M2T: Local Map Descriptor

Hanada Shogo Tanaka Kanji

***Abstract*— Map matching, the ability to match a local map built by a mobile robot to previously built maps, is crucial in many robotic mapping, self-localization, and simultaneous localization and mapping (SLAM) applications. In this paper, we propose a solution to the "map-to-text (M2T)" problem, which involves the generation of text descriptions of local map content based on scene understanding to facilitate fast succinct text-based map matching. Unlike previous local feature approaches that trade discriminativity for viewpoint invariance, we develop a holistic view descriptor that is view-dependent and highly discriminative. Our approach is inspired by two independent observations: (1) The behavior of mobile robots given a local map can often be characterized by a unique viewpoint trajectory, and (2) a holistic view descriptor can be highly discriminative if the viewpoint is unique given the local map. Our method consists of three distinct steps: (1) First, an informative local map of the robot's local surroundings is built. (2) Next, a unique viewpoint trajectory is planned in accordance with the given local map. (3) Finally, a synthetic view is described at the designated viewpoint. Because the success of our holistic view descriptor depends on the assumption that the viewpoint is unique given a local map, we also address the issue of viewpoint planning and present a solution that provides similar views for similar local maps. Consequently, we also propose a practical map-matching framework that combines the advantages of the fast succinct bag-of-words technique and the highly discriminative M2T holistic view descriptor. The results of experiments conducted using the publicly available radish dataset verify the efficacy of our proposed approach. Further, although this paper focuses on the standard 2D pointset map, we believe that our approach is sufficiently general to be applicable to a broad range of map formats, such as the 3D and general view-based maps.**

I. Introduction

Map matching, the ability to match a local map built by a mobile robot to previously built maps, is crucial in many robotic mapping, self-localization, and simultaneous localization and mapping (SLAM) applications [1]–[7]. This paper addresses a general 1-to-N matching problem in which a 2D pointset map is given as a query, and the system searches over a size N map database to find similar database maps that are relevant under rigid transformation.

The classical approach to the map-matching problem is to describe the appearance of each local map using high-dimensional local invariant feature descriptors such as shape features (e.g., polestar feature [8]), and perform feature matching between query and database maps. One major limitation of such an approach is the time consumed comparing the high-dimensional descriptors [9]. One of the most popular approaches used to address this computational cost is the bag-of-words (BoW) approach, in which an unordered collection of vector quantized feature descriptors is used for compact map representation and efficient matching to pre-built maps. Thus far, the BoW approach has been utilized in various map-matching tasks, ranging from view image sequence maps to 3D point cloud maps [5]–[7]. Our proposed approach is also built on the BoW system in [10], in which the BoW framework is successfully applied to the retrieval of 2D occupancy maps using rotation invariant polestar descriptors.

In this paper, we consider the "map-to-text (M2T)" problem, which involves the generation of text descriptions of local map content based on scene understanding to facilitate fast succinct text-based map matching (Fig.1). Unlike previous local feature approaches that trade discriminativity for viewpoint invariance, we develop a holistic view descriptor that is view-dependent and highly discriminative. Our approach is inspired by two independent observations:

- The behavior of mobile robots given a local map can often be characterized by a unique viewpoint trajectory.
- A holistic view descriptor can be highly discriminative if the viewpoint is unique given the local map.

An intuitive example is robots engaged in repetitive/predefined activities such as patrolling or delivery, for which it is natural to memorize views along a unique trajectory for future navigation and place recognition, i.e., exploiting the prior knowledge. Our method consists of three distinct steps:

1) First, an informative local map of the robot's local

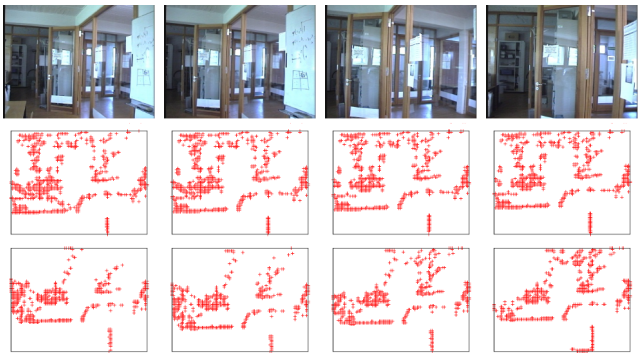


Fig. 1. Local map descriptor. The top row shows four different places in the robot's operating environment. The middle and bottom rows show two independent local maps built by the robot at different times throughout the day, each of which is warped into synthetic views by the proposed viewpoint planner. To facilitate visualization, the figure is shifted and rotated so that the viewpoint is placed at the center and the viewing direction is aligned with the upper direction of the figure. As can be seen, similar synthetic views are produced for similar local maps. Our method converts each view into a holistic view descriptor.

Our work has been supported in part by JSPS KAKENHI Grant-in-Aid for Young Scientists (B) 23700229, and for Scientific Research (C) 26330297.

S. Hanada and K. Tanaka are with Graduate School of Engineering, University of Fukui, Japan. tnkknj@u-fukui.ac.jp

surroundings is built.
2) Next, a unique viewpoint trajectory is planned in accordance with the given local map.
3) Finally, a synthetic view is described at the designated viewpoint.

The success of our holistic view descriptor is based on the assumption that the viewpoint is unique given a local map. Therefore, we also address the issue of viewpoint planning and present a solution that provides similar views for similar local maps. We also propose a practical map-matching framework that combines the advantages of the fast succinct BoW techniques (e.g., [11]), and the highly discriminative M2T holistic view descriptor. The results of experiments conducted using the publicly available radish dataset [12] confirm the efficacy of our proposed approach.

II. Related Work

Existing approaches to map matching can be classified according to which feature descriptors are used, how they are used, and whether the feature approach is global or local. A global feature approach describes the global structure of a scene using a single global feature descriptor (e.g., Gist, HOG). In contrast, a local feature approach describes a scene using a collection of local feature descriptors (e.g., SIFT). In general, both approaches can be used complementarily; however, the focus of this paper is on the latter approach. As stated above, the BoW approach [11], in which a scene is represented by an unordered collection of vector quantized local features, is one of the most popular local feature approaches. Many of the state-of-the-art map-matching systems are built on the BoW approach. There are several related works on various types of features with different scales, including texture [13], object configuration [2], point clouds [14], and polestar [8].

In this paper, we focus on methods that describe not only local feature descriptors but also the local keypoint configuration among them. Among these methods, the part model [15], in which a scene is modeled as a collection of visual parts, is very popular. The model uses information on relative positions as spatial cues to improve the discriminative power of representation. However, existing part-based models primarily focus on a small set of pre-learned parts. Our approach is somewhat similar in concept to the spatial pyramid matching approach in [16], as opposed to the focus on kernel definition and improvement to discriminative power of previous solutions.

Most of the works cited above either explicitly or implicitly assume that the viewpoint trajectory of the mapper robot w.r.t. the local map is unavailable. In contrast, we explicitly use the viewpoint information produced by our viewpoint planner as a cue to compute the holistic view descriptor. The success of our approach is based on the assumption that the viewpoint planner provides a unique viewpoint given a local map; therefore, we also consider the issue of viewpoint planning. To the best of our knowledge, these two issues have not been explored in existing work.

III. Baseline System

This section describes the baseline map-matching system, on which our proposed approach is built, and which is also used as a benchmark for performance comparison in the experimental section, Section V. The main steps in the procedure carried out by the system are as follows: (1) Extraction of appearance features from each local map, (2) translation of the extracted features to a BoW descriptor, and (3) construction/retrieval of the map database from the BoW descriptors. These three steps are explained in detail below.

A. Feature Extraction

We adopt the polestar feature for our purpose because it has several desirable properties, including viewpoint invariance and rotation independence, and has proven effective as a landmark for map matching in previous studies [10]. The extraction algorithm consists of three steps (Fig.2): (1) First, a set of keypoints are sampled from the raw 2D scan points. (2) Next, a circular grid is imposed and centered at each keypoint with different $D = 10$ radius. (3) Finally, the points falling into each circular grid cell are counted and the resulting D-dim vector outputted as the polestar descriptor.

B. BoW Descriptor

Next, we quantize each D-dim polestar vector to a 1-dimensional code termed "visual word". This quantization process consists of three steps: (1) normalization of the D-dim vector by the vector's L1 norm, (2) binarization of each i-th element of the normalized vector into $b_i \in \{0,1\}$, and (3) translation of the binarized D-dim vector into a code or a visual word: $w_a = \sum_i 2^i b_i$. Currently, the threshold for binarization is determined as the mean of all the elements of the vector. In consequence, a map is represented by an unordered collection of visual words $\{w_a \mid w_a \in [1,K]\}$, called BoW. Because we consider D-dim binarized polestar descriptors, the vocabulary size is $K = 2^{10}$.

C. Database Construction/Retrieval

We use the BoW representation for both the database construction and retrieval processes. In the former process, each local map is indexed by the inverted file system, by using each word w_a belonging to the map as an index. In the latter process, all the indexes that have words in common with the query map are accessed and the resulting candidate database maps are ranked based on the frequency or the number of words in common. A frequency histogram of visual words is represented by a K-dim vector when we have K words in the vocabulary. Similarity between a pair of BoW frequency histograms is evaluated in terms of the histogram intersection.

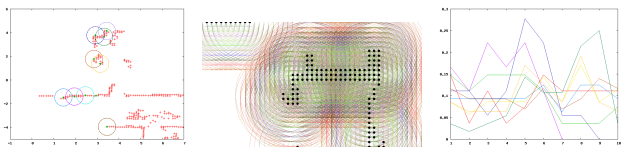


Fig. 2. Extraction of 2D polestar features from a 2D pointset map.

IV. Proposed System

In this section, we outline our proposed system. As mentioned earlier, we built on the baseline system described in Section III, and developed a novel holistic view descriptor. Our method consists of three distinct steps: (1) build a local map, (2) plan a unique viewpoint given the local map, and (3) describe a synthetic view at the planned viewpoint. These three steps as well as the modified map-matching algorithm are detailed in the ensuing subsections.

A. Map Building

We first build a local map from a short sequence of perceptual and odometry measurements; each measurement sequence must be sufficiently long to cover rich photometric and geometric information about the robot's local surroundings. In implementation, each sequence corresponds to the robot's 3 m run. Any map-building algorithm (e.g., FastSLAM, scan matching) can be used to register a measurement sequence into a local map. We start a local map every time the robot's viewpoint moves along the path. This results in a collection of overlapping local maps along the path.

B. Viewpoint Planning

We wish to design a robust planner that provides a unique viewpoint given a local map. (Note that the viewpoint is not necessarily one of the actual viewpoints.) An occupancy grid map is constructed from the 2D pointset map and used as input to our viewpoint planner. Currently, we plan the unique viewpoint near to the center of gravity (CoG) of all the occupancy grid cells. This strategy is inspired by the observation that the CoG can be unique given a local map both in narrow corridors and in rooms.

In implementation, all the viewpoints on the actual viewpoint trajectory are viewed as candidate viewpoints, and among them, the closest candidate to the CoG is selected as the viewpoint for the holistic view descriptor. Subsequently, we determine the viewing direction based on the "dominant direction" [17] of the occupancy grid cells. An intuitive example of the dominant direction is Manhattan world-like environments, where the two dominant directions should be the two orthogonal directions of the manhattans world. To estimate the dominant directions, we adapt the entropy minimization criteria in [17].

C. Holistic View Descriptor

Let us now look at the holistic view at the planned viewpoint and represent it in the BoW form. A key difference of our BoW representation from that of previous works is that we no longer need to rely on view invariant local features that trade discriminativity for view invariance. Instead, we can exploit the knowledge of viewpoint w.r.t. the egocentric local map coordinate to make the holistic descriptor view-dependent, and thus highly discriminative. Our BoW representation comprises appearance words and pose words. The former represents the appearance descriptor of each local feature w.r.t. the local map coordinate. Currently, we simply use the descriptor of each local feature and quantize it into an appearance word, as we did in Section III-B. The latter, pose word, represents the keypoint of each local feature w.r.t. the local map coordinate. During implementation, we quantize the keypoint (x, y) w.r.t. the local map's coordinate to obtain the pose word (w_x, w_y) with resolution quantization step size of 0.1 m. As a result, our visual word is in the form:

$$\langle w_x, w_y, w_a \rangle. \tag{1}$$

D. Map Matching

To index and retrieve the BoW map descriptors, we use the appearance word w_a as the primary index for the inverted file system, while using the pose word (w_x, w_y) as an additional cue for fine matching. The retrieval stage begins with a search of the map collection using the given appearance word w_a as a query to obtain all the memorized feature points with common appearance words, and filter out those feature points whose pose word (w'_x, w'_y) is distant from that of the query feature (w_x, w_y):

$$|w_x - w'_x| > D_{x,y}, \tag{2}$$

$$|w_y - w'_y| > D_{x,y}, \tag{3}$$

to obtain the final shortlist of maps. Currently, we use a large threshold, $D_{x,y} = 1[m]$, to suppress false negatives, i.e., incorrect identification of relevant maps as not being relevant.

V. Experiments

We conducted map-matching experiments to verify the efficacy of the proposed approach. In the ensuing subsections, we first describe the datasets and the map-matching tasks used in the experiments, then present the results obtained and conduct performance comparison against the baseline system.

A. Dataset

For map matching, we created a large-scale map collection from the publicly available radish dataset [12], which comprises odometry and laser data logs acquired by a car-like mobile robot in indoor environments (Fig.3). We created a collection of query/database maps using a scan matching algorithm from each of six different datasets—namely, "abuilding," "albert," "fr079," "run," "fr101," and "kwing"—which were obtained by the mobile robot's 79–295 m travel,

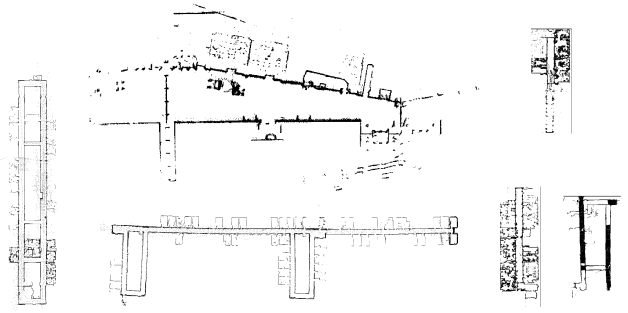


Fig. 3. Datasets used in the experiments: "abuilding," "albert," "fr079," "run," "fr101," and "kwing" from the radish dataset [12].

corresponding to 521–5299 scans. Fig.1 shows examples of the query and database maps. The map collection comprises more than 13,000 maps. Our map collections contain many virtually duplicate maps, which makes map matching a challenging task.

B. Qualitative Results

Recall that the objective of map matching is to find a relevant map from the map database for a local map given as a query. The relevant map is defined as a database map that satisfies two conditions: (1) Its pose is near the query map's pose within a predefined range, where the pose of a map is defined as the CoG of the map's pointset; and (2) its distance traveled along the robot's trajectory is distant from that of the query map, such as in a "loop-closing" situation in which a robot, after traversing a loop-like trajectory, returns to a previously explored location.

For each relevant map pair, a map-matching task is conducted using a query map and a size N map database, which consists both of the relevant map and $(N-1)$ random irrelevant maps. The spatial resolution of the occupancy map is set to 0.1m. We implemented the map-matching algorithm in C++, and successfully tested it on various maps. Figs. 4 and 5 show the results of map matching using the baseline ("BoW") and the proposed ("M2T") systems. As can be seen, fewer false positives appear in the case of the proposed M2T method than the BoW method. This is because many of the incorrect matches are successfully filtered out by the proposed feature, which uses the keypoint configuration as a cue. Quantitative evaluation results for our approach are provided in the next subsection.

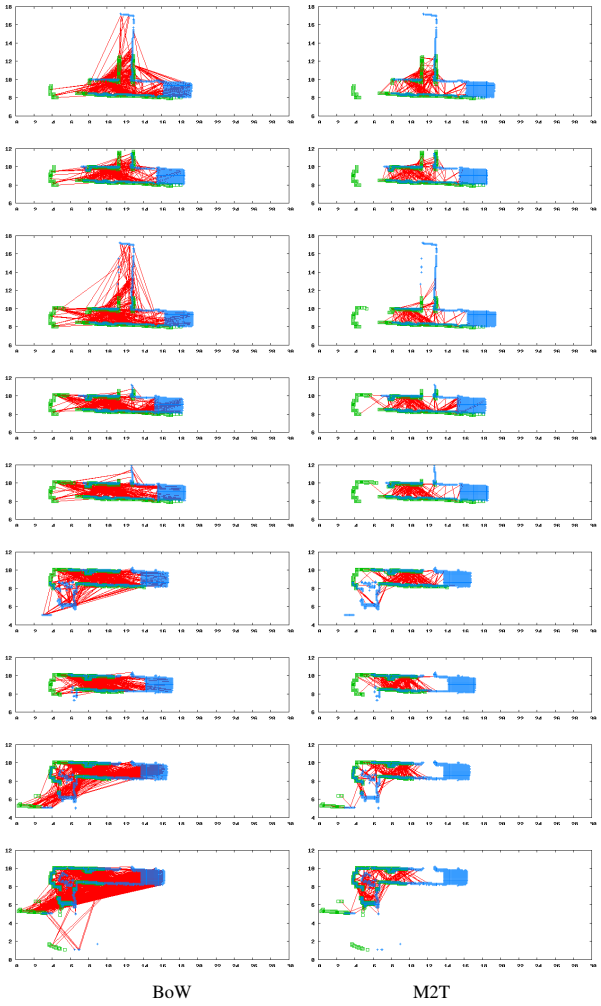


Fig. 4. Examples of matching relevant pairs. Green and blue points indicate the query and the database maps, while the red lines indicate correspondence found by either method. To facilitate visualization, both maps are aligned w.r.t. the true viewpoints.

C. Quantitative Results

For performance comparison, we evaluated the averaged normalized rank (ANR) [18] for both the BoW and M2T methods. ANR is a ranking-based performance measure

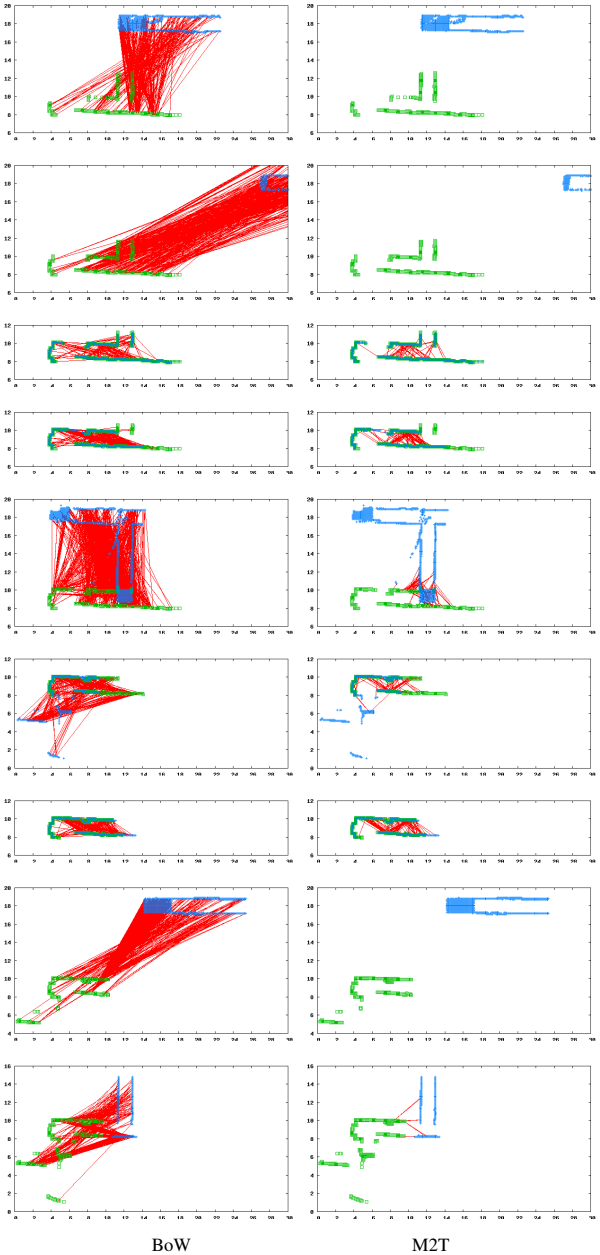


Fig. 5. Examples of matching irrelevant pairs.

in which a lower value is better. To determine ANR, we conducted a number of independent map-matching tasks with different queries and databases. For each task, the rank assigned to the ground-truth database map by a map matcher of interest was investigated and normalized by the database size N. ANR was subsequently obtained as the average of the normalized ranks over all the map-matching tasks. All map-matching tasks were conducted using 13,592 different queries and map databases.

Table I and Fig.6 summarize the ANR performance. The proposed M2T system clearly outperforms the baseline BoW system. By filtering out incorrect matches using the keypoint configuration as a cue, the M2T method was able to successfully perform map matching in many cases, as shown in the figure. In contrast, the BoW system based on appearance words alone often does not perform well, mainly because of the large number of false matches. The above results verify the efficacy of our approach. Fig.7 shows the M2T descriptors with applications to matching a relevant and an irrelevant map pairs.

VI. Conclusions

In this paper, we focused on generating text description of local map content for fast succinct text-based map matching. In particular, we presented a novel holistic view descriptor that describes a synthetic view at a planned viewpoint. We addressed the issues involved in building a local map, planning viewpoints, and computing the holistic view descriptor. The results of experiments conducted with the publicly available radish dataset confirm the efficacy of our proposed approach. In the future, we plan to use the presented M2T system for long-term operation of robots in familiar environments. Although this paper focused on the standard 2D pointset map, we believe our approach is sufficiently general to be applicable to a broad range of map formats, such as the 3D point cloud map, as well as general view-based maps.

TABLE I

Summary of ANR performance [%].

dataset	abuilding	albert	fr079	fr101	kwing1	run1
BoW	29.3	35.0	24.0	32.6	18.7	41.7
M2T	**7.0**	**26.6**	**17.1**	**16.7**	**3.6**	**15.2**

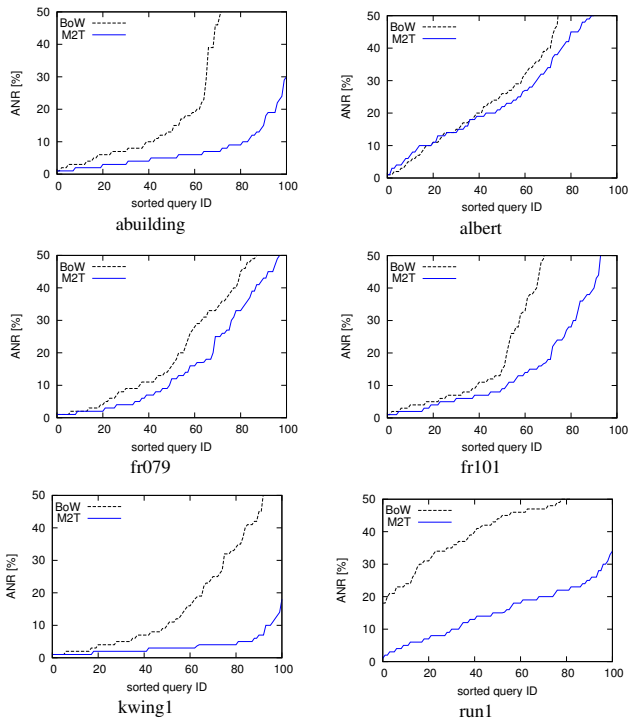


Fig. 6. ANR performance for each dataset (horizontal axis: sorted query map ID, vertical axis: ANR in [%]).

References

[1] T. Botterill, S. Mills, and R. Green. Speeded-up bag-of-words algorithm for robot localisation through scene recognition. In *IVCNZ08*, pages 1–6, 2008.

[2] Arnau Ramisa, Adriana Tapus, David Aldavert, Ricardo Toledo, and Ramon Lopez de Mantaras. Robust vision-based robot localization using combinations of local feature region detectors. *Autonomous Robots*, 27(4):373–385, 2009.

[3] John McDonald, Michael Kaess, Cesar Dario Cadena Lerma, José Neira, and John J. Leonard. Real-time 6-dof multi-session visual SLAM over large-scale environments. *Robotics and Autonomous Systems*, 61(10):1144–1158, 2013.

[4] Sebastian Thrun, Dieter Fox, Wolfram Burgard, and Frank Dellaert. Robust monte carlo localization for mobile robots. *Artif. Intell.*, 128(1-2):99–141, 2001.

[5] M. Cummins and P. Newman. Appearance-only slam at large scale with fab-map 2.0. *Int. J. Robotics Research*, 30(9):1100–1123, 2011.

[6] Adrien Angeli, David Filliat, Stephane Doncieux, and Jean arcady Meyer. A fast and incremental method for loop-closure detection using bags of visual words," conditionally accpeted for publication in. *IEEE Trans. Robotics, Special Issue on Visual SLAM*, 2008.

[7] Tom Botterill, Steven Mills, and Richard Green. Speeded-up Bag-of-Words algorithm for robot localisation through scene recognition. pages 1–6, 2008.

[8] E. Silani and M. Lovera. Star identification algorithms: Novel approach & comparison study. *IEEE Trans. Aerospace and Electronic Systems*, 42(4):1275–1288, 2006.

[9] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Packing bag-of-features. In *Proc. IEEE Int. Conf. Computer Vision*, 2009.

[10] Kanji Tanaka and Kensuke Kondo. Multi-scale bag-of-features for scalable map retrieval. *JACIII*, 16(7):793–799, 2012.

[11] Sivic J. and Zisserman A. Video google: a text retrieval approach to object matching in videos. *Proc. IEEE Int. Conf. Computer Vision*, pages 1470–1477, 2003.

[12] Andrew Howard and Nicholas Roy. The robotics data set repository (radish), 2003.

[13] Arnau Ramisa, Adriana Tapus, David Aldavert, Ricardo Toledo, and Ramon Lopez De Mantaras. Robust vision-based robot localization using combinations of local feature region detectors. *Auton. Robots*, 27(4):373–385, 2009.

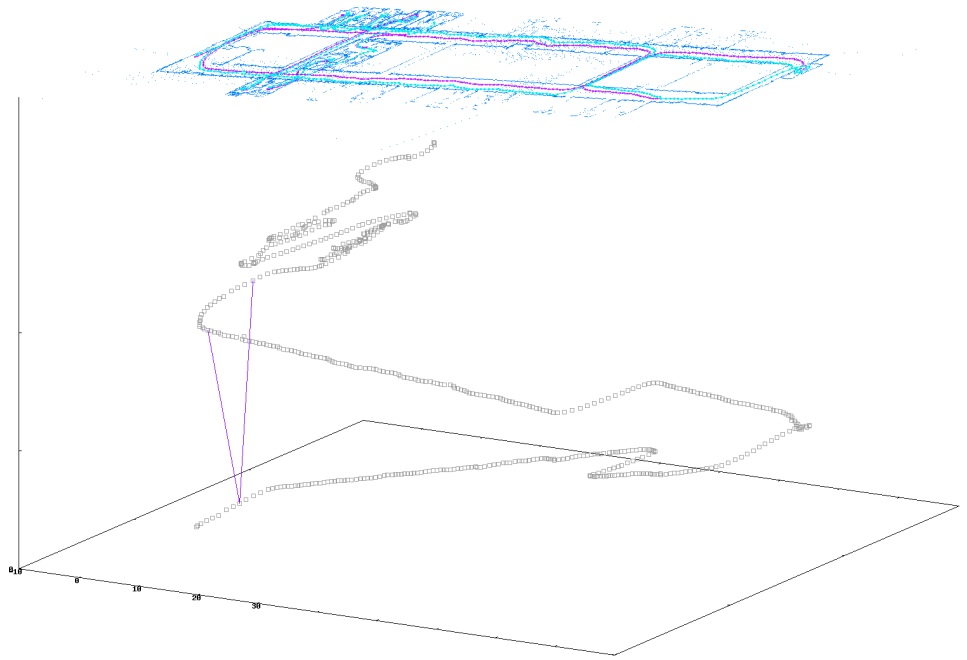
[14] Daniel Huber, Owen Carmichael, and Martial Hebert. 3d map reconstruction from range data. In *Proc. IEEE Int. Conf. Robotics and Automation*, pages 891 – 897, 2000.

[15] Dongfeng Han, Wenhui Li, and Zongcheng Li. Semantic image classification using statistical local spatial relations model. *Multimedia Tools Appl.*, 39(2):169–188, 2008.

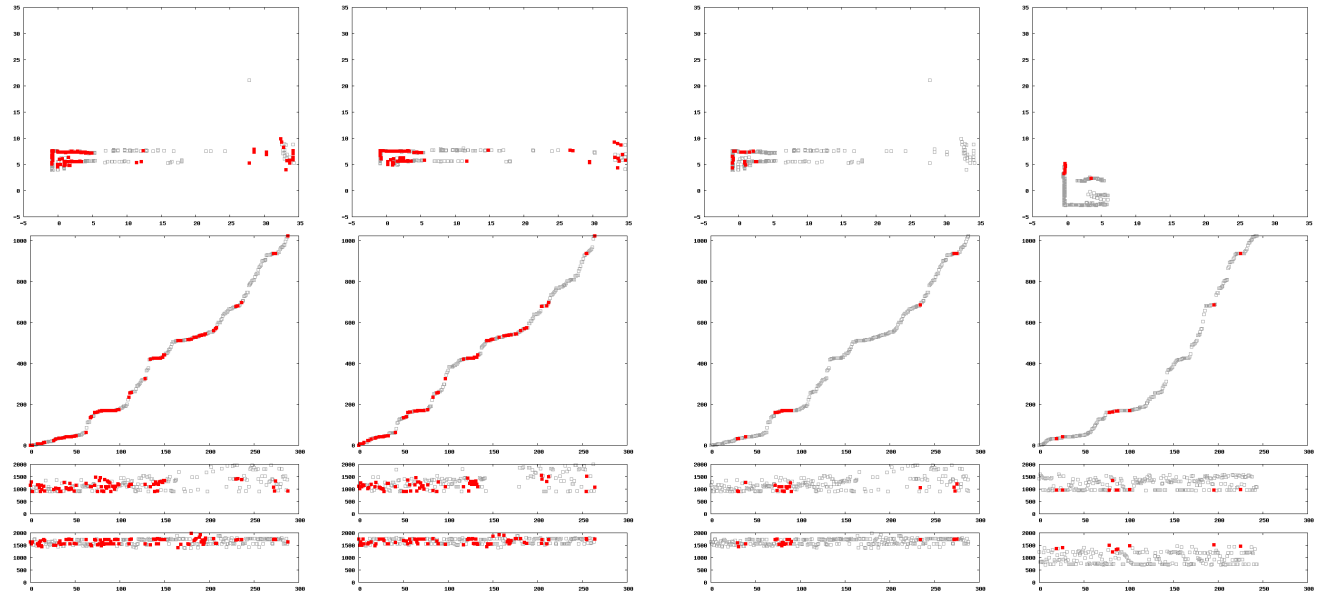
[16] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169 – 2178, 2006.

[17] Sven O. and Markus V. Robust single view room structure segmentation in manhattan-like environments from stereo vision. In *IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 5315–5322, 2011.

[18] Hanada Shogo and Tanaka Kanji. Partslam: Unsupervised part-based scene modeling for fast succinct map matching. In *IEEE/RSJ Int. Conf. IROS*, pages 1582–1588. IEEE, 2013.



a



8-th viewpoint | 378-th viewpoint | 8-th viewpoint | 360-th viewpoint

b

Fig. 7. Examples of matching M2T local map descriptors. (a) The local maps and the viewpoint trajectories. The top panel shows a collection of local maps (blue dots), the viewpoint trajectory (purple points), as well as the "unique" viewpoints planned by our framework (blue points). The bottom panel shows the sequence of planned viewpoints in the xyt space (t: viewpoint ID). (b) Matching M2T descriptors between a relevant map pair (first and second columns) and between an irrelevant map pair (third and fourth columns). The relevant and irrelevant pairs are also indicated by purple line segments in Fig. 7a. For each column, the matched visual words are highlighted in red. Each row shows from top to bottom, appearance word w_a, pose words w_x, w_y, and the text description $\{\langle w_x, w_y, w_a \rangle\}$.