The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Keyframe Detection for Appearance-Based Visual SLAM

Hong Zhang, Bo Li and Dan Yang

Abstract— **This paper is concerned with the problem of keyframe detection in appearance-based visual SLAM. Appearance SLAM models a robot's environment topologically by a graph whose nodes represent strategically interesting places that have been visited by the robot and whose arcs represent spatial connectivity between these places. Specifically, we discuss and compare various methods for identifying the next location that is sufficiently different visually from the previously visited location or node in the map graph in order to decide whether a new node should be created. We survey existing techniques of keyframe detection in image retrieval and video analysis. Using experimental results obtained from visual SLAM datasets, we conclude that the feature matching method offers the best performance among five representative methods in terms of accurately measuring the amount of appearance change between robot's views and thus can serve as a simple and effective metric for detecting keyframes. This study fills an important but missing step in the current appearance SLAM research.**

I. Introduction

Robot simultaneous localization and mapping (SLAM) attempts to solve the problem for a robot to build a map of its environment while localizing itself with respect to the map at the same time. Robotics research community has expended considerable efforts in the past two decades or so, and achieved much success, in terms of a mature understanding of – and effective solutions to – the SLAM problem [9]. Implementation of SLAM solutions initially relied heavily on range-bearing sensors (laser range finder, radar, etc.), and more recently on visual sensors (both stereo and monocular vision). Due to the rich textural information, the latter offers the advantage of more effectively handling the issues of data association and loop closing detection, than using range-bearing sensors.

Popular SLAM solutions mostly follow a Bayesian framework in which the state to be estimated consists of robot pose and landmark variables to serve as references in the environment for robot localization. As a result, the complexity of these algorithms directly depends on the number of landmarks, and high computational cost is incurred in order to maintain the probabilistic descriptions of the state variables, in spite of the efforts in efficient implementations such as submaps [4] and extended information filter [19]. In addition, major challenges exist in data association and loop closing detection where the correspondence between sensor data and landmarks of the robot map must be correctly resolved.

To address these challenges, more recently, appearance-based visual SLAM (aSLAM) has been introduced [7], [3]. In aSLAM, the environment is modeled not geometrically but topologically by a graph each node of which represents a strategically interesting location that has been visited by the robot and each arc represents spatial or visual connectivity between physical locations. aSLAM removes the need to explicitly estimate the 3D descriptions of the landmarks and replaces matching features for data association with matching images of the nodes so that spatial constraints among visual features can be exploited to the advantage of a robust matching algorithm. The complexity of matching the current robot view with all previously visited places is tackled by the bag-of-words (BoW) approach [16] in which visual features are clustered and an inverted index is built for retrieving candidate locations efficiently. In addition, spatial information between map locations can be established and maintained locally, to facilitate navigation [2].

One outstanding issue in aSLAM is keyframe detection, i.e., deciding when to introduce a new node into the topological graph. This is an important functional component of the system in order to achieve a sufficient visual coverage of the robot's environment and, at the same time, keep the representation simple for computational efficiency during, for example, path planning and loop closing detection. Current approaches to keyframe detection are *ad hoc*. Common approaches include (a) uniform sampling in space, i.e., every unit linear or angular distance traveled by the robot [7], [5], (b) uniform sampling in time, i.e., every nth frame captured by the camera [10], [14], and (c) uniform sampling in appearance [1], i.e., every fixed amount of change in appearance since the last keyframe, based on some measure of image similarity. Distance-based sampling assumes a correlation between appearance change and spatial change, but this correlation is highly sensitive to the *unknown* geometry of the environment. Similarly, time-based sampling assumes a good correlation between the time interval of successive image captures and appearance change, and this would not perform well when the robot accelerates, decelerates or comes to a stop. Appearance-based sampling measures the change in appearance directly, and is the most reasonable approach to take. In fact, appearance change via visual feature similarity calculation has been used not only in the current research in aSLAM [1] but also in research in view-based robot navigation [13]. and in loop closing detection [11]. However, there is usually little justification for the choice of the adopted method to measure appearance change.

Hong Zhang is with the Department of Computing Science, University of Alberta, Edmonton, Alberta T6G 2E8, Canada, hzhang@ualberta.ca.

Bo Li and Dan Yang are with the College of Computer Science, Chongqing University, China, boli.cqu@gmail.com and dyang@cqu.edu.cn. Bo Li visited the University of Alberta in 2009-10.

In this paper, we investigate the applicability of similarity measures in video analysis and content-based image retrieval (CBIR) algorithms to the keyframe detection problem in aSLAM. Our key contribution is a systematic comparison of keyframe detection algorithms and a recommendation of a feature matching method that has been shown to be superior to the other algorithms. The rest of the paper is organized as follows. In Section II, we will review the relevant literature in CBIR and video analysis as the basis of our analysis. In Section III we describe in detail five representative methods for measuring image similarity in CBIR and video processing due to their direct relevance to aSLAM. In Section IV, an experimental procedure is described to compare the performance of keyframe detection methods in an objective and application-independent manner. The results of applying this procedure to indoor visual SLAM datasets are provided in Section V. Finally in Section VI, conclusions are drawn and future work outlined.

II. Review of Relevant CBIR and Video Analysis Techniques

A problem similar to keyframe detection has been studied extensively in CBIR and in video analysis and segmentation, though for different purposes and under different constraints [8], [17]. CBIR is concerned with obtaining images from an image database that are similar to a query image based on an analysis of image contents, whereas video processing research addresses the problem of segmenting a video into scenes for its indexing, annotation, compression or semantic interpretation. In both cases, a reliable similarity measure between two images serves as the basis for many algorithms. As a result of their efforts, a variety of techniques have been developed. However, only a subset of these techniques are directly applicable to the keyframe detection problem in aSLAM.

There are two main approaches in CBIR – *discrete* and *continuous* [8] – and their distinction lies in whether the visual feature space is discretized in some fashion. In the continuous approach, each image is described in terms of the original visual features extracted from the image, whereas in the discrete approach the BoW (or Bag-of-Features) technique is used to map the visual features to visual words so that efficient text retrieval techniques can be applied. A great number of features have been proposed for image comparison, and they can be grouped into appearance/color, texture, local features, and shape [8]. Of interest to us are those based on appearance and local features since the texture features are not expected to work well for a robot environment cluttered with various objects, and shape extraction is both computationally expensive and unreliable.

In video processing, keyframes of a video sequence are defined as a subset of the frames in the sequence that can faithfully represent or characterize the visual contents of the video. Keyframe extraction techniques can be broadly grouped into three categories: cluster based, energy based and sequential techniques [6], [15]. Cluster and energy based methods are global; i.e., they examine the video sequence in its entirety to determine which frames among all frames can best serve as keyframes. They are therefore not applicable to aSLAM, which does not have available images from the future, not to mention the consideration of computational cost associated with global techniques. Sequential methods, on the other hand, consider the video frames one at a time [18], and are therefore appropriate for aSLAM applications. Specifically, methods based on *sufficient content change* are the most relevant to our application, and these methods make use of visual features similar to those in CBIR.

III. Image Similarity Measures for Keyframe Detection

In view of the rich research in CBIR and video processing, we choose five representative methods to study in this paper: *pixel-wise (pw)*, *global-histogram (gh)*, *local-histogram (lh)*, *feature matching (fm)*, and *Bag-of-Words (BoW)*. The pixel-wise method is a representative appearance-based technique, the two histogram-based methods are popular in both CBIR and shot-boundary detection, and both the feature matching and BoW methods use image features but one is continuous and the other discrete. These five methods account for the major applicable techniques in CBIR and video processing. Comparison among these five methods will provide us with a reasonable guideline with respect to how keyframe detection should be handled.

Each of the five methods we study computes a similarity measure S between two images, I_i and I_t, which in our case are those associated with the last node i and the current view at time t, respectively. This similarity measure is related to a distance function between the two images as shown in Equation (1).

$$S(I_i, I_t) = 1 - \frac{D(I_i, I_t) - D_{min}}{D_{max} - D_{min}} \tag{1}$$

where D is a distance function between two images based on their contents, and D_{min} and D_{max} are the minimum and maximum value that D can – or is expected to – take on. It is clear that S is normalized between [0, 1]. The five methods reviewed in this section differ only in the way in which D is defined.

The simplest and somewhat naive method uses a pixel-wise difference to compute D as shown in Equation (2).

$$D_{pw}(I_i, I_t) = \sum_x |I_i(x) - I_t(x)| \tag{2}$$

where x indexes the pixels of each image, and $|\cdot|$ represents the absolute value. Due to image noise and disregard of pixel correlations by D_{pw}, this method though simple is not expected to perform well for keyframe detection.

Another popular image similarity measure characterizes an image in terms of its histogram, and defines image difference by that between two histograms, according to some metric. Specifically, we compute the norm:

$$D_{gh} = \|H(I_i) - H(I_t)\| \tag{3}$$

where $H(I)$ is the histogram of image I, and $\|\cdot\|$ is a distance metric defined by, among others, χ^2 statistic, L_2-norm, cosine distance or K-L divergence. D_{gh} captures the difference in global intensity distribution but loses spatial information within which difference between two images may exist. This can be partially alleviated through the use of local histograms.

Image difference function via local histogram comparison divides an image into local sub-regions or blocks and computes a histogram for each block. The distance function in this case is defined as

$$D_{lh} = \sum_x \|H(I_{i,x}) - H(I_{t,x})\| \tag{4}$$

where x indexes the image blocks, and H is still the histogram function as before. $\|\cdot\|$ is once again a proper distance metric of one's choice. D_{lh} degenerates either to D_{pw} when the block size is a pixel or to D_{gh} when the image is of one block.

Difference or similarity between two images can also be computed via more sophisticated methods that involve feature extraction and matching. These methods are computationally more costly than those based on image statistics such as Equations (3) and (4), but can yield more robust measures due to the invariance properties of the visual features. In this case, visual features of the two images are first extracted, and their descriptors derived and matched. A matching score can be computed to describe how many common features are shared by the two images. The features can be matched directly or mapped to a visual dictionary first and then matched in terms of visual words [16], as commonly practiced in aSLAM.

Let N_i and N_t be the number of visual features in image I_i and I_t, respectively, and N be the number of features that match. A distance function based on feature matching is usually defined one of two ways:

$$D_{fm} = 1 - \frac{N}{min\{N_i, N_t\}} \text{ or } 1 - \frac{2N}{N_i + N_t} \tag{5}$$

where $min\{\cdot\}$ is the minimum operator. Example visual features include SIFT, MSER, Harris affine, and SURF. SIFT descriptor has been considered as superior to other feature descriptors and is a popular choice in describing the above visual features. Euclidean distance or distance ratio can be employed to match features.

BoW has become a popular technique to use in aSLAM for both loop closing detection and map representation. In this case, an image description vector V is first derived by mapping the visual features in an image to words in the visual dictionary and weighing the visual words by, for example, *tf-idf* (term-frequency inverse document frequency). A distance function can then be defined based on the description vectors of two images. Specifically,

$$D_{BoW} = dist(V(I_i) - V(I_t)) \tag{6}$$

where $V(I)$ represents the BoW description vector associated with image I, and $dist(\cdot)$ is a vector distance metric of one's choice (cosine distance, L_2-norm, or voting [3], for example). Alternatively, one could also consider V as a set, and compute the difference between them in terms of the intersection-over-union measure [1].

IV. Performance Evaluation Methodology

In this section, we describe our methodology to compare the performance of the five image similarity measures for keyframe detection in aSLAM. An image similarity measure is considered useful only if it can capture the view change objectively and accurately, so that when selected keyframes are used later for, localization or loop closing detection, they provide a sufficient coverage of the environment and do not over-sample the environment at the same time to cause unnecessary computational burden. An accurate similarity measure must truthfully reflect the degree of change in the robot's view and, in the ideal case, establish a linear relationship with the actual change in the robot's view, so that it can be tuned in an application to provide a proper sampling of the environment.

To investigate and compare performance, it is important to establish ground-truth with respect to the actual change in the robot's view. With the ground-truth, various similarity measures can be applied and evaluated in terms of their consistency with the ground-truth. For this purpose, we can choose an image sequence such as the one in Figure 3 in which the robot or camera motion is simple enough that overlap between a pair of images can be easily determined through simple one-dimensional image alignment. To achieve this, we can limit robot motion to be a pure translation parallel to the scene so that objects that remain visible experience only translation but no rotation or scale change. While this type of camera motion does not include all possible types of view change, it is the only viable way by which ground-truth can be constructed. The procedure is tedious but important in order to compare the performance of the candidate similarity measures fairly.

With such an image sequence, we build the ground-truth similarity by evaluating Equation (7) for all images in the sequence,

$$S_t = |I_0 \cap I_t|/N \tag{7}$$

where I_0 is the first image in the sequence, I_t is a subsequent image at time t, and N is the number of pixels in an image. $I_0 \cap I_t$ calculates the amount of overlap between two images, after they are aligned using a method, in our case, that maximizes normalized image correlation. $S_t \in [0,\ 1]$ measures the similarity of all images in the sequence with respect to the first.

V. Comparative Results of Keyframe Detection Algorithms

We applied the methodology described in the previous section to evaluate the five image similarity measures in Section III. The datasets used in our study come from typical

indoor environments, one in a research laboratory (*Laboratory II*) and three from the hallways of a laboratory building (*Floor 3*, *Hallway I* and *Hallway V*)[1]. The images were acquired by a Pointgrey Dragonfly firewire camera mounted on a Magellan Pro robot. There are between 200 and 500 images in each dataset, and the image resolution is 320x240 in all but one dataset, whose image resolution is 640x480. Examples of these images are shown in Figure 1. The first 50 images in *Hallway I* at approximately five centimeters per image and the four sequences, each approximately 40 images long and involving only translation in *Laboratory II* at approximately three centimeters per image, were used to create a total of five sequences of ground-truth similarity values. Figure 3 shows a sequence of five consecutive images from the *Laboratory II* dataset to illustrate the amount of robot motion in this study.

The experimental parameters are defined as follows. For the global histogram-based method, we used a 64-bin grayscale histogram. For the local-histogram based method, each image was divided in a 4x4 grid, and a 64-bin grayscale histogram was built for each sub-image. For feature matching method, we used the standard SIFT extractor and a descriptor vector of dimension 128. Finally, for the BoW method, we used a dictionary of 500 visual words obtained by running k-means to cluster the visual features in the first 70 images of the a dataset (either *Hallway I* or *Laboratory II*). For similarity calculation, *tf-idf* was used to weigh the visual words found in each query image, and the voting scheme in [3] was used to measure the distance between the two vectors or the likelihood that two images match. Equation (1) was then used to normalize the distance so as to obtain a similarity value between 0 and 1.

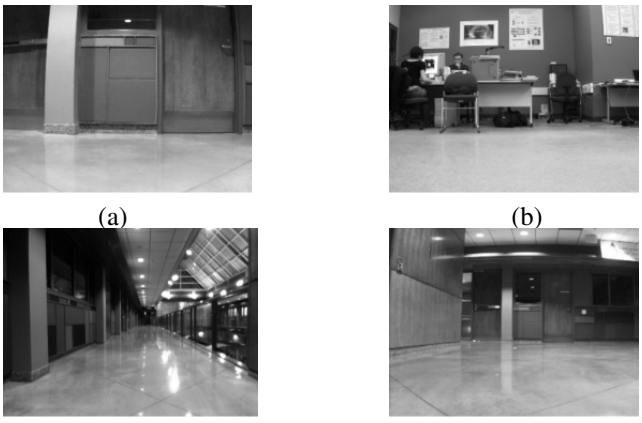

Fig. 1. Example images from the four visual SLAM datasets used in this study: (a) *Hallway I*, (b) *Laboratory II*, (c) *Floor III* and (d) *Hallway V*

For each of the five sequences for which ground-truth was created, we used all five methods to calculate image similarity between the images in the sequence and the first in the sequence. Figure 2 shows the raw data of the comparative study. Figure 2(a) shows the result from the *Hallway I* sequence and (b) from one of the four sequences in *Laboratory II*. In each figure, the horizontal axis is the ground-truth similarity value, and the vertical axis is the similarity computed by the five methods Obviously, the diagonal line corresponds to the ground-truth *gt* result. All methods track the ground-truth proportionally, but different methods exhibit quite different behaviors.

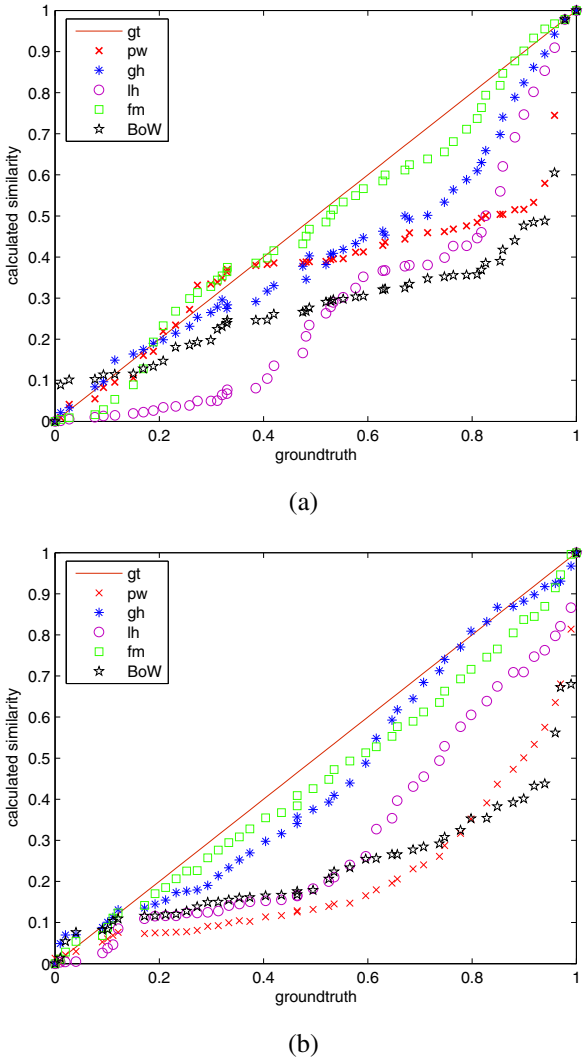

Fig. 2. Experimental results of the comparative study, showing the similarity values computed by the five methods on (a) dataset *Hallway I* and (b) one of the four sequences of dataset *Laboratory II*

Table I shows quantitatively the performance of the five methods in terms of their root-mean-squared (RMS) error with respect to the ground-truth. With the exception of the global histogram method in the case of *Hallway I* dataset, the feature-matching method is obviously superior.

To further quantify the performance evaluation, we fit linear lines through the data points of the five methods, and the slopes of their line fittings are shown in Table II for two image sequences of *Hallway I* and *Laboratory II*. In general, the closer a slope is to 1 (a 45° line), the more

[1]All datasets are available online at `http://webdocs.cs.ualberta.ca/~hajebi/datasets/`.

Fig. 3. A example sequence of the *Laboratory II* dataset used in the experimental study, to illustrate the amount of robot motion relative to the scene

TABLE I
RMS ERRORS OF THE FIVE SIMILARITY MEASURES

	S_{pw}	S_{gh}	S_{lh}	S_{BoW}	S_{fm}
Hallway I	0.164	0.137	0.212	0.172	**0.135**
Laboratory II	0.275	0.093	0.147	0.237	**0.060**

closely the corresponding method computes similarity to the ground-truth, and the more accurate the method. Feature-matching is found to be superior to the other four methods, with the BoW method performing the worst. In addition, we calculated the R-squared values of the linear fittings of the five methods and the result is summarized in Table III. R-squared is a statistical measure of how well a regression line approximates real data points, or how good one term is at predicting another. In our case, it captures in the variance of the similarity measurement. Feature matching has the largest R-value, and can be best approximated by a line whereas others are all much less predictable with a linear model.

By comparing the results from two different indoor environments, we observe that the performance of the feature matching method is the most stable. Interestingly but not surprisingly, the BoW method does not work well for keyframe detection by all measures, largely as a result of the clustering process it employs to create artificial perceptual aliasing when features from different objects are forced to match if they happen to be grouped into the same cluster. As well, in the BoW method, features in two images image are matched with each other indirectly in the space of the visual vocabulary whereas in the feature-matching method visual features in two images are matched directly with each other without the interference of visual features from other images.

TABLE II
SLOPES OF LINEAR REGRESSION MODELS

	S_{pw}	S_{gh}	S_{lh}	S_{BoW}	S_{fm}
Hallway I	0.61	0.85	0.90	0.53	**0.98**
Laboratory II	0.76	0.94	0.89	0.68	**0.96**

TABLE III
R-SQUARED VALUES OF LINEAR REGRESSION MODELS

	S_{pw}	S_{gh}	S_{lh}	S_{BoW}	S_{fm}
Hallway I	0.81	0.94	0.87	0.71	**0.98**
Laboratory II	0.72	0.89	0.92	0.79	**0.98**

A good keyframe detection method should sample the environment fairly, in such a way that a constant threshold similarity value works equally well independently of the environment. This property of keyframe detection method was examined for the five methods in our study. In the experiment, we chose a set of four similarity threshold values, each leading to the selection of a certain percentage of frames as keyframes from one image sequence. We then applied the same threshold values to another sequence, to observe the change in the proportion of the frames detected as keyframes. The two sequences come from two novel environments with similar characteristics: *Floor III* and *Hallway V* (Figure 1(c) and (d)). *Hallway V* is similar to *Hallway I* but observed the scene from a different distance, whereas in *Floor III* the robot did not move parallel to the scene but along the length of a hallway. The average change in the percentage of detected keyframes for different methods is summarized in Table IV.

TABLE IV
STABILITY OF SIMILARITY THRESHOLD FOR KEYFRAME DETECTION

	S_{pw}	S_{gh}	S_{lh}	S_{BoW}	S_{fm}
Hallway V vs. Floor III	3.70%	4.02%	3.92%	2.05%	**1.94%**
Floor III vs. Hallway V	6.13%	12.89%	7.16%	2.47%	**2.31%**

Based on the limited experimental data, the feature matching method provided the smallest variation in the percentage of keyframes selected. The BoW method worked reasonably well as well. In view of all the performance metrics, in terms of tracking the ground-truth, linearity, and stability in keyframe selection, we conclude that the feature matching method is superior to the other four representative methods. However, we should note that these results are predicated by the parameters used especially for the BoW method, and that an increased vocabulary size could cause the BoW method to improve, although that would make it approach the feature-matching method in the limit.

Finally, with respect to computational cost, histogram-based methods, i.e., *pw*, *gh*, and *lh*, are more efficient than the feature-based methods, i.e., *fm* and *BoW*. Histogram-based methods have a complexity that is linear with respect to the image size *N*. In contrast, the complexity of feature-based methods is dominated by the feature extraction process, which has a complexity of at least *O(NlogN)* for scale-space features such as SIFT – although features are almost always detected already in all appearance SLAM algorithms, so that the only additional cost for keyframe detection is for matching feature vectors. In addition, the feature-matching method has the advantage that its complexity is independent of the map size, whereas the complexity of the BoW method

increases with the map size since an increased vocabulary size is required to handle an enlarged map. Practically, the computational cost for keyframe detection is in general insignificant compared with other visual SLAM operations such as visual loop-closure detection because only two images, I_t and I_i, need to be involved in keyframe detection.

VI. Conclusions

In this paper, we have presented our comparative study of image similarity measures for keyframe detection in appearance-based visual SLAM. Appearance SLAM represents a map in terms of a graph whose nodes correspond to robot poses, and the nodes are characterized by their appearances viewed by the robot. Keyframe detection is a critical step in deciding the poses to be included in the map for an appropriate coverage of the environment. We started by reviewing literature in CBIR and video processing, and selected five representative techniques in those fields and in visual SLAM to be included in our study. A methodology was established to compare these five techniques objectively, using several performance metrics. From the experimental results, we concluded that the feature matching method performs the best among the five. Our key contribution is a systematic approach to the selection of an image similarity measure that serves as the basis for keyframe detection in appearance SLAM. The emergence of the feature matching method as the optimal method among the five we have examined can be used as a useful reference to other researchers.

One extension of this research is to design a keyframe detection algorithm in a Bayesian framework in which the image similarity measure serves as the likelihood function so that the decision of keyframe selection is made by examining an image sequence. Another extension is to integrate information about both spatial change of robot location and visual change in the robot's view to overcome the problem with view-only based methods such as when a robot undergoes pure rotation and triggers incorrect keyframe creation. Finally, additional methods in CBIR and video processing such as edge orientation histogram [12] can be further explored, and the parameters of the methods can be optimized for the keyframe detection problem in appearance SLAM.

References

[1] Adrien Angeli, Stephane Doncieux, Jean-Arcady Meyer, and David Filliat. Incremental vision-based topological slam. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1031–1036, 2008.

[2] Adrien Angeli, Stephane Doncieux, Jean-Arcady Meyer, and David Filliat. Visual topological slam and global localization. In *2009 IEEE International Conference on Robotics and Automation (ICRA 2009)*, pages 4301–4305, 2009.

[3] Adrien Angeli, David Filliat, Stephane Doncieux, and Jean-Arcady Meyer. A fast and incremental method for loop-closure detection using bags of visual words. In *IEEE Transactions On Robotics, Special Issue on Visual SLAM*, volume 24, pages 1027–1037, 2008.

[4] Tim Bailey and Hugh Durrant-Whyte. Simultaneous localization and mapping: Part ii state of the art. *IEEE Robotics and Automation Magazine*, 2006.

[5] J. Callmer, K. Granstrom, J. Nieto, and F. Ramos. Tree of words for visual loop closure detection in urban slam. In *2008 Australasian Conference on Robotics and Automation*, page 8, 2008.

[6] M. Chatzigiorgaki and A.N. Skodras. Real-time keyframe extraction towards video content identification. In *16th International Conference on Digital Signal Processing*, pages 1–6, 2009.

[7] Mark Cummins and Paul Newman. Fab-map: Probabilistic localization and mapping in the space of appearance. In *International Journal of Robotics Research*, volume 27, pages 647–665, 2008.

[8] Thomas Deselaers, Daniel Keysers, and Hermann Ney. Features for image retrieval: an experimental comparison. *Information Retrieval*, 11:77–107, 2008.

[9] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: Part i the essential algorithms. *IEEE Robotics and Automation Magazine*, pages 99–108, 2006.

[10] Kin Ho and Paul Newman. Loop closure detection in slam by combining visual and spatial appearance. In *Robotics and Autonomous Systems*, volume 54, pages 740–749, 2006.

[11] J. Kim and I.S. Kweon. Robust feature matching for loop closing and localization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems San Diego*, volume 2, pages 3905–3910, 2007.

[12] Jana Kosecka, Liang Zhou, Philip Barber, and Zoran Duric. Qualitative image based localization in indoors environments. In *Proc 2003 IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2003)*, pages 3–10, 2003.

[13] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Real time localization and 3d reconstruction. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 363–370, 2006.

[14] Paul Newman and Kin Ho. Slam-loop closing with visually salient features. In *International Conference on Robotics and Automation*, 2005.

[15] C. Panagiotakis, A.D. Doulamis, and G. Tziritas. Equivalent key frames selection based on iso-content principles. In *IEEE Transactions on Circuits and Systems for Video Technology*, volume 19, pages 447–451, 2009.

[16] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *Nineth IEEE International Conference on Computer Vision and Pattern Recognition (ICCV 2003)*, pages 1470–1477, 2003.

[17] B. T. Truong and S. Venkatesh. Video abstraction: a systematic review and classifications. In *ACM Transactions on Multimedia Computing, Communications and Applications*, volume 3, 2007.

[18] J. Vermaak, P. Perez, and M. Gangnet. Rapid summarization and browsing of video sequences. In *British Machine Vision Conference*, pages 424–433, 2002.

[19] Matthew R. Walter, Ryan M. Eustice, and John J. Leonard. Exactly sparse extended information filters for feature-based slam. *The International Journal of Robotics Research*, pages 335–359, 2007.