Rotation and Translation Invariant 3D Descriptor for Surfaces

Joshua Hampp[1] and Richard Bormann[1]

***Abstract*— We present a descriptor estimator for surface-based 3D input data for coarse localization of mobile robots. From the input pointclouds surfaces are reconstructed and simplified to detect stable keypoints which are used to evaluate rotation and translation invariant features. The invariance is achieved by transforming the triangulated input data into the frequency domain by Fourier transformation and spherical harmonics. The pipeline was evaluated against state of the art algorithms and tested to localize a mobile robot. The source code is publicly available.**

I. INTRODUCTION

In many applications mobile robotic platforms can improve the workflow, like transportation of work pieces, automatic cleaning or working at hardly accessibly places. In other cases mobility is a necessity for interaction with humans in domestic living for entertainment or health care. Beside of the manoeuvrability of the platform the software has to localize the mobile robot in the environment and refine the position continually to be able to navigate collision free. Furthermore localization is needed for following trajectories or recognizing places like workplaces, danger areas, lifts, kitchens and more. As the odometry drifts apart from the real world position over time perception systems like laser scanners are used to update the pose of the robot.

Despite the high quality standards demanded by the users, the localization solution should be cost effective. Therefore many robots like PR2, rob@work, MobiNa[1] (Fig. 1) or Care-O-bot 3 [1] use consumer hardware like the Kinect to implement the perception of the environment. In the following we will concentrate on coarse localization of a robot based on the pointclouds output from 3D cameras. This enables a robot to position itself in relation to its environment without human interaction. While the mobile platform is moving, the pose from the odometry or other sensors can also be validated and refined by the extensive pointcloud data which is aligned with a given map from CAD models or with mapped data from previously visits.

To recognize an already seen place like a part of a room the input data has to be matched with a database which contains a similar view of the scene. The input data will differ by the view point, the noise, the lighting conditions and dynamic changes to the environment as the camera will probably not be at the same spot again. For large buildings like universities, many places have to be stored, recognized and distinguished. Therefore a description of the input data is needed which satisfies following requirements:

- *Scale invariance* is achieved by the used 3D camera as the depth correlates with the scale
- *Rotation and translation invariance* increase robustness against changes of the camera pose
- *Insensitivity to density* of the sampled pointcloud is necessary as the sampling of the pointclouds heavily depends on the viewing angle and distance to the surfaces
- *Robustness* against noise and lighting conditions
- The descriptor should be *compact* and *searchable*

The approach described within this paper is designed to match all these requirements. Additional main contributions

Fig. 1: MobiNa (Mobile Emergency Assistance) used for experiments with an Asus Xtion Live Pro and an iRobot Create platform

[1]The authors are with the Fraunhofer Institute for Manufacturing Engineering and Automation IPA, 70569 Stuttgart, Germany <first name>.<last name> at ipa.fraunhofer.de; www.ipa.fraunhofer.de

[1]http://youtu.be/JfkslJbNlDU

in this paper are:

- Fast keypoint selection for surface-based data
- Novel approach to compute spherical harmonics on discontinuous functions (surfaces)
- Processing of pointclouds and CAD models as input data
- Evaluation against the state of the art methods for keypoint selection and feature estimation on the RGB-D SLAM dataset from the university of Freiburg [2]
- First experiments with a robot in indoor environments to recognize scenes with a bag-of-words approach as input for RatSLAM [3] for loop closure.
- The implementation and evaluation is publicly available [2] and provides a ROS package which uses the Point Cloud Library [4] (PCL)

The paper is structured as follows: Section 2 gives an overview of the current state of the art of related work. In Section 3 the theoretical foundation and the pipeline of the descriptor is described which is evaluated in Section 4 against five keypoint selection methods and four feature estimators on real world data. In an example application the descriptor is used to localize a robot. Finally, we conclude and give insight on future work in Section 5.

II. PREVIOUS WORK

In literature descriptors for distinguishing 3D input data are in general divided into signatures and histograms. Signatures define a local reference frame, also referred to as normalization, to achieve transformation invariance. One example for local reference frames is Signature of Histograms of Orientations [5] (SHOT) which computes unique signatures for a local surface description from a pointcloud. The reference frame is determined by Eigen Value Decomposition by the local distribution of the surface points. The normals of the relevant points are classified in bins of a sphere which leads to signatures of histograms.

In contrast histograms without reference frame define distinct bins for a transformation invariant, relational norm, like distances between points. Representative algorithms are Ensemble of Shape Functions [6] (ESF), Point Featue Histogram [7], its extension Fast Point Feature Histogram [8] (FPFH) and Viewpoint Feature Histogram [9] (VFH).

To achieve invariance against rotation and translation PFH uses a relative measure around an interest point, also called keypoint, on a pointcloud with normals. PFH computes three angles and the distance between each point pair within a user-defined radius at the query point. FFPH is a variation of PFH with the goal to speed up the computation by only visiting point pairs containing the query point and weighting the histogram in dependence of the distance of the second point from the query point.

A further extension of FPFH is VFH which is invariant to scale but not to the pose of the camera which allows retrieving the camera pose. In short, the interest point is replaced by the cameras origin and the normal of the relative point by the central viewpoint direction.

ESF also uses histograms with 10×64 bins and can be computed on pointclouds as well as CAD models. It selects random points from the input pointcloud and classifies them to be on or off the surface and the relation of the points. Different measures for the relation of points are used like distances (D2), angles (A3) or areas (D3). Compared to FPFH no pre-processing like normal estimation is necessary.

Another approach to recognize known 3D data are spherical harmonics shape descriptors [10], [11] which are meant to retrieve or categorize CAD models from a database. The method focuses on a rotational invariant representation by describing the model by a spherical function in terms of the amount of energy it contains at different frequencies. At first the model is transformed into a blurred voxel grid to provide a continuous function in $\mathbb{R}^3$. Different variations of the algorithm use local feature estimators for the transformation from the input data to the voxel grid. The voxel grid is sampled spherically at different radii. By using a Fourier transformation the weighted samples can be converted to the frequency space. The energy of the resulting vector is invariant to rotation.

To improve the matching results for local feature estimators and reducing the number of features, stable keypoints are selected beforehand to increase stability and repeatability. Two widely used approaches are Scale-Invariant Keypoints [12] (SIFT) and Harris 3D [13]. The Harris 3D keypoint selection for 3D data is an extension of the original Harris detector. The detector is based on the local autocorrelation function by measuring local changes of nearby points. For the calculation of the response different approaches (e. g. HARRIS, LOWE, TOMASI) are implemented in PCL. Points are selected as keypoints if the response is above a defined threshold.

SIFT selects keypoints at minima and maxima of the result of difference of Gaussian function applied in scale space to smoothed input data. SIFT was originally designed for 2D images and adapted for the 3D case later.

III. KEYPOINT SELECTION AND FEATURE ESTIMATION

The pipeline of the single processing steps for data acquisition and preprocessing, keypoint selection and descriptor computation is shown in Figure 2. In the first step the pointcloud is converted to a shape representation (1) which is simplified in the next step (2). On the surfaces keypoints are detected (3). Around the keypoints a local, triangulated submap (4,5) is generated. The submap is formulated as continuous function in the frequency domain (6) which is further processed by spherical harmonics (7). The result is a rotation and translation invariant descriptor at each keypoint.

A. Data acquisition

For our application we use a RGB-D camera namely Asus Xtion Pro Live which captures an ordered pointcloud representing visible surfaces of indoor environments. We

[2] https://github.com/ipa-josh/cob_environment_perception

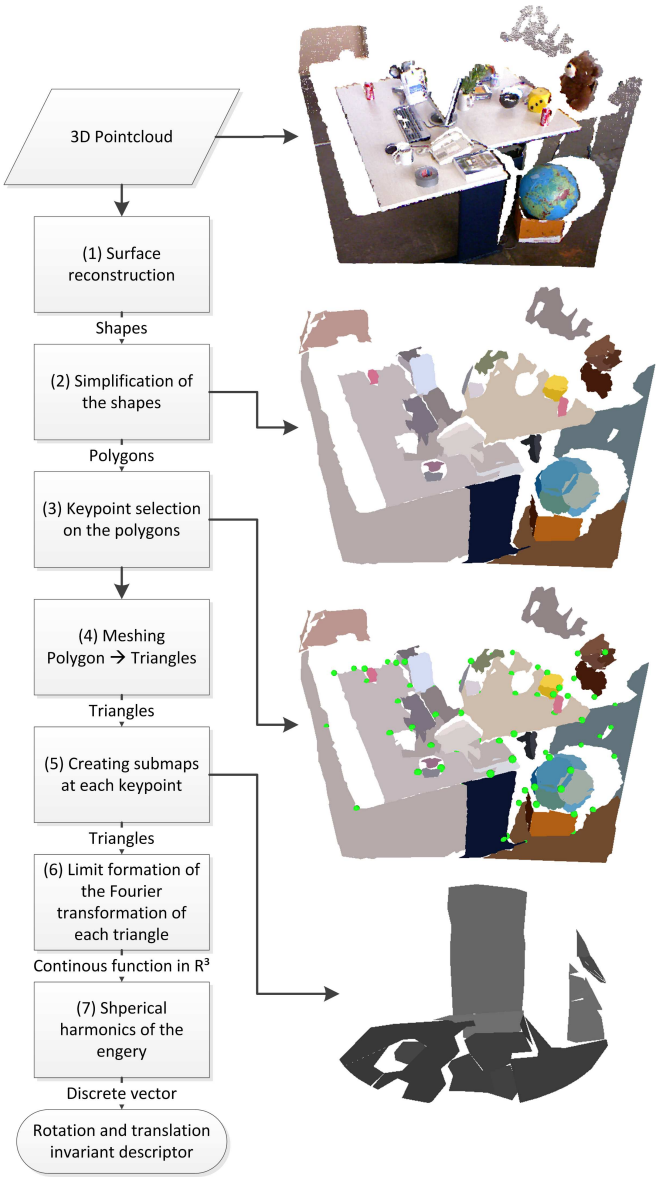


Fig. 2: Pipeline of the execution steps: First image shows the input pointcloud. Second image the reconstructed surfaces. In the third image the keypoints (green) are highlighted. In the last image a submap at a keypoint is shown.

do not use the color information. In most indoor scenarios surfaces can be found to be locally planar as our evaluation results show in [14]. By reconstructing the planar surfaces the representation of the pointcloud becomes more compact and computational efficient. We used our surface reconstruction algorithm [14] which is characterized by a high execution speed of over 30 Hz and has no needs for any preprocessing of the pointcloud. The algorithm fits polynomial functions into a pointcloud with the help of a quadtree data structure from top down. Other surface reconstruction methods (e. g. [15]) can also be used if they output delimited, planar surfaces.

In comparison to the discrete model of single points the surface representation has two main advantages for the presented feature estimation:

- *Data reduction* which leads to a significant speed up
- *Independence of point sampling density*

B. Keypoint selection

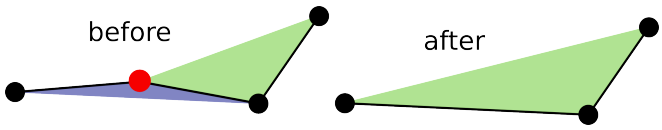


Fig. 3: Visvalingam-Whyatt algorithm: Detail view of computation of the effective area. The blue area is the effective area for the red point which is smaller than the effective are for the following point. Therefore the red point is removed.

The surface reconstruction outputs shapes from the current view of the camera. Each shape is an individual plane which is delimited by a polygon supporting holes. Small movements of the camera (around one meter) will generate resembling visible shapes which is sufficient for recognizing locations. To reduce the number of descriptors locally stable keypoints are identified. We use the polygons of the shapes to compute the keypoints by an adaption of the Visvalingam-Whyatt algorithm [16]. The Visvalingam-Whyatt algorithm is meant to simplify the polygons. In our case we use it for two purposes:

- Remove noise and reduce the size of the resulting triangulated mesh which enhances the processing speed and leads to more stable features in the next step
- To detect interest points

The requirements of an interest point are stability to stay at the same location and repeatability to be often visible. Therefore a low noise sensitivity is an advantage.

The Visvalingam-Whyatt algorithm is an area based algorithm to eliminate points with a minimal change of the whole area of the polygon. The effective area of each point of a polygon is defined by the area of the triangle of the current, next and previous point. The point with the least effective area is removed (see Fig. 3). These steps are repeated until a break condition occurs. For the simplification of the surfaces we stop the point elimination if the minimum effective area is greater than the threshold $t_{\text{min\_area}}$.

If subtle noise was removed, further deletion of points is only done virtually to identify the most stable points of the polygon with a different threshold $t_{\text{kp\_area}}$ which allows more area to be removed. The remaining points are potential keypoints. By evaluating reasonable settings on real world data we achieved the best results with the thresholds $t_{\text{min\_area}} = 12.5\,\text{cm}^2$ and $t_{\text{kp\_area}} = 225\,\text{cm}^2$ which are used throughout the paper.

The resulting interest points can be near to each other at bordering areas. For example a box will generate four interest points for each visible face. At the completely visible corner of the box there will be three potential keypoints at the same location. By thinning out random interest points near to each other within the fourth of the radius r we generate a list of

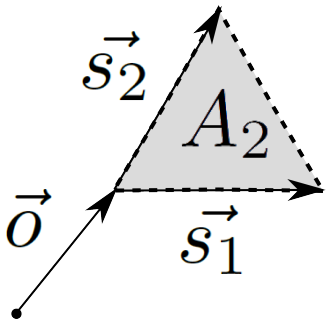


Fig. 4: Vectors of a triangle

keypoints. The radius r is defined by the feature estimation which uses the radius to limit the local influence of the input data. Further we exclude keypoints which generate features that are not completely visible within their search radius r in the current field of view.

C. Descriptor computation

Before computing the view point invariant features at the keypoints the simplified shapes are converted to single triangles (4). The triangulated mesh is a universal representation of the environment. At each keypoint the intersection of a sphere with radius r and the mesh is used to limit the influence of the input data to a local feature. Therefore only the area of the surfaces within the search radius around the point is visited. To achieve rotation invariance in literature spherical harmonics are used [10] beside other methods. In general continuous functions are needed to evaluate spherical harmonics because the integration of a visible surface yields always zero as no volume is present. Existing approaches use voxel grids to approximate pointclouds or CAD models by converting the discontinuous representation in $\mathbb{R}^3$ to a continuous representation in $\mathbb{R}^3$. The voxel grid has the drawback that the input data is discretized to a fixed resolution.

Instead our approach overcomes discretization and translation variance by converting the surfaces into the frequency space which is continuous without discretization. The transformation from surfaces to frequency domain still encounters the problem of a zero volume. Yet, in the Cartesian coordinate system the limit of the function is analytically resolvable.

Each triangle (see Fig. 4) is defined by an offset vector $\vec{o}$ and two direction vectors $\vec{s_1}$ and $\vec{s_2}$. Then there exists a function $f(s)$ which maps the surface coordinate s to a Cartesian vector.

$$f(s) = \vec{o} + \begin{bmatrix} \vec{s_1} \\ \vec{s_2} \end{bmatrix} s$$

By integrating over the delimited surface A_2 (triangle) the Fourier transformation can be computed. The evaluation of the Fourier transformation at point p is stated as follows:

$$c(p) = \frac{1}{(2\pi)^{\frac{3}{2}}} \lim_{\Delta \to 0} \left(\int_{s=A_2} 1 \cdot e^{-2\pi \mathrm{i} p(f(s)+\Delta))} \right)$$

By substitution it can be formulated as

$$A = e^{-2\pi \mathrm{i}(\vec{o}p + \vec{s_2}p + \vec{s_1}p)}$$

$$B = e^{-2\pi \mathrm{i}(\vec{o}p + \vec{s_2}p)}$$

$$C = e^{-2\pi \mathrm{i}(\vec{o}p)}$$

$$c(p) = \frac{1}{(2\pi)^{\frac{3}{2}}} \frac{-\vec{s_1}pC + (\vec{s_1} + \vec{s_2})pB - \vec{s_2}pA}{\vec{s_1}p(\vec{s_2}p)^2 + \vec{s_2}p(\vec{s_1}p)^2}$$

which simplifies further limit value considerations which have to be handled for numerical stability. The superposition of $c(p)$ allows the computation of the frequency domain for the submap by summing up the complex values for each triangle of the submap. Translation invariance is accomplished by evaluating the amplitude of the frequency domain.

In the next step (7) spherical harmonics are used for the rotation invariant description of the submap. Instead of applying the spherical harmonics on a voxel grid we use $\|c(p)\|$ as volumetric image which is sampled spherically at different radii with an uniform distribution distribution from 0 to r excluding 0.

The result is a discrete matrix which describes again the frequency domain of the translational invariant description of the mesh. Reducing the matrix to the amplitude yields a rotation invariant description. The number of radii as well as the number of the sampled frequencies of the spherical harmonics have influence on the accuracy and computational complexity. By cross validation we achieved good results with 8 radii and 32 frequencies. This parameters are used throughout the paper.

As the resulting rotation and translation invariant descriptor has a fixed size, fast nearest neighbor search methods like k-d tree is applicable. In addition the proposed approach can be used for all input data which can be converted to triangles which allows a broad application range.

IV. EVALUATION

We evaluated the proposed method for keypoint selection and descriptor computation against existing algorithms on publicly available datasets of indoor environments from the university of Freiburg. The dataset [2] is meant for comparison of SLAM algorithms for Kinect data and includes ground truth trajectories of the camera. The used recordings are listed in table II. We used every 10th frame, in total 950 pointclouds. Our keypoint selection is evaluated against SIFT and Harris (with the variations: HARRIS, TOMASI, NOBLE and LOWE). The descriptor is compared with ESF, FPFH, SHOT and VFH. All feature estimators output a vector of a fixed size. The number of bins is stated in Table I. In the following we abbreviate the proposed algorithm as Fourier Shape Descriptor (FSHD). All algorithms are implemented in PCL 1.7. The used computer system is equipped with an Intel Xeon CPU E5-2643, no GPU acceleration was used. For both execution steps the execution time and the qualitative analysis is stated. All scripts used for the evaluation can be found in the project repository[3].

[3]https://github.com/ipa-josh/cob_environment_perception

TABLE I: Evaluation results for descriptor: Bin size and execution time (including preprocessing)

Algorithm	Bin size	Execution time [s]
FSHD	128	0.019
ESF	640	0.13
FPFH	33	3.2
SHOT	352	0.071
VFH	308	0.11

TABLE II: Real world datasets from the university of Freiburg which were used

Filename	No. of frames
rgbd_dataset_freiburg2_desk	217
rgbd_dataset_freiburg2_desk_no_loop	59
rgbd_dataset_freiburg2_pioneer_360	77
rgbd_dataset_freiburg2_pioneer_slam2	131
rgbd_dataset_freiburg2_pioneer_slam3	190
rgbd_dataset_freiburg2_pioneer_slam	229
rgbd_dataset_freiburg3_long_office_household	247

A. *Keypoint Selection*

The keypoint selection algorithms estimate the keypoints for the specified datasets. The points are then transformed into a global reference coordinate system according to the ground truth transformation of the dataset. The keypoints of different frames are compared against each other. If the L2 norm is below the fourth of the feature radius the keypoint is assigned to the stable keypoints n_s. Otherwise it is an unstable keypoint n_u. For the stable point the minimum distance in Cartesian space to the best match is computed for an accuracy comparison.

Figure 5 shows the relation between stable keypoints and the total number of keypoints by $\frac{n_s}{n_s+n_u}$ in dependence of the selected search radius. A maximum is reached at 0.6 m with 30 keypoints per frame in average. Above 0.6 m the repeatability is decreasing as the viewport of the camera limits the keypoint detection. The compared keypoint selection algorithms work directly on the pointcloud with a higher granularity. For the following evaluation of the descriptor a search radius of 0.6 m was used.

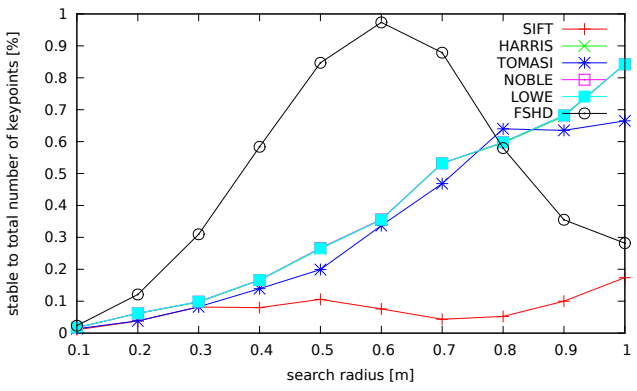


Fig. 5: Evaluation results for keypoint selection: Relation between stable and total number of keypoints in dependence of the selected search radius to state the repeatability

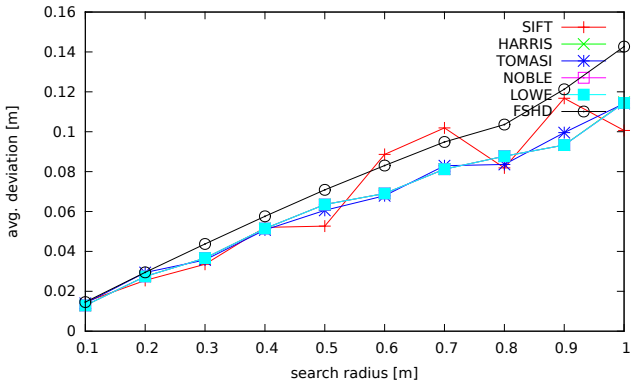


Fig. 6: Evaluation results for keypoint selection: Averaged minimum distance of the matched stable keypoints to state the stability

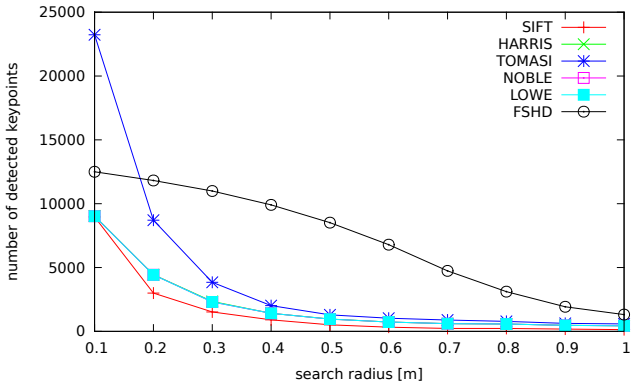


Fig. 7: Evaluation results for keypoint selection: Number of of detected keypoints in dependence of search radius r

TABLE III: Evaluation results for keypoint selection: Execution time for keypoint selection (including preprocessing)

Algorithm	Execution time [ms]
FSHD	0.035
SIFT	5.31
HARRIS	4239
TOMASI	4425
NOBLE	4237
LOWE	4252

The averaged minimum distance of the matched stable keypoints are similar for all approaches as shown in Figure 6. The proposed method shows a slightly lower accuracy as the discretization level of the shapes is higher than of the pointcloud. Yet, the reduced size of the shapes compared to the pointcloud has a great impact on the execution time as stated in Table III. The number of detected keypoints is stated in Figure 7.

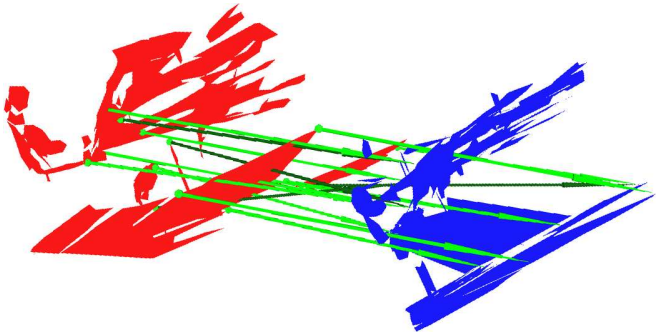


Fig. 8: Matching of our descriptor between two recordings. On the left a human is visible, working at a table. Green arrows show best matching correspondences.

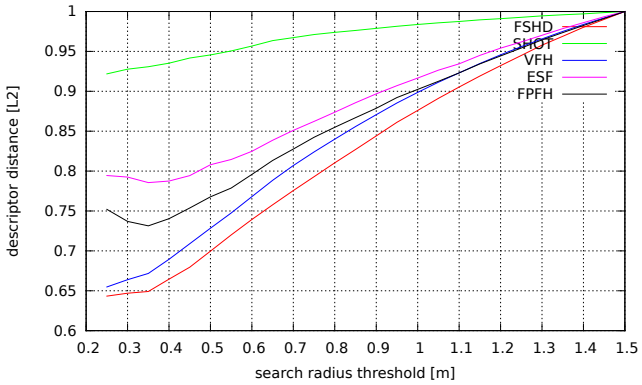


Fig. 9: Evaluation results for descriptor: L2 norm of the features are evaluated against each other which fall within the stated Euclidean distance to state the distinctiveness

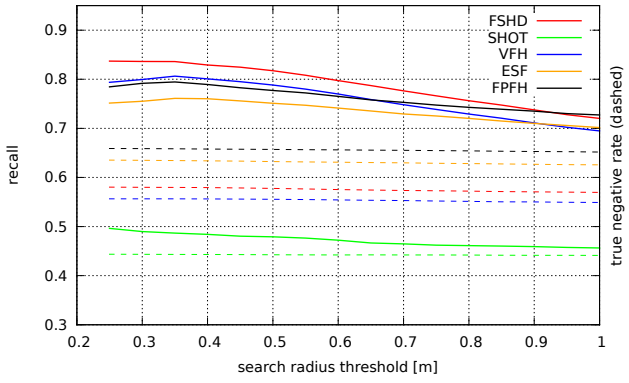


Fig. 10: Evaluation results for descriptor: recall (solid lines) and true negative rate (dashed lines) of the features with an average descriptor distance (L2) threshold of the outliers (from stated search radius up to a radius of 1.5 m)

B. Descriptor

The keypoints of our approach are used as common basis for the descriptor evaluation. Like in the evaluation of the keypoint selection each local point is transformed into a global coordinate system. The L2 norm of the features are evaluated against each other which fall within the stated Euclidean distance. Figure 8 shows the correspondences of a typical match between two scenes using our algorithm. Figure 9 visualizes the Euclidean distance on the abscissa and the corresponding and normalized average L2 norm of the feature distances on the ordinate. The normalization is necessary because each descriptor has a different range of values. The normalized values are stated relative to the feature distance value at a radius of 1.5 m.

A high distinctiveness is achieved by a low feature distance within the search radius of the feature estimators. As it can be seen the feature distance of our approach is lower than the results of the other algorithms. Therefore a good match of the descriptor corresponds well with the feature distance. As the curve approximates steadily but slow against 1 over the Euclidean distance the descriptor is stable against transformations.

The recall and true negative rate for different search radii are stated in Figure 10. The features within the stated search radius are the positive samples and samples above the search radius up to 1.5 m are used as outliers. The matching threshold for the feature distance was chosen from the average L2 norm of the outliers. FSHD and VFH have both a high recall rate with a similar distance to the true negative rate which indicates a higher distinctiveness than the compared feature estimators. The true negative rate of around 0.5 results from the adaptive threshold. Furthermore the specificity is limited because of similar and ambiguous geometries of the indoor scenes like repeating walls or furniture. Failure cases of our approach are results of the limited bandwidth of the descriptor which is true for all 3D descriptors and the dependence on the surface reconstruction. The execution time for all approaches is listed in Table I. Our descriptor estimation based on surfaces instead of pointclouds requires the lowest execution time.

C. Example Application: Real-world experiments with RatSLAM

In conclusion the descriptor was used for a first example application with a SLAM approach on the robot MobiNa. We used the open source implementation of RatSLAM [3] for localization and mapping. RatSLAM is a bio-inspired SLAM system based on the insights of the functioning of the brain of rodents (therefore the name ”rat”) which uses a grid-based attractor network dynamics for integrating odometry and landmark sensing to form a topological map. The outputted map describes a set of relative transitions of poses which can recover from major path integration errors.

Unlike many other SLAM systems RatSLAM does not depend on pose refinement by registration of the input data against a map. Instead it uses local scene identification for loop closure and rough pose correction. Local scene

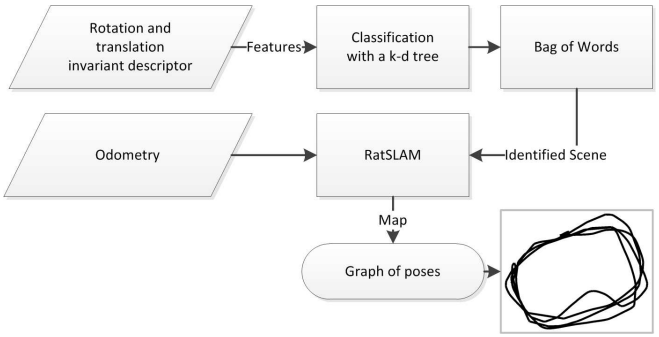


Fig. 11: Pipeline of the scene recognition and SLAM system: First the computed features are classified by a k-d tree to a distinct vocabulary. Second, all features of a scene generate a unique word of the recognized scene. The recognized scene is used by RatSLAM for localization and loop closure which leads to a non-Cartesian pose map.

(a) Two captures of the same scene from rgbd_dataset_freiburg2_desk

(b) Two captures of the same scene from rgbd_dataset_freiburg2_pioneer_slam

Fig. 12: Correct scene identifications from different poses show the transformation invariance of the descriptor, however are misleading for loop closures

identification is achieved by exteroceptive sensor input like visual features. For the proposed descriptor we used a bag-of-words approach to retrieve a scene identifier from multiple features of a single-shot scene. Figure 11 shows the pipeline of the implemented scene identification. At first the features are classified to a sparse vector of occurrences by a nearest-neighbour classifier (k-d tree for the L2 norm of the descriptor). To identify a scene the similarity is computed for the perceived features F_{perc} compared to all known scenes F_{scene}. The similarity score is defined by $\frac{|F_{\text{perc}} \cap F_{\text{scene}}|}{|F_{\text{perc}}|}$. The thresholds for the L2 norm (80) and the minimum similarity score (0.2) were determined by cross validation and the previous evaluation results. The default settings of RatSLAM were used.

(a) Picture of floor scene (b) Picture of office scene

Fig. 13: Pictures of indoor scenes used for experiments

To evaluate the average precision and recall rate for correct loop closures we compared the camera poses of the scene matches from the groundtruth dataset of the university of Freiburg. Poses which lie within $0.4\ \mathrm{m}$ and have a maximal angle deviation of $0.2\,\mathrm{rad}$ are considered to be correct loop closures. For the application of SLAM the higher precision rate of 0.76 is more important than the lower average recall rate of 0.36. False positives are partly results from the view point invariance of the descriptor. Two examples are shown in Figure 12 of unsuccessful loop closures. Yet, the scene was identified successfully.

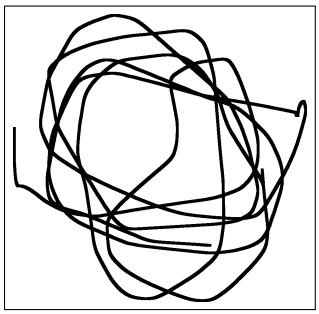
The real-world scenes were captured by the service robot MobiNa in indoor environments. The telepresence robot is based on an iRobot Create platform and is equipped with an Asus Xtion Pro Live and an embedded computing system (Exynos 4412: 4×1.7 GHz, 2 GB RAM). The odometry information has an significant average linear deviation of 4.9% because of the heavy setup. Surface reconstruction and feature estimation was computed online on the robot with around 10 Hz.

In the first experiment the robot was driven manually up- and downwards in an office floor (Fig. 13a, duration: 649 s). The floor has a regular geometry with some side doors and houseplants. In the second experiment a circular route was chosen within an office (Fig. 13b, duration: 443 s). The office has a more complex geometry and is equipped with tables, chairs and other equipment. For both experiments the course was driven 10 times round to provide enough possibilities to recognize the scene again.

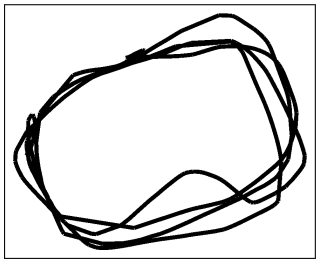
Figure 14 shows the comparison of the pose map with and without scene recognition. The odometry drift is drastically reduced through the pose correction by exploiting the additional information of the features. In the office experiment two false loop closures resulted in ”jumps” in the pose map from which the SLAM system recovered successfully after the next correct loop closure. From 2179 scenes 400 were recognized in the office experiment. In the floor experiment 1061 scenes of 1990 were identified.

V. CONCLUSIONS

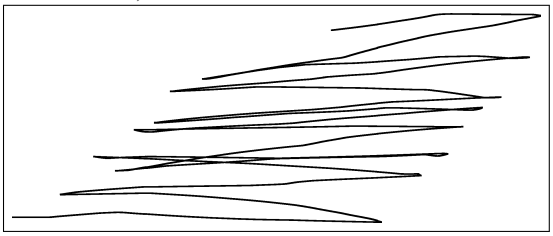
We presented processing steps to retrieve keypoints and rotation and translation invariant descriptors for surface-



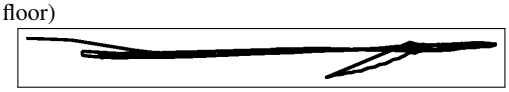
(a) Pose map from odometry only shows drift (round course in office)



(b) Pose map generated from odometry and features can recover from drift (round course in office)



(c) Pose map from odometry only shows drift (patrol in floor)



(d) Pose map generated from odometry and features can recover from drift (patrol in floor)

Fig. 14: Comparison of RatSLAM results with and without the usage of the proposed descriptor

based input data. By using the efficient data representation of the shapes and their transformation to the frequency domain it is possible to compute spherical harmonics descriptors online. The evaluation results show low computation time of $19\,\mathrm{ms}$ and good distinctiveness compared to the state of the art. First experiments on a mobile robot were successful. The next steps include extensive experiments for different applications like localization and object recognition. In further developments the descriptor will be extended to include additional information like color or intensity.

REFERENCES

[1] B. Graf, U. Reiser, M. Hägele, K. Mauz, and P. Klein, "Robotic Home Assistant Care-O-bot® 3 - Product Vision and Innovation Platform," in *Proceedings of the 13th International Conference on Human-Computer Interaction. Part II: Novel Interaction Methods and Techniques*, Tokyo, Japan, 2009, pp. 312–320.

[2] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-d SLAM systems," in *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[3] M. Milford, G. Wyeth, and D. Prasser, "Ratslam: a hippocampal model for simultaneous localization and mapping," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 1, April 2004, pp. 403–408 Vol.1.

[4] R. B. Rusu and S. Cousins, "3d is here: Point cloud library (PCL)," in *International Conference on Robotics and Automation*, Shanghai, China, 2011.

[5] F. Tombari, S. Salti, and L. Di Stefano, "Unique signatures of histograms for local surface description," in *Proceedings of the 11th European Conference on Computer Vision Conference on Computer Vision: Part III*, ser. ECCV'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 356–369.

[6] W. Wohlkinger and M. Vincze, "Ensemble of shape functions for 3d object classification," in *2011 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2011, pp. 2987–2992.

[7] R. B. Rusu, Z. C. Marton, N. Blodow, and M. Beetz, "Persistent point feature histograms for 3d point clouds," in *Proceedings of the 10th International Conference on Intelligent Autonomous Systems (IAS-10), Baden-Baden, Germany*, 2008.

[8] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3d registration." IEEE, May 2009, pp. 3212–3217. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5152473

[9] R. B. Rusu, G. Bradski, R. Thibaux, and J. Hsu, "Fast 3d recognition and pose using the viewpoint feature histogram." IEEE, Oct. 2010, pp. 2155–2162. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5651280

[10] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3d shape descriptors," in *Symposium on Geometry Processing*, June 2003.

[11] L. Zhang, M. J. da Fonseca, A. Ferreira, and Recuperação, "Survey on 3D Shape Descriptors."

[12] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004. [Online]. Available: http://link.springer.com/10.1023/B:VISI.0000029664.99615.94

[13] 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, Nov. 2011. [Online]. Available: http://link.springer.com/10.1007/s00371-011-0610-y

[14] J. Hampp and R. Bormann, "Quadtree-based polynomial polygon fitting." IEEE, Nov. 2013, pp. 4207–4213. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6696959

[15] G. Arbeiter, S. Fuchs, J. Hampp, and R. Bormann, "Efficient segmentation and surface classification of range images," in *IEEE ICRA 2014, International Conference on Robotics and Automation*, 2014.

[16] W. Shi and C. Cheung, "Performance evaluation of line simplification algorithms for vector generalization," *The Cartographic Journal*, vol. 43, no. 1, pp. 27–44, Mar. 2006. [Online]. Available: http://www.maneyonline.com/doi/abs/10.1179/000870406X93490