X-View: Graph-Based Semantic Multi-View Localization

Abel Gawel*, Carlo Del Don*, Roland Siegwart, Juan Nieto and Cesar Cadena

Abstract—Global registration of multi-view robot data is a challenging task. Appearance-based global localization approaches often fail under drastic view-point changes, as representations have limited view-point invariance. This work is based on the idea that human-made environments contain rich semantics which can be used to disambiguate global localization. Here, we present X-View, a Multi-View Semantic Global Localization system. X-View leverages semantic graph descriptor matching for global localization, enabling localization under drastically different view-points. While the approach is general in terms of the semantic input data, we present and evaluate an implementation on visual data. We demonstrate the system in experiments on the publicly available SYNTHIA dataset, and a realistic urban dataset recorded with a simulator. On these data, our findings show that X-View is able to globally localize aerial-to-ground, and ground-to-ground robot data of drastically different view-points. Our approach achieves an accuracy of up to 85% on global localizations in the multi-view case, while the benchmarked traditional appearance based method reaches up to 50%.

I. INTRODUCTION

Global localization between heterogeneous robots is a difficult problem for classic place-recognition approaches. Visual appearance-based approaches such as [1, 2] are currently among the most effective methods for re-localization. However, they tend to significantly degrade with appearance changes due to different time, weather, season, and also view-point [3]. In addition, when using different sensor modalities, the key-point extraction becomes an issue as they are generated from different physical and geometrical properties, for instance intensity gradients in images vs. high-curvature regions in point clouds.

Relying on geometrical information, directly from the measurements or from a reconstruction algorithm, on the other hand shows stronger robustness on view-point changes, seasonal changes, and different sensor modalities. However, geometrical approaches typically do not scale well to very large environments, and it remains questionable if very strong view-point changes can be compensated while maintaining only a limited overlap between the localization query and database [4, 5].

Recently, topological approaches to global localization regained interest by the community, as a way to efficiently encode relations between multiple local visual features [6, 7]. On the other hand, the computer vision community has made great progress in semantic segmentation and classification, resulting in capable tools for extracting semantics from visual and depth data [8].

Based on the hypothesis that semantics can help to mitigate the effects of appearance changes, we present X-View, a novel approach for global localization based on building graphs of semantics. X-View introduces graph descriptors that efficiently represent unique topologies of semantic objects. These can be matched in much lower computational effort, therefore not suffering under the need for exhaustive subgraph matching [9].

By using semantics as an abstraction between robot viewpoints, we achieve invariances to strong view-point changes. Furthermore, with semantics understanding of the scene, unwanted elements, such as moving objects can naturally be excluded from the localization.

We evaluate our global localization algorithm on publicly available datasets of simulated urban outdoor environments and report our findings on localizing under strong viewpoint changes. Specifically, this paper presents the following contributions:

• A novel graph representation for semantic topologies.
• Introduction of a graph descriptor based on random walks that can be efficiently matched with established matching methods.
• A full pipeline to process semantically segmented images into global localizations.

* The authors contributed equally to this work.

Authors are with the Autonomous Systems Lab, ETH Zurich. gawela@ethz.ch, delдонc@student.ethz.ch, {rsiegwart, nietoij, cesarc}@ethz.ch
• Open source implementation of the X-View algorithm\footnote{We will make the X-View source code publicly available with this publication under https://github.com/gawela/X-View}
• Experimental evaluation on publicly available datasets.

The remainder of this paper is structured as follows: Sec. II reviews the related work on global localization, followed by the presentation of the X-View system in Sec. III. We present our experimental evaluation in Sec. IV and conclude our findings in Sec. V.

II. RELATED WORK

In this section we review the current state-of-the-art in multi-robot global localization in relation to our proposed system.

A common approach to global localization is visual feature matching. A large amount of approaches have been proposed in the last decade, giving reliable performance under perceptually similar conditions \cite{1,2,10}. Several extensions have been proposed to overcome perceptually difficult situations, such as seasonal changes \cite{11,12}, daytime changes \cite{13}, or varying view-points using Convolutional Neural Network (CNN) landmarks \cite{14,15}. However, drastic view-point invariance, e.g., between views of aerial and ground robots continues to be a challenging problem for appearance-based techniques.

In our previous work, we demonstrated effective 3D heterogeneous map merging approaches between different viewpoints from camera and LiDAR data, based on overlapping 3D structural descriptors \cite{4,5}. However, 3D reconstructions are still strongly viewpoint dependent. While these techniques do not rely on specific semantic information of the scenes, the scaling to large environments has not yet been investigated, and computational time is outside real-time performance with large datasets.

Other approaches to global localization are based on topological mapping \cite{16,17}. Here, maps are represented as graphs $G = (V, E)$ of unique vertices $V$ and edges $E$ encoding relationships between vertices. While these works focus on graph merging by exhaustive vertex matching on small graphs, they do not consider graph extraction from sensory data or ambiguous vertices. Furthermore, the computationally expensive matching does not scale to larger graph comparisons.

With the recent advances in learning-based semantic extraction methods, using semantics for localization is a promising avenue \cite{18,19,20}. In \cite{19,20} the authors focus on the data association problem for semantic localization using Expectation Maximization (EM) and the formulation of the pose estimation problem for semantic constraints as an error minimization. The semantic extraction is based on a standard object detector from visual key-points.

Stumm et al. \cite{6} propose to use graph kernels for place recognition on visual key-point descriptors. Graph kernels are used to project image-wise covisibility graphs into a feature space. The authors show that graph descriptions can help localization performance as to efficiently cluster multiple descriptors meaningfully. However, the use of large densely connected graphs sets limitations to the choice of graph representation. Motivated, by these findings, we propose to use graph descriptors on sparse semantic graphs for global localization.

III. X-VIEW

In this section, we present our Graph-Based Multi-View Semantic Global Localization system, coined X-View. It leverages graph extraction from semantic input data and graph matching using graph descriptors. Fig. 2 illustrates the architecture of the proposed global localization algorithm, focusing on the graph representation and matching of query semantic input data to a global graph. The localization target map is represented as the global graph. X-View is designed to operate on any given odometry estimation system and semantic input cue. However, for the sake of clarity, we present our system as implemented for semantically segmented images, but it is not limited to it.

A. System input

We use semantically segmented images containing pixel-wise semantic classification as input to the localization algorithm. Also instance-wise segmentation, i.e., unique identifiers for separating overlapping objects of same class in the image space can be considered for improved segmentation, but is not strictly necessary for the approach to work. Furthermore, we assume the estimate of an external odometry system. Finally, we also consider a database semantic graph $G_{db}$, as it could have been built and described on a previous run of our graph building algorithm as described in the next sub-sections.

B. Graph extraction and assembly

In this step, we convert a sequence of semantic images $I_q$ into a query graph $G_q$. We extract blobs of connected regions, i.e., regions of the same class label $l_j$ in each image. Since semantically segmented images often show noisy partitioning of the observed scene (holes and disconnected edges), we preprocess them by dilating and eroding the boundaries of each blob. This process removes unwanted noise in the semantically segmented images. Furthermore, the center location $p_j$ of the blobs are extracted and stored alongside the blob labels as vertices $v_j = \{l_j, p_j\}$. In the case that also instance-wise segmentation is available, it can be considered in the blob extraction step, otherwise the extraction operates only on a class basis.

The undirected edges $e_{ij}$ between vertices are formed when fulfilling a proximity requirement, which can be either in image or 3D space. Using a depth channel or the depth information from, e.g., optical flow, the neighborhood can be formed in 3D space, using the 3D locations of the image blobs to compute a Euclidean distance. The process is illustrated for image data in Fig. 3 (top). Then, several image-wise graphs are merged into $G_q$ by connecting vertices of consecutive images using their Euclidean distance, see Fig. 3.
close instances in $G_q$ are merged into a single vertex, at the location of the vertices’ first observation. The strategy of merging vertices into their first observation location is further motivated by the structure of continuous semantic entities, such as streets. This strategy leads to evenly spaced creation of continuous entities’ vertices in $G_q$.

C. Descriptors

X-View is based on the idea that semantic graphs hold high descriptive power, and that localizing a sub-graph in a database graph can yield good localization results. However, since sub-graph matching is an NP-complete problem [9], a different regime is required to perform the graph registration under real-time constraints, i.e., in the order of seconds for typical robotic applications. In this work, we extract random walk descriptors for every node of the graph [21], and match them in a subsequent step. This has the advantage that the descriptors can be extracted and matched in constant or linear time, given a static or growing database-graph, respectively.

Each vertex descriptor is an $n \times m$ matrix consisting of $n$ random walks of depth $m$. Each of the random walks originates at the base vertex $v_j$ and stores the class labels of the visited vertices. Walk strategies, such as preventing from immediate returns to the vertex that was visited in the last step, and exclusion of duplicate random walks can be applied to facilitate expressiveness of the random walk descriptors. The process of random walk descriptor extraction is illustrated in Fig. 4.

D. Descriptor Matching

After both $G_q$ and $G_{db}$ are created, we find associations between vertices in the query graph and the ones in the database graph by computing a similarity score between the corresponding graph descriptors. The similarity measure is computed by matching each row of the semantic descriptor of the query vertex to the descriptor of the database vertex. The number of identical random walks on the two descriptors reflects the similarity score $s$, which is normalized between 0 and 1. In a second step, the $k$ matches with highest similarity score are selected for estimating the location of the query graph inside the database map.

E. Localization Back-End

The matching between query graph and global graph, the robot-to-vertex observations, and the robot odometry measurements result in constraints $\theta_i \subseteq \Theta(p_i, c_i)$ on the vertex positions $p_i$ and robot poses $c_i$ with $\theta_i = e_i^T \Omega_i e_i$, the measurement errors $e_i$, and associated information matrix $\Omega_i$. Specifically these three types of constraints are denoted as $\Theta_M(p_i)$, $\Theta_V(p_i, c_i)$, and $\Theta_O(c_i)$ respectively.
The matching constraints $\Theta_M(p_i) \text{ stem from the semantic descriptor matching of the previous step, while the robot odometry constraints } \Theta_O(c_i) \text{ are created using the robots estimated odometry between consecutive robot poses associated to the localization graph. The robot-to-vertex constraints encode the transformation between each robot-to-vertex observation. Using these constraints, we compute a Maximum a Posteriori (MAP) estimate of the robot pose $c_i$ by minimizing a negative log-posterior $E = \sum \Theta_i$, i.e.,

$$c_i^* = \arg\min_{c_i} \sum \Theta(p_i, c_i) \quad (1)$$

with $\Theta(p_i, c_i) = \{\Theta_M(p_i), \Theta_V(p_i, c_i), \Theta_V(p_i)\}$. This optimization is carried out by a non-linear Gauss-Newton optimizer. Optionally, the algorithm also allows to reject matching constraints in a sample consensus manner, using RANSAC on all constraints between $G_q$ and $G_{db}$, excluding the specific constraints from the optimization objective. We initialize the robot position at the mean location of all matching vertices’ locations from $G_{db}$.

IV. EXPERIMENTS

We evaluate our approach on two different synthetic outdoor datasets, of varying viewpoints, namely forward to rear view, and forward to aerial view. In this section, we present the experimental set-up, the results, and a discussion.

A. Datasets

The first of the used datasets is the public SYNTHIA dataset [22]. It consists of several sequences of simulated sensor data from a car travelling in different dynamic environments and under varying conditions, e.g., weather and daytime. The sensor data provides RGB, depth and pixel-wise semantic classification for 8 cameras, with always 2 cameras facing forward, left, backwards and right respectively. The segmentation provides 13 different semantic classes which are labelled class-wise. Additionally, dynamic objects, such as pedestrians and cars are also labelled instance-wise. We use sequence 4, which features a town-like environment. The total travelled distance is 970 m.

In the absence of suitable public aerial-ground semantic localization datasets, we use the Airsim framework [23] to generate photo-realistic datasets of a simulated rural environment. This environment is explored with a top-down viewing Unmanned Aerial Vehicle (UAV) and a car traversing the streets with forward-facing sensors. Both views provide RGB, depth and pixel-wise semantic classification data in 13 different classes with instance-wise labelling. Furthermore, both trajectories are overlapping with only an offset in z-direction and have a length of 500 m each.

While the travelled distance between two image locations in the Airsim dataset is always 1 m, it varies between 0 m to 1 m in the SYNTHIA dataset. Please note that we used a pre-built environment, i.e., the objects in the environment have not specifically been placed for enhanced performance. Sample images of both datasets are depicted in Fig. [1].

Our approach relies on good semantic representations of scenes. While we do not propose contributions on semantic extraction from raw sensor data, recent advances on semantic segmentation show ever increasing accuracies on visual and depth data [8, 24, 25]. We therefore believe that assuming semantically segmented data as provided appears sensible. Nevertheless, we will discuss noisy input data in Section IV-E.

B. Experimental Setup

We evaluate the core components of X-View in different experimental settings. In all experiments, we evaluate X-View on overlapping trajectories and the provided depth and segmentation images of the data. We focus our evaluation of the different graph settings on the SYNTHIA dataset. We then perform a comparative analysis on both SYNTHIA and Airsim.

In SYNTHIA, we use the left forward camera for building a database map and then use the left backward camera for localization. Furthermore, we use 8 semantic classes of SYNTHIA: building, street, sidewalk, fence, vegetation, pole, car, and sign, and reject the remaining five classes: miscellaneous, sky, pedestrian, cyclist, lanemarking.

Analogously, we use the forward-view of the car in the Airsim dataset to build the database map and then localize the UAV based on the downward camera. Here we use six classes (street, building, car, fence, hedge, tree) and reject the remaining from insertion into the graph (powerline, pool, sign, wall, bench, rock, miscellaneous), as these are usually only visible by one of the robots, or their scale is too small to be reliably detected from the aerial robot.

No assumptions are made on the prior alignment between the data. The ground-truth alignment is solely used for performance evaluation.

C. Localization performance

Firstly, we evaluate the matching performance of the random walk descriptors and matching as described in Sec. III. We therefore evaluate the PR of the localization based on two thresholds. The localization threshold $t_L$ is applied on the distance between the estimated robot position $c_i^*$ and the ground truth position $c_{gt}$. It is set as true, if the distance between $c_i^*$ and $c_{gt}$ is smaller than $t_L$, i.e., $\|c_i^* - c_{gt}\| \leq t_L$, and to false for $\|c_i^* - c_{gt}\| > t_L$. The margin $t_L$ on the locations is required, since $G_q$ and $G_{db}$ do not create vertices in the exact same spot. The same node can be off by up to twice the distance that we use for merging vertices in a graph. Here, we use $t_L = 20$ m. For the PR curves, we vary the consistency threshold $t_s$ that is applied on the RANSAC-based rejection, i.e., the acceptable deviation from the consensus transformation between query and database graph vertices.

The localization estimation yields a positive vote for an estimated consensus value $s$ of $s \leq t_s$ and a negative vote otherwise.
Recall

Precision

n=200, m=3
n=500, m=3
n=500, m=5
BoW

(a) Descriptor parameters.

Recall

Precision

15 frames
30 frames
BoW

(b) Query length.

Recall

Precision

Medium coarse
Coarse
Dense
BoW SYNTHIA
BoW Airsim
BoW SYNTHIA f-f
BoW Airsim

(c) Graph coarseness.

Recall

Precision

X-View SYNTHIA
X-View Airsim
BoW SYNTHIA
BoW Airsim

(d) Datasets comparison.

Fig. 5: PR curves for localization of the rear view semantic images against a database graph built from the forward view on the SYNTHIA dataset. For all plots we accept a localization if it falls within a distance of 20 m from the ground-truth robot position. This threshold corresponds to the value up to which query graph vertices of the same semantic instance can be off from their corresponding location in the database graph, caused by the graph construction technique. Fig. 5a illustrates the effect of different descriptor settings on the localization performance. Fig. 5b shows the effect of increasing the amount of frames used for query graph construction, while Fig. 5c depicts the effect of using coarser graphs, i.e., a large distance in which we merge vertices of same class label. The appearance-based technique used is visual BoW [2]. Finally, in Fig. 5d we show the PR curves for the final descriptor settings on both datasets. The operating points are indicated as red dots.

Fig. 6: Localization performance of X-View on the SYNTHIA and the Airsim data compared to the appearance-based method [2]. The operation points are chosen according to Fig. 5d.

We show the effect of different options on the description and matching using the random walk descriptors, i.e., random walk parameters, graph coarseness, and number of query frames. To illustrate the contrast to appearance-based methods, we also present results on the visual feature place recognition technique based on BoW as implemented by Gálvez-López and Tardos [2] on the datasets’ RGB data.

Furthermore, we show the performance of the full global localization algorithm on the operating point taken from the PR curves. Our performance metric is defined as the percentage of correct localizations over the Euclidean distance between the $c^*_i$ and the $c_{gt}$. As for BoW, we take localization as the best matching image. The localization error is then computed as the Euclidean distance between associated positions of the matched image and the ground truth image. To improve performance of the appearance-based method, we select the operating point with highest performance, i.e., the highest precision in the PR graph.

D. Results

Fig. 5 illustrates localization results in the form of PR curves. While we illustrate the effects of different attributes of X-View in Fig. 5a - 5c as evaluated on SYNTHIA, we then show a comparison on both datasets in Fig. 5d. Fig. 5a depicts the effect of varying the random walk descriptors on the graph. Here, a descriptor size with number of random walks $n = 200$ and walk depth $m$ between $3 - 5$, depending on the size of $G_q$ perform best. Both decreasing $n$ or increasing $m$ leads to a decrease in performance. These findings are expected, considering query graph sizes ranging between $20 - 40$ vertices. Under these conditions, the graph can be well explored with the above settings. Especially a further increase of $m$ leads to a decrease in performance. Descriptors with larger walk depth $m$ significantly diverge between $G_q$ and $G_{db}$, as the random walk reaches the size limits of the $G_q$ and continues exploring already visited vertices, while it is possible to continue exploring $G_{db}$ to greater depth.

Secondly, Fig. 5b presents PR curves for different sizes of $G_q$, i.e., different numbers of frames used for the construc-
TABLE I: Timing results in ms, reporting the means and standard deviations per frame on the best performing configurations on SYNTHIA and Airsim. The timings were computed on a single core of an Intel Xeon E3-1226 CPU @ 3.30GHz.

| Module                      | SYNTHIA       | Airsim        |
|-----------------------------|---------------|---------------|
| Blob extraction             | 2.73 ± 0.65   | 1.76 ± 0.26   |
| Construction of $G_q$       | 337.39 ± 92.81| 257.40 ± 28.30|
| Random Walks Generation     | 1.38 ± 0.82   | 1.07 ± 0.56   |
| Matching $G_q$ to $G_{db}$  | 7.30 ± 4.51   | 4.33 ± 1.25   |
| Localization Back-End       | 22.50 ± 9.71  | 5.15 ± 0.63   |
| Total                       | 371.3 ± 108.5 | 269.71 ± 31.0 |

An increase in the query graph size leads to a considerable increase of the localization performance. Also this effect is expected as $G_q$ contains more vertices, forming more unique descriptors. However, it is also desirable to keep the size of $G_q$ limited, as a growing query graph size requires larger overlap between $G_q$ and $G_{db}$. Furthermore, the computational time for descriptor calculation and matching grows with increased query graph size.

Thirdly, Fig. 5c shows the impact of increased graph coarseness, i.e., larger distances of merging vertices. Here, the coarseness cannot be arbitrarily scaled to low or high values, as it leads to either over- or under-segmented graphs. Our best performing results were obtained with a vertex merging distance of 10 m for the SYNTHIA dataset and 15 m for the Airsim dataset, respectively.

The evaluation using success rates over the localization error is depicted in Fig. 6. X-View has a much higher success rate in multi-view experiments than the appearance-based technique on both datasets at our achievable accuracy of 20 m for SYNTHIA and 30 m on Airsim. These accuracies are considered successful as node locations between $G_q$ and $G_{db}$ can differ by twice the merging distance with our current graph merging strategy. On the considered operation point of the PR curve, X-View achieves a localization accuracy of 85% within 30 m on Airsim, and 85% on SYNTHIA within 20 m. For reference, we also plot the success rate on SYNTHIA for same view-point localization in Fig. 6 denoted f-f. Here we used the left forward facing camera to build the map and localize in it using the right forward facing camera. While the appearance-based technique performs well as expected, also X-View succeeds to achieve a 95% success rate at our success-threshold of 20 m.

Finally, we also report timings of the individual components of our system in Table II. Here, the construction of $G_q$ has by far the largest contribution, due to iteratively matching and merging frames into $G_q$. As the graphs in SYNTHIA consider more classes and smaller merging distances, these generally contain more vertices and therefore longer computational times.

E. Discussion

Global registration of multi-view data is a difficult problem where traditional appearance based techniques fail. Semantic graph representations can provide significantly better localization performance under these difficult perceptual conditions. Our results obtained with X-View show a better localization performance than an appearance-based method, such as BoW. One of the open questions is the effective setting of the system parameters, which we have analysed, but not yet fully explored.

During our experiments, we observed that some of the parameters are dependent on each other. Intuitively, the coarseness of the graph has an effect on the random walk descriptors as a coarser graph contains fewer vertices and therefore deeper random walks show decreasing performance as $G_q$ can be explored with short random walks. On the other hand, an increasing amount of frames used for localization has the reverse effect on the descriptor depth as $G_q$ potentially contains more vertices, and deeper random walks do not show a performance drop as they do for smaller query graphs.

Also the success rate curves indicate that X-View outperforms the appearance based methods particularly in the presence of strong view-point changes. While the appearance-based method fails to produce interesting results for the Airsim dataset, it has a moderate amount of successful localizations on SYNTHIA. On the other hand, X-View has generally higher localization performance and does not show a significant drop in performance between the datasets. While computational efficiency has not been the main focus of our research, the achieved timings are close to the typical requirements for robotic applications.

Finally, our experiments were performed on clean segmentation data, and our current research leads are towards introducing noisy data, e.g., by introducing noise on the graph level or using a semantic segmentation algorithm on the image data. In the graph structure, a set percentage of false vertices and edges can be included, or removed to investigate the performance under noisy data. However, an analysis of noise effects is outside the scope of this paper.

V. CONCLUSIONS

In this paper we presented X-View, a multi-view global localization algorithm leveraging semantic graph descriptor matching. The approach was evaluated on two simulated urban outdoor datasets with drastically different view-points. Our initial results show the potential of using graph representations of semantics for large-scale robotic global localization tasks. Alongside further advantages, such as compact representation and real-time-capability, the presented method is a step towards view-point invariant localization.

Our current research includes the investigation of more sophisticated graph construction methods, and the integration of X-View with a full semantic segmentation and SLAM system to generate loop closures.

ACKNOWLEDGMENT

This work was supported by European Union’s Seventh Framework Programme for research, technological development and demonstration under the TRADR project No. FP7-ICT-609763.
REFERENCES

[1] M. Cummins and P. Newman, “Fab-map: Probabilistic localization and mapping in the space of appearance,” The International Journal of Robotics Research, vol. 27, no. 6, pp. 647–665, 2008.

[2] 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.

[3] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, “Visual Place Recognition: A Survey,” IEEE Transactions on Robotics, vol. 32, no. 1, pp. 1–19, 2016.

[4] A. Gawel, T. Cieslewski, R. Dubé, M. Bosse, R. Siegwart, and J. Nieto, “Structure-based vision-laser matching,” in Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on. IEEE, 2016, pp. 182–188.

[5] A. Gawel, R. Dubé, H. Surmann, J. Nieto, R. Siegwart, and C. Cadena, “3d registration of aerial and ground robots for disaster response: An evaluation of features, descriptors, and transformation estimation,” in Safety, Security, and Rescue Robotics (SSRR), 2017 IEEE International Symposium on, 2017.

[6] E. Stumm, C. Mei, S. Lacroix, J. Nieto, M. Hutter, and R. Siegwart, “Robust visual place recognition with graph kernels,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4535–4544.

[7] Y. Su, F. Han, R. E. Harang, and X. Yan, “A fast kernel for attributed graphs,” in Proceedings of the 2016 SIAM International Conference on Data Mining. SIAM, 2016, pp. 486–494.

[8] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, “A review on deep learning techniques applied to semantic segmentation,” arXiv preprint arXiv:1704.06857, 2017.

[9] S. A. Cook, “The complexity of theorem-proving procedures,” in Proceedings of the third annual ACM symposium on Theory of computing. ACM, 1971, pp. 151–158.

[10] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, “Visual place recognition: A survey,” IEEE Transactions on Robotics, vol. 32, no. 1, pp. 1–19, 2016.

[11] M. J. Milford and G. F. Wyeth, “Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights,” in Robotics and Automation (ICRA), 2012 IEEE International Conference on. IEEE, 2012, pp. 1643–1649.

[12] T. Cieslewski, E. Stumm, A. Gawel, M. Bosse, S. Lyken, and R. Siegwart, “Point cloud descriptors for place recognition using sparse visual information,” in Robotics and Automation (ICRA), 2016 IEEE International Conference on. IEEE, 2016, pp. 4830–4836.

[13] M. Bürki, I. Gilitschenski, E. Stumm, R. Siegwart, and J. Nieto, “Appearance-based landmark selection for efficient long-term visual localization,” in Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on. IEEE, 2016, pp. 4137–4143.

[14] N. Sünderhauf, S. Shirazi, A. Jacobson, F. Dayoub, E. Pepperell, B. Upcroft, and M. Milford, “Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free,” Proceedings of Robotics: Science and Systems XII, 2015.

[15] Z. Chen, A. Jacobson, N. Sünderhauf, B. Upcroft, L. Liu, C. Shen, I. Reid, and M. Milford, “Deep learning features at scale for visual place recognition,” 2017.

[16] W. H. Huang and K. R. Beavers, “Topological map merging,” The International Journal of Robotics Research, vol. 24, no. 8, pp. 601–613, 2005.

[17] D. Marinakis and G. Dudek, “Pure topological mapping in mobile robotics,” IEEE Transactions on Robotics, vol. 26, no. 6, pp. 1051–1064, 2010.

[18] I. Kostavelis and A. Gasteratos, “Semantic mapping for mobile robotics tasks: A survey,” Robotics and Autonomous Systems, vol. 66, pp. 86–103, 2015.

[19] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, “Probabilistic data association for semantic slam,” in Robotics and Automation (ICRA), 2017 IEEE International Conference on. IEEE, 2017, pp. 1722–1729.

[20] N. Atanasov, M. Zhu, K. Daniilidis, and G. J. Pappas, “Semantic localization via the matrix permanent.” in Robotics: Science and Systems, 2014.

[21] B. Perozzi, R. Al-Rfou, and S. Skiena, “Deepwalk: Online learning of social representations.” ACM, 2014, pp. 701–710.

[22] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, “The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3234–3243.

[23] S. Shah, D. Dey, C. Lovett, and A. Kapoor, “Airsim: High-fidelity visual and physical simulation for autonomous vehicles,” in Field and Service Robotics, 2017.

[24] H. Noh, S. Hong, and B. Han, “Learning deconvolution network for semantic segmentation,” in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1520–1528.

[25] V. Badrinarayanan, A. Kendall, and R. Cipolla, “Segnet: A deep convolutional encoder-decoder architecture for image segmentation,” arXiv preprint arXiv:1511.00561, 2015.