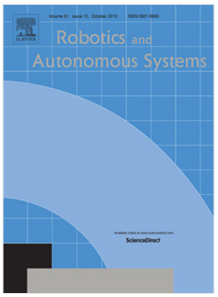
Accepted Manuscript



Robust visual semi-semantic loop closure detection by a covisibility graph and CNN features

Silvia Cascianelli, Gabriele Costante, Enrico Bellocchio, Paolo Valigi, Mario L. Fravolini, Thomas A. Ciarfuglia

PII: S0921-8890(16)30490-0
DOI: http://dx.doi.org/10.1016/j.robot.2017.03.004
Reference: ROBOT 2803

To appear in: *Robotics and Autonomous Systems*

Received date : 26 August 2016

Please cite this article as: S. Cascianelli, et al., Robust visual semi-semantic loop closure detection by a covisibility graph and CNN features, *Robotics and Autonomous Systems* (2017), http://dx.doi.org/10.1016/j.robot.2017.03.004

Robust Visual Semi-Semantic Loop Closure Detection by a Covisibility Graph and CNN Features

Silvia Cascianelli[a,*], Gabriele Costante[a], Enrico Bellocchio[a], Paolo Valigi[a], Mario L. Fravolini[a], Thomas A. Ciarfuglia[a]

[a]*Department of Engineering, University of Perugia, via Duranti 93, 06125, Perugia, Italy*

Abstract

Visual Self-localization in unknown environments is a crucial capability for an autonomous robot. Real life scenarios often present critical challenges for autonomous vision-based localization, such as robustness to viewpoint and appearance changes. To address these issues, this paper proposes a novel strategy that models the visual scene by preserving its geometric and semantic structure and, at the same time, improves appearance invariance through a robust visual representation. Our method relies on high level visual landmarks consisting of appearance invariant descriptors that are extracted by a pre-trained Convolutional Neural Network (CNN) on the basis of image patches. In addition, during the exploration, the landmarks are organized by building an incremental covisibility graph that, at query time, is exploited to retrieve candidate matching locations improving the robustness in terms of viewpoint invariance. In this respect, through the covisibility graph, the algorithm finds, more effectively, location similarities by exploiting the structure of the scene that, in turn, allows the construction of *virtual locations i.e.*, artificially augmented views from a real location that are useful to enhance the loop closure ability of the robot. The proposed approach has been deeply analysed and tested in different challenging scenarios taken from public

☆This work has been partly supported by funds under the project SEAL [SCN-398]. We also gratefully acknowledge the support of NVIDIA Corporation with the donation of the Tesla K40 GPU used for this research.

*Corresponding author

Email addresses: silvia.cascianelli@studenti.unipg.it (Silvia Cascianelli), gabriele.costante@unipg.it (Gabriele Costante), enrico.bellocchio@unipg.it (Enrico Bellocchio), paolo.valigi@unipg.it (Paolo Valigi), mario.fravolini@unipg.it (Mario L. Fravolini), thomas.ciarfuglia@unipg.it (Thomas A. Ciarfuglia)

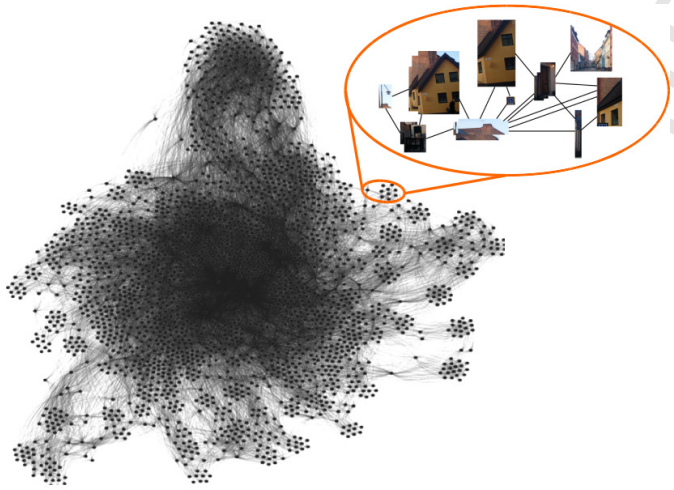


Figure 1: Graph Of Covisible CNN-Extracted features for semi-semantic visual Place Recognition: exemplar created graph.

datasets. The approach has also been compared with a state-of-the-art visual navigation algorithm.

Keywords: Place Recognition, Loop Closing, CNN Features, Graph, Semantic

1. Introduction

In the last decade, vision-based navigation systems have achieved impressive results [1, 2], considerably extending the application area of many robotic platforms. However, it is well known that, during long term operations, the localization performance may drop due to the drift of the estimation procedures, which can lead to a critical failure of most state-of-the-art systems. As a consequence, place recognition capabilities are crucial functions for loop closure detection and to increase the robustness of the overall estimation process.

Most of the existing place recognition strategies have been developed considering image sequences characterized by small viewpoint and lighting variations [3, 4, 5] and, within these scenarios, the results obtained are very promising. However, these simplified conditions do not hold in real life autonomous exploration contexts, where

the visual scene is typically affected by a number of challenging problems. For instance, seasonal or weather changes, natural or artificial daily illumination variations may severely affect the global appearance of the scene; further, dynamic elements, *e.g.*, pedestrians, vehicles or new static objects may cause appearance changes, since they can occlude or alter portions of the scene. In addition, traversing the same environment with different orientations can change the scene viewpoint, which may alter significantly the relative position of objects in the scene.

Place recognition algorithms that exploit low level visual features [3, 4] are typically very sensitive to strong image variations and, therefore, they do not provide good place recognition performance. Recent works [6, 7, 8] have shown that high level visual features, *i.e.*, semantic cues, provide a more robust representation of the scene since they also encode information about object categories and their mutual relations. In fact, semantic features provide a better characterization of the scene, which may facilitate the place recognition process by an autonomous robot. However, the detection of different objects may not be enough to unequivocally identify a specific place (*e.g.*, cars and buildings could be not discriminative in an urban environment). In these scenarios, the capability to discriminate between different spatial configurations and different views of the objects is crucial.

Motivated by the previous considerations, we have worked out a vision-based place recognition system that relies on a graph of semantic visual objects (see Figure 1, where it is shown the graph produced by our algorithm using 623 images taken from the IDOL dum_sunny3 + dum_cloudy1 dataset [9]) that is built incrementally during navigation. In order to improve the robustness with respect to appearance changes, the graph was built in such a way that the nodes collect similar image patches that are represented by high level descriptors extracted by the inner convolutional layer of a public CNN trained specifically for object recognition purposes [10].

Furthermore, to handle viewpoint changes and to ease the place recognition task, the edges of the graph are used to encode covisibility information, that is edges are created to connect the objects that have been observed together from the same point of view. The result is a *covisibility graph* [11, 12] that takes into account mutual object arrangements. In addition, the graph structure is exploited to build *virtual locations*

[13] in a new strategy that relies only on graph algebraic properties. Virtual locations represent synthetic views of the scene that are not present in the image database. As a consequence, the algorithm has the potential ability to recognize places even in the presence of strong viewpoint changes.

To summarize, the main contributions of this work are:

- The employment of semi-semantic features extracted by a pre-trained CNN on the basis of image patches, which are robust to appearance changes, in a covisibility graph-based model of the environment, which enhances the viewpoint robustness of the place recognition algorithm.
- The development of a procedure for the construction of artificial virtual locations via a novel parameter-free approach that exploits the covisibility graph properties to face critical loop closure detection situations.
- The extension of the work in [14], with a different strategy for virtual location construction and with a deeper experimental analysis on the performance of each part of the proposed algorithm, which was evaluated on an extended number of datasets with respect to the work [14].

To the best of our knowledge, apart from [14], there are no previous applications that use high level features extracted by a CNN as nodes of a graph to build an incremental model of the environment during the exploration. Another important specific novelty of this study is the development of a parameter-free procedure for inferring artificial views on the basis of the developed graph model.

The remainder of this paper is organized as follows. In Section 2, related work is discussed, while in Section 3 the graph construction procedure is described. Section 4 describes the pipeline of the algorithm and Section 5 provides a detailed description of the experimental results. Conclusion and future development are discussed in Section 6.

2. Related Work

Place recognition and loop closure detection are strictly related problems that are particularly important for autonomous robotic navigation in unknown environments. The main challenges for autonomous visual navigation in real life scenarios are viewpoint and appearance changes. A short categorization of the main research directions is provided below.

2.1. Appearance invariant approaches

The appearance change issue is typically faced via change removal methods, as in [15], via change prediction, as in Neubert *et al.* in [16], or by computing visual descriptors that exhibit invariance properties to appearance, as in [17], where the authors trained a multi-layer perceptron model to learn an appearance invariant set of descriptors. Among appearance invariant descriptors, features obtained from the inner layers of CNNs (that were pre-trained for object recognition tasks) have shown their effectiveness, as shown for instance in [18]. In particular, the authors in [15] and [19] were able to reduce significantly the effects of daily shadow and sunlight by transforming images in an illumination invariant colour space. The authors in [16] exploited the repeatability of the seasonal appearance changes, and built a super-pixel dictionary specific for each season and opportunely translated images captured in different seasons before matching. Authors in [17] studied the local changes of appearance of image patches subject to variation in lighting conditions and trained a multi-layer perceptron model and a convolutional multi-layer perceptron model for learning an appearance invariant feature descriptor. In [18, 20] the authors extensively studied the appearance and viewpoint invariance properties of the outputs produced by different layers of pre-trained convolutional neural networks, specifically designed for object recognition and scene categorization. They demonstrated that the inner convolutional layer outputs provide robust appearance invariant features, while higher fully connected layers provide viewpoint robust features.

2.2. Viewpoint invariant approaches

Viewpoint changes are usually more critical than appearance changes. Some successful Simultaneous Navigation And Mapping (SLAM) systems exploit, as loop closure detection modules, Place Recognition methods that are based on local invariant features. Some examples are FAB-MAP [3], which is based on SURF [21] features and ORB-SLAM [22], which is based on ORB [23] features. However, for visual Place Recognition algorithms viewpoint change is still a critical issue. Viewpoint invariance is generally addressed in an application dependant fashion, either by applying image rectification methods in case of mild viewpoint changes [24], or by considering the specific type of changes in the viewpoint that will be encountered while performing a specific task *e.g.*, [25, 26, 27]. In particular, the authors in [24] estimate and normalize affine parameters of local transformations in the images, but their approach is applicable only to objects with regular structure, as *e.g.*, buildings. Some heuristics or solutions designed for specific environments are applied to perform visual Place Recognition in case of specific severe viewpoint changes, such as in case of lane traversal in [25], panoramic vision in [26] or air-ground viewpoint change in [27].

2.3. Appearance and viewpoint invariant approaches

Scenarios characterized both by viewpoint and appearance changes are particularly challenging for the loop closure detection task. Promising solutions usually rely on CNNs specifically designed for place recognition [28] or on features extracted from a CNN designed for object recognition [6], or viewpoint synthesis [29], or exploiting robust sequence matching techniques [25].

2.4. Graph-based approaches

Modelling the environment as a graph requires the definition of what ”a node is” and of a criterion that defines the node connection mechanism. In order to preserve geometric information, in [7, 8] a geometric graph based on the distance between centres of 3D point clouds or 2D patches around a landmark was proposed. A recent work by Pepperell *et al.* [30] focused on maze urban environments and used roads as

directed edges connecting intersections to facilitate sequence matching in place recognition. Another general criterion for building graphs of the environment, while dealing with bidimensional images, is based on the covisibility of the landmarks, *i.e.*, an edge is created between landmarks if they are present in the same image. This approach was proposed in [13] and is also adopted in this work, with the important difference that, instead of using hand-crafted descriptors, we use features extracted by a convolutional layer of a pre-trained CNN that receives as input unprocessed image patches. Using a graph to model the environment allows the integration of additional information from other sources, such as robots or other intelligent systems. Hence, it provides a framework that can be easily integrated with network information, and with other environment specific visual object galleries following a transfer learning paradigm [31].

3. Incremental Covisibility Graph Construction

In this study we assume that the autonomous robot does not have at its disposal any prior information on the environment, that is, the visual exploration starts from scratch. As a new image is captured, patches containing objects are extracted and then processed by a CNN. The outputs of an inner layer of the CNN, along with the dimensions of the patches, are used to build a graph-based representation of the environment and to enrich the collection of landmarks encountered as the exploration progresses. In Figure 2 a block diagram of the operations performed in this knowledge acquisition phase is shown; below, the building blocks of this scheme are described in detail.

3.1. Semi-semantic Landmarks Extraction

The model of the environment is here obtained using high level visual landmarks extracted from the the scene acquired by the robot during navigation. For each new image the landmarks are derived from the processing of image patches that are likely to contain a generic object. In this work the number of extracted patches per frame is constant and fixed at 50. To obtain these patches we apply the algorithm by Zitnick *et al.* proposed in [32], named Edge Boxes, which efficiently detects a bounding box around a patch (of variable size and dimensions) that contains a high number of internal

contours compared to the number of contours exiting from the box. This fact indicates the presence of an intelligible object in the enclosed patch. The visual content of these patches, however, is not associated with an 'object label' *i.e.*, the Edge Boxes algorithm does not provide any object categorization for the object within the patches. For this reason our method can be considered a "semi-semantic" approach.

The 2D patches extracted by the Edge Boxes algorithm are directly processed by a pre-trained CNN and the output produced by an inner layer of the CNN is used as descriptor vector of the patch.

The strategy of using, as descriptors, the outputs provided by inner layers of a pre-trained CNN was proposed by some authors as in [33, 18, 34] thanks to the high representational power of deep nets.

In this study, we use the pre-trained AlexNet CNN [10], that is a well-known CNN used for Object Recognition, and select the output of the conv3 layer as descriptor vector. This choice is mainly motivated by the study reported in [18], where the output descriptors provided by the different layers of some CNNs for Object Recognition and Place Recognition were compared in order to find the best descriptor vector for the Place Recognition task. In particular, the authors of [18] demonstrated that in case of viewpoint changes, AlexNet has a slight performance improvement compared to CNNs trained on location-based images if considering the whole images. The same authors in [6] demonstrated that using region-based features rather than whole-image features provides a benefit in terms of viewpoint robustness. Since our region-based features are extracted on the basis of image patches containing objects, we decide to use AlexNet as feature extractor.

AlexNet works on fixed size images, while Edge Boxes produces patches with arbitrary dimensions, therefore we resize them in order to fit the AlexNet input dimensions. In order not to lose the original size information, the height and width of the patch are considered as additional descriptors, together with the conv3 output vector.

The conv3 layer output is a vector of $13 \times 13 \times 384 = 64896$ elements that provides a redundant representation of the input image which is useful to better discriminate between classes of objects. Considering that in the robotic exploration it is important to limit the real time computational load we decide to reduce the dimensionality of

conv3 output by applying the Gaussian Random Projection method [35] obtaining a reduced vector of length 2048. This reduction does not significantly deteriorate matching performance, since Gaussian Random Projection provides a good approximation of radial metrics that are typically used to measure the similarity between vectors (as the Euclidean distance or the cosine similarity). The choice of the size for the reduced dimension of the conv3 output has been made considering both the results of the study in [6] and additional parametric studies that were carried out on the Gardens Point day-left and day-right dataset [18].

The Edge Boxes patches, described by the reduced AlexNet conv3 output p_q and their width w_q and height h_q (i.e., by triples $< p_q, w_q, h_q >$), constitute the semi-semantic landmarks that are used as basic components of the graph-based representation of the environment.

3.2. Graph Nodes and Edges

The characterization of a graph requires the definition of its nodes and edges. Inspired by the work of Stumm *et al.* [13], we build a covisibility graph that models the environment as a structured collection of visual landmarks, acquired sequentially during the environment exploration.

In particular, the nodes of our graph are built on the basis of the semi-semantic landmarks (described in Section 3.1) using the procedure described in details in Section 3.3.

Covisibility information is modelled by connecting the nodes belonging to the same image by an unweighted edge, *i.e.*, nodes observed from the same point of view are connected. Landmarks in the same image are therefore fully connected, forming a complete subgraph for that image.

This node connection policy encodes proximity relations among patches (and their enclosed objects), but it is not strictly related to any metric distance information, that is objects that are metrically distant may be connected in the covisibility graph and metrically close objects (because of visual occlusions) can be not connected. Hence, our method does not rely on the metric position of the patches but uses only visual information.

3.3. Mapping Landmarks into Nodes

In the previous section we described how we build the covisibility subgraph of a new acquired image during the exploration.

Now, in order to incrementally build the graph of the whole environment, we need to specify how to connect each new subgraph to the current graph. This is carried out by mapping the landmarks extracted from a new image in nodes of the graph. For the first image (*i.e.*, at the beginning of the exploration), a node is created for each of the extracted landmarks. For the following images, new nodes are added only for new landmarks, while the landmarks having small distance from existing nodes are considered as "already seen landmarks" and are therefore mapped in the best matching existing node. An illustration of the graph building process is shown in Figure 3 while in Figure 4 we report an example of nodes that are generated by our algorithm on the Gardens Point day-left and day-right dataset [18] and the visual patches contained in these nodes.

In this study the similarity between landmarks is measured using the scalar cosine distance d_{ij} between the feature vector $p_{q,i}$ of the i-th landmark in the current image and the one it is most similar to, $p_{c,j}$ taken among all the landmarks in the previous images.

To speed up the search for the most similar landmark, we exploit the KD-Tree algorithm proposed in [36]. This algorithm works only with distance metrics that are component-wise additive and monotonically increasing with components addition, as in the case of the Euclidean distance. Cosine similarity is more suitable than Euclidean distance for high dimensional data, but does not exhibit the characteristics requested by the KD-Tree algorithm. This technical problem is overcome by first calculating the Euclidean distance between l_2-normalized feature vectors and then applying the following transformation:

$$d_{ij} = 1 - \frac{d_{Euclidean,ij}}{2} \tag{1}$$

where $d_{Euclidean,ij}$ is the Euclidean distance and d_{ij} is the scalar cosine distance between the landmarks $p_{q,i}$ in the current image and $p_{c,j}$ in the previous images.

For each most similar pair of landmarks we also calculate the "dissimilarity" measure of the geometric shape of their bounding boxes s_{ij}. The definition of s_{ij} is taken from [6]:

$$s_{ij} = exp\left\{\frac{1}{2}\left(\frac{|w_{q,i} - w_{c,j}|}{max\{w_{q,i}, w_{c,j}\}} + \frac{|h_{q,i} - h_{c,j}|}{max\{h_{q,i}, h_{c,j}\}}\right)\right\} \quad (2)$$

Values of s_{ij} that are close to 1 indicate that bounding boxes are similar, while larger values indicates differences in their area and shape.

The overall similarity between landmarks in the current image and the most similar landmarks in the previous images is then computed as:

$$P_{ij} = 1 - d_{ij} \cdot s_{ij} \quad (3)$$

Values of P_{ij} that are close to 1 indicate that the two considered landmarks have both very similar shape and conv3 feature descriptor, while small values indicate a difference that can be due to both shape and conv3 features; negative values indicate a relevant difference in the shape of the patches. Using the shape dissimilarity coefficient s_{ij} as a multiplicative factor enhances the cosine distance d_{ij} between the conv3 features. This allows the information on the shape of patches, that is lost (as explained in Section 3.1) because of the resizing of the patches that is requested to use the AlexNet CNN, to be taken into account.

Finally, landmarks are considered to be "the same landmark" (and therefore mapped in the same node of the graph) when the overall similarity P_{ij} is larger than a user defined threshold. The higher this threshold is, the more similar are the landmarks contained in the same node. However, the algorithm becomes slower because of the fast growth of the whole covisibility graph, while the overall recognition performances are not significantly improved.

It is important to note that a new image produces at most as many new nodes as the maximum number of patches extracted by the Edge Boxes algorithm (50 in this study) since very similar (overlapping) patches are mapped in a unique node.

The analysis of Figure 4 highlights some important characteristics of the nodes that are built with the above procedure. Specifically, different nodes can include scaled versions of the same landmarks (*e.g.*, nodes A, B and C); the same node can include some

outlier patches (*e.g.*, nodes F and J) because of the resizing needed to feed AlexNet; in the same node there can be clusters of patches, similar to each other, since we associate new landmarks to a node computing the similarity with the whole set of landmarks associated to that node and not simply with a "centroid" landmark for that node (*e.g.*, node G).

3.4. Graph Representation

In practice, the computed covisibility graph is encoded and managed using a sparse clique matrix, M_{clique}, whose rows represent nodes and whose columns represent image indices, so that a 1 in $M_{clique}[p, f]$ means that the node p is present in the image f.

The graph growth due to the allocation of a new node is implemented by the following matrix update:

$$M_{clique}\big|_{k-1} = \begin{pmatrix} \dots & 1 \\ \vdots & 1 \\ \vdots & 1 \\ \dots & 1 \end{pmatrix} \rightarrow M_{clique}\big|_{k} = \begin{pmatrix} \dots & 1 & 1 \\ \vdots & 1 & 1 \\ \vdots & 1 & 1 \\ \vdots & 1 & 0 \\ \dots & 0 & 1 \end{pmatrix} \tag{4}$$

where in (4) a new column is added for the current image, which has 1s in the existing rows corresponding to already observed landmarks. In addition, when a landmark is assumed to be new, then a new row is allocated, having a 1 in the column associated to the last image, where the landmark was observed (allocated) the first time.

The representation via a sparse matrix also provides an efficient indexing for the image dataset. In fact, considering the definition of the M_{clique} matrix, we know that the rows that are associated with a specific landmark contain ones in positions corresponding to the indices of images where that landmark has been observed, and, conversely, for each image we can know which landmarks belong to that image. This information can be obtained in constant time.

It is instructive to look at the 2D geometry of the clique matrix. For this purpose we generate the clique matrix from the City Centre benchmark dataset [3], that is char-

acterized by a trajectory that is traversed twice. In this representation, zeros are white dots, while ones are black dots. The corresponding clique matrix (shown in Figure 5) presents a repeating nodes pattern in the image indices corresponding to images collected during the two traversals of the same path. This indicates a loop, since the algorithm recognizes many landmarks allocated during the first traversal, along with a few new nodes that are specific of the second traversal.

It is also observed that, due to the presence of already acquired landmarks, the M_{clique} matrix has a growth rate slower than 50 new nodes per image: for example, in the City Centre Dataset, which contains 1237 images, our algorithm creates 8326 nodes instead of $50 \times 1237 = 416300$ nodes. It is expected that the continuous exploration of the same environment will tend to decrease the allocation rate of new nodes over time. This aspect is very important for robotic applications because, for a space constrained environment, we expect a sort of saturation effect to slow down the graph growing process, thus limiting memory consumption of our system.

4. Place Recognition Algorithm

In this section the proposed place recognition algorithm whose block diagram is shown in Figure 6 is described. The purpose of this algorithm is to find possible matchings between the current image (that in this phase is called "query image") and a subset of the most promising images in the set of images (called "image collection") that has been acquired previously (also named as "candidate images"). In particular, the place recognition algorithm is based on the visual modelling of the environment described in Section 3. The matching score between images is computed taking into account two aspects: the mean similarity of landmarks in the query and candidate images and the similarity between images subgraphs. In addition, in order to facilitate the detection of possible loop closures in critical points along the path, a mechanism that produces artificial "enlarged views" (also named "virtual locations") on the basis of the candidate images is proposed.

4.1. Candidates Retrieval

In this section the first block of the system which exploits the covisibility graph is described. Considering a query image, we select, from the whole image collection only a subset of images to be further analysed for the detection of possible loop closures. In particular we retrieve the images that share at least a minimum number of nodes (this number is a free design parameter) with the query image. The sparseness of the clique matrix allows us to efficiently identify (in constant time) the candidate images that fulfil this retrieval criterion.

In this work, the retrieval criterion is "unselective" and all the images that share at least one node with the query one are retrieved. It should be noted that a more selective criterion could be used, improving the speed and precision of the entire algorithm. In fact, a more selective criterion automatically excludes from the analysis many true negative matching images, so that the retrieved images are only those sharing a large number of landmarks with the query image, thus the loop closure detection system would prove to be more precise. However, a selective criterion also has the potential drawback of inducing a possible recall drop (*i.e.*, the fraction of relevant images that are effectively considered) due to the exclusion of many true positive matchings along with the true negative matchings. This side effect is more relevant with the increase in the minimum number of shared nodes requested by the algorithm. This trend is clearly confirmed in Table 1, which shows the percentage of true positive and true negative matching images that have been excluded from analysis due to the retrieval criterion in the four datasets that are used for the experiments (described in Section 5.1). Note that the New College and City Centre Dataset contain images from different environments (such as gardens, archways, squares, alleys and inner urban areas), *i.e.*, high "intra dataset" diversity. Thus, even a loose retrieval criterion is favourable in terms of a priori excluded true negatives. Conversely, the Malaga parking 6L dataset contains images that are more similar to each other. Thus, the positive effect, in terms of a priori excluded true negatives, of not strict retrieval criteria is less evident. Finally, unlike the other datasets, the IDOL dum_sunny3+dum_cloudy1 dataset was collected in an indoor environment and exhibits high sensitivity to the retrieval criterion. In particular, it is observed that the negative effects of strict criteria in terms of a priori excluded true

Minimum Number of Shared Nodes	New College		City Centre		IDOL dum_sunny3+dum_cloudy1		Malaga parking 6L	
	Excluded TP	*Excluded TN*	*Excluded TP*	*Excluded TN*	*Excluded TP*	*Excluded TN*	*Excluded TP*	*Excluded TN*
1	0.10%	44.50%	0.48%	8.53%	0.00%	0.18%	0.00%	0.14%
5	12.23%	72.50%	13.10%	92.36%	74.92%	93.14%	4.51%	22.55%
10	44.33%	94.37%	43.29%	98.71%	91.67%	99.10%	50.93%	95.53%
20	99.29%	99.96%	96.13%	99.97%	98.14%	99.85%	84.38%	99.95%

Table 1: Percentage of true positive (TP) and true negative (TN) matching images a priori excluded from matching due to the retrieval criterion (Minimum Number of Shared Nodes) in the four tested public datasets. A strict criterion causes the exclusion of many True Negatives, thus augmenting the precision, but it also causes the exclusion of many True Positives, thus reducing appreciably the recall.

positives are visible also for less strict criteria that, conversely, do not severely affect outdoor datasets.

Finally, the choice of a reasonable minimum number of shared nodes is application dependent: for example, for a localization and mapping task, precision is critical and a strict retrieval criterion (*e.g.*, minimum number of shared nodes equal to 10) is advisable.

4.2. Unstructured Similarity Between Images

In this section we analyse the block that computes the similarity between landmarks to establish whether a candidate image from the image collection matches with the current query image. This block is based on the algorithm proposed by Sunderhauf *et al.* [6] and does not consider the covisibility information.

The similarity measure between the query and candidate images is derived as a function of the landmarks' feature vectors and of the shape parameters of their bounding box. The algorithm computes a similarity score, P_{ij} (via equation (3)), between each landmark in the query image and the most similar landmark in the candidate image under investigation. The matching score is then assigned to a candidate image as the mean value of individual scores of its landmarks:

$$\hat{S}_{Q,C_n} = \frac{1}{N_P} \sum_{ij} P_{ij} \tag{5}$$

Note that, since the considered landmarks in this phase are those of the query and a candidate image, the similarity score between a pair of landmarks can be smaller

than the threshold that has been fixed in Section 3.3 to map them in the same node of the graph. This is reasonable because in this phase the similarity between images is computed on the basis of landmarks appearance, without exploiting the covisibility graph information.

4.3. Subgraph Matching

The purpose of the Subgraph Matching block is to exploit the information embedded in the covisibility graph in order to refine the previously computed matching score $\hat{S}_{Q,C_n}$, which is based only on similarity between landmarks (Section 4.2). In particular, we exploit the graph Adjacency matrix to take into account the neighbouring information of the nodes in each image subgraph. The Adjacency matrix is obtained on the basis of the graph clique matrix M_{clique}.

As the exploration proceeds, the covisibility graph grows, thus, except in the initial phase, our system deals with a large clique matrix. In order to manage efficiently the large dimensionality, we implement an ad-hoc procedure (see the pseudo code in: Algorithm 1) that exploits the definition of the Adjacency matrix for its calculation, thus limiting significantly the computational cost needed to obtain it (*i.e.*, $O(N^2)$, that is further reduced to $O(N)$ thanks to the sparsity of the clique matrix).

Note that during the graph construction the nodes maintain their order (that is the order in which they have been allocated during the exploration as explained in Section 3.4), thus the row and column indices of the Adjacency matrix are the same for the query and candidate images subgraphs. This implies that the subgraphs are aligned [37], with the great advantage that they can be directly compared by means of their Adjacency matrices. The similarity between the candidate and the query Adjacency matrices is measured by means of the normalized cross correlation as follows:

$$\gamma_{Q,C_n} = \frac{\sum_{ij} A_{ij}^{Q} \cdot A_{ij}^{C_n}}{\sqrt{\sum_{ij}(A_{ij}^{Q})^2 \cdot \sum_{ij}(A_{ij}^{C_n})^2}} \tag{6}$$

where in (6) A_{ij}^{Q} and $A_{ij}^{C_n}$ are the Adjacency matrix entries relative to landmarks p_i and p_j in the subgraphs of query location Q and candidate location C_n respectively.

Then we maintain only normalized cross-correlation values that are lower than a defined fraction α (set at 0.1 in this study) of the normalized cross-correlation between

Algorithm 1 Obtain Adjacency matrix

Input : M_{clique}

Output : A

$A \leftarrow \mathbf{0}_{N \times N}$

for $x \leftarrow 1$ *to* N **do**

▷ isolate M_{clique} columns having 1 in row index x

$x\_columns \leftarrow M_{clique[x=1,:]}$

▷ set to 0 $x\_columns$ element in row index x

$x\_columns(x) \leftarrow 0$

▷ collect indices of node x's neighbours

$x\_neighbours \leftarrow indexOf(x\_columns = 1)$

$A[x, x\_neighbours] \leftarrow 1$

end for

the query image and the previous one C_{k-1}, which is reasonably the most correlated with the current query image, as:

$$\hat{\gamma}_{Q,C_n} = \begin{cases} \gamma_{Q,C_n} & \text{if } \gamma_{Q,C_n} < \alpha \cdot \gamma_{Q,C_{k-1}} \\ 1 & \text{if } \gamma_{Q,C_n} \geq \alpha \cdot \gamma_{Q,C_{k-1}} \end{cases} \tag{7}$$

Note that α can assume any value between 0 and 1. The choice of setting $\alpha = 0.1$ is guided by the consideration that a small value implies a small cross-correlation between the Adjacency matrices of the query and candidate images. In fact, the obtained $\hat{\gamma}_{Q,C_n}$ value is used to weight the similarity score $\hat{S}_{Q,C_n}$ (5) of each candidate location, thus filtering out matching scores of candidate location whose landmark arrangement is too different from that of the query location.

The resulting matching score between images is thus computed as follows:

$$S_{Q,C_n} = \hat{\gamma}_{Q,C_n} \cdot \hat{S}_{Q,C_n} \tag{8}$$

4.4. Virtual Locations

Each new acquired query image is compared to a subset of images from the Image Collection which have been retrieved as described in Section 4.1. In this block each candidate image is "virtually" expanded using the visual information of neighbouring images.

This can be very useful in situations where viewpoint changes are critical. When a place is revisited it is reasonable to assume that the viewpoint is different, this especially in proximity of $90°$ corners or in stretches traversed with lateral displacement. In such a situation some detected landmarks can have a very different relative position, others can be occluded and some new ones can enter the current view. Thus, the place recognition algorithm can benefit from the generation of virtual locations in order to compensate viewpoint changes.

A possible strategy to build virtual locations is to temporarily add nodes (landmarks) to the current candidate image under investigation. Previous works, such as [12], [13] and [14], obtained virtual locations by "merging" subgraphs of candidate images that share a user-defined number of nodes. In this work, we remove this parameter and propose a strategy based on the spectral properties of the covisibility graph. In particular, the nodes to be added to the current candidate image are selected following an agglomerative clustering approach [38]. The agglomerative clustering algorithms start from a seed subgraph and iteratively include nodes among its neighbours (*i.e.*, nodes that are connected at least to one node that already belongs to the seed). In this respect, the subgraph of the current candidate image is used as seed and its neighbourhood contains nodes belonging to other candidate images. Our node selection criterion is based on the graph connectivity metrics, which is computed exploiting the algebraic graph theory as explained below.

Considering the Adjacency matrix A of the graph, its Degree matrix D can be immediately derived. This is a diagonal matrix with as many rows and columns as the number of nodes and, for an undirected graph (such as our covisibility graph) D contains the number of each node's neighbours in the corresponding diagonal positions,

that is:

$$D_{ij} = \sum_{j=1}^{N} A_{ij} \tag{9}$$

where N is the total number of nodes in the graph.

On the basis of the Adjacency matrix and of the Degree matrix it is possible to compute the Laplacian matrix L of the graph as:

$$L = D - A \tag{10}$$

By construction L is singular, symmetric and positive semidefinite in case of a undirected graph. Eigen-decomposition of the Laplacian matrix induces a clustering of the nodes of the graph (in particular it makes it possible to identify a specific number of groups of nodes depending on the eigenvector we select for clustering purpose).

The N ordered eigenvalues of L are defined as $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_N$. The sum of each row and column of L is zero, thus, by construction, the eigenvalue λ_1 is equal to zero and its associated eigenvector is $\mathbf{u}_1 = \mathbf{1}$, in fact $L\mathbf{u}_1 = \mathbf{0}$.

In this study we exploit the second eigenvector, $\mathbf{u}_2$, associated to eigenvalue λ_2 , since it provides a measure of the graph connectivity as explained in [39], [40]. For instance, Figure 7 shows the components of the eigenvector $\mathbf{u}_2$ mapped on the nodes of a sample covisibilty graph computed on the first 20 images of the City Centre Dataset. It may be observed that the components of $\mathbf{u}_2$ vary smoothly from the smallest ones (in blue) to the largest ones (in red), thus inducing a natural ranking of the nodes of the graph.

Based on the previous considerations, each candidate image can be expanded by adding nodes, one by one, as a function of similarity measure provided by the $\mathbf{u}_2$ component value. This strategy reflects the fact that the node to be added is the most connected to those actually contained in the candidate image subgraph.

After the addition of a node to the candidate seed subgraph, the matching score of the expanded candidate location is recalculated. The expansion process is stopped if the similarity measure between query and candidate images decreases. The process is also stopped if a predefined maximum number of nodes is added to the seed location (we set this limit to 50% of the number of Edge Boxes extracted, which is equals to 25 in this work). The role of this additional stopping criterion is twofold. First, it

limits the time complexity of the virtual location construction procedure and second, it prevents false positive matches. In fact, if the expansion were uncontrolled, a candidate image would likely obtain a high matching score because of the addition of many nodes not belonging to its original subgraph, thus the matching score might prove misleading. The pseudo-code of the virtual location construction process is reported in the Algorithm 2 table.

To have an idea of the positions where the virtual locations are actually generated along the paths, in Figure 8 we report the 2D GPS trajectories for the four test datasets datasets where the red dots represent the GPS coordinates of the candidate images that were used as seeds for virtual locations. It can be observed that virtual locations are created near curves, 90° angles and stretches traversed in opposite directions or in cases of significant lateral displacement. Those points are particularly critical in terms of viewpoint changes, since even small variations in the trajectory (and thus, in the viewpoint) may cause a very different arrangement of the visible landmarks in the acquired scene, thus making the loop closure detection particularly challenging.

4.4.1. Computational Complexity

The computational complexity of the procedure for computing a virtual location is quadratic in the number of nodes of the graph, *i.e.*, $O(N^2)$, in the worst case. In the average case the complexity is linear in the number of nodes *i.e.*, $O(N)$. In fact, the actual number of allocated nodes is much less than the product between the number of stored images and the fixed number of patches extracted in each query image *i.e.*, N_P (see Section 3.4). In addition, the number of candidate images is much less than the number of the database images thanks to the selection carried out by the retrieval criterion (see Section 4.1).

The construction of a virtual location is performed for each one of the retrieved candidate images, which is equal to the number of database images in the worst case. The most time consuming part of the virtual location construction algorithm is mainly due to the eigenvector decomposition procedure used to compute the $\mathbf{u}_2$ vector. This procedure is cubic in the number of nodes in the graph, *i.e.*, $O(N^3)$ but it is performed

only once for all retrieved candidate images.

A possible strategy to limit the computational load is to use odometry information to ”activate” the construction of virtual locations only in particular situations, such as during turns, where they proved to be particularly useful.

5. Experiments and Results

In this section we describe the experimental setup and the public datasets selected for testing. In previous works [6, 14], the superiority of semi-semantic feature based methods over low-level feature based methods has been clearly shown. For this reason, in this study the analysis is carried out with the purpose of highlighting the importance and the role of the different blocks of the overall algorithm based on semi-semantic features, and to perform a deep experimental evaluation of the performance in different operative scenarios.

5.1. Tuning and validation Datasets

The parameters of the proposed algorithm were tuned on the Gardens Point day-left and day-right dataset used, for example, in [6]. To achieve a fair comparison, this dataset was not used for testing. This dataset presents both indoor and outdoor sections, repeating patterns along the path, dynamic objects such as pedestrians, many corners and curves along the trajectory, illumination condition variations such as shadows and sunlight and a typical scenario of viewpoint variation such as lateral displacement.

The main purpose of the tuning phase is the setting of the threshold value defining the minimum similarity score between landmarks in order to map them in a unique node (see Section 3.3). This threshold is set to 0.3 in our implementation. In light of the considerations made in Section 3.3, the selected value for the threshold value was deemed to provide a reasonable trade-off between speed and accuracy.

The performance evaluation was carried out using the following four public datasets.

City Centre Dataset. This dataset [3] consists of left and right view images collected ”roughly” with a spatial frequency of $1.5m$ by a Segway robot along a $2km$ path in a

urban environment. Right and left images are acquired at the same time, thus we concatenated each pair and considered the new "panoramic" images in our experiments. This dataset is characterized by the presence of dynamic objects such as pedestrians and vehicles, mild illumination variation mainly due to shadows and sunlight and mild viewpoint variation due to lateral displacement while traversing the same path.

New College Dataset. This dataset [3] consists of left and right images collected with a spatial frequency of $1.5m$ by a Segway robot along a $1.9km$ path in a university campus. Since independent right and left images are acquired also in this case, we concatenated each pair and considered the new "panoramic" images in our experiments. The trajectory is articulated and presents many loops and straight segments traversed also in opposite directions. Also this dataset contains many dynamic elements, such as pedestrians, and repeated elements, since it was acquired in an area characterized by similar walls, archways and bushes.

Malaga Parking 6L Dataset. This dataset [41] was acquired in a university parking area using an electric car equipped with two Firewire colour cameras. For our experiments we considered the rectified images of the left camera. The sequence of images was subsampled at sampling rate 3, thus retaining a third of the entire number of images in the sequence. The explored area covers about $17920m^2$ and images used here are taken every $0.4s$. The environment of this dataset presents moving vehicles and pedestrians and significant sunlight variations. The trajectory presents many loops, stretches traversed in opposite directions and many intersections, thus viewpoint changes are particularly severe in this dataset.

IDOL dum_sunny3 + dum_cloudy1 Dataset. This dataset [9] was acquired in a research laboratory consisting of five rooms, in different seasons, hours of the day and weather conditions, by a PowerBot robot equipped with a monocular camera whose height above the floor is $36cm$. In order to have significant illumination variation, we concatenated two sequences one taken on a sunny summer day and the other n a cloudy winter day. The two sequences have been concatenated after subsampling them at sampling rate 3, thus retaining a third of the entire number of images in each sequence. The

Dataset	Radius [m]	Min. indices difference
City Centre	10	40
New College	10	40
Malaga Parking 6L	2	135
IDOL dum_sunny3 + dum_cloudy1	1.5	300

Table 2: Radius and minimum difference between indices used for ground truth construction for each test dataset

same trajectory is traversed twice, with mild differences that however produce critical viewpoint changes in an indoor environment.

5.1.1. Ground truth

Although some of the above public datasets provide image matching information, it was decided to recompute the image matching matrix in order to use a consistent criterion for all the considered datasets.

The ground truth was computed on the basis of the GPS coordinates of the images. Namely, we considered two images to be matching if they were acquired within a small distance radius. Further, to avoid "trivial matchings" between consecutive images, a minimum difference between the index value of the matching images was also defined. In fact, it is obvious that the most similar images to the current query image are the ones acquired immediately before, but this similarity should be disregarded in the procedure for loop closure detection.

The IDOL dum_sunny3 + dum_cloudy1 dataset is the only indoor dataset that was used in our experiments. Due to the significant viewpoint variation caused by even small trajectory variations, for this indoor dataset we decided to match images of the first traversal with those of the second traversal. Thus, we imposed a minimum difference between matching images equal to 300, so that only images belonging to different traversals are considered.

Table 2 reports the parameters that were used for the computation of the ground truth for each dataset.

5.2. Plan for the Experiments

In order to evaluate the performance of the different blocks of the proposed algorithm and to compare the overall performance with those of a state-of-the-art method we considered the following scenarios:

- A state-of-the-art technique that is based on the high level features extracted by Edge Boxes and AlexNet conv3, that are also used in our work, but does not use any graph based representation of the environment (named 'HOCE' - Heap Of CNN Extracted features - in this work). This is essentially the approach proposed by Sunderhauf *et al.* in [6]. This algorithm was here re-implemented and used in an incremental fashion to be consistent with our approach.

- Our complete approach (named 'GOCCE' - Graph Of Covisible CNN Extracted features), that exploits the covisibility graph as described in Section 4.

- A simplified version of the approach (named 'GOCCE_{R}') that uses only the covisibility graph for the retrieval of matching candidates, selected in case they share at least 10 nodes with the current query image (in the following this criterion will also be referred to as 'strict retrieval').

- Another simplified version of our approach (named 'GOCCE_{RS}') that uses the covisibility graph for matching candidates retrieval, selected if they share at least a node with the current query image, and for refining the matching score of candidate locations via subgraph comparison.

As for the settings, we used for each scenario the same values for the maximum number of patches extracted in each image and for the minimum similarity score between landmarks in order to be included in the same node.

5.3. Performance analysis

In a localization and mapping application, the loop closure detection module is essential since it allows an autonomous agent to self-relocalize and to adjust the map of the environment. This section reports the results of a detailed study that is mainly focussed toward the evaluation of the loop closure detection performance of the proposed

method. Considering a generic loop closing problem, it is generally more important to avoid wrong matchings along the trajectory, rather than not to miss a matching, *i.e.*, precision is usually a more critical requirement then recall.

To have a synthetic comparison of the performance provided by the considered variants of our method, in Figure 9 the precision-recall curves obtained on the four test datasets are reported, while Table 3 shows the precision and recall values obtained at maximum recall and precision respectively. It is observed that in the case of strict retrieval (*i.e.*, for GOCCE_R) the precision is higher in every dataset (note in particular the performance for the City Centre dataset in Figure 9a and for the Malaga Parking 6L dataset in Figure 9c). This is mainly due to the fact that the strict retrieval criterion excludes a priori many true negative matchings, thus the precision is higher (see Section 4.1). The main drawback of this approach is that 100% recall is never reached. This is because true positive matchings with lower matching scores (that are considered by the other analysed methods) are a priori excluded by GOCCE_R. An important difference between the graph-free approach (*i.e.*, HOCE) and the graph-based approaches with unselective retrieval criterion (*i.e.*, GOCCE_RS and GOCCE) was also observed. In fact, especially on the City Centre Dataset the precision obtained by HOCE at high recall is almost 10% inferior to the precision achieved by GOCCE_RS and GOCCE. This fact confirms clearly the beneficial role of the subgraph matching score (Eq. 7) as additional information to refine the overall matching score between images. Performance obtained by HOCE in the remaining datasets was comparable (just slightly inferior) to that of GOCCE_RS and GOCCE.

To evaluate the performance of the loop closing module it is also important to evaluate the metric error produced by wrong matches. Indeed, in order to build a consistent map of the environment, a wrong loop closure detection can be considered somewhat useful if the metric error is small. In fact, it is reasonable that images having a similar visual content are acquired at close distance each other, thus the localization error produced by their matching can be considered acceptable for a coarse localization. In other words, errors of a few meters can still allow a reliable localization producing a consistent map of the environment. The results of the metric study are reported in Figure 10, which shows the average metric error, *i.e.*, the average Euclidean distance

	City Centre		New College		Malaga parking 6L		IDOL dum_sunny3+dum_cloudy1	
	recall at 100% precision	*precision at max recall*	*recall at 100% precision*	*precision at max recall*	*recall at 100% precision*	*precision at max recall*	*recall at 100% precision*	*precision at max recall*
HOCE	15.64%	45.18%	10.09%	41.40%	08.17%	00.56%	03.44%	68.73%
GOCCE$_{R}$	15.64%	90.89% at 91.74%	10.30%	63.44% at 90.31%	18.27%	15.12% at 62.50%	03.13%	12.09% at 68.52%
GOCCE$_{RS}$	16.18%	45.85%	07.30%	43.21%	08.17%	00.56%	03.44%	68.73%
GOCCE	16.00%	45.68%	10.30%	43.53%	08.17%	00.56%	03.44%	68.73%

Table 3: Precision and recall values at maximum recall and precision respectively comparing the different techniques on the four considered datasets

between coordinates of false positive matching images, as a function of the threshold value applied to the matching score for assessing a loop closure. Analysing Figure 10 it can be observed that a low threshold leads mainly to spatially distant false positive matches, while large threshold values do not produce false positive matches (this implies a high precision). Some differences among the methods were highlighted by this metrical study: the higher precision of the GOCCE$_{R}$ approach is confirmed also in metric terms, while the method of Sunderhauf *et al.* in [6] (HOCE) produces less precise results compared to the methods exploiting the covisibility graph, especially in metric terms.

Finally, to evaluate the role of the virtual locations, we carried out an additional study considering only candidate images that served as seed for the construction of a virtual location (shown in Figure 8). In other words, we considered only those matches between a query image and candidate images that included nodes from other images, *i.e.*, were used as seeds for a virtual location. Precision-recall curves obtained considering only this subset of images are reported in Figure 11. It may be observed that, especially in the Malaga Parking 6L dataset (Figure 11c) our complete approach, GOCCE, obtains good performances thanks to the virtual locations construction function (note the difference with respect to the GOCCE$_{RS}$ approach that does not calculate virtual locations). The Malaga Parking 6L dataset exhibits an articulated trajectory, with many curves, intersections and stretches traversed in opposite directions. In these

critical scenarios a localization and mapping system can benefit from the virtual location construction in terms of loop closure detection performance.

In light of this, we use the Malaga Parking 6L dataset for an additional study to evaluate the benefits of the virtual locations in terms of loop closure detection performance. In particular, we consider the GPS position where virtual locations have been constructed, these are shown with dots in Figure 12. For each one of the four methods under investigation, the threshold on the matching score is set at a value that guarantees at least 85% of precision and 20% of recall. With these settings, the output of the loop closure detection methods is evaluated. In Figure 12, GPS positions of virtual locations are marked with coloured dots along the paths. In particular, black dots are used in the case of a correct output (i.e., true positives or true negatives), green dots in the case of a false negative output and red dots in the case of a false positive output. It can be observed that GOCCE produces the smallest number of false positive matches: 4 FP against 6 FP for GOCCE_{R}, 7 FP for GOCCE_{RS} and 27 FP for HOCE. These results highlight the fact that virtual locations are useful in scenarios that are particularly challenging in terms of viewpoint changes, such as curves and oppositely traversed stretches. Exploiting virtual locations in these cases makes the loop closure detection system more precise.

6. Conclusion

In this work, we proposed an appearance and viewpoint invariant place recognition system. The method relies only on machine vision images and does not need any specific training when operating in new unexplored environments.

These characteristics are achieved by modelling inter object geometric relations in the environment by means of a covisibility graph, whose nodes are high level, semi-semantic landmarks. These landmarks are image patches containing generic objects and are described by means of features extracted by an inner convolutional layer of a pre-trained CNN, that are particularly robust to appearance changes.

We proposed novel specific algorithms that leverage the covisibility graph representation for a fast and robust retrieval of the most likely matching candidate images.

The covisibility graph is also exploited for refining images matching score based on the co-presence of landmark contained in the images. We also proposed a novel strategy for synthesising virtual locations via a parameter-free approach that is based on a local graph clustering method which exploits covisibility graph connectivity information.

Experimental validation carried out on four public datasets has shown that, with regard to precision and recall, our approach provides performance that is comparable (or superior) with respect to a state-of-the-art place recognition technique that does not rely on any graph representation of the environment.

In addition, the construction of virtual locations is useful in specific but critical situations such as turning near 90° corners or traversing a stretch in opposite directions. In these scenarios, virtual locations construction provides an improvement in terms of precision of the loop closure detection system.

Considering metric error (i.e., the metric distance between mismatched images co-ordinates), our graph-based technique outperformed a state-of-the-art graph-free approach that was considered as benchmark.

A possible extension of this work would be the implementation of a strategy that compares sequences of images, rather than single images. This directly translates in the comparison of bigger subgraphs.

[1] J. Engel, T. Schöps, D. Cremers, LSD-SLAM: Large-Scale Direct Monocular SLAM, in: European Conference on Computer Vision (ECCV), 2014, pp. 834–849.

[2] C. Forster, M. Pizzoli, D. Scaramuzza, SVO: Fast Semi-Direct Monocular Visual Odometry, in: IEEE International Conference on Robotics and Automation (ICRA), 2014.

[3] M. Cummins, P. Newman, Fab-map: Probabilistic localization and mapping in the space of appearance, The International Journal of Robotics Research 27 (6) (2008) 647–665.

[4] M. J. Milford, G. F. Wyeth, Seqslam: Visual route-based navigation for sunny

summer days and stormy winter nights, in: Robotics and Automation (ICRA), 2012 IEEE International Conference on, IEEE, 2012, pp. 1643–1649.

[5] T. A. Ciarfuglia, G. Costante, P. Valigi, E. Ricci, A Discriminative Approach for Appearance Based Loop Closing, in: IEEE International Conference on Intelligent Robots and Systems (IROS), 2012.

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

[7] R. Finman, L. Paull, J. J. Leonard, Toward object-based place recognition in dense rgb-d maps, in: ICRA Workshop Visual Place Recognition in Changing Environments, Seattle, WA, 2015.

[8] J. Oh, J. Jeon, B. Lee, Place recognition for visual loop-closures using similarities of object graphs, Electronics Letters 51 (1) (2014) 44–46.

[9] J. Luo, A. Pronobis, B. Caputo, P. Jensfelt, Incremental learning for place recognition in dynamic environments, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, 2007, pp. 721–728.

[10] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems (NIPS), 2012.

[11] C. Mei, G. Sibley, P. Newman, Closing loops without places, in: Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on, IEEE, 2010, pp. 3738–3744.

[12] E. S. Stumm, C. Mei, S. Lacroix, Building location models for visual place recognition, The International Journal of Robotics Research 35 (4) (2016) 334–356.

[13] E. Stumm, C. Mei, S. Lacroix, M. Chli, Location graphs for visual place recognition, in: Robotics and Automation (ICRA), 2015 IEEE International Conference on, IEEE, 2015, pp. 5475–5480.

[14] S. Cascianelli, G. Costante, E. Bellocchio, P. Valigi, M. L. Fravolini, T. A. Ciarfuglia, A robust semi-semantic approach for visual localization in urban environment, in: Smart Cities Conference (ISC2), 2016 IEEE International, IEEE, 2016, pp. 1–6.

[15] C. McManus, W. Churchill, W. Maddern, A. D. Stewart, P. Newman, Shady dealings: Robust, long-term visual localisation using illumination invariance, in: Robotics and Automation (ICRA), 2014 IEEE International Conference on, IEEE, 2014, pp. 901–906.

[16] P. Neubert, N. Sünderhauf, P. Protzel, Superpixel-based appearance change prediction for long-term navigation across seasons, Robotics and Autonomous Systems 69 (2015) 15–27.

[17] N. Carlevaris-Bianco, R. M. Eustice, Learning visual feature descriptors for dynamic lighting conditions, in: Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on, IEEE, 2014, pp. 2769–2776.

[18] N. Sunderhauf, S. Shirazi, F. Dayoub, B. Upcroft, M. Milford, On the performance of convnet features for place recognition, in: Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on, IEEE, 2015, pp. 4297–4304.

[19] W. Maddern, A. Stewart, C. McManus, B. Upcroft, W. Churchill, P. Newman, Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles, in: Proceedings of the Visual Place Recognition in Changing Environments Workshop, IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 2014.

[20] Z. Chen, O. Lam, A. Jacobson, M. Milford, Convolutional neural network-based place recognition, arXiv preprint arXiv:1411.1509.

[21] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (surf), Computer vision and image understanding 110 (3) (2008) 346–359.

[22] R. Mur-Artal, J. Montiel, J. D. Tardós, Orb-slam: a versatile and accurate monocular slam system, IEEE Transactions on Robotics 31 (5) (2015) 1147–1163.

[23] E. Rublee, V. Rabaud, K. Konolige, G. Bradski, Orb: An efficient alternative to sift or surf, in: 2011 International conference on computer vision, IEEE, 2011, pp. 2564–2571.

[24] H. Yang, S. Cai, J. Wang, L. Quan, Low-rank sift: an affine invariant feature for place recognition, in: Image Processing (ICIP), 2014 IEEE International Conference on, IEEE, 2014, pp. 5731–5735.

[25] M. Milford, C. Shen, S. Lowry, N. Suenderhauf, S. Shirazi, G. Lin, F. Liu, E. Pepperell, C. Lerma, B. Upcroft, et al., Sequence searching with deep-learnt depth for condition-and viewpoint-invariant route-based place recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015, pp. 18–25.

[26] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, J. J. Yebes, S. Gámez, Bidirectional loop closure detection on panoramas for visual navigation, in: Intelligent Vehicles Symposium Proceedings, 2014 IEEE, IEEE, 2014, pp. 1378–1383.

[27] A. L. Majdik, D. Verda, Y. Albers-Schoenberg, D. Scaramuzza, Air-ground matching: Appearance-based gps-denied urban localization of micro aerial vehicles, Journal of Field Robotics 32 (7) (2015) 1015–1039.

[28] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, J. Sivic, Netvlad: Cnn architecture for weakly supervised place recognition, arXiv preprint arXiv:1511.07247.

[29] D. Mishkin, M. Perdoch, J. Matas, Place recognition with wxbs retrieval, in: CVPR 2015 Workshop on Visual Place Recognition in Changing Environments, 2015.

[30] E. Pepperell, P. Corke, M. Milford, Routed roads: Probabilistic vision-based place recognition for changing conditions, split streets and varied viewpoints, The International Journal of Robotics Research.

[31] G. Costante, T. A. Ciarfuglia, P. Valigi, E. Ricci, A transfer learning approach for multi-cue semantic place recognition, in: Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on, IEEE, 2013, pp. 2122–2129.

[32] C. L. Zitnick, P. Dollár, Edge boxes: Locating object proposals from edges, in: Computer Vision–ECCV 2014, Springer, 2014, pp. 391–405.

[33] A. Sharif Razavian, H. Azizpour, J. Sullivan, S. Carlsson, Cnn features off-the-shelf: an astounding baseline for recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 806–813.

[34] J. L. Long, N. Zhang, T. Darrell, Do convnets learn correspondence?, in: Advances in Neural Information Processing Systems, 2014, pp. 1601–1609.

[35] E. J. Candes, T. Tao, Near-optimal signal recovery from random projections: Universal encoding strategies?, Information Theory, IEEE Transactions on 52 (12) (2006) 5406–5425.

[36] M. Muja, D. G. Lowe, Fast approximate nearest neighbors with automatic algorithm configuration, in: International Conference on Computer Vision Theory and Application VISSAPP'09), INSTICC Press, 2009, pp. 331–340.

[37] S. Feizi, G. Quon, M. Medard, M. Kellis, A. Jadbabaie, Spectral alignment of networks.

[38] S. E. Schaeffer, Graph clustering, Computer science review 1 (1) (2007) 27–64.

[39] C. Godsil, G. F. Royle, Algebraic graph theory, Vol. 207, Springer Science & Business Media, 2013.

[40] M. Newman, Networks: an introduction, Oxford university press, 2010.

[41] J.-L. Blanco, F.-A. Moreno, J. González, A collection of outdoor robotic datasets with centimeter-accuracy ground truth, Autonomous Robots 27 (4) (2009) 327–351. doi:10.1007/s10514-009-9138-7.
URL http://www.mrpt.org/Paper:Malaga_Dataset_2009

Silvia Cascianelli received the B.Sc. degree in Electronic and Information Engineering in 2013, from University of Perugia, with a thesis on System Fault Detection and Accomodation for UAV's anemometers. Since then she collaborates with the Intelligent Systems, Automation and Robotics Laboratory (ISARLab). In 2015 she received the M.Sc. *magna cum laude* degree in Information and Automation Engineering with a thesis on Nuclear Image based Computer Aided Diagnosis systems for Alzheimer's Disease from the University of Perugia. She then joined the ISARLab in 2015 as a Ph.D. student. Her research interests are mainly machine learning and computer vision applied to robotics.

Gabriele Costante received the B.Sc. *magna cum laude* degree in Electronic and Information Engineering and the M.Sc. *magna cum laude* degree in Information and Automation Engineering from the University of Perugia respectively in 2010 and 2012. He then joined the Service and Industrial Robotics and Automation Laboratory (SIRALab) in 2012 and in 2016 he received the Ph.D. degree in Robotics from the University of Perugia. His research interests are mainly robotics, computer vision and machine learning.

Enrico Bellocchio received his B.Sc. degree in Electronic and Information Engineering in 2012 from University of Perugia. In 2014 received M.Sc. degree in Information and Automation Engineering From University of Perugia. His Master thesis work was about the implementation of an Human-Robot Interface using a Transfer Learning algorithm on a UAV platform. He joined the Intelligent Systems Automation and Robotics Laboratory (ISARLab) in 2014 as a researcher engineer. His current research activity is about machine learning, computer vision and robotics.

Paolo Valigi received the Laurea degree in 1986 from University of Rome La Sapienza and the Ph.D. degree from University of Rome Tor Vergata in 1991. From 1990 to 1994 he worked with the Fondazione Ugo Bordoni. From 1998 to 2004 he was associate professor at the University of Perugia, Department of Electronics and Informatics Engineering, where since 2004 he has been full professor of System Theory and Optimization and Control. His research interests are in the field of robotics and systems biology. He has authored or co-authored more than 130 journal and conference papers and book chapters.

Mario Luca Fravolini received his PhD degree in Electronic Engineering from the University of Perugia in 2000. Currently He is associate Professor at the Department of Engineering, University of Perugia. In 1999 He was with the Control Group at the School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, USA. He has been a visiting Research Assistant Professor at the Department of Mechanical and Aerospace Engineering West Virginia University, USA for several years. His research interests include: Fault Diagnosis, Intelligent and Adaptive Control, Predictive Control, Optical Feedback, Biomedical Imaging, Modeling and Control Biomedical Systems and Active Control of Structures.

Thomas Alessandro Ciarfuglia received the M.Sc. *magna cum laude* degree in Electronics Engineering from the University of Perugia in 2004. He worked as HW/FW/SW designer engineer for various companies from 2004 to 2006. He then obtained an M.Sc. in Mechatronics and a Ph.D. degree in Robotics from the University of Perugia in 2008 and 2011 respectively. He joined the Service and Industrial Robotics and Automation Laboratory (SIRALab) in 2008 and he is currently working as a Post-Doc there. His research interests are machine learning and computer vision applied to robotics.

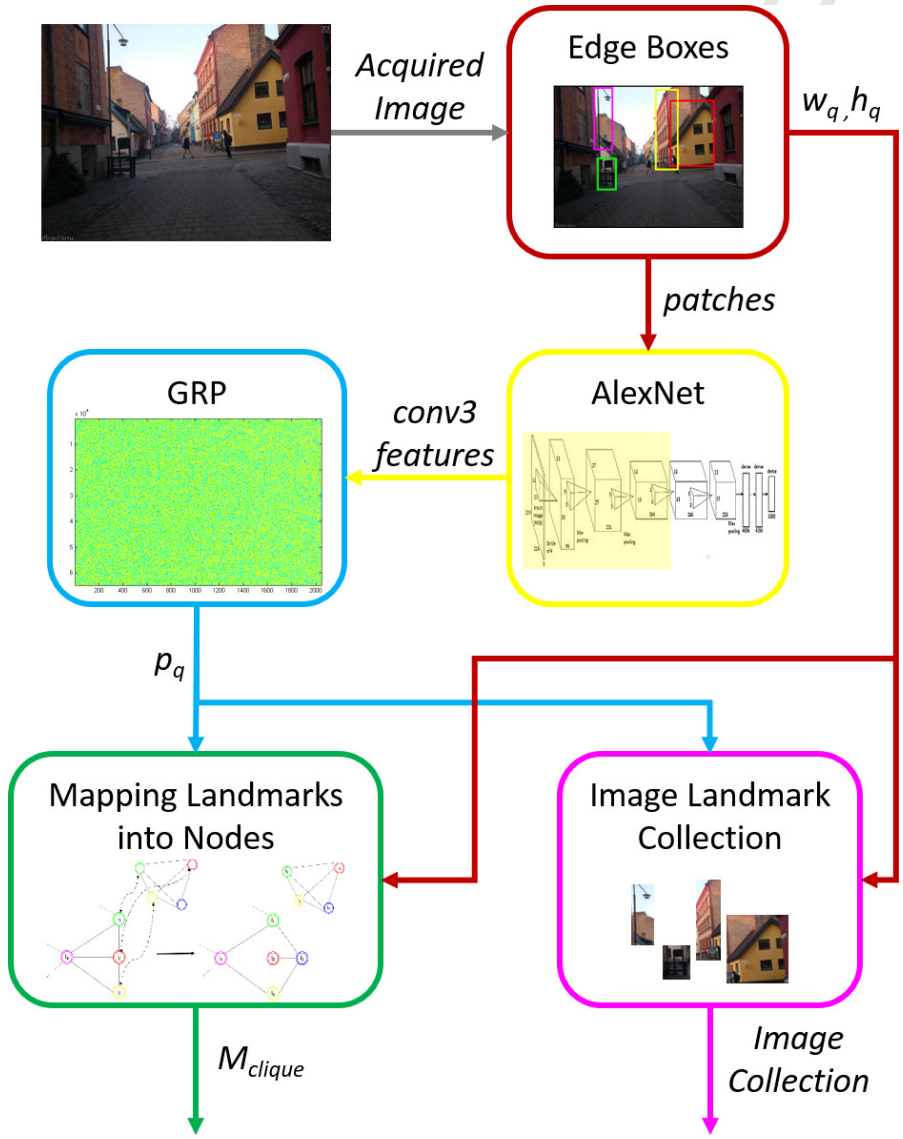


Figure 2: Schematic representation of the visual information processing blocks used during the exploration. When a new image is acquired, the Edge Boxes algorithm (dark red block) extracts a pre-defined number of image patches. These are fed to AlexNet (yellow block), from which the output of conv3 layer is retained. The dimensionality of this output vector is reduced via Gaussian Random Projection (cyan block). Information about each patch enriches the incremental database of images (magenta block) and extends the covisibility graph by mapping landmarks in existing or new nodes (green block).

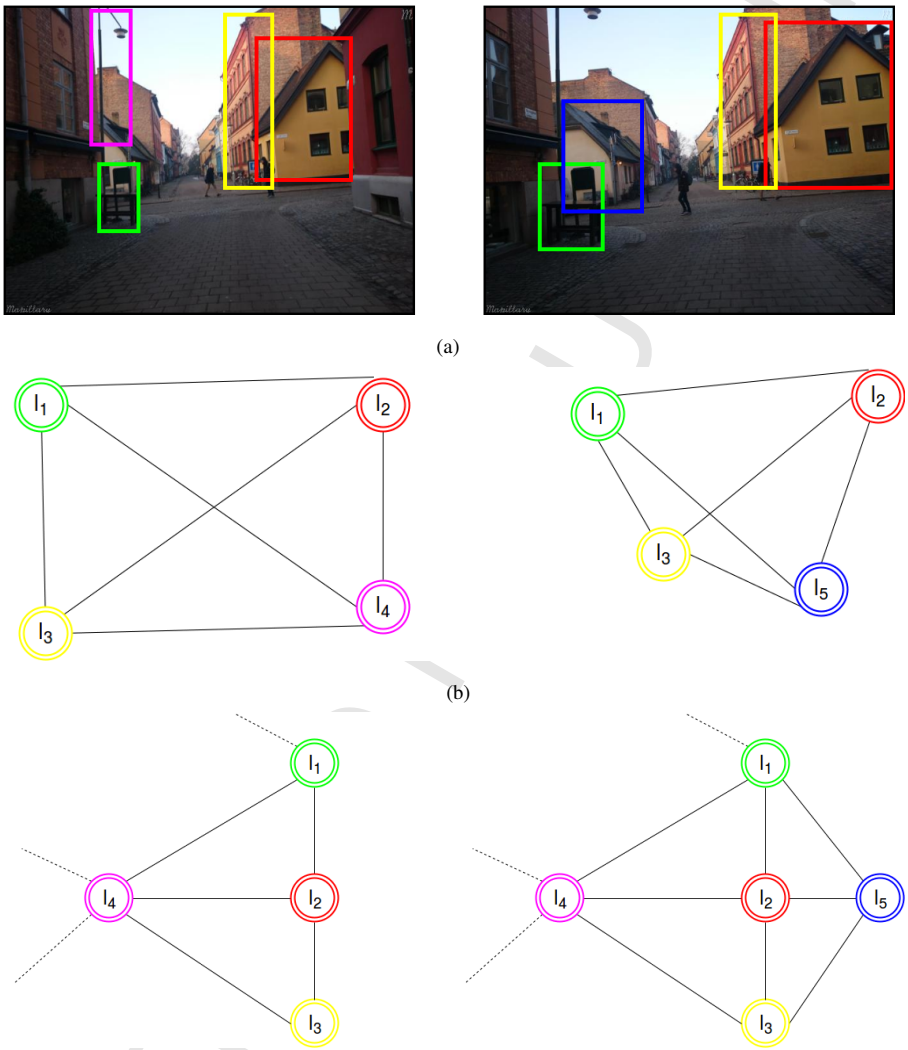


Figure 3: Incremental covisibility graph construction during the environment exploration. Examples of Edge Boxes landmarks extracted from images at time $k-1$ (left) and at time k (right) respectively are shown in (3a). Relative landmark covisibility subgraphs of images visible at time $k-1$ (left) and at time k (right) respectively are shown in (3b): landmarks acquired in the same image are connected in a dense graph. Landmark covisibility whole graph at time $k-1$ (left) and at time k (right) respectively are shown in (3c): similar landmarks are mapped in the same node, while different landmarks produces new nodes.

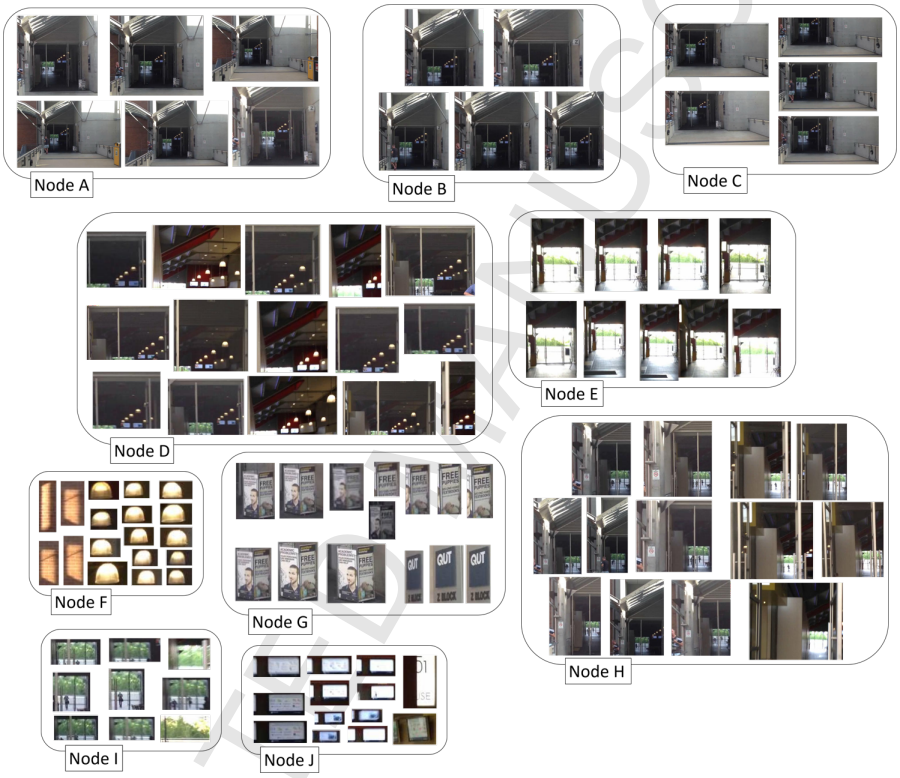


Figure 4: Landmarks belonging to some sample nodes: different nodes can contain scaled versions of the same landmark (*e.g.*, nodes A, B and C), the same node can contain a small number of different outlier patches (*e.g.*, nodes F and J) and in the same node there can also be clusters of patches, smoothly similar each other.

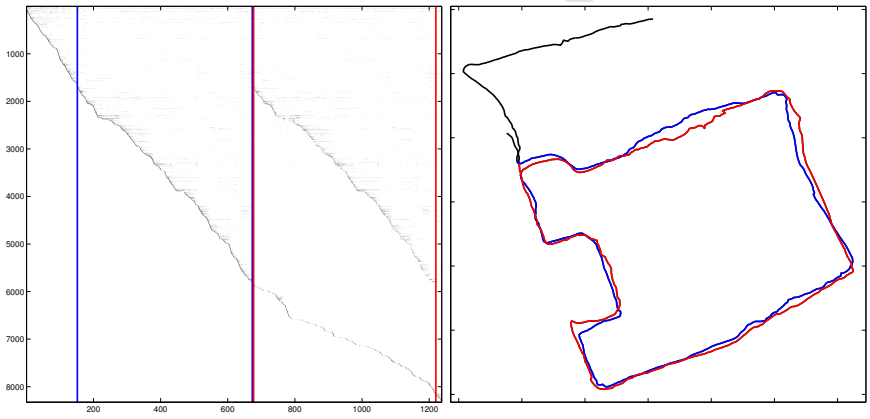


Figure 5: Trajectory and Clique matrix M_{clique} relative to the City Centre Dataset. This dataset presents a circular trajectory traversed two times starting form image 152 to image 674 and from image 675 to image 1220 and its clique matrix presents a repeating nodes pattern in the corresponding image indices, along with few new nodes that are specific of the second traversal. The allocation rate of new nodes is inferior in the second traversal with respect to the first traversal because the robot sees many landmarks belonging to already allocated nodes and only a small number of new nodes is allocated.

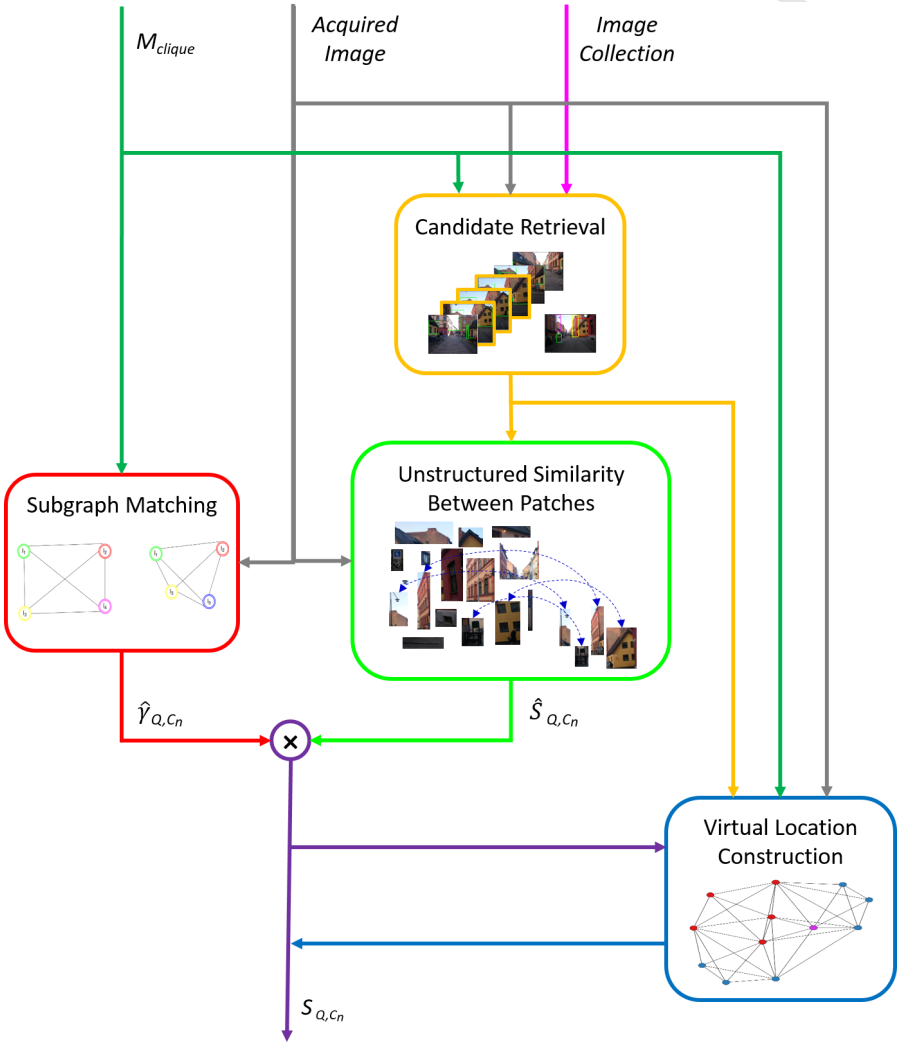


Figure 6: Schematic representation of the proposed Place Recognition system. The covisibility graph is exploited to retrieve the most relevant candidate images (orange block). For each one of the retrieved images, it is calculated the landmarks similarity score (light green block) and the subgraph matching score (red block). Those values are multiplied and used as baseline score in the process of virtual location construction (blue block). Using this latter block the final similarity score for each candidate image is assessed.

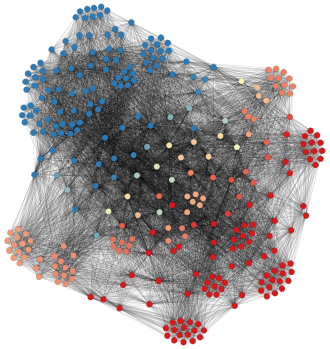


Figure 7: Eigenvector $\mathbf{u}_2$ components associated to each node a subgraph of the City Centre Dataset, made of the first 20 images: note the inducted partition in two subsets.

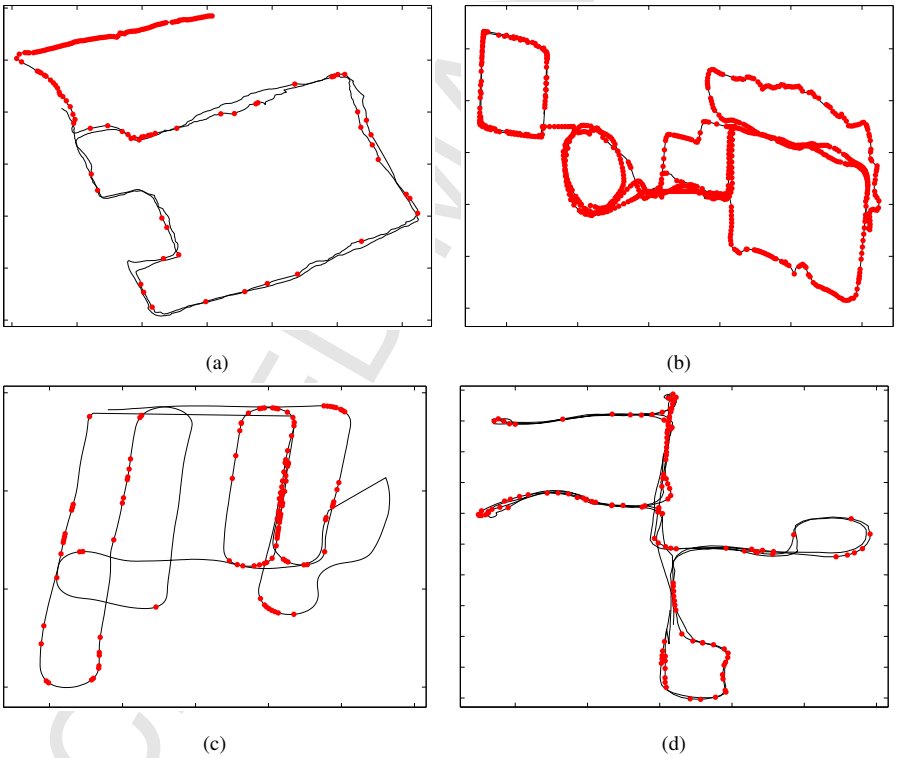


Figure 8: GPS positions of candidate images (red dots) that are used as seed for the construction of a virtual location on the four tested datasets, namely the City Centre dataset (8a), the New College dataset (8b), the Malaga Parking 6L dataset (8c) and the IDOL dum_sunny3 + dum_cloudy1 dataset (8d). Virtual locations are created near curves, 90° angles and stretches traversed in opposite directions or in cases of a severe lateral displacement.

Algorithm 2 Obtain Virtual Location

Input : $\mathbf{u}_2$, M_{clique}, S_{Q,C_n}

Output : S^*_{Q,C_n}, $M^{C_n^+}_{clique}$ ▷ $C_n^+ \doteq$ expanded candidate

$S^*_{Q,C_n} \leftarrow S_{Q,C_n}$

$M^{C_n^+}_{clique} \leftarrow M_{clique}[:, C_n]$

$M^{\hat{C}_n^+}_{clique} \leftarrow M_{clique}[:, C_n]$

▷ $\hat{C}_n^+ \doteq$ temporary expanded candidate

$added \leftarrow 0$

while $added < \dfrac{N_P}{2}$ **do** ▷ $N_P = 50$ in this study

▷ collect indices of nodes in seed subgraph

$seed \leftarrow indexOf(M^{C_n^+}_{clique} = 1)$

▷ collect indices of nodes not in seed subgraph

$\mathcal{N}(seed) \leftarrow indexOf(M^{C_n^+}_{clique} = 0)$

▷ $\mathcal{N}(seed) \doteq$ seed neighbourhood

▷ find the index of the best node to add to seed subgraph

$$best\_neighbour \leftarrow \underset{j \in \mathcal{N}(seed)}{\operatorname{argmin}} \left\{ \sum_{i \in seed} (\mathbf{u}_2[i] - \mathbf{u}_2[j])^2 \right\}$$

▷ add $best\_neighbour$ to the current seed subgraph

$M^{\hat{C}_n^+}_{clique}[best\_neighbour] \leftarrow 1$

calculate $S_{Q,\hat{C}_n^+}$

if $S^*_{Q,C_n} \geq S_{Q,\hat{C}_n^+}$ **then**

break

else

$M^{C_n^+}_{clique} \leftarrow M^{\hat{C}_n^+}_{clique}$

$S^*_{Q,C_n} \leftarrow S_{Q,\hat{C}_n^+}$

$added + 1$

end if

end while

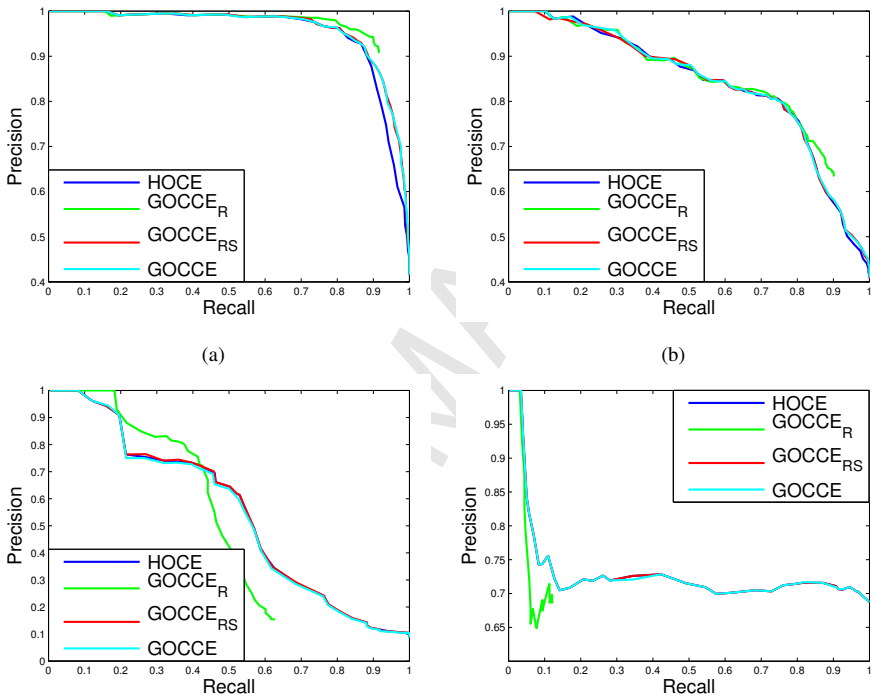


Figure 9: Precision-recall curves comparing the different techniques with our novel approach on the four test datasets, namely. the City Centre dataset 9a, the New College dataset 9b, the Malaga Parking 6L dataset 9c and the IDOL dum_sunny3 + dum_cloudy1 dataset 9d.

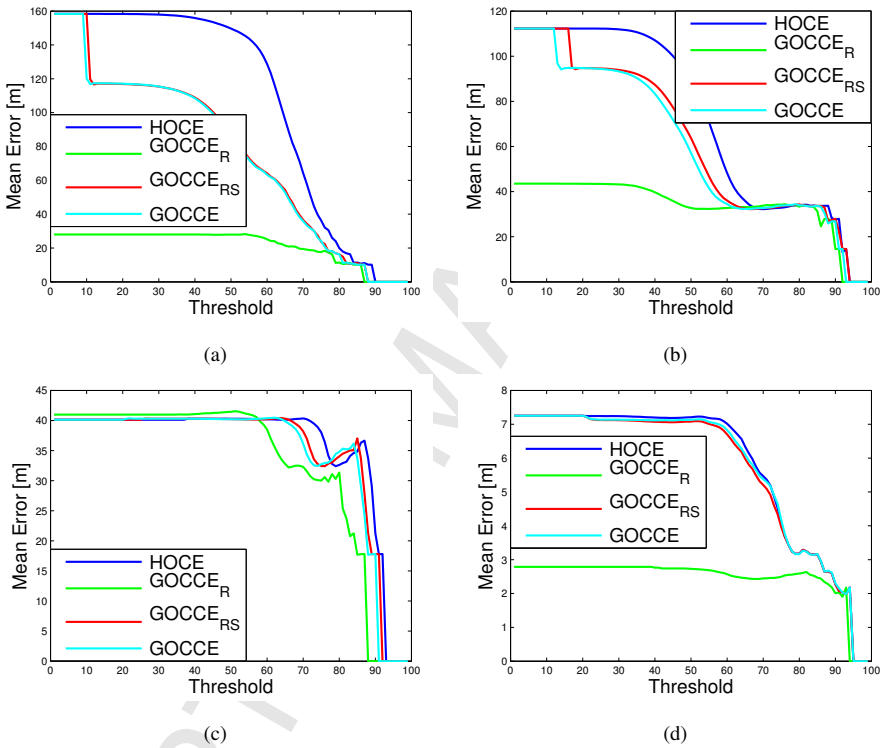


Figure 10: Average metric error curves, relative to false positive matching errors comparing the different techniques on the four considered datasets, namely the City Centre dataset (10a), the New College dataset (10b), the Malaga Parking 6L dataset (10c) and the IDOL dum_sunny3 + dum_cloudy1 dataset (10d).

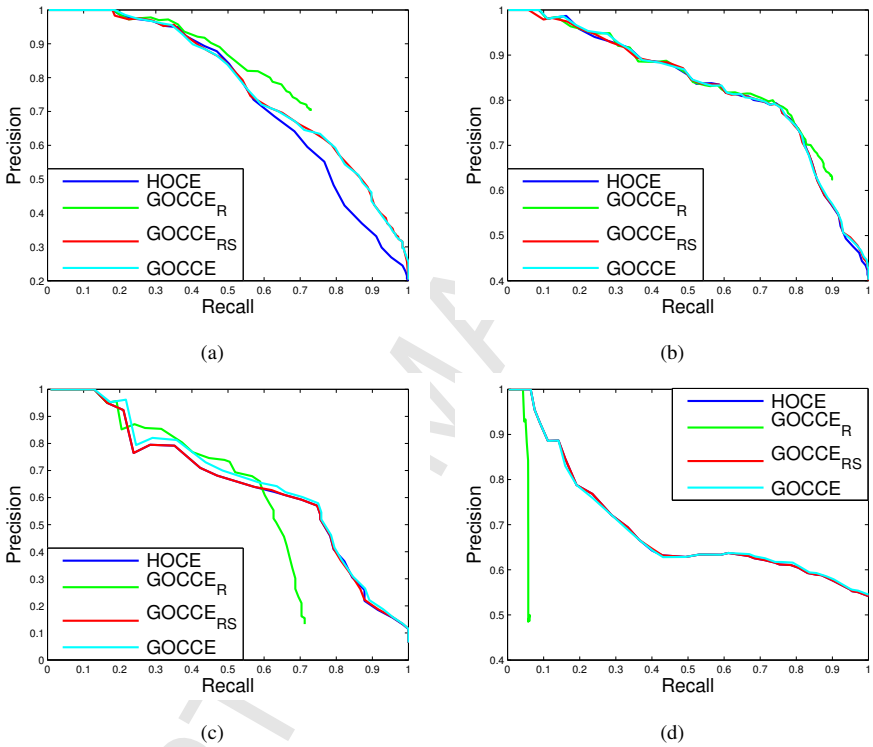


Figure 11: Precision-recall curves comparing the different techniques with respect to our novel approach considering only candidate images that were used as seed for the construction of a virtual location on the four tested datasets, namely the City Centre dataset 11a, the New College dataset 11b, the Malaga Parking 6L dataset 11c and the IDOL dum_sunny3 + dum_cloudy1 dataset 11d.

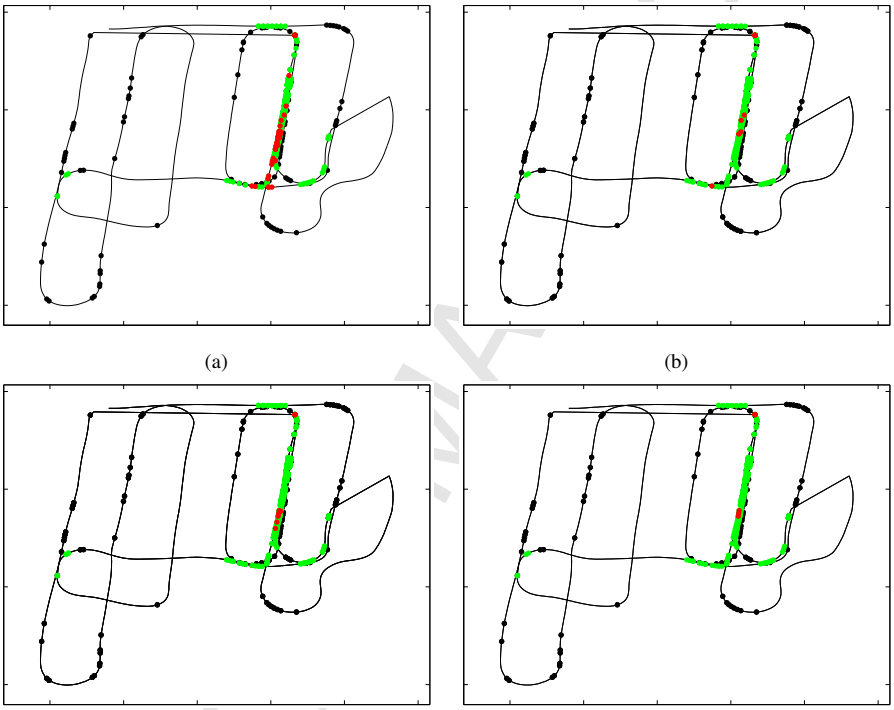


Figure 12: GPS position of candidate images that produced a loop closure detection error on the Malaga Parking 6L dataset for the four considered methods: HOCE (12a), GOCCE$_{R}$ (12b), GOCCE$_{RS}$ (12c) and GOCCE (12d). Black dots represent GPS positions of correctly matched images, green dots are GPS positions of false negative matching images and red dots are GPS positions of false positive matching images. Note that GOCCE has the smallest number of false positive matches.

Highlights

- A training-free appearance and viewpoint robust Place Recognition system is proposed
- The method uses CNN features and preserves scene structure via a covisibility graph
- A novel approach for synthesising virtual views of the environment is proposed
- Virtual views are particularly useful to face critical situations of viewpoint change