Localization in urban environments using a panoramic gist descriptor

A. C. Murillo, G. Singh, J. Košecká, J.J. Guerrero

***Abstract*—Vision based topological localization and mapping for autonomous robotic systems are topics of increased interest in recent years. The need for mapping larger environments requires models at different levels of abstraction and additional abilities to deal with large amounts of data efficiently. Most successful approaches for appearance based localization and mapping with large datasets typically represent locations using local image features. We study the feasibility of performing these tasks in urban environments using global descriptors instead and taking advantage of the more and more common panoramic datasets. This paper describes how to represent a panorama using the global gist descriptor [1], while maintaining desirable invariance properties for location recognition and loop detection. We propose different gist similarity measures and algorithms for appearance based localization and an on-line loop closure detection method, where the probability of loop closure is determined in a Bayesian filtering framework using the proposed image representation. The extensive experimental validation in this paper shows that their performance in urban environments is comparable to local feature based approaches when using wide field of view images.**

***Index Terms*—computer vision, appearance based localization, recognition, gist descriptor, omnidirectional images**

I. Introduction

Generating metric and topological maps from streams of visual data has in recent years become an active area of research. This increased interest has been facilitated to a large extent by improvements in large scale wide-baseline matching techniques and advances in appearance based localization by means of place recognition. Place recognition, for purely appearance based strategies, is typically formulated as an image based retrieval task; given a database of views from certain geographical area, and a new query view, the goal is to determine the closest view from the reference database. The related loop closure detection task aims to recognize previously visited locations either in an on-line or batch manner. One of the key issues in both tasks is the choice of image representation and similarity measure between two images.

In this paper, we investigate the suitability of the gist descriptor, proposed by Oliva and Torralba [1], [2], for panoramic image representation and propose how to build this representation. Along with this representation, we introduce and compare several similarity measures between panoramic views captured at individual locations. We evaluate the proposed omnidirectional gist descriptor on a large scale place recognition task given the database of reference panoramas of an urban environment, such as the one shown in Fig. 1. Given the proposed image representation, we introduce an on-line loop closure detection method, where the probability of loop closure is determined in a Bayesian filtering framework.

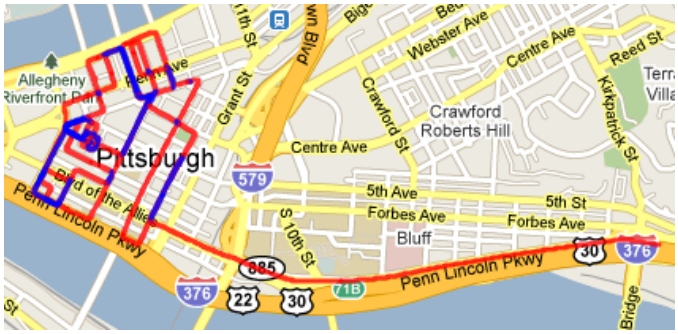


Fig. 1: Google Maps Visualization of Street View dataset.

Experimental results in this work show that despite the simplicity and compactness of the gist descriptor, its effectiveness and discrimination capability in urban settings is quite high. This is partly due to the 360^o field of view. We also present a discussion of efficiency and scalability trade-offs between gist descriptor and local feature based methods. Our extensive experiments applied in panoramic images demonstrate similar or superior performance and higher efficiency of gist descriptor for both location recognition and loop closure detection compared to local feature based methods.

In the rest of this paper, Section II briefly discusses the related work. The proposed panoramic gist representation and associated similarity measures are detailed in Section III. Our approach for place recognition is evaluated on two different urban panoramic datasets in Section IV, including comparisons with local feature based methods. The proposed method for loop closure detection is detailed and experimentally validated in Section V. The conclusions are in Section VI.

II. Related Work

Appearance based localization and mapping has been studied extensively in robotics, using a large variety of different approaches and camera systems. In recent years, notable scalability and robustness have been achieved in acquiring both metric and topological maps. This progress has been fueled by an increase in computational speed, capabilities for handling and discriminating large amounts of image data and advances in effective image representations. An essential component of vision based localization is the choice of image

A.C. Murillo and J.J. Guerrero are with DIIS, Instituto de Investigación en Ingeniería de Aragón, University of Zaragoza, Spain. e-mail: {acm,jguerrer}@unizar.es

G. Singh and J. Košecká are with the Computer Science Department, George Mason University, USA. e-mail:{gsinghc,kosecka}@gmu.edu

representation and associated similarity measure, which has to be sufficiently invariant yet discriminative and enable efficient search in large databases. Another key component deals with means of modeling temporal and online aspects of the loop detection and localization process. Next we review some related approaches.

Image representation. The existing image representations vary depending on the choice of local image measurements and means of aggregating them spatially. Most of the recent advances in appearance based localization are based on local scale invariant features [3], [4] and a geometric verification stage [5], [6], [7]. The effectiveness of local features for wide-baseline matching is facilitated by approximate nearest neighbor methods and quantization schemes, such as k-means clustering, vocabulary trees and inverted file structures [8], [9], [10], which enable scalability of these methods to large datasets with large appearance variations. For example, Cummins and Newman presented an appearance based localization method handling extremely large trajectories with their FABMAP approach [11], and Valgren and Lilienthal [12] evaluated localization across different season variations.

Alternative image representations are based on global image descriptors, which often forgo the feature location information and aggregate various image statistics globally or over large support regions. Early examples of these, both using conventional or omnidirectional images, were global color histograms, histograms of gradient orientations, Fourier transform components or color invariant moments [13], [14], [15], [16], [17]. These works typically dealt with small scale datasets. More recently, the gist descriptor [1] has been shown to work well for scene categorization, scene alignment or duplicate detection in large datasets of millions of images [18], [19], [20]. The gist of an image captures image statistics of responses to a filter bank, while weakly spatially integrating the responses over a coarse image grid. More recent attempts to use the gist descriptor in the context of robotic localization include [21], [22]. In [22] initial location recognition is achieved using gist descriptor and is refined by tracking salient image regions. In [21], vanishing points are used as additional cues to obtain more robust place recognition results. Some previous works combine local feature information with the global descriptors to augment a gist-based place recognition approach with object recognition [23] or to re-rank the top candidates selected by other types of global descriptor similarity [16], [24]. Some of the known disadvantages of purely global descriptors include lower invariance, difficulties with occlusions and inability to incorporate stronger geometric constraints. Their main advantage is the efficiency of computation and compact representation, allowing enhancements in storage and computational speed, facilitating working with millions of images [25]. The majority of the above mentioned works consider conventional monocular images. While the representations based on local scale invariant features can be naturally extended to an omnidirectional setting, the computation and the number of local features per location increases bringing down the efficiency of the matching stage [26].

In our work, we explore the effectiveness of the gist descriptor and show how to compute "the gist of a location" as captured by a panoramic image. We will demonstrate the use of this representation in the context of location recognition and loop closure detection. In order to maintain some of the invariance properties required for image based localization and place recognition, we propose a similarity measure which weakly exploits a Manhattan world property [27] assuming that camera/vehicle headings at revisited locations are related by multiple of 90^o degrees. This assumption is reasonable for urban outdoors and indoors environments which can be often viewed as networks of streets/corridors and intersections, with the preferred directions of travel being related by 90^o.

Our initial location recognition and loop closure detection experiments using the proposed representation were shown in [28] and [29]. In this work we extend them and incorporate an on-line temporal model for loop closure detection which computes the probability of loop closure in a Bayesian filtering framework. We compare the proposed methods with state of the art techniques based on local features and discuss in detail the tradeoffs between local and global representations and associated retrieval strategies as the size of the dataset increases. We can find initial experiments towards adapting the proposed panorama representation for different types of omnidirectional images in [30].

Appearance based localization and loop closure. Given the image representation and associated similarity measure, the simplest version of appearance based localization can be accomplished by means of place recognition. In this setting, the environment is represented by a set of images acquired at previously visited locations. In the localization stage, given the query view, one seeks the closest view from the reference set. In our approach, the model of the environment is simply an unorganized database of images, as done for instance by Fraundorfer et al. [5]. One can endow the model of the environment with additional topological structure, which captures neighboring relationships between the locations, as in the approach for catadioptric systems by Goedeme at al [17] or the proposed method for conventional images from Li and Košecká [31]. The methods for inferring the topology vary from supervised to unsupervised settings. We find many place recognition approaches built on vocabulary tree based methods, such as the work from Schindler et al [32], who showed that the recognition performance improves when using more informative features and heuristics in nearest neighbor search.

In the map building stage, another important problem is the loop closure detection. The existing loop closure strategies can be broadly partitioned into on-line and off-line (batch) methods. Among the on-line methods, the FAB-MAP [33] uses a bag of words image representation and explicitly models the dependencies between different visual words. In FAB-MAP, each view is considered as a separate location and the probability of loop closure is determined for each view at runtime. Ranganthan at al [34] present a representation called probabilistic topological maps (PTMs) that approximates posterior distribution over possible topologies using odometry and appearance measurements. Other examples for topological

map building and loop closure detection which integrate metric information in hybrid topological-metric approaches can be found in [35], [36]. They model the environment with a global topological map that relates local metric maps with each node of the topological model. In Angeli et al [37], loop closures are detected using "bag of features" representation in the Bayesian setting, where at each instance the most likely sequence of loop/no loop hypotheses is computed on-line. Our temporal model for loop detection is closely related to [37], using a different model for image likelihoods and transition probabilities. The off-line methods typically compute the similarity matrix between all pairs of views acquired during the run. Ho and Newman [38] detect loop closures directly from the similarity matrix, by detecting statistically significant sequences from this matrix. Anati and Daniilidis [26] formulate the loop closure detection in a Markov Random Field (MRF) framework and propose a novel similarity measure for comparing two panoramas. The rotational invariance with respect to changes in heading is achieved by alignment of local features projected on the horizontal plane using a dynamic programming approach. There is also a group of place recognition methods which try to obtain a globally consistent set of matches given a set of local loop closure hypotheses. These methods can be seen as a post-processing step to filter out the false positives. For example, Olson presented an approach [39] which uses spectral clustering [40] for efficiently determining the globally correct matches.

III. GIST IN PANORAMAS

Panoramic images are becoming a popular way of visually mapping large environments. Both in the process of building these models and at the time of using them, a measure for the similarity between two given panoramas is needed. Finding similar images is essential to build a topological map, detect revisited areas or localize new measurements with regard to reference ones. In this work, we investigate if the more detailed place information and configuration contained in panoramic images compensate for the lower discriminability of global descriptors. This section describes our proposed gist based panorama representation and similarity evaluation approaches.

A. Image representation

The gist descriptor [1], [2] aggregates image statistics of the responses of several filters combined with the input image. The advantage of the descriptor is that it is very compact and fast to compute. In the standard configuration, each image is represented by a 320 dimensional vector per color band, resulting in a 960 dimensional descriptor per image. The feature vector corresponds to the mean response to steerable filters at different scales and orientations. Sample responses in a conventional image to these filters are shown in Fig. 2. This resulting descriptor vector coarsely encodes the distribution of orientations and scales in the image. To get an intuition of what this descriptor captures, images are clustered together according to their gist descriptor and reference views from some of these clusters are visualized in Fig. 3. One can note that images with similar scene structure have the same type of

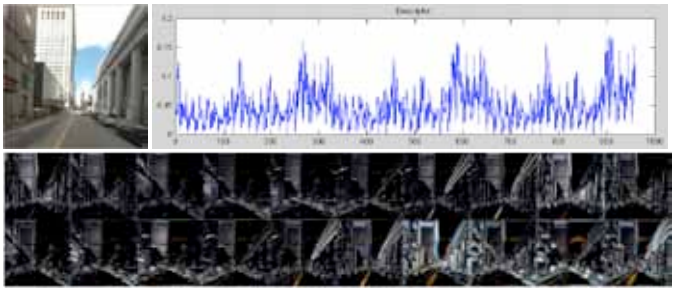


Fig. 2: Gist descriptor extraction. Example of intermediate responses of an image to the 20-filter bank used to build the descriptor.

Fig. 3: Clustering reference view gists into a $k = 40$ vocabulary. Sample views from three of the clusters show how views with similar structure get clustered together.

gist descriptors. The gist descriptor has been demonstrated to be a good solution for scene categorization problems [1] and it has been used effectively for retrieving nearest neighbors from large scale image databases, both for place and object recognition [41], suggesting how it could be combined with local features to further improve the recognition system.

In our experiments, each location is represented with a StreetView™panorama acquired by a 360^o field of view multi-camera system. A single panorama is obtained by warping the five radially undistorted perspective images onto the sphere assuming one virtual optical center. One virtual optical center is a reasonable assumption considering that the structure around the sensor is very far compared to the discrepancy between optical centers of all the cameras. The sphere is backprojected into a quadrangular prism to get a piecewise perspective panoramic image, see Fig. 4a. The top camera acquisition is discarded as it does not provide much information. Then, our panorama is composed of four perspective images covering 360^o horizontally and 127^o vertically. We discard the bottom part of all views, which always contains parts of the vehicle acquiring the panoramas.

The gist descriptor for the entire panorama is obtained by

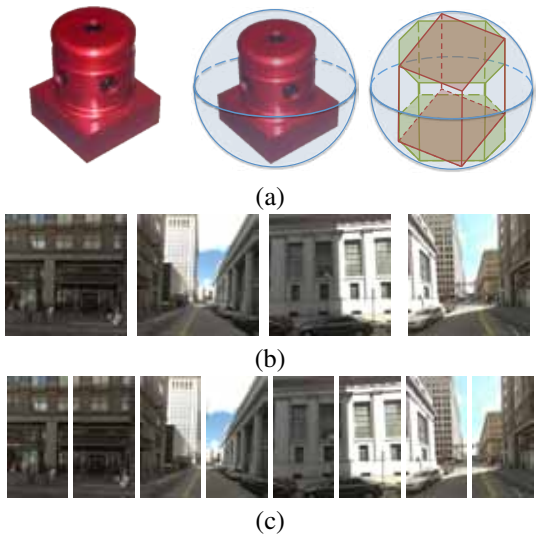


Fig. 4: Panorama acquisition. (a) Multi-camera system with 5 side cameras, whose composed panorama can be re-projected into any kind of *f*-faced prism, providing f sub-views from the whole scene. (b) 4 sub-views from panoramic piecewise perspective image as an outer surface of the quadrangular prism. (c) Finer partitioning of the panorama into 8 sub-views.

computing the standard gist descriptor for each of the 4 views and stacking them together. The panorama is then represented by a 4-tuple of gist descriptors computed for left, front, right and back portion of the panorama denoted by:

$$\mathbf{g} = [g_1, g_2, g_3, g_4] = \mathbf{g}_{1234}. \tag{1}$$

The aforementioned back-projection could be done to any arbitrary number of faces. In urban environments, it is more natural to follow the Manhattan world directions and use four individual views. Partitioning the gist descriptor into the four parts as described above, along with an appropriate similarity measure detailed next, will enable us to strike a good balance between discriminability of the proposed descriptor, viewpoint invariance and compact representation. This partitioning is suitable for urban indoors and outdoors environments which can be well described as networks of roads/corridors and intersections, such that the possible camera headings at a particular location are related by multiple of 90^o degrees. Deviations from these assumptions are discussed in Section IV, where we can see that obtaining finer panorama partitioning does not provide significant improvements but it does carry significant computational overload.

B. Image Similarity Measure

Given the proposed representation, this subsection details how to compare two panoramas with three different methods. They are all run and evaluated in the following section to study the advantages and issues in each of them. We first introduce a similarity measure for exact neighbor search followed by two approximate more efficient methods.

1) Exact Gist distance - E_{gist}: We are given the 4-tuple of gist descriptors computed for the reference panoramic image, denoted by $\mathbf{g}^r = [g_1^r, g_2^r, g_3^r, g_4^r] = \mathbf{g}_{1234}^r$, and the query image with corresponding gist descriptor of $\mathbf{g}^q = [g_1^q, g_2^q, g_3^q, g_4^q] = \mathbf{g}_{1234}^q$, where the short hand index $_{1234}$ denotes the order of individual components of the descriptor. In order to compare the two panoramic images, we want to take into account the possibility that they have been taken at different orientation headings. To accommodate this level of viewpoint invariance, we propose to consider in the matching the following descriptor permutations, obtained by circular shifts, $\mathbf{g}_{1234}$, $\mathbf{g}_{2341}$, $\mathbf{g}_{3412}$, $\mathbf{g}_{4123}$. The similarity measure between two panoramic images is then defined in the following way:

$$dist(\mathbf{g}^q, \mathbf{g}^r) = \min_m d_e(\mathbf{g}^q, \pi_m(\mathbf{g}_{1234}^r)), \tag{2}$$

where π_m is the m^{th} circular permutation of the gist component vectors (m = 1,2,3,4) and d_e is the sum of the norms of differences between the gist vector components,

$$d_e(\mathbf{g}^q, \mathbf{g}^r) = \sum_{i=1}^{4} \|g_i^q - g_i^r\|. \tag{3}$$

When using the exact distance measure, the computational complexity of the problem of finding nearest neighbor in the image database is linear with the size of the database. While this strategy is acceptable for relatively small databases, it does not scale to large ones, where sub-linear strategies for nearest neighbor have to be sought. We now present two strategies based on methods commonly used for sub-linear nearest neighbor search: descriptor quantization and approximate nearest neighbor methods.

2) Gist vocabulary - VQ_{gist}: Following the commonly used bag of words approach [8], the space of all gist descriptors is quantized to build a vocabulary of k gist words. A subset of training reference panoramas is used to build the gist vocabulary $\mathbf{V}_{gist} = \{w_1, w_2, ..., w_k\}$. k-means clustering is run on gist descriptors from each of the four parts of all these reference images. The k cluster centroids are considered to be the *words*, w, of the gist vocabulary. Typical values for k and their impact in the results are discussed later at the experiments in Section IV. Fig. 3 shows how views with similar basic structure get clustered together. Fig. 5 presents the average image of all views that belong to each cluster. Notice that qualitatively different features of urban areas are revealed by individual clusters.

We define the similarity measure, $dist_{VQ}$ between two panoramas using the gist vocabulary $\mathbf{V}_{gist}$. It is defined using the distance between the closest cluster (word) assigned to the each of the original gist descriptors. Each of the four gist descriptors, composing the panoramic image descriptor, is assigned to the nearest gist word from $\mathbf{V}_{gist}$:

$$\begin{aligned} \mathbf{g}^q &= [g_1^q, g_2^q, g_3^q, g_4^q] \leftarrow [w_a, w_b, w_c, w_d] \\ \mathbf{g}^r &= [g_1^r, g_2^r, g_3^r, g_4^r] \leftarrow [w_e, w_f, w_g, w_h] \end{aligned}$$

The complexity of this nearest neighbor search depends on the number of words in the gist vocabulary - k - since the nearest word for each gist descriptor needs to be found. Once the gist

Fig. 5: Average view in several gist-vocabulary ($k = 40$) clusters built from $\approx$ 9000 reference panoramas (36000 views).

word assignments are obtained, the distance between the two panoramas is computed as described in (2):

$$dist_{VQ}(\mathbf{g}^q, \mathbf{g}^r) = dist([w_a, w_b, w_c, w_d], [w_e, w_f, w_g, w_h]) \quad (4)$$

In case the size of the vocabulary is small, further efficiency can be gained by pre-computing the distance matrix between all gist words: $D_w(i,j) = \|w_i - w_j\|$. Using only $dist_{VQ}$, we can efficiently retrieve a small set of likely candidates including the best alignment of each candidate with the query panorama. In a second stage, exact gist similarity measure E_{gist}, as described before, is used to re-rank these candidates with regard to the query.

3) k-d tree based similarity - KD_{gist}: Another commonly used approach for speeding up nearest neighbor search are *k-d* trees. In the basic *k-d* tree algorithm, the splitting dimension at a node is the one with the highest variance and the threshold for split is set to be the median value along that dimension. The first candidate for the nearest neighbor is obtained from a traversal through the tree by comparison to the thresholds at each level. This can be optionally followed by the process of backtracking in which other unexplored branches of the tree are searched for better candidates. The search efficiency was improved by [42] who described a priority queue based backtracking known as Best Bin First (BBF). An improved version of the *k-d* tree algorithm in which multiple randomized *k-d* trees are used was proposed by [43]. The randomized trees are built by choosing the split dimension randomly from the set of dimensions with high variance. The aforementioned priority queue is maintained across all these randomized trees and the nearest neighbor is found through simultaneous independent searches in different trees.

In previous works such as [44], it has been shown that given a desired precision for an approximate nearest neighbor method, the efficiency decreases dramatically with the increased dimensionality making them comparable to linear search methods. Therefore, we perform Principal Component Analysis (PCA) for dimensionality reduction. Given a 4-tuple of standard gist descriptors ($4 \times 960 = 3840$ dimensions), we compute the principal components for the set of reference views and select the top principal components such that they explain 99% of the variance in the data. In our experiments, we kept the first 500 components. Projecting the original descriptors on the principal components, we get a lower dimensional representation $\mathbf{g}_p$ for each composite gist descriptor.

The randomized *k-d* tree is built from the projected descriptor values $\mathbf{g}_p$ of the reference image set. In order to exploit the Manhattan world assumption in the image comparison stage, the *k-d* tree is queried with the projected gist descriptor from all four possible permutations of the query image 4-tuple,

$$\min_m d_e(\mathbf{g}^{kd}, \pi_m(\mathbf{g}_p^q))$$
$$d_e(\mathbf{g}^{kd}, \mathbf{g}^q) = \|\mathbf{g}^{kd} - \mathbf{g}_p^q\| \quad (5)$$

where $\mathbf{g}^{kd}$ is the approximate nearest neighbor returned by the *k-d* tree method and π_m is the m^{th} circular permutation of the 4-tuple gist elements ($m = 1, 2, 3, 4$) in the full gist descriptor. The permutation yielding the smallest distance d_e is kept as result.

IV. Appearance Based Localization

Appearance based localization (location recognition) and loop closure detection are closely related tasks. All of them share the basic goal of finding the closest view in the reference set, given some similarity measure.

In the localization problem, we evaluate the similarity of a query view with respect to a given set of reference images that cover the whole considered environment. For the loop detection task, we evaluate the similarity in an on-line manner, comparing the current view only with the set of images acquired so far. This section evaluates our proposed image representation and similarity measures for localization. The techniques and results for loop closure detection are described in Section V.

A. Experimental settings

Experiments in this section are designed to evaluate the proposed panorama representation and image similarity measures. We present results for the three similarity measures proposed in the previous section (E_{gist}, VQ_{gist} and KD_{gist}) and compare them to standard local feature "bag of words" representation. All our experiments were run in Matlab. The vocabulary and *k-d* tree for the VQ_{gist} and KD_{gist} approaches are built using the VLFeat open source library [45] and gist descriptors were computed using the code provided by the authors[1].

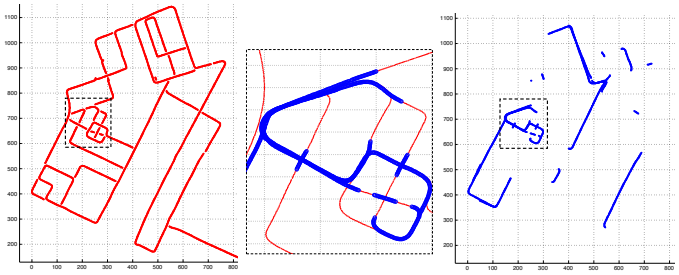
We evaluate the performance as a function of the top-k retrieved neighbors using the different similarity measures. Quantitative evaluation of the localization accuracy is done using the ground truth GPS annotations.

B. Localization in Street View dataset

This section analyzes and demonstrates several key design issues and evaluates the performance of the proposed approaches on a large dataset of Street View images[2]. The set consists of 12,000 panoramas acquired from a vehicle along a 13 mile long run in an urban environment shown in Fig. 1. All the locations in the dataset were provided with GPS coordinates of the vehicle. The dataset is divided into a

[1] http://people.csail.mit.edu/torralba/code/spatialenvelope/

[2] Dataset provided for research purposes by Google™.



(a) (b) (c)

Fig. 6: Street View dataset. A general overview of this dataset is shown in Fig. 1. Red locations (a) are the reference set; blue locations (c) are later revisits to different areas used as test set for the localization experiments and ground truth for the loop closure experiments. The zoomed view of the rectangular dashed area (b) shows some superposed reference and test locations, where the vehicles traverses the same streets at different time in different travel directions.

reference set and a test set (Fig. 6 visualizes the train-test split of the dataset):

- The reference set contains all the images from the first time the vehicle passes by a particular location ($\approx$ 9000 panoramas).
- The test set is composed by the rest of images acquired at a later time, when the vehicle passed through those places again($\approx$ 3000 panoramas).

In the following results, we provide the accuracy for correct localization (defined as the number of correctly localized query views divided by the total amount of query views) using different distance ($dist$) thresholds (10, 20 or 40m). Each column presents the results for considering the nearest retrieved location within the top-k results. We present the results for top-$k = 1, 10, 20, 40$. Top 1 is the localization result provided by our approach, while the other columns evaluate results if we consider a set of top candidates.

1) Panorama partitioning: The first set of experiments, summarized in Table I, analyzes the proposed partitioning of the panoramas into four views. We also evaluate if it can be improved by finer grained image partitioning with the corresponding alignment estimation. Localization is obtained for all the test set images using the basic approach, E_{gist}, and a finer partitioning into 8 segments (See Fig. 4). It is observed that the finer image partitioning does not lead to any major improvement with changes up to 2% increase in the recognition rates. However, the time for processing each query view increases. In our Matlab implementation, the time for computing the matches increased from 5 to 15 seconds when a finer grained image partitioning is used. Therefore, for the rest of our localization experiments, we use a partitioning of the panoramas into 4 views.

We have also run this experiment without computing the distance over all circular permutations, i.e., with similarity which is not invariant to any rotation. For this dataset, even with just a few locations which are revisited from opposite direction, it decreased localization performance by 5%.

TABLE I: Localization results varying the partitioning

		top 1	top 10	top 20	top 40
E_{gist} (4-parts)	dist $<$ 10m	0.88	0.92	0.93	0.94
	dist $<$ 20m	0.93	0.94	0.95	0.96
	dist $<$ 40m	0.94	0.96	0.97	0.97
	Average query search time: **5.23 s**				
E_{gist} (8-parts)	dist $<$ 10m	0.90	0.93	0.94	0.96
	dist $<$ 20m	0.93	0.95	0.96	0.98
	dist $<$ 40m	0.95	0.97	0.98	0.99
	Average query search time: **15.44 s**				

TABLE II: Localization results varying the field of view: 360^{o}FOV (all faces) vs standard FOV (single face)

		top 1	top 10	top 20	top 40
E_{gist} (All faces)	dist $<$ 10m	0.88	0.92	0.93	0.94
	dist $<$ 20m	0.93	0.94	0.95	0.96
	dist $<$ 40m	0.94	0.96	0.97	0.97
	Average query search time: **5.23 s**				
E_{gist} (Single face)	dist $<$ 10m	0.75	0.82	0.84	0.86
	dist $<$ 20m	0.78	0.85	0.87	0.89
	dist $<$ 40m	0.81	0.88	0.90	0.92
	Average query search time: **1.42 s**				

2) Field of view impact: In this set of experiments, summarized in Table II, we evaluate the importance of using panoramic images in our approach. We compare localization results using wide field of view panoramas and narrow field of view images. Localization in both cases is obtained using the E_{gist} approach. To facilitate the comparison, we run the localization experiments on the entire panorama (360^o FOV) and on a single face of the panorama. As expected, using the entire panorama has a higher computational cost compared to using a single face (search for a single query needs about 5 and 1.5 seconds respectively). However, using the whole field of view provides a significantly higher accuracy, with an increase of more than 12%. Notice that localization for conventional field of view images, using the proposed image representation, achieves a performance above 90% when considering a set of top-40 candidates, and therefore, a final re-ranking step of these candidates would be required for the case of conventional field of view images.

3) Exact vs approximate gist similarity: This subsection provides an evaluation of the different similarity measures, which use either the exact gist similarity or the approximate approaches described in Section III-B, namely, E_{gist}, VQ_{gist} and KD_{gist}. The results, summarized in Table III and Fig. 7, point that the most accurate localization, among the gist based approaches, is achieved by an exhaustive search on the exact gist representation, E_{gist}. However, accuracy with the other approaches is only slightly lower and presents advantages with regard to memory and computational time requirements, as described in more detail later in Table V.

We tested different vocabulary sizes k in the VQ_{gist} experiments (k = 40, 100, 1000). As the size of the vocabulary increases, the accuracy of localization improves. We only evaluated up to a vocabulary size of 1000 words since a vocabulary of a larger order of magnitude will be the same size as the number of images in the reference set.

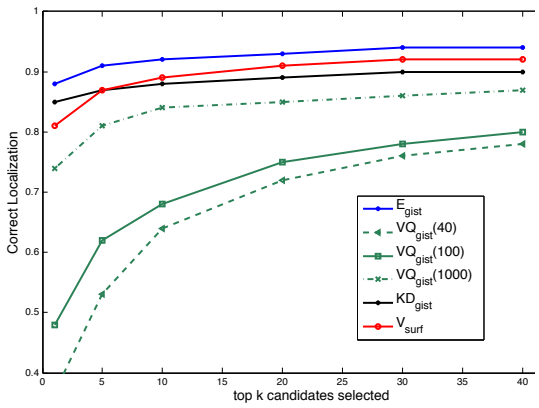


Fig. 7: Correct localization results for all approaches run in Street View dataset, with acceptance threshold of 10 meters.

The KD_{gist} approach uses a descriptor of reduced dimensionality (500 dimensions). We used a single *k-d* tree in our experiments. We experimented with a higher number of trees and there was no performance improvement but the execution time per query increased from 0.09s (using a single tree) to 0.17s (using a forest of five trees). We also evaluated the approach for using the entire descriptor instead of the reduced one obtained through PCA. The performance was similar but the execution time increased to 0.29s per query.

4) Gist-based vs. local features: This subsection compares results obtained for the three gist based localization approaches with a standard local feature based approach, V_{surf}. The V_{surf} method uses SURF local features [4], extracted from the whole panorama, in a bag of features approach which is based on the hierarchical k-means clustering proposed by Nistér and Stewénius [8]. Each of the obtained cluster centers represents a *visual-word*, and the local features of an image are quantized by assigning them to the nearest visual word. An image is then represented by a weighted histogram of visual words where the individual bin weights are determined by the inverse frequency of the visual word across all reference images. Normalized histogram distance is used to evaluate image similarity and select the top-k most similar images as most likely locations. As in our previous experiments, we have used the publicly available VLFeat library [45]. We use hierarchical k-means to build a vocabulary tree with the following parameter values for the tree construction: tree depth of 4 and branching factor of 10, resulting in 10 000 leaves. Table III and Fig. 7 include the results obtained with this approximated local feature representation. The quality of the localization in urban panoramic datasets using the local features is comparable to the gist descriptor based approaches proposed in our work. A discussion of why we have chosen quantized local features as baseline for our comparisons is included in the following Section IV-D together with a detailed analysis of memory and computational requirements presented for all approaches.

Fig. 8 shows examples of the top two retrieved results for two different queries. Note that even though the matched panoramas may look quite alike at the shown scale, sequence

TABLE III: Localization results varying the image similarity evaluation (exact vs approximate search; global vs local features)

		top 1	top 10	top 20	top 40
E_{gist}	dist $<$ 10m	0.88	0.92	0.93	0.94
	dist $<$ 20m	0.93	0.94	0.95	0.96
	dist $<$ 40m	0.94	0.96	0.97	0.97
VQ_{gist} ($k = 40$)	dist $<$ 10m	0.36	0.61	0.68	0.74
	dist $<$ 20m	0.47	0.66	0.72	0.77
	dist $<$ 40m	0.52	0.69	0.75	0.80
VQ_{gist} ($k = 1000$)	dist $<$ 10m	0.74	0.84	0.85	0.87
	dist $<$ 20m	0.84	0.87	0.88	0.89
	dist $<$ 40m	0.86	0.89	0.90	0.91
KD_{gist}	dist $<$ 10m	0.85	0.89	0.90	0.91
	dist $<$ 20m	0.90	0.91	0.92	0.93
	dist $<$ 40m	0.93	0.93	0.94	0.95
V_{surf}	dist $<$ 10m	0.81	0.89	0.91	0.92
	dist $<$ 20m	0.87	0.92	0.94	0.95
	dist $<$ 40m	0.88	0.93	0.95	0.97

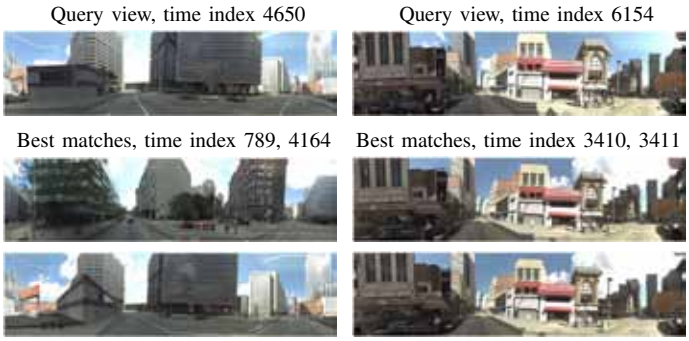


Fig. 8: Street View dataset. The two most similar panoramas found for two query views. Left column shows one test where the most likely candidate is not correct but the second is.

frame index shows that these panoramas are actually far from each other (in time). As can be seen in the examples, the result provided using only gist-based representation, i.e. first match, may not always be the correct localization as we can see in the figure on the left. If we would consider a set of top candidates selected by our approach, results can be refined by post-processing the top candidates with exact nearest neighbor matching using local features (for instance SURF [4]) and choosing the candidate with the highest number of correspondences. We validate in our experiments that selecting the candidate (from the top-10 retrieved results) with the highest number of SURF correspondences, without any spatial verification, finds the correct solution 98% of the time if a correct solution existed within the top-10 retrieved results.

C. Localization in New College Dataset

We have also evaluated our place recognition algorithms on the New College dataset [46][1]. It provides panoramic data acquired from a multi-camera system mounted on a mobile robot platform. The trajectories are performed along pedestrian paths around college buildings and parks. The provided dataset contains images which are stitched panoramas. The dataset

[1]http://www.robots.ox.ac.uk/NewCollegeData/

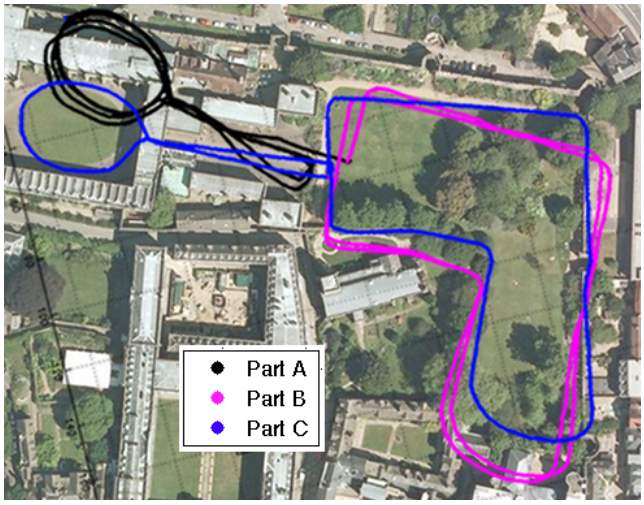


Fig. 9: New College dataset. Plot of GPS tags from its three parts (provided by the dataset authors [46]), manually aligned with an aerial view of the traversed environment. Data detailed description at authors' website. (Best viewed in color)

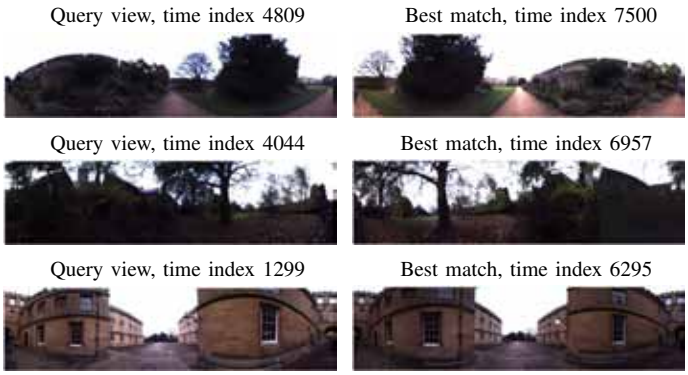


Fig. 10: New College dataset. Each row visualizes a query view (left column) and its corresponding top retrieved result (right column). In the top two rows, besides lightning changes, the query's best match occurs rotated by 180^o. Our proposed similarity measure detects the match correctly in the three examples.

consists of three parts, shown in Fig. 9. The locations in the dataset have been provided with GPS tags but as can be seen in the figure some revisited parts of the dataset contain GPS errors of up to 40m even though they correspond to same locations, e.g., the small loops on the top left.

Part C, which covers the entire area once, is used as the reference set and images from trajectories A and B are used as test images. Fig. 10 shows three examples of test images and the most similar reference image retrieved. Fig. 11 presents a visual summary of the localization results in this experiment. A line is drawn between each test image (black colored) and the most similar reference image (blue colored) found using E_{gist}. We can observe that generally the most similar panorama selected corresponds to a correct location. Results in the figure are shown for a GPS acceptance threshold of 40m. Table IV shows the quantitative localization results. Note that as we

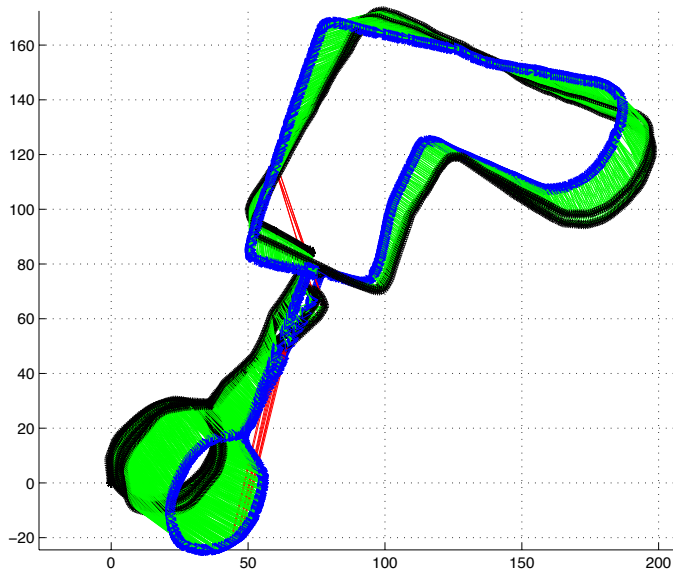


Fig. 11: Localization results. Test images (colored in black) are connected with the most similar reference image found (in blue). Green and red lines visualize correct and incorrect results respectively, using an acceptance threshold of 40m.

TABLE IV: Localization results with New College dataset

		top 1
E_{gist}	dist $<$ 10m	0.25
	dist $<$ 20m	0.73
	dist $<$ 30m	0.92
	dist $<$ 40m	**0.99**

increase the threshold distance for accepting a localization result, the performance improves drastically. For example, when distance is set at 10m, the accuracy is 25%. However when the distance threshold is increased to 40m, it improves to 99%. This accounts for the errors in the GPS annotations which is corroborated by the visualization in Fig. 11.

D. Space and Time Complexity Issues

The previous subsections evaluated the quality of the localization results for the different proposed approaches. We have seen in the summary from Table III and Fig. 7 that results based on the gist representation, E_{gist} and KD_{gist}, are comparable or outperform local feature "bag of words" model V_{surf} in our setting. In this section, we discuss in more detail the space and time of execution tradeoffs for the different proposed approaches.

Gist based representation provides advantages in memory requirements and execution times. The evaluation by Douze et al [20] illustrated considerable difference in using exact local feature image representation compared to gist based approaches with a difference of more than an order of magnitude for the memory requirements. We have chosen quantized local features as a baseline for our comparisons because although quantized approaches are less accurate than exact methods, the representations using the quantized local features are comparable with gist representations with regards to memory and efficiency. As shown in the aforementioned evaluation,

TABLE V: Memory bytes required and execution time per test for a 10,000 image dataset

	Approximate storage requirements	Average execution time per test (one image search)	d (f) length
E_{gist}	$(N \times f \times d)$: 146 MB	$O(N)$: 5.23 s	960(4)
KD_{gist}	$(N \times f \times d)$: 20 MB	$O(\log N)$: 0.09 s	500(1)
VQ_{gist} (k=40)	$(k \times d + N \times k)$: 1.7 MB	$O(\log k)$: 0.8 s	960
VQ_{gist} (k=1000)	$(k \times d + N \times k)$: 42 MB	$O(\log k)$: 1.02 s	960
V_{surf} (k=10000)	$(k \times d + N \times k)$: 382 MB	$O(\log k)$: 5.5 s	64

when using the exact methods for local features, scalability turns unfeasible for standard computers due to memory and computational requirements.

The approximate requirements of the different approaches for a dataset of 10 000 images (a size comparable to our Street View dataset) are provided in Table V. The table includes the parameters used to estimate the required storage - N the number of images, f the number of features per image, d the length of the feature descriptor and if applicable, k the size of the vocabulary used. It also provides the time complexity of the execution of a single search in the different methods. From the results, we note that the *k-d* tree based method provides the best compromise between accuracy, memory requirements and speed of the retrieval. The advantage of "bag of words" approaches (VQ_{gist} and V_{surf}) is the flexibility to handle bigger datasets: they require lower memory and search time when large datasets are used, since their time search growth is constant with the size of the database (N) and the storage depends less significantly on it. However, performance of the "bag of words" method critically depends on the size of the vocabulary. The performance improves as we augment the size of the vocabulary, but it also raises the computational cost and memory storage requirements, making VQ_{gist} similar to KD_{gist} for our settings without reaching as good recognition rates and efficiency.

It was already pointed out in previous evaluations of local features vs. gist based representations in conventional images [20], that memory and efficiency wise, the gist image representation presents clear advantages for scalability to large datasets. Their work showed that quality of recognition for global descriptor image representation was only comparable to local feature approaches when searching for near-duplicates. Our current problem is related to the duplicate detection problem because, in spite of weather, lighting or occlusion condition changes that make it harder, we consider that the reference information covers the whole environment and we can usually expect a good solution among the reference information. This, together with the fact that large FOV are more discriminative than conventional images, makes the good performance of gist based approaches to find the most similar location not surprising.

The work in [47] presents a detailed analysis of complexity and storage requirements for approaches similar to ours. Following their analysis, with datasets of 50K images, using exact local feature descriptors memory requirements start to be problematic for a conventional computer and "bag of words" methods are recommended with vocabularies about 1 million words. In our case, due to the smaller memory requirements of global descriptors; 150KB for the conventional image local feature descriptors vs 15KB for our panorama gist descriptor; we could then handle datasets up to one order of magnitude higher before using quantized descriptor spaces.

As a summary, we state that in our settings the approximate nearest neighbor technique based on *k-d* tree is the best compromise between performance (execution time and discriminability) and storage requirements. It runs around 8 times faster to localize a new query in Matlab and also has the advantage of reduced dimensionality of the descriptor. However, if the size of the reference image database grows by more than an order of magnitude, memory requirements of this approach would start to be problematic and the quantized version of local or global descriptors should be used. Local descriptors may provide better discriminability in some cases, although global descriptors decrease computation requirements since far less features have to be evaluated per image and smaller representative vocabularies can be obtained.

V. Loop Closure Detection

As previously mentioned, appearance based localization and loop closure detection are closely related tasks. They share the basic goal of finding the closest view in the reference set, but loop detection considers as reference set for a given view all images acquired previously during current trajectory. Loop detection requires to evaluate the similarity in an on-line manner. This framework raises different issues and possibilities that can be exploited through spatio-temporal constraints. In this section, we first present a basic approach to detect revisited places (Section V-A1). We then describe steps to incorporate temporal context for an improved loop detection method which explicitly models the probability of loop closure detection in a Bayesian filter framework (Section V-A2).

A. Loop closure detection approaches

1) Basic revisited places detection: The problem of loop closure detection involves detecting a return to a previously visited location while traversing the environment. In our basic loop closure detection method, we extend the place recognition approaches described in the previous section to an on-line setting. Given a query view, to decide if it corresponds to a revisited place, we search for similar views among the locations visited so far. From the previously described gist-based similarity measures, we use E_{gist} since it provided the best accuracy in the localization experiments. At each location of the vehicle's traversal of the environment, the following criteria is used to detect a revisit at that location.

Gist similarity distances are computed between the current location and past locations. The closest match to the current location (at time t) is defined as the location which has the

minimum gist similarity distance, i.e.,

$$d_{min} = \min_{i \in [0, t-p]} (\quad dist(\mathbf{g}^t, \mathbf{g}^i)),$$

where $dist(\mathbf{g}^t, \mathbf{g}^i)$ is computed using (2). If the gist similarity distance to the closest match (d_{min}) is below a threshold (τ_{loop}), we predict a loop closure at that location. Otherwise, it is considered a visit of a new location. The search is not performed for the last p locations to discard immediately preceding locations since the appearances of scenes from neighboring locations are very similar to each other.

A drawback to this approach is that it requires a user specified threshold for predicting loop closure detections. We now propose a method which overcomes this drawback and also incorporates temporal information.

2) Bayes filter for loop closure detection: Instead of considering only the current view for loop closure detection, we introduce a simple temporal model for determining the probability of loop closure. This model captures the fact that if the current view has a high probability of being a loop closure, the subsequent view is more likely to be a loop closure. As the vehicle traverses through the environment and acquires new images, the gist based panorama representation is used to evaluate the probability of loop closure in a Bayesian filtering framework.

We now describe how the state is represented and estimated at each time instance. Our approach is motivated by [37]. Let S_t be the random variable that represents the loop closure event at time *t*. The observable output is the visual appearance I_t of the current location x_t, which is represented as the tuple of gist descriptors of the current location's image. $S_t = i$ represents the event that the location with image I_t is a revisit of previously traversed location x_i with image I_i. $S_t = 0$ represents the event that no past location is being revisited at time *t*, hence a new location is visited. In the Bayesian framework, the problem of loop closure detection can be formulated as searching for a past location *j* which satisfies:

$$j = \arg\max_{i \in [0, t-p]} p(S_t = i | I^t), \tag{6}$$

where $I^t = I_1, I_2, ..., I_t$ are the images of the locations visited so far in chronological order. Therefore, we need to estimate the posterior probability $p(S_t = i | I^t)$ for all $i \in [0, t-p]$ in order to detect if a loop closure has occurred or not. Expanding the expression for the posterior probability we get

$$\begin{aligned} p(S_t = i | I^t) &= p(S_t = i | I_1, I_2, ..., I_t). \\ &= p(S_t = i | I^{t-1}, I_t) \end{aligned} \tag{7}$$

Applying Bayes rule to the right hand side of (7),

$$\begin{aligned} p(S_t = i | I^t) &= \frac{p(I_t | S_t = i)\, p(S_t = i | I^{t-1})}{p(I_t | I^{t-1})} \\ &= \alpha\, p(I_t | S_t = i)\, p(S_t = i | I^{t-1}) \end{aligned} \tag{8}$$

where α is a normalization constant (since the denominator $p(I_t | I^{t-1})$ is constant relative to the state variable). The conditional probability $p(I_t | S_t = i)$ is the likelihood of the currently observed image I_t (represented by the tuple of gist descriptors) given $S_t = i$. The right part of (8) can be further decomposed as

$$\alpha\, p(I_t | S_t = i) \sum_{j=0}^{t-p} p(S_t = i | S_{t-1} = j)\, p(S_{t-1} = j | I^{t-1}), \tag{9}$$

where $p(S_t = i | S_{t-1} = j)$ is the state transition probability for observing event $S_t = i$ given $S_{t-1} = j$. We now describe the state transition probabilities and how to estimate the likelihood term of this model.

Likelihood. The likelihood function for loop closure event S_t is based on the similarity between the panoramas of the two locations:

$$p(I_t | S_t = i) = exp\left(\frac{-dist(g^t, g^i)}{\sigma^2}\right), \tag{10}$$

with $dist(g^t, g^i)$ computed as in (2) and σ^2 a user-defined variance. This assumes an exponential decay with the gist similarity distance for the observation likelihood.

We also need to compute the likelihood for the non-loop closure event, $S_t = 0$, which does not have any image associated with it. Therefore, to compute the likelihood for this event, we construct a *virtual* image using the gist descriptors of traversed locations. The virtual image at time *t*, denoted by I_{0t}, is built as the average of the gist tuples of the past *K* locations:

$$I_{0t} = \frac{\sum_{i=t-K}^{t-1} g^i}{K}, \tag{11}$$

where g^i is the 4-tuple described in Section III. For locations at time $t < K$, all the past locations are used for calculating the virtual image. The virtual image I_{0t} represents the average appearance of the environment at time t. Its construction reflects the idea that the gist similarity distance associated with this virtual image will change according to the current location. When a new location is visited (i.e. no loop-closure occurs), the image at the current location - I_t - should appear more similar to the average appearance of the environment I_{0t} than an image I_i from a previous timestep. However, when the vehicle returns to a previously visited location x_i, the image I_t should appear more similar to the image I_i since I_i is more specific to that location than the average appearance described by I_{0t}.

State Transition. The probability $p(S_t | S_{t-1})$ is used to model all possible transitions between states at times $t-1$ and t. The state transition diagram is summarized in the diagram in Fig. 12 and we use the following transition probabilities:

- $p(S_t = 0 | S_{t-1} = 0) = p_{0 \to 0}$ is the probability of non-loop closure event at time t given that no loop closure occurred at $t-1$.
- $p(S_t = i | S_{t-1} = 0) = \frac{1 - p_{0 \to 0}}{t-p}$, with $i \in [1, t\text{-}p]$, stands for the probability of a loop closure event at time t given that none occurred at $t-1$. The possible states for a moving vehicle are either the visit of a new location or the revisit of the past locations (except the immediately preceding p locations). This implies that $\sum_{i=0}^{t-p} p(S_t =$

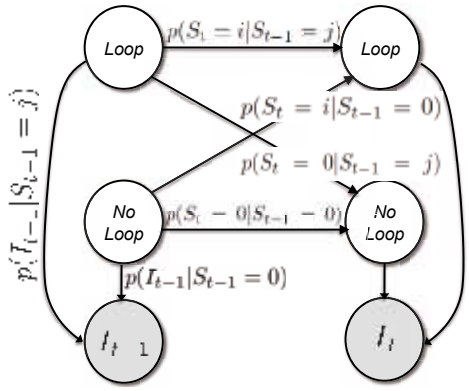


Fig. 12: Diagram of the HMM modeling of the loop closure or non-loop closure events.

$i|S_{t-1} = 0) = 1$ and we assign equal probabilities to all the $t - p$ possible loop closure events.

- $p(S_t = 0|S_{t-1} = j) = p_{j\to 0}$, with $j \in [1, t\text{-}p]$, the probability of non loop closure event at time t given that loop closure occurred at $t - 1$.
- $p(S_t = i|S_{t-1} = j) = p_{j\to i}$, with $i, j \in [1, t\text{-}p]$ is the probability of loop closure at time t at viewpoint x_i given that loop closure occurred at $t-1$ at viewpoint x_j. This probability is represented by a Gaussian distribution based on the distance between viewpoints x_i and x_j. It uses the assumption that among loop closure events, the probability of transitioning to the nearest neighbor locations will be higher and it will decrease as their distance to the current location increases. The variance of the Gaussian distribution is chosen in such a way that it is non-zero for exactly $2w$ neighbors of viewpoint x_j i.e. $p_{j\to i} > 0$ if $i \in [j - w, j + w]$. This represents the varying image similarity between neighboring viewpoints according to the distance between them. The non-zero probabilities in this case must sum to $1 - p_{j\to 0}$ since $\sum_{i=0}^{t-p} p(S_t = i|S_{t-1} = j) = 1$.

B. Experiments on loop detection

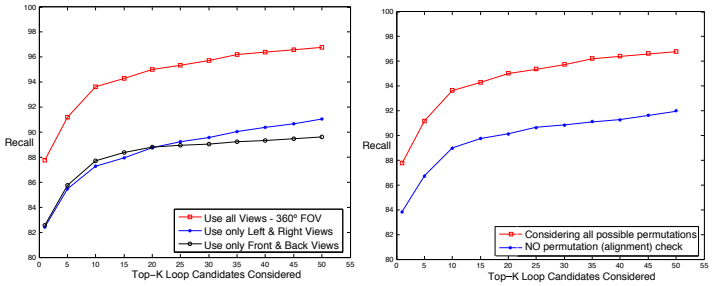
This section evaluates the proposed image representation and methods for loop closure detection. We first discuss the metrics used for evaluation. We then show the effect of the main design decisions in our proposed panorama representation for loop closure detection. This basic approach is run on two different datasets, including a comparison with a local feature based image representation. We then illustrate the advantages and improvements obtained with the Bayes filter loop detection approach.

1) Evaluation: Our metrics for evaluating the performance of the framework are precision and recall:

$$Precision = \frac{TP}{TP + FP}\,100\% \qquad Recall = \frac{TP}{TP + FN}\,100\%$$

where TP is true positive, FP is false positive and FN is false negative. A location is considered a true positive if loop closure is successfully detected at that location. Loop closure locations in the ground truth for which loop closure is not detected successfully are false negatives, while locations incorrectly predicted as loop closure are false positives. It is trivial to obtain perfect recall by predicting loop closure at every query location but it will lead to poor precision due to the high number of false positives. Hence, it is essential to achieve high precision, while trying to maintain a good recall.

To evaluate our results, for a given query view, locations within a threshold distance are considered to be correct loop closure detections. In order to avoid considering immediately preceding locations as loop closures, we use a window of p preceding frames so that views taken within short time of each other are not considered for loop closure evaluations.



(a) (b)

Fig. 13: Street View dataset. Performance of E_{gist} for basic loop closure detection using the top-k nearest neighbors. Our approach results are shown in red. (a) Maximum recall rate of our approach (whole panoramic view) vs. a subset of the views; (b) Maximum recall rate of our approach (doing circular permutation check) vs not checking the views alignment.

2) Influence of image representation decisions: Similar to our experiments for the localization problem, we analyze the impact of choices for panorama representation: the use of *wide field of view* images and the alignment analysis, i.e., using the four circular permutation possibilities when computing the gist similarity as described in (2).

These experiments are run on the previously described Street View dataset (Section IV-B). To set the ground truth of actual revisited locations, we used a threshold distance of 10m and a window of size $p = 25$ yielding a ground truth set of 3362 revisited locations (the same as the test set locations in Fig. 6). The effect of using 360^o field of view for loop closure detection is demonstrated in Fig. 13a. Notice that having the full FOV significantly improves the discrimination capability of the loop closure detection. Considering a smaller portion of the panorama, e.g., only two faces, which resembles the localization with traditional cameras, is shown to be detrimental to the overall performance. Fig. 13b shows the effectiveness of the proposed similarity measure, which compares each query panorama with all rotational permutations of the reference views instead of using one single alignment between the panoramas.

3) Comparison to local features based methods: The comparison is carried out on the New College dataset (detailed previously in Section IV-C) since results for a local features

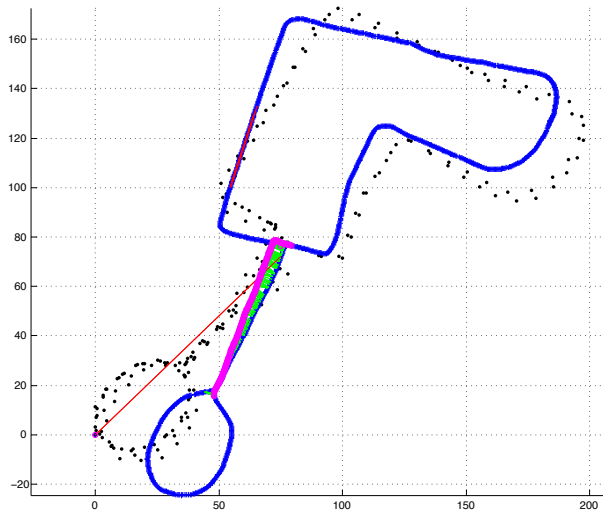


Fig. 14: New College dataset loop detection results. Green lines represent correct detected loops. Locations are plotted according to the GPS tags available: trajectory used for this experiment (part C) uses blue dots; locations shown in pink are the ground truth for revisited locations. The black dots correspond to GPS tags of images from other parts of the dataset. They were not used in this experiment but visualize the available GPS accuracy.

based method are publicly available and they provide us an opportunity to carry out a direct comparison. We use only part C of the dataset, because errors between GPS tags of the other parts are too high. Loop detection results for this experiment are obtained by running the basic loop closure detection from Section V-A1. We discard a window of $p = 50$ preceding frames as non-valid candidates for loop closure. A loop detection is predicted by setting the threshold $\tau_{loop} = 0.5$. This value for the threshold provides the maximum recall for a precision of 100%. Figure 14 visualizes the loop closure detection results, by drawing a line between later images and the previous location they are revisiting. The loop detection is considered correct if the two views are located within 15m according to the reference GPS tags. Successful detections are shown with green lines and incorrect ones with red lines.

Table VI presents the precision for our system in comparison to the results in [48] on the same dataset[3]. We achieve a higher recall than the basic FAB-MAP approach, and are competitive with FAB-MAP 3D [48], where visual information is augmented with range data. The proposed gist representation does not suffer from the ambiguities, responsible for matching errors, generated by approaches using local feature quantization methods. An additional advantage is the compactness of gist-based descriptors.

4) Bayesian Loop Closure Detection: We now present the evaluation of the Bayesian filtering approach which uses temporal context in comparison to the basic loop detection method. We also illustrate the impact of different parameters

[3]The test sets for the final recall numbers may differ as the exact test set used was not provided in [48]

TABLE VI: E_{gist} based basic loop detection compared to published results for the New College dataset for precision=100%

	Gist-based	FAB-MAP [48]	FAB-MAP 3D [48]
Recall	0.71	0.42	0.74

to the task of loop closure detection. The experiments have been performed on the Street View dataset.

As we gradually evaluate individual frames, the posterior probability for non-loop and all possible loop closure events at each query location is computed with (8). The event with the maximum posterior probability is then chosen as its best match. Experiments were carried out varying the state transition probabilities and the method for virtual image generation. In the final configuration for all the experiments shown next, the virtual image was computed as the average 4-tuple gist descriptor of the past 1,000 locations i.e. when computing I_{0t}, K = 1,000. The number of neighbors which take non-zero values for transition from loop closure event to loop closure event $p(S_t = i|S_{t-1} = j)$ was set to 10.

The results for varying $p_{j\to 0}$ i.e. $p(S_t = 0|S_{t-1} = j)$ are shown in Table VII. A high value for $p_{j\to 0}$ should lead to a low recall rate because the possibility of a non-loop closure event at time t after a loop closure event at time $t-1$ is considered more probable compared to loop-closure. The table shows that the results are consistent with the above assumption. It can be observed that a relatively high value detected fewer loop closure locations. Consequently, a decrease in $p_{j\to 0}$ lead to higher recall rates, but also lower precision.

Similarly a high value for $p_{0\to 0}$ i.e. $p(S_t = 0|S_{t-1} = 0)$ should lead to detection of fewer loop closure locations. By increasing $p_{0\to 0}$, we increase the probability of non-loop closure at time t after no loop closure is observed at time $t-1$ or in other words, decrease the possibility of observing

TABLE VII: Precision-recall by varying $p_{j\to 0}$ ($p_{0\to 0}$= 0.8, σ^2= 1)

$p(S_t = 0\|S_{t-1} = j)$	**Precision**	**Recall**
0.1	100	15.56
0.01	79.64	71.33
0.005	71.40	72.99
0.002	64.14	74.48

TABLE VIII: Precision-recall varying $p_{0\to 0}$ ($p_{j\to 0}$= 0.01, σ^2= 1)

$p(S_t = 0\|S_{t-1} = 0)$	**Precision**	**Recall**
0.9999	94.51	55.29
0.999	93.42	57.82
0.99	91.45	61.12
0.9	86.92	67.79
0.8	79.64	71.33
0.7	70.38	73.71

TABLE IX: Precision-recall varying σ^2 ($p_{j\to 0}$= 0.1, $p_{0\to 0}$= 0.8)

σ^2	**Precision**	**Recall**
1.0	100	15.56
0.81	99.19	36.32
0.64	97.35	56.72
0.49	94.88	68.32
0.25	80.43	77.28

(a) (b)

Fig. 15: Green colored locations are TP, Blue - TN, Red - FN and Yellow - FP. (a) Analysis of loop detection in areas not fitting the "Manhattan world" assumption. Some of the locations are still correctly identified by using the Bayesian loop detection method. (b) Sample panoramas from the same location with a relative rotation violating the "Manhattan world" assumption.

a loop closure. But by increasing $p_{0\to0}$, we can achieve a higher precision since the possibility of predicting an incorrect loop closure will decrease leading to fewer false positives. The results shown in Table VIII validate this assumption.

By changing the variance σ^2 used to compute $p(I_t|S_t = i)$ in (10), we vary the probability of observing the image given the state. If σ is decreased, the value of the likelihood function subsequently increases. This could lead to a higher recall due to the increase in the likelihood values. Table IX shows the result at different values of σ. As expected, the recall rate increases as we decrease σ but precision drops.

The robustness of the proposed image representation and similarity measure along with HMM can be illustrated in the following example. Fig. 15a corresponds to a zoomed area of top left corner of the whole trajectory where the vehicle turns on streets which are not intersecting at 90^o. Note that due to the temporal model, if the loop closures were detected successfully in the past, the model continues to correctly detect loop closure despite the relatively oblique turns. Fig. 15b shows two panoramas segmented into the four parts from the same location but more oblique relative orientation. Note that the content in each part is quite different and not related by a simple circular permutation of the 4 views. The gist similarity based on permutations of the views will not give a high score between these two images. As the vehicle progresses the views will be eventually better aligned with the HMM helping to overcome some of these issues.

5) Discussion: We have shown that in the case of urban panoramic data, the proposed approach provides comparable or better performance to previously published results which use local features. Besides, using temporal information proves useful compared to a standalone loop detection method.

Fig. 16 provides the precision-recall curve comparing the basic loop detection method to the Bayesian loop detection method. To generate the precision-recall curve for the basic loop detection method, we vary τ_{loop} for the basic loop detection method described in Section V-A1. As τ_{loop} increases, the recall increases (more locations satisfy the gist similarity distance threshold) and precision decreases. For the

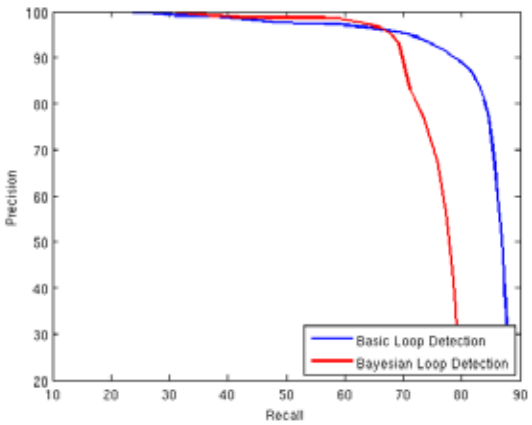


Fig. 16: Precision-recall curves comparing basic loop detection method to the Bayesian loop detection.

(a) (b)

Fig. 17: Comparison of loop detection results at 100% precision. (a) Basic loop detection method (b) Bayesian loop detection method. There were three areas in the city where using temporal information improved the loop detection. The color scheme is the same as Fig. 15. (Best viewed in color)

Bayesian loop detection precision-recall curve, we set σ^2 = 0.81 and $p_{j\to0}$= 0.1. The precision-recall values are computed for varying $p_{0\to0}$. As $p_{0\to0}$ is decreased, precision decreases and recall increases. At 100% precision, the basic loop detection achieves a recall of 20.9%. At the same precision, the Bayesian loop detection method had 29.4% recall, an increase of more than 8% over the basic loop detection method. This proves the usefulness of temporal information for such methods. A visualization comparing the basic loop detection method results to that of Bayesian loop detection at 100% precision is provided in Fig. 17.

The Bayesian filtering framework presented further improves loop closure detection compared to a non-Bayesian approach. However, the current design of the Bayesian approach will not be able to deal efficiently with very large datasets. From (8), we note that after the traversal of N locations, the posterior probability is computed for all previous locations except for the last p locations. To compute this, the posterior

from all the past time steps have to be kept at runtime. Therefore, the time complexity is $O(N)$ and the memory cost is $O(N^2)$. This makes it computationally expensive for large scale experiments. Instead of the exact method described here, we can use an approximate method which will reduce the number of computations at runtime. One such model used in the past is the particle filtering based approximation presented in [49] which had a constant runtime and $O(N)$ memory requirements.

VI. Conclusions and Future Work

In this paper, we have presented how to adopt a global gist descriptor for panoramas and proposed associated similarity measures which strike a good performance between discriminability and computational complexity in urban environments. For the proposed representation and similarity measures, we have evaluated performance and scalability of location recognition using different strategies and introduced a novel algorithm for loop closure detection. The effectiveness of the approach has been demonstrated on extensive experiments with 360^o field of view panoramas, comparing them with local feature based approaches.

For location recognition, the best results were obtained using a *k-d* tree based approach with a PCA reduced version of the panoramic gist descriptor. The performance of the proposed representation was comparable or better than local feature based approaches, with the advantage of higher efficiency and smaller memory storage requirements for datasets up to 100K images. We also compared the proposed approach with a state of the art technique for loop closure detection based on local features, reporting favorable or comparable performance. Moreover, we described an approach to incorporate temporal consistency, where the probability of a loop closure is determined in a Bayesian filtering framework with a HMM model.

The presented work emphasizes the issue of a compact image representation and its effect on scalability and efficiency for image-based localization. Additional improvements can be achieved by endowing the database of images acquired along the route with additional topological structure to achieve more efficient loop closure detection framework and incorporating stronger geometric constraints in the final verification stage.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments. This project has been supported by National Science Foundation grant IIS-0347774, Army Research Office ARO 60054-CS and Spanish projects DPI2012-31781 and DPI2012-32100.

References

[1] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. of Computer Vision*, vol. 42, no. 3, pp. 145–175, May 2001.

[2] ——, "Building the gist of a scene: The role of global image features in recognition," *Visual Perception, Progress in Brain Research*, vol. 155, 2006.

[3] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[4] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proc. of ECCV*, 2006, http://www.vision.ee.ethz.ch/ surf/.

[5] F. Fraundorfer, C. Engels, and D. Nistér, "Topological mapping, localization and navigation using image collections," in *Proc. of IEEE/RSJ IROS*, 2007, pp. 3872–3877.

[6] Z. Zivkovic, O. Booij, and B. Krose, "From images to rooms," *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 411–418, 2007.

[7] K. Konolige and J. Bowman, "Towards lifelong visual maps," in *Proc. of IEEE/RSJ IROS*, 2009, pp. 1156–1163.

[8] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proc. of IEEE CVPR*, 2006, pp. 2161–2168.

[9] J. Sivic and A. Zisserman, "Video Google: Efficient visual search of videos," in *Toward Category-Level Object Recognition*, ser. LNCS. Springer, 2006, vol. 4170, pp. 127–144.

[10] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. of IEEE CVPR*, 2007.

[11] M. Cummins and P. Newman, "Highly scalable appearance-only SLAM - FAB-MAP 2.0," in *Robotics Science and Systems*, Seattle, USA, 2009.

[12] C. Valgren and A. J. Lilienthal, "Sift, surf & seasons: Appearance-based long-term localization in outdoor environments," *Robotics and Autonomous Systems*, vol. 58, no. 2, pp. 149–156, 2010.

[13] I. Ulrich and I. Nourbakhsh, "Appearance-based place recognition for topological localization," in *Proc. of IEEE ICRA*, 2000, pp. 1023–1029.

[14] J. Košecká and F. Li, "Vision based topological Markov localization," in *In Proc. of IEEE ICRA*, 2004, pp. 1481–1486.

[15] E. Menegatti, T. Maeda, and H. Ishiguro, "Image-based memory for robot navigation using properties of the omnidirectional images," *Robotics and Autonomous Systems*, vol. 47, no. 4, pp. 251–267, 2004.

[16] A. C. Murillo, C. Sagüés, J. J. Guerrero, T. Goedemé, T. Tuytelaars, and L. Van Gool, "From omnidirectional images to hierarchical localization," *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 372–382, 2007.

[17] T. Goedemé, M.Nuttin, T. Tuytelaars, and L. Van Gool, "Omnidirectional vision based topological navigation," *Int. J. of Computer Vision*, vol. 74, no. 3, pp. 219–236, 2007.

[18] B. C. Russell, A. Torralba, C. Liu, R. Fergus, and W. T. Freeman, "Object recognition by scene alignment," in *Advances in Neural Information Processing Systems*, 2007.

[19] A. Torralba, R. Fergus, and Y. Weiss, "Small codes and large databases for recognition," in *Proc. of IEEE CVPR*, 2008.

[20] M. Douze, H. Jégou, H. Sandhawalia, L. Amsaleg, and C. Schmid, "Evaluation of gist descriptors for web-scale image search," in *Int. Conf. on Image and Video Retrieval*, july 2009.

[21] O. Saurer, F. Fraundorfer, and M. Pollefeys, "Visual localization using global visual features and vanishing points," in *CLEF*, 2010.

[22] C.-K. Chang, C. Siagian, and L. Itti, "Mobile robot vision navigation & localization using gist and saliency," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010.

[23] P. Viswanathan, T. Southey, J. Little, and A. Mackworth, "Place classification using visual object categorization and global information," in *Canadian Conf. on Computer and Robot Vision*, may 2011, pp. 1 –7.

[24] J. Courbon, Y. Mezouar, and P. Martinet, "Autonomous navigation of vehicles from a visual memory using a generic camera model," *Intelligent Transport System*, vol. 10, pp. 392–402, 2009.

[25] A. Torralba, R. Fergus, and W. T. Freeman, "80 million tiny images: a large database for non-parametric object and scene recognition," *IEEE PAMI*, vol. 30, no. 11, pp. 1958–1970, November 2008.

[26] R. Anati and K. Daniilidis, "Constructing topological maps using markov random fields and loop closure detection," in *Proc. of NIPS*, 2009, pp. 37–45.

[27] J. M. Coughlan and A. L. Yuille, "The manhattan world assumption: Regularities in scene statistics which enable bayesian inference." in *Proc. of NIPS*, 2000.

[28] A. C. Murillo and J. Košecká, "Experiments in place recognition using gist panoramas," in *9th IEEE Workshop OMNIVIS, held with ICCV*, 2009, pp. 2196–2203.

[29] G. Singh and J. Košecká, "Visual loop closing using gist descriptors in manhattan world," in *Omnidirectional Robot Vision workshop, held with IEEE ICRA*, 2010.

[30] A. C. Murillo, P. Campos, J. Košecká, and J. J. Guerrero, "Gist vocabularies in omnidirectional images for appearance based mapping and localization," in *10th Workshop OMNIVIS, held with RSS*, 2010.

[31] F. Li and J. Košecká, "Probabilistic location recognition using reduced feature set," in *In Proc. of IEEE ICRA*, 2006, pp. 3405–3410.

[32] G. Schindler, M. Brown, and R. Szeliski, "City-scale location recognition," in *Proc. of IEEE CVPR*, 2007, pp. 1–7.

[33] M. Cummins and P. Newman, "Fab-map: Probabilistic localization and mapping in the space of appearance," *Int. J. of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.
[34] A. Ranganathan, E. Menegatti, and F. Dellaert, "Bayesian inference in the space of topological maps," *IEEE Trans. on Robotics*, vol. 22, pp. 92–107, 2006.
[35] N. Tomatis, I. Nourbaksh, and R. Siegwart, "Hybrid simultaneous localization and map building: a natural integration of topological and metric," *Robotics and Autonomous Systems*, vol. 44, no. 1, pp. 3–14, 2003.
[36] M. Bose, P. Newman, J. Leonard, M. Soika, W. Feiten, and S. Teller, "An atlas framework for scalable mapping," in *In Proc. of IEEE ICRA*, 2003, pp. 1899–1906.
[37] A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer, "Fast and incremental method for loop-closure detection using bags of visual words," *IEEE Trans. On Robotics, Special Issue on Visual SLAM*, vol. 24, no. 5, pp. 1027–1037, 2008.
[38] K. L. Ho and P. Newman, "Detecting loop closure with scene sequences," *Int. J. of Computer Vision*, vol. 74, no. 3, pp. 261–286, September 2007.
[39] E. Olson, "Recognizing places using spectrally clustered local matches," *Robotics and Autonomous Systems*, vol. 57, no. 12, pp. 1157–1172, 2009.
[40] E. Olson, M. Walter, S. J. Teller, and J. J. Leonard, "Single-cluster spectral graph partitioning for robotics applications," in *Robotics: Science and Systems*, 2005, pp. 265–272.
[41] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *Proc. of IEEE ICCV*, 2003, pp. 273–280.
[42] J. S. Beis and D. G. Lowe, "Shape indexing using approximate nearest-neighbour search in high-dimensional spaces," in *Proc. of IEEE CVPR*, 1997, pp. 1000–1006.
[43] C. Silpa-Anan and R. Hartley, "Optimised kd-trees for fast image descriptor matching," in *Proc. of IEEE CVPR*, 2008.
[44] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *Int. Conf. on Computer Vision Theory and Application*. INSTICC Press, 2009, pp. 331–340.
[45] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," http://www.vlfeat.org/, 2008.
[46] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman, "The new college vision and laser data set," *Int. Journal of Robotics Research*, vol. 28, no. 5, pp. 595–599, May 2009. [Online]. Available: http://www.robots.ox.ac.uk/NewCollegeData/
[47] M. Aly, M. Munich, and P. Perona, "Indexing in Large Scale Image Collections: Scaling Properties and Benchmark," in *IEEE Workshop on Applications of Computer Vision (WACV)*, January 2011.
[48] R. Paul and P. Newman, "Fab-map 3d: Topological mapping with spatial and visual appearance," in *In Proc. of IEEE ICRA*, Anchorage, Alaska, 2010, pp. 2649–2656.
[49] A. Ranganathan, "Pliss: Detecting and labeling places using online change-point detection," in *Robotics Science and Systems*, Zaragoza, Spain, June 2010.

Ana C. Murillo obtained her PhD in Computer Science at the University of Zaragoza, Spain, in 2008 as a member of Robotics, Perception and Real Time group. Since then, she is researcher and assistant professor of the same institution. Her current research interests are in the area of computer vision, in particular, place recognition, semantic mapping and scene understanding, with special interest in omni-directional vision systems and robotics and assistive devices applications.

Gautam Singh received the Bachelor in Technology degree in Computer Science with honours from the International Institute of Information Technology, Hyderabad in 2007. He is currently working towards the PhD degree at George Mason University, Fairfax, Virginia. His research interests include computer vision, robotics and machine learning with an application to scene understanding from images and videos.

Jana Košecká is an Associate Professor at the Department of Computer Science, George Mason University. She obtained her M.S.E. in Electrical Engineering and Computer Science from Slovak Technical University and Ph.D. in Computer Science from University of Pennsylvania in 1996. She is the recipient of David Marr's prize (with Y. Ma, S. Soatto and S. Sastry) and received the NSF CAREER Award. Her general research interests are in Computer Vision, Machine Learning and Robotics.

J. J. Guerrero received the M.S. degree in Electrical Engineering and the Ph.D. degree from the Universidad de Zaragoza, Zaragoza, Spain, in 1989 and 1996, respectively. He is currently an Associate Professor and the Deputy Director of the Department of Informatics and Systems Engineering at Universidad de Zaragoza. His research interests are in the area of computer vision, particularly in 3-D visual perception, photogrammetry, visual control, robotics, and vision based navigation.