3D Keypoint Repeatability for Heterogeneous Multi-Robot SLAM

Elizabeth R. Boroson and Nora Ayanian

***Abstract*— For robots with different types of sensors, loop closure in a multi-robot SLAM scenario requires keypoints that can be matched between sensor measurement point clouds with different properties such as point density and noise. In this paper, we evaluate the performance of several 3D keypoint detectors (Harris3D, ISS, KPQ, KPQ-SI, and NARF) for repeatability between scans from different sensors towards building a heterogeneous multi-robot SLAM system. We find that KPQ-SI and NARF have the best relative repeatability, with KPQ-SI finding more keypoints overall and a higher number of repeatable keypoints, at the cost of significantly worse computational performance. In scans of the same area from different poses, both detectors find enough keypoints for point cloud registration and loop closure. For heterogenous multi-robot SLAM applications with computational or bandwidth restrictions, the NARF detector consistently finds repeatable keypoints while also allowing for real-time performance.**

I. INTRODUCTION

In multi-robot Simultaneous Localization and Mapping (SLAM), a group of robots explore and map an unknown area. Each robot may localize on its own and build its own map, but it is more efficient for their maps to be combined so each robot only need explore part of the area. Additionally, point cloud registration can improve an individual robot's localization using shared information, as it can observe other robots or use their observations to constrain its own pose.

Many recent approaches for multi-robot SLAM improve localization for a robot team [1]–[4], but most consider homogeneous groups, meaning all robots have the same sensors. Thus, detecting that two robots are observing the same scene is similar to detecting single-robot loop closure.

For heterogeneous robots with different sensors, these shared observations are much more difficult to detect. Consider a scenario where large ground vehicles map an outdoor region with the aid of small Unmanned Aerial Vehicles (UAVs). The ground vehicles may have high-quality Inertial Measurement Units (IMUs) and heavy, expensive sensors such as LIDAR, which provide an accurate point cloud of the scene but no visual information. The UAVs are too small to carry a LIDAR, but have lightweight, low-quality IMUs and one or more small cameras. They can reconstruct a point cloud using stereo vision, but it has very different error properties to the LIDAR scans. Ideally, the UAVs initially explore to construct a coarse map, then important areas of the map are refined using ground vehicles. Meanwhile, the UAVs use their observations of areas already mapped by the ground vehicles to reduce drift and refine their pose history.

Authors are with the Department of Computer Science, University of Southern California, USA {boroson, ayanian}@usc.edu. This work was supported by a NASA Space Technology Research Fellowship and by ARL DCIST CRA W911NF-17-2-0181.

Finding correspondences between LIDAR scans and stereo cameras is challenging since LIDAR does not provide imagery. While both can provide 3D point clouds (as can other sensors like RGB-D cameras), the point clouds generated have different resolutions and error models, which make dense methods for data association challenging (see Fig. 1). However, correspondences can be found by matching keypoints observed in all types of point clouds. These keypoints are identified using keypoint detectors, or methods to select unique and repeatable points in the point cloud.

Once keypoints have been detected in point clouds, using them for multi-robot SLAM would require identifying matches, or pairs of keypoints corresponding to the same physical location. These matches are typically found using descriptors, which uniquely describe each selected point. Several types of descriptors exist for keypoints in point clouds, including descriptors associated with keypoint detectors (e.g., NARF [5] and ISS [6]), and standalone descriptors (e.g., Spin Images [7] and variations on Point Feature Histograms [8], [9]). Alexandre [10] and Guo *et al.* [11] evaluated 3D descriptors on point clouds from various sensors, and found that all descriptors worked significantly better when applied to detected keypoints rather than a uniform sampling. However, this descriptor matching approach assumes that those keypoints are the same between point clouds. Thus, before detected keypoints can be matched and used in SLAM, it is necessary to identify keypoint detectors with this property, which is the focus of this paper.

The contribution of this paper is an evaluation of 3D keypoint detectors for loop closure in heterogeneous multi-robot SLAM. Specifically, we seek a keypoint detector that:

1) Finds repeatable keypoints across different sensor types;
2) Finds repeatable keypoints in observations made from different poses;
3) Detects relatively few high-quality keypoints that can be shared even with low-bandwidth communication; and
4) Detects keypoints efficiently from LIDAR and stereo camera point clouds to enable real-time SLAM.

To the best of our knowledge, this is the first comparison of 3D keypoint detectors for SLAM or visual odometry.

We evaluate 5 commonly used 3D keypoint detectors for these properties using point clouds from LIDAR and stereo camera scans in the KITTI dataset [12], which are representative of measurements made by mobile robots in SLAM applications. We evaluate repeatability of each detector, measuring how likely it is that a keypoint detected in a scan from one sensor will be found in a scan from the other sensor, and we consider scans from the same pose and from different nearby poses. Overall, our evaluations

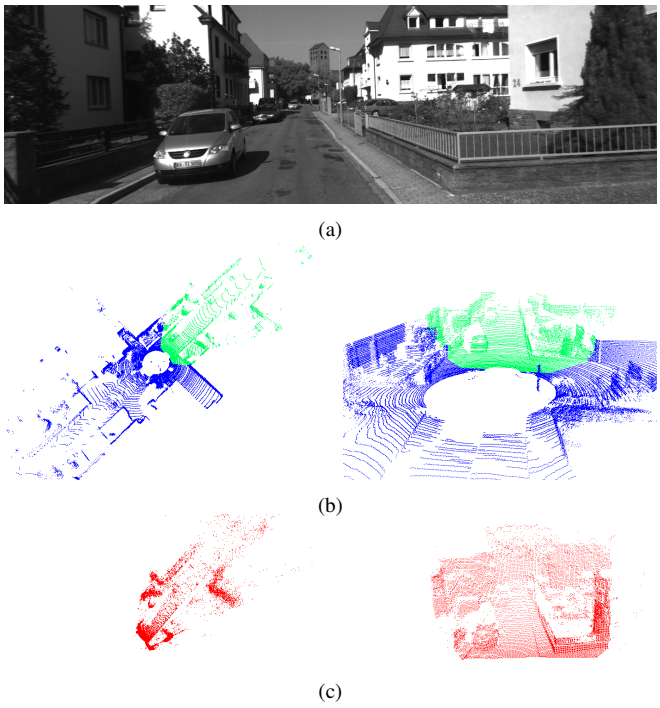

Fig. 1. Several views of frame 2181 from track 0 of the KITTI data. (a) Image from the left camera. (b) Point cloud from LIDAR scan, shown from two viewpoints.The area also visible to the stereo camera is shown in green. (c) Point cloud from stereo camera images, shown from two viewpoints. The different properties of the two sensors are visible in the point clouds. LIDAR scans are less dense, but they have little noise and a 360-degree field of view. The camera images have a denser view of part of the scene, but also significant noise along the axis toward the camera. They also have erroneous points in the "sky," which can challenge keypoint detectors.

indicate that NARF [5] and KPQ-SI keypoints [13] have the best relative repeatability. The KPQ-SI detector finds many more keypoints, but has significantly worse computational performance on all scans. We conclude that both detectors can work well, but the NARF keypoint detector is the best choice if computational power or bandwidth is limited.

II. 3D KEYPOINT DETECTION

In applications where information is extracted from high-dimensional sensor measurements such as images or LIDAR scans of the environment, it can be helpful to use *keypoints* rather than considering the entire scan, which may contain thousands of points. Keypoints are points that contain most of the relevant information from the measurement while reducing the data that must be stored to a manageable size.

In feature-based SLAM methods, robot poses are reconstructed using observations of those keypoints in several images and the correspondences between them. Thus, an important requirement for feature-based SLAM is features that are both repeatable, meaning they are found in the same place over many scans, and unique. It is common to first extract repeatable keypoints, then compute a unique descriptor for each keypoint and use this descriptor to find correspondences between images. With visual sensors, such algorithms often use FAST or ORB keypoints [14]–[17].

Several types of keypoints exist for 3D data, including object models or point clouds from 2.5D sensors (which make depth measurements from a single viewpoint, e.g., LIDAR and stereo or RGB-D cameras) [5], [6], [13], [18]–[22]. Many were developed for object recognition, where a 2.5D scan is matched against a known 3D model of the object. In contrast, in feature-based SLAM, correspondences are computed using features from individual scans. To this end, we selected 5 keypoint detection methods, which we now describe in detail, to compare based on the requirements described above and their performance in other studies. Other methods we considered are also briefly described.

A. Extensions from Image Keypoints

Common 2D keypoint detectors evaluate functions of the image intensity. Some of those methods can be adapted to 3D point clouds, either by evaluating the same expressions over volumetric data or by computing surface normals and evaluating them over the surface. A volumetric model of the scene cannot be accurately reconstructed from a 2.5D single scan. However, normals can be computed, so methods evaluated over surfaces are particularly applicable.

One such keypoint type is Harris corners [23]. In 2D, Harris corners are defined at the maxima of an autocorrelation function that describes how similar a patch of the image is to a slightly shifted copy of itself. This is an efficient way to detect corners, as patches containing them vary rapidly in both directions. In the *Harris3D* detector, a 3-dimensional version of Harris corners, the normal at each point is computed by fitting a plane to nearby points, then the same autocorrelation operator is applied on these normals and peaks are selected as Harris3D keypoints [18].

B. Keypoints from Surface Curvature

Intrinsic Shape Signatures (ISS) are a keypoint and descriptor for a region of a 3D surface. The detector finds keypoints with a unique orientation and neighbors with large surface normal variation in all 3 dimensions [6]. The ISS detector did well in previous comparison studies in efficiency and repeatability over viewpoint changes [24]–[26].

Mian *et al.* [13] define another keypoint detector, which we refer to here as *KPQ (KeyPoint Quality)*. In the KPQ detector, points' neighborhoods are aligned with the local frame and keypoints are selected with the largest ratio in how far that neighborhood patch extends along each axis. This selects points where the surface has much more curvature in one direction than the other. A surface is then fitted to the neighborhood points to compute a quality measure. KPQ uses a fixed neighborhood size, but a scale-invariant method is also described by identifying keypoints at multiple scales. We refer to the scale-invariant method as *KPQ-SI*. KPQ-SI also performed well in previous surveys [24], [25].

C. Range Image Keypoints

Steder *et al.* [5] propose a very different approach. Rather than work with the full 3D point cloud, they reduce the point cloud to a *range image* by projecting it onto a spherical camera plane located at the viewpoint and coloring each pixel by its depth. The range image is a distorted view of the full 360-degree surroundings. If the point cloud is

a 2.5D measurement, like the scans in SLAM scenarios, and the origin used for the projection is the same as the pose where the scan was collected, no information is lost in this projection. The *Normal Aligned Radial Feature (NARF)* keypoint extractor finds points in the range image which are on stable surfaces that are not sensitive to viewpoint changes, but are still unique [5]. Each point in the range image is assigned an interest value, which is high when the point's immediate neighbors (within a fixed neighborhood) have similar normals, but more distant neighbors have large normal variations. Keypoints are selected at maxima of this interest value. A descriptor is also proposed for these keypoints, but we do not evaluate it here.

D. Keypoints we did not evaluate

Our evaluation does not include 3D keypoints that performed poorly with data similar to ours in other studies or that have requirements that are unreasonable for robots in a SLAM system. In particular, the MeshDOG [19], Salient Points [20], Laplace-Beltrami Scale Space [21], and Heat Kernel Signature [22] detectors require a mesh surface to be fitted to the point cloud. This is computationally expensive and slow, particularly for the scans of very large areas used in SLAM. The Local Surface Patches detector [27] identifies points with different surface curvature from their neighborhood, but it performed very poorly on point clouds from laser scans in other comparison studies [24], [25].

III. RELATED WORK

While each keypoint has been analyzed for repeatability and matching performance in the literature, the analyses typically use different data and methods. Several studies have directly compared different types of keypoints. Salti *et al.* [24] and Tombari *et al.* [25] evaluated a variety of keypoint detectors for object recognition over 5 datasets. Each dataset had 3D models of objects and scans of those objects from different viewpoints and in scenes. They compared all keypoint detectors described in Sec. II except the NARF detector and found that KPQ-SI, MeshDOG, and ISS had the best performance. ISS was most efficient, but performed poorly when the object was partially occluded. Filipe and Alexandre [26] evaluated keypoints for object detection in an RGB-D dataset, and also found that ISS keypoints had the best performance. These studies used datasets with only one sensor, so they could not evaluate matching across sensor types, as is required for heterogeneous multi-robot SLAM. Additionally, they evaluated matching of 3D models with 2.5D scans, which applies well to object recognition but less so to SLAM, where models of the scene are unlikely to exist.

Yu *et al.* [28] compared several types of keypoint detectors on volumetric data for object detection. They worked with both synthetic and real data. They extended typical 2D image keypoints like Harris corners and SURF to 3D, and found that MSER features had the best performance but were inefficient to compute. The conversion to occupancy grids and evaluation over those grids can be slow, and occupancy is difficult to determine from a 2.5D scan, so these detectors are not appropriate for SLAM scenarios.

Kostusiak [29] evaluated 2D keypoint detectors and descriptors (e.g., SURF, KAZE, ORB) in RGB-D data for visual odometry. They used the depth channel only for 3D positions of those keypoints, which is common in SLAM with RGB-D cameras [30], [31]. While these detectors perform well in RGB-D SLAM, they require image-based sensors, which may not apply in a heterogeneous group.

Systems that perform feature-based SLAM or visual odometry typically use 2D image keypoints [16], [17]. Most SLAM systems without visual sensors use dense matching between scans or LIDAR-specific methods like SegMatch [3], which are sensitive to changes in the the sensor model. Because point clouds from different kinds of sensors have very different properties, dense matching cannot be used for heterogeneous sensors. To the best of our knowledge, no comparison of 3D keypoints for SLAM or visual odometry currently exists in the literature.

IV. METHODS

As described in Section II, we compared the 3D keypoint detectors that were reasonable for the data and computational limitations in heterogeneous multi-robot SLAM: Harris3D, NARF, ISS, KPQ, and KPQ-SI. We used implementations of Harris3D, ISS, and NARF available in the open-source Point Cloud Library (PCL) [32]. KPQ and KPQ-SI were implemented in C++ to operate on PCL point clouds.

We modified KPQ and KPQ-SI to better apply to SLAM. These methods have two parts: an initial identification step based on the keypoint's neighborhood, followed by a filtering step in which a mesh is fit to that neighborhood and used to determine the keypoint's quality. Surface fitting a set of points is computationally intensive for large point clouds generated by robotic sensors, particularly on embedded processors. Instead, we use all the keypoints identified in the first step, which is equivalent to setting the filtering threshold to 0. Including the selection step results in fewer features with a larger fraction repeatable, but the full implementation may not be feasible to perform on robots in real time. Additionally, the KPQ-SI detector requires evaluation of keypoints at several scales. To limit computational time, we evaluated only 3 scales. Keypoints were detected at all 3 scales. More scales may have resulted in a larger number of keypoints, but significantly worse efficiency.

Many of the detectors we evaluate have performed well in past comparison studies for object recognition, as described in Sec. III. In object recognition, it is important to retain all model information in order to recognize the object from any viewpoint. However, the NARF detector works on a range image from a particular pose, so any parts of the point cloud obscured from that pose are lost. For that reason, the NARF keypoint detector was not evaluated in those comparison studies. In SLAM, however, observations are 2.5D, thus reduction to a range image does not lose any information. Therefore, the NARF keypoint extractor is appropriate for SLAM applications, so it is included in our evaluation.

We evaluate keypoint repeatability for pairs of scans observed using different sensors, from different viewpoints,

or both. Keypoints consist of only a location within the scan, with no other identifying information. Thus, we define keypoints in different scans to be the same point if the Euclidean distance between their locations is less than some search radius r. Additionally, we allow each keypoint from a scan to have only one match in another scan. For example, for keypoint p_i^1 in scan 1, if keypoints p_j^2 and p_k^2 in scan 2 both have distance less than r from p_i^1, with p_j^2 closer, only the one that is closer is considered a match. Thus, p_i^1 and p_j^2 are matched and p_k^2 cannot have p_i^1 as a match.

For each keypoint type, we evaluate both absolute and relative repeatability. Absolute repeatability gives the total number of keypoints repeated between scans. This number depends on the total number of keypoints detected, which may also depend on parameters such as scales and thresholds in the detector. A higher number here may be helpful, as more keypoints will continue to be visible even if part of the scene is obscured. However, a high number may indicate lower quality keypoints, so future matching using other methods such as descriptors may be more difficult.

Relative repeatability is the fraction of keypoints that are repeatable. Since scans from different sensors may have very different numbers of keypoints detected (due to different resolutions or qualities of the underlying data), we compute relative repeatability for both sensor types.

For all experiments, we consider only keypoints in the region visible in both scans.

We do not evaluate the distribution of repeatable keypoints within each scan. This is important when applying this work, such as in a loop closure application, as well distributed matches give better observability of relative pose. If the keypoints are not well distributed, it might be due to inherent properties of the sensors or asymmetries in the environment. For example, points farther from a sensor might have more error, making those keypoints more difficult to detect. Before using SLAM, it would be necessary to calibrate the sensors used to determine any regions of the scans with larger errors and adjust the system parameters accordingly. Bias may also be introduced when some parts of the scene have more keypoints; this is commonly overcome by dividing the scan into regions and searching the regions individually. While these are important concerns in using the detected keypoints, such an analysis of the sensors and environment is outside the scope of this paper.

V. RESULTS

To evaluate keypoint repeatability in data representative of robotic applications, we selected several trajectories from the KITTI odometry dataset [12]. We selected tracks 0, 2, 6, and 9, which gave a large number of frames with varied scenery and a relatively small number of other moving objects observed. Track 0 also allowed for evaluations over different views of the same scene, as it has segments where the same area is observed from multiple orientations.

A. Scans from same viewpoint

For each time step in the data, we have the true pose and scans from each sensor recorded at that time. We use the LIDAR scans recorded from a Velodyne HDL-64E LIDAR, and grayscale images from two Point Grey Flea 2 cameras. Prior to our analysis, some preprocessing was done on the LIDAR data so that all points in each scan are valid at the timestamp of the corresponding pose. Additionally, the images have already been undistorted and rectified.

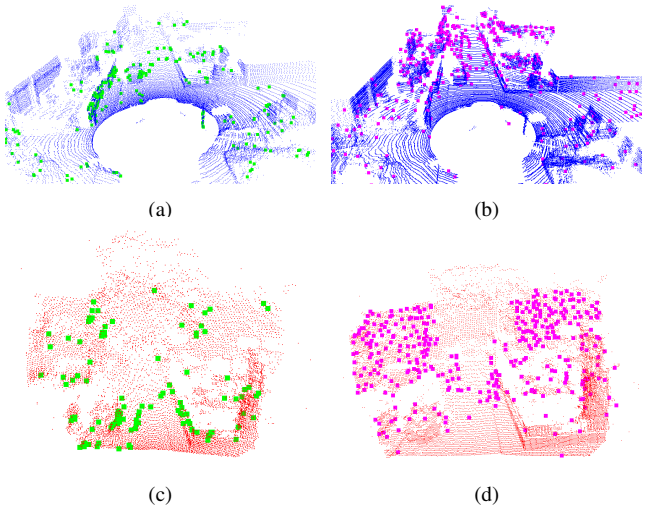

Fig. 2. Examples of detected keypoints in frame 2181 from track 0. (a) NARF keypoints in the LIDAR scan (b) KPQ-SI keypoints in the LIDAR scan (c) NARF keypoints in the stereo camera scan (d) KPQ-SI keypoints in the stereo camera scan

TABLE I

MEAN NUMBER OF KEYPOINTS DETECTED BY EACH METHOD

Scan Type	Harris	ISS	KPQ	KPQ-SI	NARF
LIDAR	92	135	130	250	71
Stereo Camera	173	168	223	318	102

For each track selected, we evaluated the keypoint repeatability between sensors at each frame. To do this, we created point clouds from the stereo camera scans using the OpenCV 3D reconstruction library to compute a disparity map between the two images, then project each pixel to a point in space, with depth determined by the disparity. We then downsampled to a density similar to the LIDAR point cloud. As shown in Fig. 1, the LIDAR and stereo camera point clouds have very different noise and types of errors.

For each pair of scans from the same viewpoint, we detected keypoints in each scan and computed the repeatability as described in Section IV; results in Fig. 3 are averages over the four tracks. We evaluate repeatability as a function of search radius, so the plots show, for a given search radius on the x-axis, what fraction of keypoints in one scan have a matching keypoint within that search radius in the other scan. Examples of some detected keypoints are shown in Fig. 2.

The KPQ-SI detector finds the most features and the NARF detector the fewest (Table I), resulting in quite different absolute repeatability (Fig. 3). Using a search radius below 0.25m, the NARF detector finds the most repeatable keypoints, but still only finds an average of about 5 keypoints in each frame. For larger search radii, the KPQ-SI detector finds a larger number of repeatable keypoints. However, fewer keypoints may be preferred for SLAM, as the band-

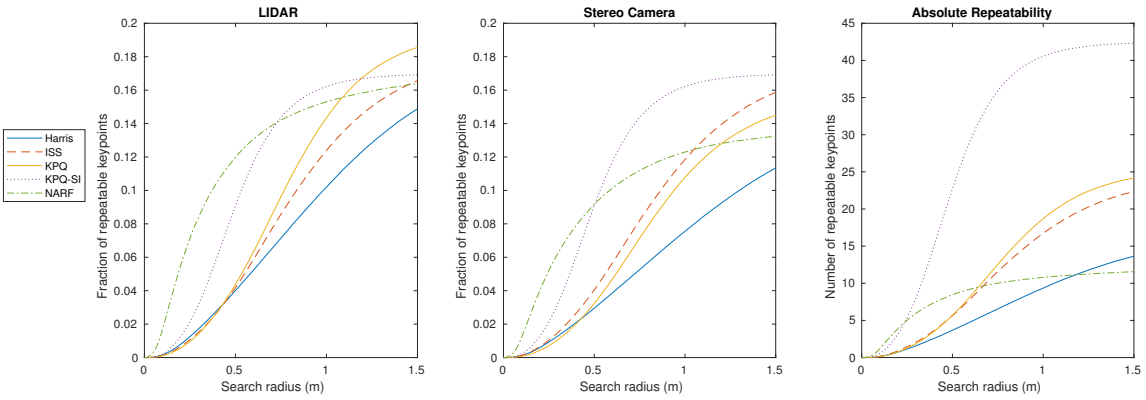

Fig. 3. Repeatability of each keypoint detector between scans taken with different sensors from the same pose. The left (LIDAR) and center (stereo camera) plots show the fraction of keypoints from that type of scan with a match in the other type of scan within the search radius. The right plot shows the total number of repeatable keypoints between the scans. Values shown are means over all poses evaluated.

width required for robots to share keypoints and the time required for matching increase with the number of keypoints. While it may be possible to identify keypoints with better chances of matching (e.g., with the KPQ mesh-fitting step), the selection process increases computation time so tradeoffs in selecting the best keypoints must be considered.

For a small search radius (<0.5m), NARF keypoints have significantly higher relative repeatability than the other keypoint detectors (Fig. 3). For larger search radii, KPQ-SI features have better relative repeatability, but all methods reach similar values. The best search radius depends on factors like the quality of the point cloud from each sensor, the localization accuracy of each detector, and the density of keypoints. Noise and localization errors may increase the appropriate search radius. However, too large of a search radius results in unrelated keypoints being matched.

B. Scans from different viewpoints

In previous keypoint analyses for tasks such as object detection, keypoints from scans of scenes are compared against a full 3-D model. In that case, while only some regions of the object are visible in the scan, all of those regions will be visible in the model. In SLAM applications, however, differing poses for two scans may lead to limited or no overlap of the visible parts of the scene.

Comparing two scans from the same pose, as in Section V-A, guarantees that the same parts of the scene are visible, but this is not realistic for multirobot SLAM. Even if two robots can detect a loop closure, they may still not be close enough to assume that the scene is observed from the same pose. To understand how the system might perform in such scenarios, we evaluated keypoint repeatability in different frames, selecting sections of tracks 0 and 6 for cross-frame comparison. Based on the repeatability observed between scans from the same sensor, we select 0.5m as a search radius. We evaluated repeatability between scans measured in nearby frames and in an area where the track intersected itself. Track 0 is in a crowded neighborhood, with many buildings and cars close to the street, while track 6 is on a larger street, with any visible buildings and cars farther away (Fig. 4 shows sample images). Thus, the point clouds in track 6 cover a much larger area than those in track 0.

Fig. 4. Sample frames from track 0 (top) and track 6 (bottom).

For matching between nearby frames, we chose sections of tracks 0 and 6 where the track is straight and the vehicle has a near-constant velocity, so difference in frame numbers can be associated with a displacement distance. In track 0, we used frames 0-80 and 3770-3870; the vehicle averages 1 meter of forward motion per frame. In track 6, we used frames 0-180 and 330-615; the vehicle averages 1.2 meters of forward motion per frame. All poses had nearly the same orientation (cameras facing forward, in the direction of motion). We measured absolute repeatability for scans separated by up to 20 meters in track 0 and 12 meters in track 6 (Fig. 5).

For comparison, we also computed repeatability for scans from the same sensor in these pose pairs, shown in Fig. 6. Though most of the keypoint detectors have not previously been evaluated in outdoor scenes of this scale, performance is similar to that reported in other comparison studies [25].

Over the pairs of scans with the same orientation, both the NARF and KPQ-SI detectors find a significant number of repeatable keypoints. The KPQ-SI detector again has a higher absolute number, but the relative repeatabilities are similar. As expected, the number of repeatable keypoints decreases as the distance between the frames compared increases. However, even at a distance of 10 meters, the NARF detector still typically finds about 10 matching keypoints, and the KPQ-SI detector finds 20-30. This is enough to compute point cloud

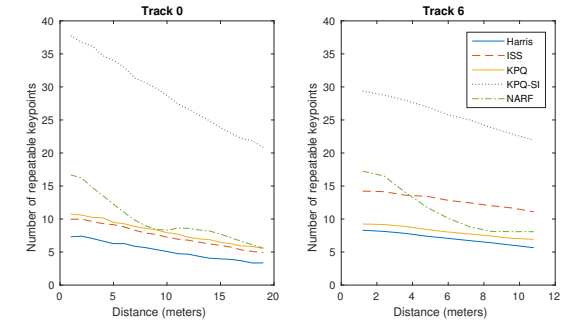

Fig. 5. Absolute repeatability of each keypoint detector between LIDAR and camera frames from different poses in tracks 0 (left) and 6 (right). The distance between poses is given on the x-axis. Repeatability decreases as the distance increases, both because the area covered by the scans has less overlap and because the viewpoints for the scans differ more.

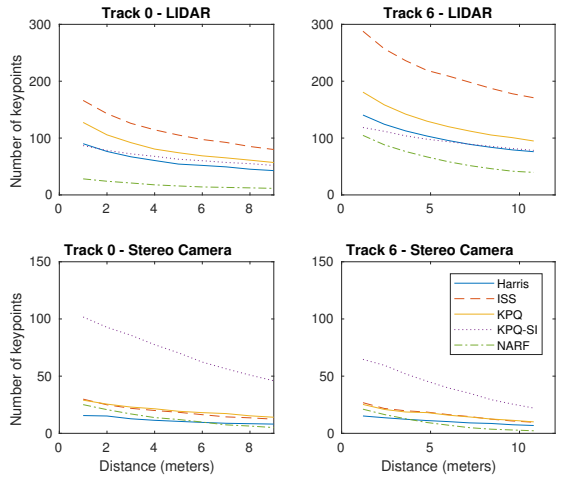

Fig. 6. Absolute repeatability of each keypoint detector between scans from the same sensor from different poses.

registration and indicates that loop closure would be possible between different 3D sensors using these detectors.

In the area where the track crossed itself, we selected two segments where the same area was visible but the vehicle had different orientations. We used track 0, frames 2326-2340 and 3255-3269; here, the vehicle encounters an intersection on one road and later returns to it on another road. The average numbers of repeatable keypoints between LIDAR scans in the first segment and stereo camera frames in the second are given in Table II. While all detectors find some repeatable keypoints, only NARF and KPQ-SI find enough to consistently estimate relative poses and perform loop closure.

We also evaluated the computational performance of each detector. We randomly selected 100 frames from the four tracks and measured the mean time for each detector to find keypoints in each type of scan; Table III reports measured time. The evaluations were performed on an Intel Xeon 3.5 GHz processor with 32 GB of RAM, and all methods were written in C++ to interface with PCL. All detectors were significantly slower on the LIDAR point clouds; this can be somewhat mitigated by only searching the areas visible in both scans. However, NARF was the only detector that could perform keypoint detection in real time for a system making scans at 1 Hz, a standard rate for outdoor SLAM systems.

TABLE II

MEAN REPEATABLE KEYPOINTS BETWEEN LIDAR AND STEREO CAMERA SCANS FROM POSES WITH DIFFERENT ORIENTATIONS

Harris	ISS	KPQ	KPQ-SI	NARF
3	5	5	28	11

TABLE III

MEAN TIME TO EXTRACT KEYPOINTS (IN SECONDS)

Scan Type	Harris	ISS	KPQ	KPQ-SI	NARF
LIDAR	7.77	25.9	31.0	48.4	0.22
Stereo Camera	0.24	0.51	0.68	1.05	0.05

Overall, the NARF and KPQ-SI detectors find keypoints with the highest repeatability between scans from different sensors. This suggests that they are good choices for heterogeneous SLAM applications. The NARF detector is very efficient, but its repeatability decreases more quickly with increased distance between poses (Fig. 5). Thus, it is likely the best choice for most applications. If high bandwidth for inter-robot communication and significant computing power are available, it may be better to use the KPQ-SI detector, particularly if keypoint quality can be estimated.

VI. CONCLUSION AND FUTURE WORK

In this paper, we evaluated 3D keypoint detectors for repeatability of detected keypoints in point clouds from heterogeneous sensors for multi-robot SLAM. To the best of our knowledge, this is the first comparison of 3D keypoints for SLAM or visual odometry. We evaluated the Harris3D, ISS, KPQ, KPQ-SI, and NARF detectors, comparing their performance on LIDAR and stereo camera scans from the KITTI dataset. We found that NARF and KPQ-SI had the highest relative repeatability, with both detectors finding about 10% repeatable keypoints in scans from different sensors in the same pose. While KPQ-SI found more keypoints overall, it had the lowest computational efficiency. In scans observing the same area with different sensors and in different poses, NARF and KPQ-SI consistently found enough matching keypoints to perform point cloud registration and find loop closures, while the other methods did not. For building a feature-based heterogeneous multi-robot SLAM system, NARF is the best choice of keypoint if computing power or bandwidth are limited. If they are not, KPQ-SI may be a better choice due to the higher absolute number of keypoints.

In the future, we plan to work toward building a heterogeneous multi-robot SLAM system. To that end, we plan to evaluate these same keypoints, particularly NARF, on sensors other than those used here. A heterogeneous multi-robot SLAM system must identify keypoint matches without true pose information, thus we plan to evaluate keypoint descriptors to understand if NARF descriptors work well with hetergenous sensors or if another type of descriptor is better. We also plan to study how NARF keypoints might be used more effectively: for example, if many point clouds are merged into a map, that map could be projected to a range image with any pose as the origin. This way, keypoints could be extracted for a pose that has not been visited.

REFERENCES

[1] S. Saeedi, M. Trentini, M. Seto, and H. Li, "Multiple-robot simultaneous localization and mapping: A review," *Journal of Field Robotics*, vol. 33, no. 1, pp. 3–46, 2016.

[2] L. Luft, T. Schubert, S. I. Roumeliotis, and W. Burgard, "Recursive decentralized collaborative localization for sparsely communicating robots." in *Robotics: Science and Systems*, 2016.

[3] R. Dubé, A. Gawel, H. Sommer, J. Nieto, R. Siegwart, and C. Cadena, "An online multi-robot SLAM system for 3D LiDARs," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 1004–1011.

[4] S. Choudhary, L. Carlone, C. Nieto, J. Rogers, H. I. Christensen, and F. Dellaert, "Distributed mapping with privacy and communication constraints: Lightweight algorithms and object-based models," *International Journal of Robotics Research*, vol. 36, no. 12, pp. 1286–1311, 2017.

[5] B. Steder, R. B. Rusu, K. Konolige, and W. Burgard, "Point feature extraction on 3D range scans taking into account object boundaries," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 2601–2608.

[6] Y. Zhong, "Intrinsic shape signatures: A shape descriptor for 3D object recognition," in *IEEE International Conference on Computer Vision (ICCV) Workshops, 2009*, 2009, pp. 689–696.

[7] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 5, pp. 433–449, 1999.

[8] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, "Aligning point cloud views using persistent feature histograms," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2008, pp. 3384–3391.

[9] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2009, pp. 3212–3217.

[10] L. A. Alexandre, "3D descriptors for object and category recognition: a comparative evaluation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) Workshop on Color-Depth Camera Fusion in Robotics*, vol. 1, no. 3, 2012.

[11] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, and N. M. Kwok, "A comprehensive performance evaluation of 3D local feature descriptors," *International Journal of Computer Vision*, vol. 116, no. 1, pp. 66–89, 2016.

[12] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[13] 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.

[14] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2564–2571.

[15] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European Conference on Computer Vision*, 2006, pp. 430–443.

[16] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[17] D. Gálvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[18] I. Sipiran and B. Bustos, "Harris 3D: a robust extension of the Harris operator for interest point detection on 3D meshes," *The Visual Computer*, vol. 27, no. 11, pp. 963–976, 2011.

[19] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, "Surface feature detection and description with applications to mesh matching," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 373–380.

[20] U. Castellani, M. Cristani, S. Fantoni, and V. Murino, "Sparse points matching by combining 3D mesh saliency with statistical descriptors," in *Computer Graphics Forum*, vol. 27, no. 2, 2008, pp. 643–652.

[21] R. Unnikrishnan and M. Hebert, "Multi-scale interest regions from unorganized point clouds," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2008, pp. 1–8.

[22] J. Sun, M. Ovsjanikov, and L. Guibas, "A concise and provably informative multi-scale signature based on heat diffusion," in *Computer Graphics Forum*, vol. 28, no. 5, 2009, pp. 1383–1392.

[23] C. Harris and M. Stephens, "A combined corner and edge detector." in *Alvey Vision Conference*, vol. 15, no. 50, 1988, pp. 147–151.

[24] S. Salti, F. Tombari, and L. Di Stefano, "A performance evaluation of 3D keypoint detectors," in *International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT)*, 2011, pp. 236–243.

[25] F. Tombari, S. Salti, and L. Di Stefano, "Performance evaluation of 3D keypoint detectors," *International Journal of Computer Vision*, vol. 102, no. 1-3, pp. 198–220, 2013.

[26] S. Filipe and L. A. Alexandre, "A Comparative Evaluation of 3D Keypoint Detectors in a RGB-D Object Dataset," in *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2014, pp. 476–483.

[27] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches," *Pattern Recognition Letters*, vol. 28, no. 10, pp. 1252–1262, 2007.

[28] T.-H. Yu, O. J. Woodford, and R. Cipolla, "A performance evaluation of volumetric 3D interest point detectors," *International Journal of Computer Vision*, vol. 102, no. 1-3, pp. 180–197, 2013.

[29] A. Kostusiak, "The comparison of keypoint detectors and descriptors for registration of RGB-D data," in *Challenges in Automation, Robotics and Measurement Techniques*. Springer, 2016, pp. 609–622.

[30] S. A. Scherer and A. Zell, "Efficient onboard RGBD-SLAM for autonomous MAVs," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 1062–1068.

[31] A. Pumarola, A. Vakhitov, A. Agudo, A. Sanfeliu, and F. Moreno-Noguer, "PL-SLAM: Real-time monocular visual SLAM with points and lines," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4503–4508.

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