPUT: A $(pq + lt)$ -vector d for point cloud P

- 1: Compute the centroid of point cloud P , $\bar{P} = \text{mean}(P)$.
 - 2: $P \leftarrow P - \bar{P}$.
 - 3: Compute the PCA of P . Align the x-axis with the first PC. Align the y-axis with the second PC. Initialize a $pq \times lt$ matrix A with zero values.
 - 4: **for** $\theta = 0$ to π **do**
 - 5: **for** $\phi = 0$ to $\frac{\pi}{2}$ **do**
 - 6: Construct a 2D plane X with normal vector express m as a function of θ and ϕ algebraically, i.e., $m = [\cos \theta \cos \phi, \cos \theta \sin \phi, \sin \theta]^T$.
 - 7: Project P and x-axis to X . Denote the projected points as P_x and the x-axis also as x-axis. For each point u_x in P_x and its corresponding point u in P , $u_x = u - \frac{u^T m}{\|m\|_2^2} m$.
 - 8: Generate the bins on X as shown in Fig. 2 with parameters l and t .
 - 9: Count the number of points of P_x in each bin. Use the counts to form a $lt \times 1$ signature vector v_x .
 - 10: Augment A by a row with v_x .
 - 11: $\phi + \frac{\pi}{2q} \rightarrow \phi$.
 - 12: **end for**
 - 13: $\theta + \frac{\pi}{p} \rightarrow \theta$.
 - 14: **end for**
 - 15: Run SVD on A , $A = USV^T$. Concatenate the first left and right singular vectors as d , $d = [u_1^T \ v_1^T]^T$.
-

generate l concentric circles, with the radii $[r, 2^2r, \dots, l^2r]$, and set the maximum radius to be identical to the distance between the centroid and the farthest point (see Fig. 2). We then divide each ring into t bins, and index these bins according to the x-axis. This way of dividing a plane generates a total of $l \times t$ bins.

For every bin, similar to Spin Image, we simply count the number of points lying inside that bin. Thus we have an $lt \times 1$ signature vector v_x to describe the projection of the 3D point cloud on X , where $\sum_{i=1}^{lt} v_x(i) = n$ holds.

There are several benefits of using the proposed 2D descriptor, such as efficiency in computation and accuracy in description as the result of avoiding the computation of surface normals, a process that is prone to noise in input data. Another point-count-based descriptor similar to ours is Spin Image. Spin Image is designed to be a local shape descriptor where a grid is built near a keypoint, and each cell in this grid holds the weighted sum of nearby points. Since the number of points in Spin Image is linearly interpolated and there is only one grid in producing the descriptor, much spatial information of a point cloud is lost through the description process of its Spin Image.

Compared with many normal-based local/global descriptors, we do not need the calculation of point normals. The

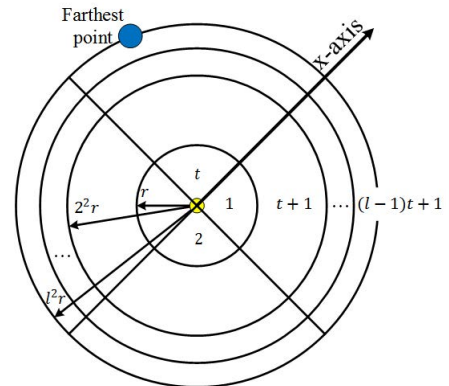


Fig. 2: Illustration of bins in 2D plane

normal-based local/global descriptors have been successfully applied to shape matching problems. However, the computational cost of normals on a large number of points is costly in computation, and most normal-based descriptors often use a downsampling step to limit the number of points. One problem of this downsampling step is the risk of loss of accuracy in representing the cloud. The advantage of our method that count point numbers projected to multiple planes gives it the ability for an accurate description of the source point cloud, and it also has very low computational cost compared with those that require calculating the normals.

D. Multiview 2D Projection Descriptor

We produce multiple 2D planes by employing p different azimuth angle θ s and q different elevation angle ϕ s, where the stride on azimuth is $\frac{\pi}{p}$ and that on elevation $\frac{\pi}{2q}$. The total number of 2D planes hence is pq . For each 2D plane, we generate a 2D signature described in the previous section, which is a $lt \times 1$ vector. We build the $pq \times lt$ matrix A whose each row corresponds to one 2D signature, to represent the cloud. We further run SVD on A , and concatenate the first left and right singular vectors as our final descriptor.

IV. EXPERIMENTAL EVALUATION

In this section, the proposed method is validated through experiments. We compare our proposed M2DP with five methods: the global descriptor VFH [12], ESF [14], Z-projection [15], the local descriptor SHOT [8] and Spin Image [9]. The proposed M2DP is implemented in Matlab and C¹. The Z-projection method is written in Matlab, while we use the C++ implementation of VFH, ESF, SHOT and Spin Image in the Point Cloud Library (PCL)². Our platform running these experiments is a laptop equipped with an Intel quad-core 2.50 GHz CPU and an 8 GB RAM.

In general, there are two ways to use a local descriptor, such as SHOT, to represent an entire point cloud. The first is the bags-of-words method, which uses the histogram of vector-quantized local descriptors as the global descriptor. The second considers the whole cloud as the support of one reference point, such as the centroid, and computes the local descriptor with respect to this point. In our experiments, we adopt the second way.

A. Dataset and Experimental Settings

We evaluate the proposed method using lidar data sequences in KITTI³ [22], Freiburg Campus⁴ and Ford Campus⁵ [21]. We use sequences 00, 05, 06 and 07 in KITTI dataset. The 06 sequence (KITTI06) is a narrow oval-shaped loop, with a total of 1101 point clouds. The number of point clouds in 00, 05 and 07 sequences are 4541, 2761 and 1101, respectively. Freiburg Campus consists of 77 point clouds and Ford Campus dataset contains 3817 point clouds. Two

locations are considered as the ground truth loop closure if their distance is less than 10m.

For each point cloud, we calculate its descriptor and search for the nearest neighbour as the matching candidate. The ± 50 (± 5 for Freiburg Campus) frames before/after the current frame are excluded from the matching process to avoid matching neighboring point clouds. We then vary a distance threshold on the nearest neighbour distance to determine the match. If the distance between one frame and its nearest neighbour is less than the threshold, we take this match as loop closure. Using this threshold, the recall and precision curve is generated to compare the performance of different methods.

We compare the proposed M2DP with six other competing methods, i.e., Spin Image, SHOT, VFH, ESF, Z-projection and GIST [23]. GIST is employed here as a visual-based benchmark descriptor for comparison. GIST is absent in Freiburg Campus test because no visual image is provided in that dataset.

All the surface normal-based methods require the parameter of the radius to calculate normals. In our experiments, this value is set to be $5 \times res$, where res is the average distance of points to their nearest neighbours in the first frame. Spin Image requires three parameters: the radius to compute the normal, the radius of the cylinder and the number of bins. We set the cylinder radius to cover the farthest point, which turns the local descriptor into a global one. We use 8 bins in Spin Image. SHOT requires two parameters: the normal radius and the radius to compute SHOT. Similar to Spin Image, the second parameter is set to cover the whole cloud. VFH only requires the normal radius as the input. Z-projection and the ESF are parameter-free, while in Z-projection we set the number of bins as 202. By using the default parameters, the length of GIST descriptor is 512.

The proposed M2DP method needs to set the number of bins for each 2D plane, or the l (number of circles) and t (bins in one ring), and the number of 2D planes to use, or the p (the azimuth) and q (the elevation). In our experiments, we optimize parameters in M2DP on KITTI07 sequence and fix the values as $l = 8$, $t = 16$, $p = 4$ and $q = 16$ for all tests. This leads to a 64×128 matrix A and the final descriptor is a 192 vector.

B. Experimental Results

In the first experiment, we employ the raw point clouds as the input to verify the performance of competing methods. The recall-precision curves for different methods are plotted in Fig. 3. In addition, the recall values at 100% precision of all methods are shown in Tab. I. Tab. II shows respectively the time costs of generating descriptor and nearest neighbour searching of all methods on KITTI00 sequence.

The surface normal-based descriptors, VFH, SHOT and Spin Image, always incur a heavy computational burden for a large cloud. As suggested in PCL, one popular solution is to downsample an point cloud. We split the input 3D space of KITTI06 sequence into several grids and in one

¹Source code of M2DP: <https://github.com/LiHeUA/M2DP>

²<http://pointclouds.org>

³http://www.cvlibs.net/datasets/kitti/eval_odometry.php

⁴<http://ais.informatik.uni-freiburg.de/projects/datasets/fr360>

⁵<http://robots.engin.umich.edu/SoftwareData/Ford>

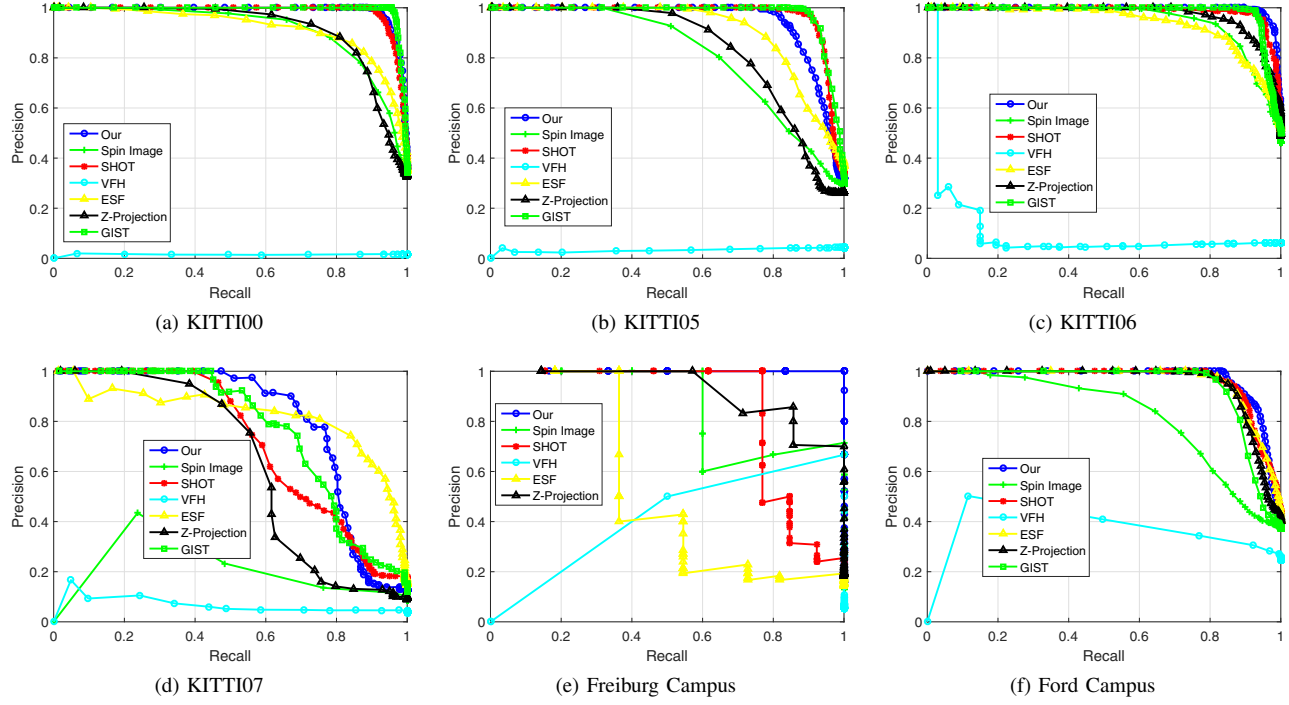


Fig. 3: Recall-Precision curves on KITTI, Freiburg Campus and Ford Campus datasets

TABLE I: Recall at 100% precision on KITTI, Freiburg Campus and Ford Campus datasets

	Spin Image	SHOT	VFH	ESF	Z-projection	M2DP	GIST
KITTI00	0.271	0.898	0	0.174	0.456	0.896	0.957
KITTI05	0.510	0.816	0	0.557	0.513	0.764	0.855
KITTI06	0.606	0.666	0.029	0.233	0.737	0.899	0.885
KITTI07	0	0.466	0	0.098	0.383	0.510	0.451
Freiburg Campus	0.6	0.769	0	0.363	0.714	1	-
Ford Campus	0.177	0.798	0	0.774	0.702	0.838	0.789

TABLE II: Time costs of calculating descriptors and searching loop closures on KITTI00 for each point cloud in seconds

	Calculating descriptor	Searching loop closure
Spin Image	1.4549	0.0052
SHOT	1.5279	0.0113
VFH	1.5086	0.0101
ESF	0.7851	0.0177
Z-projection	0.4943	0.006
M2DP	0.3598	0.0057

grid we employ the mean of points in that grid as the representative. To compare the robustness of competing methods with respect to different downsample sizes, in the second experiment, we set the grid size as $res \times [10, 20, 50, 100]$ and run all competing methods on these downsampled point clouds. The recall-precision curves are shown in Fig. 4 and the corresponding recall at 100% precision values are listed in Tab. III.

The third experiment focuses on the robustness of all methods against noise. In this experiment, we add uniform noise in $[-range, range]$ to each point location in KITTI07

sequence, where $range = res \times [0.01, 0.05, 0.1, 0.2, 0.5]$. We use the relative descriptor error, err , to measure the robustness of all competing methods. Denote the descriptor of clean data as d_{clean} and that of noise data as d_{noise} . Then, $err = \|d_{clean} - d_{noise}\|_2 / \|d_{clean}\|_2$. Fig. 5 (a) shows the relative errors and Fig. 5 (b) illustrates the corresponding recall at 100% precision values.

From Tab. 1-3 and Fig. 3-5, we can observe the followings:

1) The performance of VFH and Z-projection methods is poor. Since VFH builds the histogram on the angles between point normals and the centroid direction, it only takes the angles into account, and the information from point locations is hence lost. If the pdf's of normal vectors in different clouds are similar, as in the case of lidar clouds, the absence of spatial information may lead to non-discriminating VFH descriptors. As a consequence, both the recall and precision values of VFH are very low in our evaluation. A similar problem can be found in the Z-projection method, in which the z-component in normal vectors is not informative enough for matching. ESF and Spin Image methods both show relatively good results in our experiments. However, for loop closure detection, the recall at 100% precision of both

TABLE III: Recall at 100% precision with different downsampling sizes, KITTI06

Grid size	Spin Image	SHOT	VFH	ESF	Z-projection	M2DP
$\times 10$	0.154	0.945	0	0.071	0.007	0.996
$\times 20$	0.109	0.834	0	0.037	0.008	0.996
$\times 50$	0.047	0.415	0	0.056	0.023	0.949
$\times 100$	0.021	0.087	0	0.005	0.027	0.718

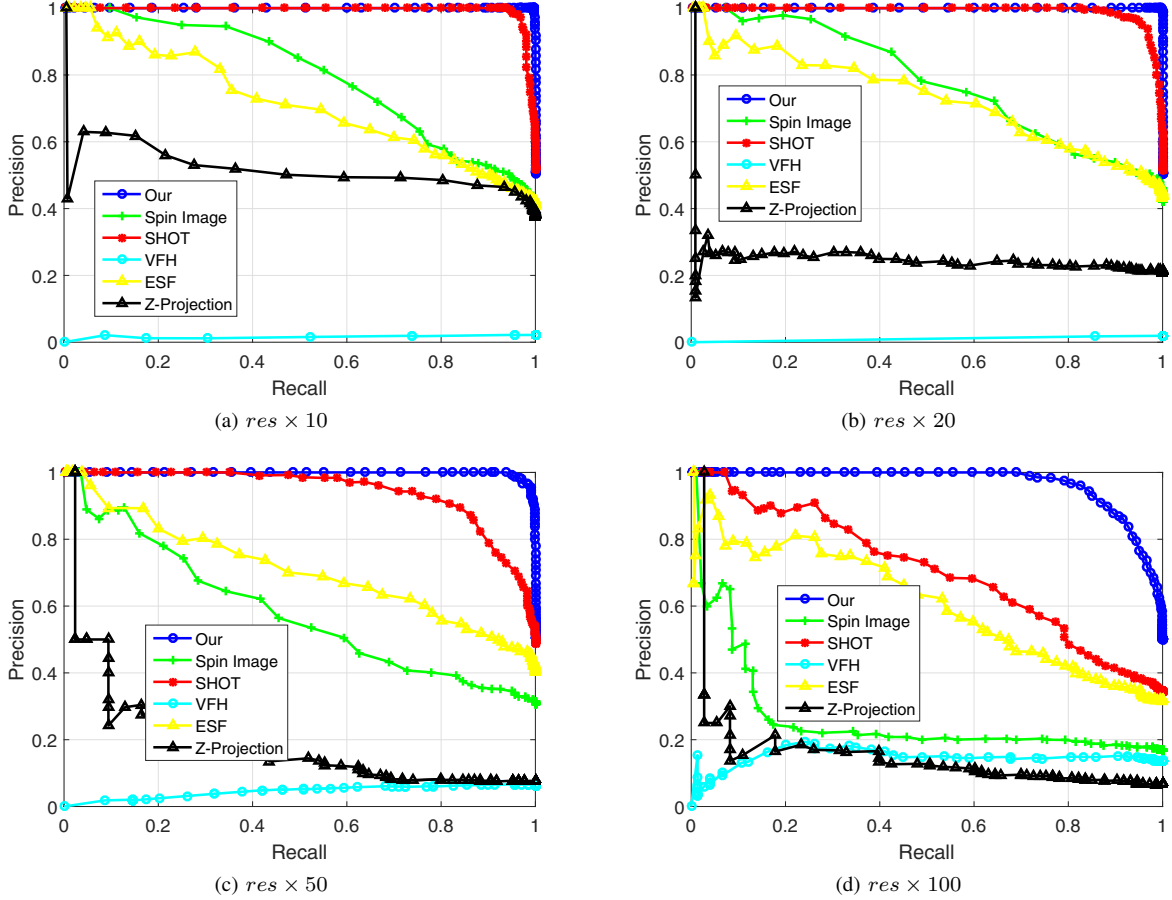


Fig. 4: Recall-Precision curves on KITTI06 with different downsampling grid sizes

methods is not competitive with respect to SHOT and our method.

2) SHOT shows good result as can be seen in Fig. 3, giving a healthy balance in recall and precision. However, the recall at 100% precision of the SHOT descriptor highly depends on the cloud size, as shown in Tab. III. There is a notable loss of precision if the downsampling grid size increases from $res \times 10$ to $res \times 100$ in KITTI06 sequence. To use SHOT, therefore, it is important to select the downsampling size carefully to obtain a good balance between efficiency and accuracy.

3) The proposed M2DP method shows its advantages in matching point clouds among all competing methods especially with small downsampled cloud sizes. On KITTI06 with grid size of $res \times 50$, M2DP detects most of the loop closures (recall of 94.9%) with no false alarm (precision of unity). The proposed method also shows its robustness against noise

in KITTI07. Finally, the computational time of generating M2DP descriptor is the least among all the competing 3D descriptors. Compared with the visual benchmark descriptor of GIST, M2DP shows comparable performance in loop closure detection tasks.

V. SUMMARY

In this paper, we have proposed a novel global descriptor, named M2DP, for a 3D point cloud, and applied it to lidar-based loop closure detection. The M2DP descriptor is built from the projection of the 3D point cloud to multiple 2D planes and computation of the signatures of the cloud on these planes before SVD is applied to reduce the dimensions of the final descriptor. Experiments on the benchmark datasets KITTI, Freiburg Campus and Ford Campus verify that the proposed method is superior to existing 3D point cloud descriptors in both accuracy and computational ef-

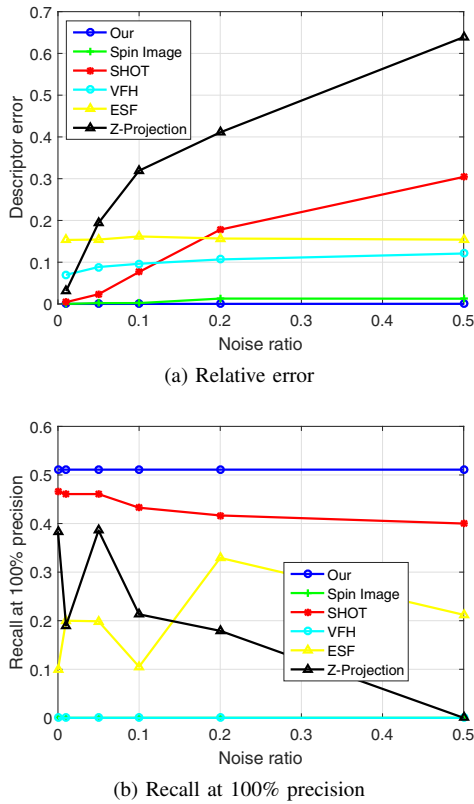


Fig. 5: Robustness of competing methods against noise, KITTI 07

iciency. This superiority is particularly pronounced when sparse point clouds are involved. The ability to handle sparse point clouds is an important advantage when inexpensive lidar is used with low spatial resolution or if computational time is a major concern. In the future, we will examine M2DP's applicability to other types of depth data such as RGB-D and depth map from stereo vision.

VI. ACKNOWLEDGEMENT

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) through the NSERC Canadian Field Robotics Network (NCFRN), and by the Frontier and Key Technology Innovation Special Funds of Guangdong Province (Grant No. 2016B 090910003).

REFERENCES

- [1] Besl, Paul J., and Neil D. McKay. "Method for registration of 3-D shapes." In *Robotics-DL tentative*, pp. 586-606. International Society for Optics and Photonics, 1992.
- [2] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *International Conference On 3-D Digital Imaging and Modeling*, 2001.
- [3] Segal, Aleksandr, D. Haehnel and S. Thrun, "Generalized-ICP," in *Robotics: Science and Systems*, vol. 2, no. 4. 2009.
- [4] Aiger, Dror, J. M. Niloy and C. Daniel, "4-points congruent sets for robust pairwise surface registration," in *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, p. 85. ACM, 2008.
- [5] Mellado, Nicolas, D. Aiger and N. J. Mitra, "Super 4PCS Fast Global Pointcloud Registration via Smart Indexing," in *Computer Graphics Forum*, vol. 33, no. 5, pp. 205-215. 2014.
- [6] Wang, X., Zhang, H. and Peng, G., "3-DOF point cloud registration using congruent triangles". In *Intelligent Robots and Systems (IROS)*, 2015 IEEE/RSJ International Conference on, pp. 1943-1948.
- [7] Krishnan, K. Aravindhan and S. Saripalli, "Point cloud registration using congruent pyramids," in *Intelligent Robots and Systems (IROS 2014)*, 2014 IEEE/RSJ International Conference on, pp. 1812-1817. IEEE, 2014.
- [8] Tombari, Federico, Samuele Salti, and Luigi Di Stefano. "A combined texture-shape descriptor for enhanced 3D feature matching." In *Image Processing (ICIP)*, 2011 18th IEEE International Conference on, pp. 809-812. IEEE, 2011.
- [9] Johnson, Andrew E., and Martial Hebert. "Using spin images for efficient object recognition in cluttered 3D scenes." *Pattern Analysis and Machine Intelligence*, IEEE Transactions on 21, no. 5 (1999): 433-449.
- [10] Sun, Jian, Maks Ovsjanikov, and Leonidas Guibas. "A Concise and Provably Informative Multi-Scale Signature Based on Heat Diffusion." In *Computer graphics forum*, vol. 28, no. 5, pp. 1383-1392. Blackwell Publishing Ltd, 2009.
- [11] Inaba, Mary, and Naoki Katoh. "Variance-based k-clustering algorithms by Voronoi diagrams and randomization." *IEICE Transactions on Information and Systems* 83, no. 6 (2000): 1199-1206.
- [12] Rusu, Radu Bogdan, Gary Bradski, Romain Thibaux, and John Hsu. "Fast 3d recognition and pose using the viewpoint feature histogram." In *Intelligent Robots and Systems (IROS)*, 2010 IEEE/RSJ International Conference on, pp. 2155-2162. IEEE, 2010.
- [13] Aldoma, Aitor, Markus Vincze, Nico Blodow, David Gossow, Suat Gedikli, Radu Bogdan Rusu, and Gary Bradski. "CAD-model recognition and 6DOF pose estimation using 3D cues." In *Computer Vision Workshops (ICCV Workshops)*, 2011 IEEE International Conference on, pp. 585-592. IEEE, 2011.
- [14] Wohlkinger, Walter, and Markus Vincze. "Ensemble of shape functions for 3d object classification." In *Robotics and Biomimetics (ROBIO)*, 2011 IEEE International Conference on, pp. 2987-2992. IEEE, 2011.
- [15] Muhammad, Naveed, and Simon Lacroix. "Loop closure detection using small-sized signatures from 3d lidar data." In *Safety, Security, and Rescue Robotics (SSRR)*, 2011 IEEE International Symposium on, pp. 333-338. IEEE, 2011.
- [16] Frome, Andrea, Daniel Huber, Ravi Kolluri, Thomas Blow, and Jitendra Malik. "Recognizing objects in range data using regional point descriptors." In *Computer Vision-ECCV 2004*, pp. 224-237. Springer Berlin Heidelberg, 2004.
- [17] Mian, Ajmal S., Mohammed Bennamoun, and Robyn A. Owens. "A novel representation and feature matching algorithm for automatic pairwise registration of range images." *International Journal of Computer Vision* 66, no. 1 (2006): 19-40.
- [18] Stein, Fridtjof, and Grard Medioni. "Structural indexing: Efficient 3-D object recognition." *IEEE Transactions on Pattern Analysis & Machine Intelligence* 2 (1992): 125-145.
- [19] Knopp, Jan, Mukta Prasad, Geert Willems, Radu Timofte, and Luc Van Gool. "Hough transform and 3D SURF for robust three dimensional classification." In *Computer Vision/ECCEV 2010*, pp. 589-602. Springer Berlin Heidelberg, 2010.
- [20] Bronstein, Alexander M., Michael M. Bronstein, Leonidas J. Guibas, and Maks Ovsjanikov. "Shape google: Geometric words and expressions for invariant shape retrieval." *ACM Transactions on Graphics (TOG)* 30, no. 1 (2011): 1.
- [21] Pandey, G., McBride, J. R., and Eustice, R. M. "Ford campus vision and lidar data set." *The International Journal of Robotics Research* 30, no. 13 (2011): 1543-1552.
- [22] Geiger, A., Lenz, P. and Urtasun, R. "Are we ready for autonomous driving? the kitti vision benchmark suite." In *Computer Vision and Pattern Recognition (CVPR)*, 2012 IEEE Conference on pp. 3354-3361. IEEE.
- [23] Oliva, A. and Torralba, A. "Modeling the shape of the scene: A holistic representation of the spatial envelope." *International Journal of Computer Vision*, 2001, 42(3), pp.145-175.
- [24] Magnusson, M., Andreasson, H., Nüchter, A. and Lilienthal, A.J. "Automatic appearance-based loop detection from three-dimensional laser data using the normal distributions transform." *Journal of Field Robotics*, 2009, 26(11-12), pp.892-914.
- [25] Stoyanov T, Magnusson M and Lilienthal A.J. "Fast and accurate scan registration through minimization of the distance between compact 3D-NDT representations." *The International Journal of Robotics Research* 2012, 31: 1377-1393.