Image Sequence Partitioning for Outdoor Mapping

Hemanth Korrapati, Jonathan Courbon, Youcef Mezouar, Philippe Martinet

***Abstract*— Most of the existing appearance based topological mapping algorithms produce dense topological maps in which each image stands as a node in the topological graph. Sparser maps can be built by representing groups of visually similar images as nodes of a topological graph. In this paper, we present a sparse topological mapping framework which uses Image Sequence Partitioning (ISP) techniques to group visually similar images as topological graph nodes. We present four different ISP techniques and evaluate their performance. In order to take advantage of the afore mentioned maps, we make use of Hierarchical Inverted Files (HIF) which enable efficient hierarchical loop closure. Outdoor experimental results demonstrating the sparsity, efficiency and accuracy achieved by the combination of ISP and HIF in performing loop closure are presented.**

I. INTRODUCTION

Many powerful loop closing techniques for topological maps have been introduced recently [10], [11], [9], [12]. Most of them produce dense topological maps, in which every acquired image stands as a node in the topological graph. In dense maps, loop closure time increases linearly with the increase in number of nodes/images. Consequently, in maps with huge number of nodes loop closure becomes demanding in terms of computational time. The problem can be tackled by sparse topological maps - maps that represent the same environment with fewer nodes. Each node in a sparse map must represent a group of sequential and visually similar images. These maps with fewer nodes enable efficient loop closure and map merging. Each node in a sparse topological map can be understood as a place - region of an environment throughout which visual appearance remains more or less constant. Breaking a sequence of images acquired by the robot into nodes/places is called Image Sequence Partitioning (ISP).

We present four ISP techniques, and evaluate each of them in terms of generating sparse and accurate maps. Out of four the first two techniques namely GIST [14], [13] and Optical flow [15] were adapted from existing work. The remaining two techniques namely Local Feature Matching (LFM) and Common Important Words (CIW) are our contributions.

To facilitate efficient loop closure on sparse topological maps, we propose a special data structure called Hierarchical Inverted File (HIF) for feature storage. As opposed to the traditional inverted files [18], [17], HIFs store features hierarchically in two levels - node level and image level. HIFs enable loop closure at two resolutions - a coarse node level loop closure which finds the most similar node and a finer image level loop closure which pin-points the most similar image inside a node.

Experimentation was performed in outdoor urban environments using an omnidirectional camera. Sparsity and accuracy of maps constructed using different ISP techniques are evaluated and the power of HIF representation in time efficient loop closure is demonstrated.

II. RELATED WORK

Scene Change Detection and Key Frame Selection for video segmentation and abstraction [7], [6] have similar goals as that of ISP. They try to represent a video with fewer images called key frames whenever there is a sufficient change in the scene and most of them focus on video compression domain. The major difference between these video abstraction problems and mapping is that mapping demands localization of a query image which is obtained at a previously visited place, but with varied illumination, viewpoint, and a possible occlusion. Hence, video segmentation techniques using pixel-wise intensity measures and global image features like histograms, motion based segmentation cannot be applied to our problem.

Quite a few loop closure techniques for topological maps have been proposed recently for both indoor [4], [12], [10], [11], [2], [22] and outdoor environments [9], [8], [13], [15]. However, only a few of them concentrate on generating sparse maps.

In [4], [5] topological maps are built for indoor environments. They segment the topological graph of the environment using normalized graph-cuts algorithm resulting in subgraphs corresponding to convex areas in the environment. In [2] SIFT features were used to perform matching over a sequence of images. They detect transitions between individual indoor locations depending on the number of SIFT features which can be successfully matched between the successive frames. In [3] fingerprint of an acquired image is generated using omnidirectional image and laser readings, and these fingerprints are used in loop closure. If the similarity is above a threshold the image is added to the existing node and if not a new node is formed. Change point detection has been used for indoor place segmentation in [19]. All the above works experimented on indoor environments which contain convex spaces (like rooms) and are relatively easier to be partitioned when compared to outdoor environments.

A sparse topological mapping framework using incremental spectral clustering has been presented in [1]. Nodes are constructed using incremental spectral clustering over the affinity matrix of the images, producing a sparse topological graph. Another ISP technique was presented in [15] which used optical flow to discover change in environmental appearance.

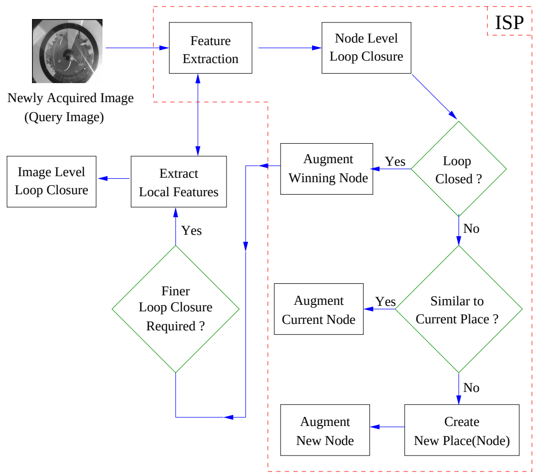


Fig. 1: A global modular view of our topological mapping framework.

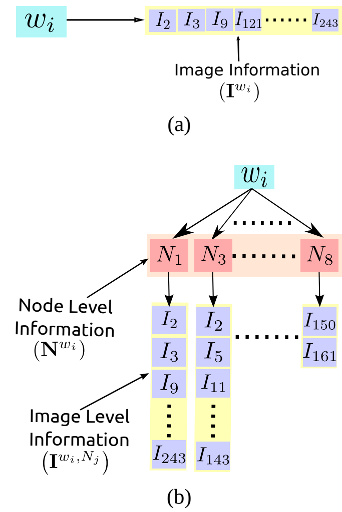


Fig. 2: (a) A traditional inverted file with its flat structure. (b) A hierarchical inverted file with place node indexes $\{N_1, N_2, \cdots\}$ at the first level and their associated images $\{I_1, I_2, \cdots\}$ at the second level.

III. HIERARCHICAL MAPPING FRAMEWORK

In our framework a topological map is formulated as a graph $\mathbf{T} = (\mathbf{N}, \mathbf{E})$ where $\mathbf{N} = \{N_1, N_2, \ldots\}$ is the set of nodes and $\mathbf{E}$ the set of edges. We suppose that visual features extracted from images can be quantized into visual words of a vocabulary $\mathbf{V} = \{w_1, w_2, \ldots\}$ where each w_i is a visual word. In the rest of the paper, variables representing sets and vectors are depicted in bold letters and all other variables in normal font.

The overview of our topological mapping framework is depicted in figure 1. Given a query image (a newly acquired image), we extract appearance features which are used to perform a node level loop closure. On a successful node level loop closure, the image is added to the winning node. A finer image level loop closure is performed if better accuracy is desired. However, if a node level loop closure does not occur, the image is compared with the current place node. If similar it is augmented to the current node. Otherwise a new node is created with the query image as its initial member. The steps discussed thus far are collectively referred as Image Sequence Partitioning (ISP), shown by a red bounding box in figure 1. Each node consists of a set of representative features constructed from all the member images' features. On addition of each new image to a node, the representative features set is updated. The update procedure differs based on the ISP technique used. Representative feature sets are used in computing the similarity of a node with that of the query image which we refer to as node-image similarity.

Hence, given a query image, we should be able to perform node level loop closure, image level loop closure and update representative feature set of the similar node. We make use of Hierarchical Inverted Files for performing these tasks. The following subsections describe in detail how each of these tasks is carried out.

A. Hierarchical Inverted Files

Hierarchical Inverted Files are derivatives of traditional inverted files which are being used in various recent vision based loop closure [10], [11], [12] and object recognition applications [17]. Inverted files encode the occurrence frequency information of visual words in images, and can be used for efficient loop closure. Each visual word of the vocabulary is associated with an inverted file which enlists the images containing the word. Given a query image I_q and its visual word set $\mathbf{W_q} = \{w_1, w_2, \cdots, w_p\}$, inverted files corresponding to each $w_i \in \mathbf{W_q}$ are used to generate similarities scores of reference images (images already in the map) with I_q. These similarities are used for further filtering. Detailed descriptions can be found in [8], [10], [11].

Inverted files discussed above are better suited for loop closure for maps stored as a list of images. However, our map is stored hierarchically as nodes which in turn store corresponding images. To suit our needs we propose a modified hierarchical version of inverted files called Hierarchical Inverted Files (HIF) illustrated in figure 2.

Every visual word w_i in the vocabulary has an associated HIF. HIFs possess two levels of hierarchy. The first level or node level contains node information - a set of nodes/places $\mathbf{N}^{w_i}$ in which the visual word w_i previously occurred. Each node entry $N_j \in \mathbf{N}^{w_i}$ in the first level is associated with a set of images $\mathbf{I}^{w_i,N_j}$ which belong to the node N_j in the map and contain the word w_i. These sets of images $\{\mathbf{I}^{w_i,N_j}\}$ constitute the second level of Hierarchy referred as image level. HIF representation enables efficient real-time loop closure at coarse (node level) and fine (image level) resolutions which are discussed in the next subsections.

B. Node Level Loop Closure

Node level loop closure aims to find the node (place) in the map which is most similar to the query image. For each visual word of the query image $w_i \in \mathbf{W_q}$, its node level occurrence information $\mathbf{N}^{w_i}$ is retrieved from the associated HIF. As mentioned before, $\mathbf{N}^{w_i}$ is a set of nodes in wich the visual word w_i has previously occurred. This node information is used to generate node similarity (likelihood) scores using Tf-Idf (Term Frequency-Inverse Document Frequency [23]) in a way similar to [10]. The nodes with similarity scores above a threshold are chosen as the set of winning nodes $\mathbf{N}^*$. If a finer resolution is desired, the winning nodes are further sent for image level loop closure.

Node level loop closure is performed using only a part (node level information) of the HIF and hence faster than the traditional techniques. The computational gain offered by this technique becomes prominent in huge maps holding thousands of images.

C. Image Level Loop Closure

Image level loop closure finely pin-points the reference images in the topological graph which are highly probable to be similar to the query image. The reference image set $\mathbf{I^R}$ is the union of sets of images belonging to the nodes in $\mathbf{N}^*$ obtained as in (1).

$$\mathbf{I^R} = \{\mathbf{I^{N_k}} \mid \forall_k N_k \in \mathbf{N}^*\} \tag{1}$$

Where, $\mathbf{I}^{N_k}$ is the set of images belonging to the node N_k. The reference images $\mathbf{I^R}$ and their previous occurrence information obtained from HIFs are used to compute Tf-Idf based image similarity scores. These similarity scores are smoothed and used as likelihoods for a recursive bayesian filter to generate the posterior probabilities. To keep the discussion simple, we do not present the details of the bayesian filter model that has been adapted from [10]. The image(s) with posterior probability more than 95% are selected to be the winning images $\mathbf{I}^*$. If $\mathbf{I}^*$ contains more than one image, a RANSAC based epi-polar geometry verification is used to select the right image.

IV. IMAGE SEQUENCE PARTITIONING - TECHNIQUES

Image sequence partitioning mainly consists of three modules - node level loop closure and query image similarity to current place and representative feature set update. In this section, we briefly introduce four ISP techniques and discuss how each of the three afore mentioned modules are executed.

A. GIST

GIST is a global image feature introduced by [14] as a part of a context-based vision system for object and place recognition. GIST produces a feature vector corresponding to the average response to steerable filters employed at different scales and orientations computed over small sub-windows. The advantages of GIST are fast descriptor computation and compact representation of the spatial structure of images. We use omni-gist, a variant of GIST as proposed in [13] by dividing each image into 4 parts and computing four corresponding GIST descriptors. Hence an omni-gist feature takes the form $g = \{g_1, g_2, g_3, g_4\}$, where each element g_i is a GIST feature. The representative feature of a node is the centroid of all member image omni-gist features. Please note that the node level loop closure of section III-B is not directly applicable to the GIST based ISP technique as GIST uses global image features.

Node Level Loop Closure: Computed in two steps. In step I the image's four gist descriptors are quantized using a gist vocabulary. In stage II cyclic-euclidean distance [13] is computed between the image's visual word centroids and the representative feature of the node. Minimum cyclic-euclidean distance less than a threshold T_{gist}, indicates a match. Similarity of the query image to the current node is also evaluated in the same way.

B. Optical Flow

Optical flow based key place detection algorithm of [15] has been used. The basis of this algorithm is the fact that change in scene triggers significant change in optical flow. Image corners are detected using Canny edge detector and Lucas-Kanade algorithm is used to track the corners across frames. Optical flow is only computed using features that are present over several consecutive frames. The vehicle velocity is subtracted from the optical flow values to normalize the optical flow vectors. Optical flow vectors are converted into polar coordinates and the mean optical flow vector length is computed. An image is considered similar to the current node if the change in mean optical flow vector length is less than a threshold T_{of}.

For the node level loop closure, representative feature set is the union of quantized visual words of all the member images of the node.

C. Local Feature Matching

Local Feature Matching method splits an image sequence based on the percentage of common local image features over an image subsequence. Starting with an initial image, every subsequent image's features are matched with that of the initial image. As long as the match percentage is below a certain threshold T_{lfm}, the matched images including the initial image are assigned to the same partition(node). This is the way by which similarity with the current node is computed.

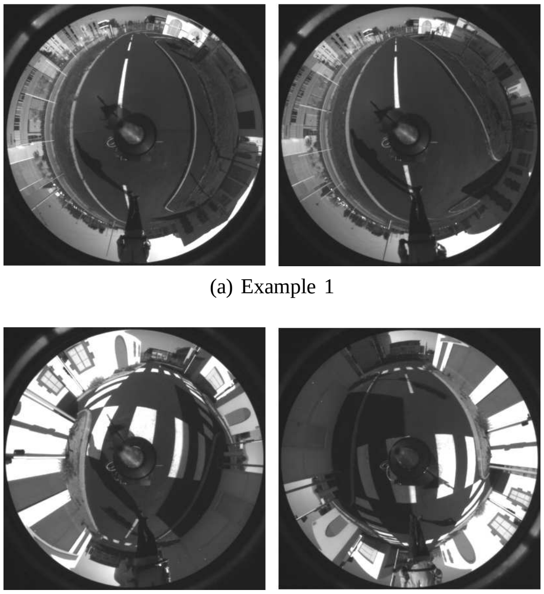
Representative feature set for node level loop closure is the union of member image visual words as in the above technique.

D. Common Important Words

This technique exploits the fact that a set of common features (visual words) are shared by all the images of a place. A common visual word set $\mathbf{C_1}$ is initiated with the words from the first image $I_{initial}$. Thereafter with each subsequent image I_j, the common word set is updated to $\mathbf{C_j}$ such that it satisfies the condition in 2

$$\forall_l w_l \in \mathbf{C_j} \mid freq(w_l) - j \leq \delta \tag{2}$$

where, $freq(w_l)$ is the occurrence frequency of the word w_l in the images from $I_{initial}$ through I_j. The variable δ is the cushion variable which accounts for word missed out in a few images due to noise or quantization errors. With



(a) Example 1

(b) Example 2

Fig. 3: Two loop closure situations of Dataset-6560. Figure 3a shows a loop closure with the robot traversing in the same direction as that of previous traversal and figure 3b shows a loop closure with the robot traversing in reverse direction.

each update of the common word set certain features are added or deleted pertaining to the condition 2. Every set $\mathbf{C_j}$ is compared with the initial set $\mathbf{C_1}$ and if the matching score is above a threshold Th_{ciw}, the image I_j is added to the node or else a new node is formed. Representative feature set is obtained in the same way as the above techniques.

V. EXPERIMENTS

Our experimental platform comprises of a Pioneer P3DX robot equipped with an omnidirectional camera. A laptop equipped with an Intel Centrino dual core processor running linux is used for data processing. The experiments were carried out in our artificial urban environment - PAVIN. PAVIN contains roads, artificial buildings, and a variety of real-world road settings like junctions, traffic lights, round abouts, curved roads and dead ends.

Omnidirectional images were acquired at a frame rate of 2 fps, as the robot moves along a manually controlled trajectory. Two datasets - Dataset-6560 and Dataset-11200 containing 6560 and 11200 images respectively are acquired over two days under varying illuminations. There are 52 and 71 loop closure situations in Dataset-6560 and Dataset-11200 respectively. The number of loop closures were determined by manually examining the datasets. Two loop closures situations on PAVIN are shown in figure 3.

A. ISP - Parameters

This subsection discusses the parameters and thresholds used by the four ISP techniques.

GIST features are computed using 4×4 blocks and 8 orientations per scale, and quantized into gist visual words using a 25 word gist vocabulary. The cyclic-euclidean distance threshold $T_{gist} = 0.30$ has been used for node-image similarity(for node level loop closure) evaluation.

For optical flow based technique, we used the same parameters as in [15] except for the change threshold $T_{of} = 0.11$. This value was an optimal choice, as values lesser or greater than this were over or under segmenting the environment.

The feature matching threshold $T_{lfm} = 60\%$, has been used in local feature matching technique. In common important words based method, we used $Th_{ciw} = 40\%$ and $\Delta = 5$.

$128-$dimensional non-rotation invariant SURF features are used to construct a visual vocabulary of size 7000 as in [17]. Average number of SURF features per image are 370 and 330 in Dataset-6560 and Dataset-11200 respectively.

TABLE I: NUMBER OF NODES IN TOPOLOGICAL MAPS

	GIST	OF	LFM	CIW
DATASET-6560	523	402	473	705
DATASET-11200	895	689	723	1016

B. ISP - Sparsity

The number of nodes in a topological map indicate its sparsity. Table I shows the number of nodes obtained by using omni-gist (GIST), Optical Flow (OF), Local Feature Matching (LFM) and Common Important Words (CIW). We can see that optical flow (OF) generates the most sparse maps. However, the number of nodes is much greater than that of the [15]. The explanation lies in the fact that in [15], an UAV has been used for mapping where as we used a ground robot. A ground robot passes several structures like buildings,etc.., which are close to the camera, and as a result environmental appearance changes more frequently than in [15]. In [15], an UAV flies at a greater height and also distant from the structures present in the environment, as a result of which scene change detection is less often. Local feature matching (LFM) technique produced the second most sparse maps while omni-gist (GIST) common important words (CIW) follow.

C. Accuracy

The most sparse map may not guarantee an accurate map. Only those maps with accurate place partitioning are accurate and can lead to accurate loop closures. Thus a good mapping technique is one which provides an optimal combination of sparsity and accuracy. In this subsection, we discuss the accuracy of the maps produced by the four techniques, based on the number of loop closures and the false positives.

Given a query image, first we perform loop closure at node level and then at image level if more accuracy is required. We analyse the accuracy of loop closures at node and image level separately. Tables II(a) and II(b) show the number of loop closures detected and the number of false positives obtained on Dataset-6560 and Dataset-11200 respectively. We can see that Local Feature Matching (LFM) algorithm detected most

TABLE II: NODE LEVEL LOOP CLOSURE ACCURACY

(a) Dataset-6560

	#(LC)	#(FP)
GIST	34	3
OF	44	7
LFM	49	4
CIW	40	5

(b) Dataset-11200

#(LC)	#(FP)
47	5
59	11
68	6
53	8

TABLE III: COMPUTATION TIMES (in ms)

(a) LFE, QUANT, ILLC

	LFE+QUANT	ILLC
GIST	200 + 85	29
OF	200 + 85	19
LFM	200 + 85	22
CIW	200 + 85	25

(b) NLLC

Dataset-6560	Dataset-11200
150 + 90	150 + 140
196 + 60	196 + 55
60	55
40	50

of the loop closures with a few false positives. Optical flow (OF) and common important word (CIW) techniques have almost similar performance but slightly more false positives. Omni-gist (GIST) technique detects the least number of loop closures and hence the least performant of the four techniques.

The success of LFM can be attributed to its direct dependence on the same features for both node level and image level loop closures. Optical flow(OF) depends on different features for node level and image level loop closures because of which the loop closure accuracy is affected. CIW technique also uses local features for both levels of loop closures but still does not achieve the best performance. CIW uses only a set of important visual words to represent a place(node) which only form a fraction of total number of image features. Another problem is that the visual words are not stable over time because of quantization errors which are sensitive to view changes when objects are very close to the camera. Omni-gist (GIST) depends upon GIST features which encode rough spatial information. But most of the places in our dataset contain similar spatial structure with a road bordered by curbs and grass, buildings on the sides of the road. The less discriminative power of GIST made it incapable to achieve high accuracy in loop closure on our data.

Image level loop closure accuracy depends on the accuracy of node level loop closure. If node level loop closure selects an inaccurate set of winning nodes, consequently image level loop closure also becomes inaccurate. However in case of an accurate node level loop closure, we have observed that 99% accuracy was possible in image level loop closure irrespective of the ISP technique.

D. *Computational Time*

Table IIIa shows average computational time (in milliseconds) taken by Local Feature Extraction (LFE), Quantization (QUANT) and Image Level Loop Closure (ILLC) to process each query frame. For each ISP technique the times taken by each of these processes stay almost constant. Obviously, local feature extraction ($200ms$) and quantization ($85ms$) time is constant for all the techniques. Computation time of image level loop closure using all the ISP techniques are too low because of reduced search space (only the winning nodes from node level loop closure).

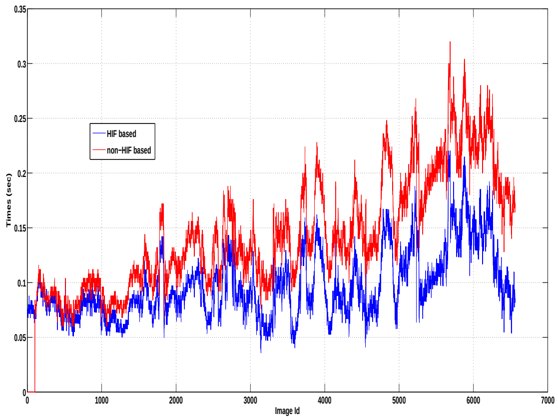
Node Level Loop Closure (NLLC) times are discussed separately as it varies with map size and ISP technique as shown in table IIIb. The values in the table are the node level loop closure times for processing each query image after the maps contained 6560 and 11200 images respectively. For GIST based mapping, NLLC involves extracting omni-gist features for the query image (takes $150ms$) and some additional time to pick the best matches from all the nodes of the graph sequentially. In OF based mapping, NLLC involves computing mean optical flow vector length, (takes $196ms$) and some additional time to perform a node level loop closure. In LFM and CIW, NLLC involves performing a node level loop closure using the local features and visual words.

Local features have to be computed for all the mapping techniques in order to be able to perform a fine level loop closure. But LFM and CIW techniques use local features also for NLLC as opposed to the other two techniques which use additional features. As LFM and CIW escape the additional computational burden they are faster than the other two map building techniques.

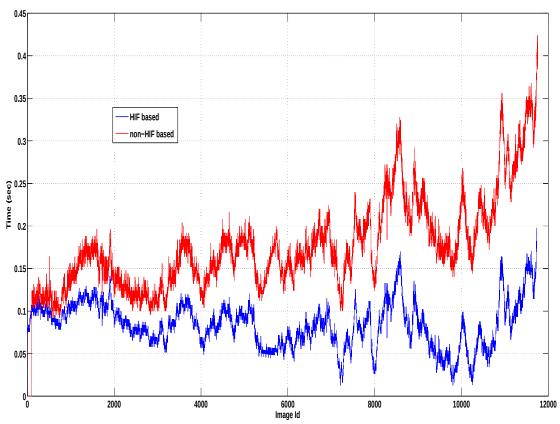
From the above discussion, one can notice that LFM has the best performance in terms of loop closure accuracy and computational time. However its not the best technique in terms of false positive count and sparsity of topological graph produced. In terms of sparsity, which is one of the main objectives of our mapping, LFM is the second best loosing the first place by a very small margin. By looking at the overall performance, LFM when combined with HIF produces optimally sparse topological maps with high loop closure speeds. Hence, we show some more results obtained by using LFM with HIF-based representation and compare them to the traditional loop closure technique [10]. Figures 4(a) and 4(b) show graphs comparing the loop closure times of our HIF-based method and non-HIF based method discussed in section III-A. It can be seen that the loop closure time of non-HIF based method increases relatively steeply with the increase in the number of images in the map, while HIF method increases much slowly. We can also observe that the non-HIF loop closure time for Dataset-11200 increases less steeply than that of Dataset-6560. This happened because the average number of features of Dataset-11200 is lesser than that of Dataset-6560 and as a result it takes lesser time to process each reference frame. Efficiency of our HIF-based method can be attributed to the combination of sparse topological mapping and HIFs for efficient map storage. The representational power of HIFs save a lot of computation involved in loop closure especially when the maps are huge.

VI. CONCLUSION

We proposed a sparse topological mapping framework involving two levels of loop closure. We used Image Se-



(a) Loop Closure time - Dataset-6560



(b) Loop Closure time - Dataset-11200

Fig. 4: Loop closure computation times of non-HIF based loop closure and HIF based loop closure on maps generated using LFM on our datasets.

quence Partitioning for building sparse topological maps. We compared the loop closure performance for four ISP methods. For efficient loop closure over the sparse maps, we introduced novel indexing structures called Hierarchical Inverted Files, which enabled loop closure at desired resolution at decent speeds. From our experiments, we have seen that the local feature matching approach provides good accuracy and sparsity in performing loop closures at high speeds.

ACKNOWLEDGMENT

This work has been funded by The French National Research Agency (L'Agence nationale de la recherche) under the projects CityVip (ANR-07-TSFA-013-01) and R-Discover (ANR-08-CORD-019).

REFERENCES

[1] C. Valgren, T. Duckett, and A. Lilienthal., *Incremental spectral clustering and its application to topological mapping*, Proceedings of IEEE Intl. Conf. on Robotics and Automation (ICRA), pages 42834288, 2007.

[2] J. Kosecka, F. Li, and X. Yang, *Global localization and relative positioning based on scale-invariant keypoints*, Robotics and Autonomous Systems, vol. 52, no. 1, pp. 2738, 2005.

[3] A. Tapus and R. Siegwart, *Incremental robot mapping with fingerprints of places*, Proceedings of IEEE/RSJ Intl. Conf. on Intelligent Robotics and Systems (IROS), pp. 2429-2434, 2005.

[4] Z. Zivkovic, O. Booij, and B. J. A. Kröse, *From images to rooms*, Robotics and Autonomous Systems, vol. 55(5), pp. 411418, 2007.

[5] Z. Zivkovic, B. Bakker, and B. Kröse, *Hierarchical map building using visual landmarks and geometric constraints*, Proceedings of IEEE/RSJ Intl. Conf. on Intelligent Robotics and Systems (IROS), pp. 2480-2485, 2005.

[6] Cees G.M. Snoek and Marcel Worring *Multimodal Video Indexing: A Review of the State-of-the-art* Springer Journal of Multimedia Tools and Applications, volume 25, pp. 5-35, 2005.

[7] Ba Tu Truong, Svetha Venkatesh, *Video abstraction: A systematic review and classification*, ACM Transactions on Multimedia Computing, Communications, and Applications, Volume 3 Issue 1, February 2007.

[8] Mark Cummins and Paul Newman, *Highly Scalable Appearance-Only SLAM - FAB-MAP 2.0*, RSS 2009, Seattle.

[9] Mark Cummins and Paul Newman. *FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance*, The International Journal of Robotics Research, vol. 27(6), PP. 647665, 2008.

[10] Angeli, A., Filliat, D., Doncieux, S., Meyer, J.-A., *A Fast and Incremental Method for Loop-Closure Detection Using Bags of Visual Words.*, IEEE Transactions On Robotics, Special Issue on Visual SLAM.

[11] Angeli, A., Filliat, D., Doncieux, S., Meyer, J.-A., *Visual topological SLAM and global localization.* Proceedings of Intl. Conf. on Robotics and Automation (ICRA), pp. 4300-4305, 2009.

[12] Friedrich Fraundorfer, Christopher Engels, David Nistér, *Topological mapping, localization and navigation using image collections.*, Proceedings of IEEE/RSJ Intl. Conf. on Intelligent Robotics and Systems (IROS), pp. 3872-3877, 2007.

[13] C. Murillo, P. Campos, J. Kosecka, J. Guerrero, *Gist vocabularies in omnidirectional-images for appearance based mapping and localization*, OMNIVIS 2010.

[14] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin., *Context-based vision system for place and object recognition*, Proceedings of IEEE Intl. Conf. on Computer Vision (ICCV), pp. 273-280, 2003.

[15] Nourani Vatani, Navid, Pradalier, Cedric, *Scene Change Detection for Vision-Based Topological Mapping and Localization*, Proceedings of IEEE/RSJ Intl. Conf. on Intelligent Robotics and Systems (IROS), 2010.

[16] Herbert Bay, Tinne Tuytelaars, Luc Van Gool, *SURF: Speeded Up Robust Features*, Proceedings of European Conference on Computer Vision (ECCV), Vol. 3951 (2006), pp. 404-417.

[17] David Nister, Henrik Stewnius, *Scalable Recognition with a Vocabulary Tree*, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2161-2168, 2006.

[18] Nivio Ziviani, Edleno Silva de Moura, Gonzalo Navarro, and Ricardo Baeza-Yates, *Compression: A Key for Next-Generation Text Retrieval Systems*, IEEE Computer, 33(11):37-44, November 2000.

[19] A. Ranganathan, *PLISS: Detecting and Labeling Places Using Online Change-Point Detection*, Proceedings of Robotics: Science and Systems, 2010, June, Zaragoza, Spain.

[20] Salton, G. and M. J. McGill, *Introduction to modern information retrieval*, McGraw-Hill, 1983, ISBN 0070544840.

[21] Sparck Jones, Karen, *A statistical interpretation of term specificity and its application in retrieval*, 1972, Journal of Documentation 28 (1): 1121. doi:10.1108/eb026526.

[22] Aravindhan K Krishnan and K Madhava Krishna, *A Visual Exploration Algorithm using Semantic Cues that Constructs Image based Hybrid Maps*, Proceedings of IEEE/RSJ Intl. Conf. on Intelligent Robotics and Systems (IROS), 2010.

[23] Salton, G. and M. J. McGill, *Introduction to modern information retrieval*, McGraw-Hill, 1983.