To appear in *Advanced Robotics*
Vol. 31, No. 5, March 2017, 1–18

FULL PAPER

LexToMap: Lexical-based Topological Mapping

José Carlos Rangel[a,b], Jesus Martínez-Gómez[c], Ismael García-Varea[c] and Miguel Cazorla[a,*]

[a] *Computer Science Research Institute. University of Alicante*
P.O. Box 99. E-03080. Alicante. Spain;
[b] *Universidad Tecnológica de Panamá*
Santiago, Veraguas. Panamá;
[c] *University of Castilla-La Mancha*
Albacete. Spain

(*v1.0 released June 2015*)

Any robot should be provided with a proper representation of its environment in order to perform navigation and other tasks. In addition to metrical approaches, topological mapping generates graph representations in which nodes and edges correspond to locations and transitions. In this article, we present LexToMap, a topological mapping procedure that relies on image annotations. These annotations, represented in this work by lexical labels, are obtained from pre-trained deep learning models, namely CNNs, and are used to estimate image similarities. Moreover, the lexical labels contribute to the descriptive capabilities of the topological maps. The proposal has been evaluated using the KTH-IDOL 2 dataset, which consists of image sequences acquired within an indoor environment under three different lighting conditions. The generality of the procedure as well as the descriptive capabilities of the generated maps validate the proposal.

Keywords: Topological Mapping, Deep Learning, Localization, Image Annotations, Lexical Labels

1. Introduction

Building an appropriate representation of the environment in which an autonomous robot operates is still a widely addressed problem in the robotics research community. This problem is usually known as map building or mapping since maps are considered the most common and appropriate environment representation [1]. A map is useful for robot localization, navigation [2] and path-planning tasks [3], but also for a better understanding of the robot's surroundings [4]. That is, a map may not be limited to metric (e.g. specific poses of objects/obstacles) and topological information (e.g. paths from one place to others), but it can also integrate semantic information (e.g. symbolic representations of objects, expected behaviors for specific locations, or even situated dialogues, to name a few) corresponding to the objects, agents, and places represented on it. Three different type of maps are graphically presented in Fig. 1, where can be observed the bridge between metric and semantic representations.

Topological mapping consists in generating a graph-based representation of the environment, where nodes represent locations and arcs transitions between adjacent locations [5]. When using images as input data the topological map construction process requires several image-to-image or image-to-nodes (set of images) comparisons in order to incrementally build the topological map.

This problem has been widely studied in robotics, and most of the state-of-the-art approaches

*Corresponding author. Email: miguel.cazorla@ua.es

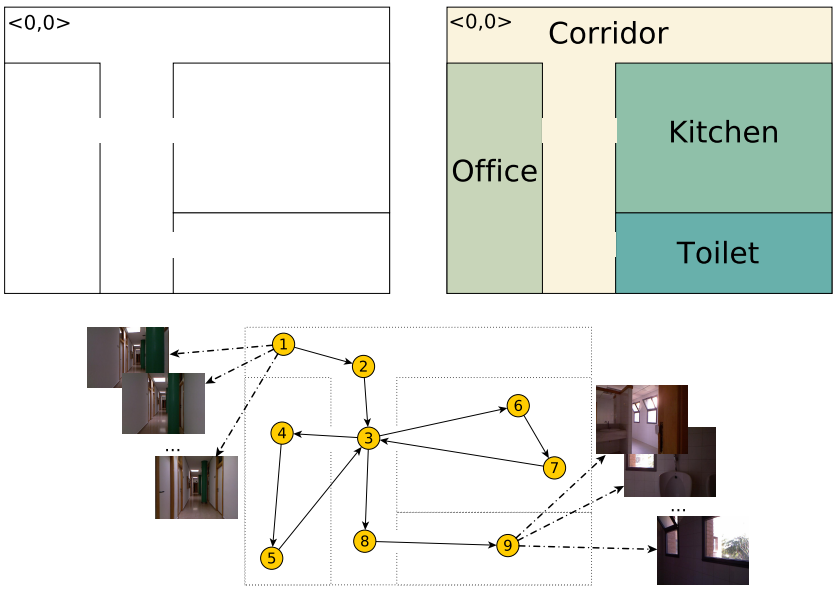


Figure 1.: Metric (top-left), metric-semantic (top-right), and topological exemplar maps (bottom)

rely on the use of computer vision techniques to estimate the similarity between robot perceptions, which are usually in the form of images [6, 7]. This standard approach, however, presents an important drawback: the poor interpretability of the generated maps. Furthermore, two images can be visually different while representing a similar location, due to changes in the viewpoint or structural modifications.

To cope with these two drawbacks, in this article we propose the use of image annotations as input for topological map generation instead of usual visual features extracted from the image. By image annotations we refer to the set of tags or lexical labels used to characterize an input image. While the annotation process has been traditionally an expensive or even unapproachable task, the recent availability of deep learning models allows for efficient real-time annotations of any input image. These models are trained using huge datasets, such as ImageNET [8, 9] or Places [10], where images are annotated using a large and heterogeneous set of lexical labels.

The advantages of using lexical labels to describe/represent an image (in our case obtained from deep learning classification models) are twofold:

- First, the similarity between images can be computed without the use of any computer vision technique. That avoids selecting the optimal set of image features to be extracted (e.g. SIFT [11], SURF [12], HoG [13], ...), to make use of dimensionality reduction techniques, as well as carrying out further parameter tuning processes, which typically rely on proposals that are too specific and environment dependent.
- Second, the locations or nodes of the generated topological map can be described by means of the lexical labels associated to their contained images.

This novel map representation allows automatic objective-driven navigation, since a robot can understand a sentence such as “bring me a cup of coffee” without the need of making any explicit reference to the location where coffee cups are expected to be (typically in the kitchen) or where the beneficiary of the the action is currently located.

The main contribution of this work is the generalist framework for generating descriptive topological maps. The proposal has been evaluated on the KTH-IDOL 2 dataset, which consists of sequences of images acquired under three different lighting conditions: sunny, cloudy, and night. Moreover, the descriptive capabilities of the maps have also been shown and discussed for future

applications. The rest of the paper is organized as follows. In Section 2 we review some related works and state-of-the-art solutions to the topological mapping problem. The process for extracting annotations and computing the similarity between images based on lexical labels is presented in Section 3. Section 4 describes the procedure for generating topological maps from lexical labels. Experimental results and the descriptive capabilities of the LexToMap proposal are presented in Section 5. Finally, the main conclusions of this work as well as some future research directions are outlined in Section 6.

2. Related Work

The similarity between images has been widely used for several robotic tasks such as object recognition [14], navigation [15] and semantic localization [16]. Regarding topological mapping, large image collections [5] are the traditional main source of information. This fact increases the computational time when applying image matching approaches, and this encourages the search for alternative approaches, capable of coping with large sequences of images. The visual similarity between images has traditionally been computed from invariant local features [6, 17], and global image descriptors [18, 19], mainly generated by following bag-of-words approaches [20]. From these image representations, the spatial distribution of the map has been modeled using graph representations [21], as well as hierarchical proposals [22]. More concretely, [21] provides a way to detect loop clousure, but the proposed system needs to learn the visual features in the environment. Our method differs from the former in twofold: first, we do not need to learn the environment and, second, our aim is not only to detect loop clousure but also to build the map at the same time, which is not achieved by [21].

OpenRatSlam [23] and ORBSlam [24] are well-known current SLAM solutions, which rely on the use of matching and bag-of-words approaches respectively, but their requirements (visual images should be provided in conjunction with the camera rotational and translational velocity) and limitations (poor descriptive capabilities of the generated maps) encourage the search for novel approaches related to topological mapping.

The emergence of deep learning in the robotic community has opened up new research opportunities in the last few years. In addition to model generation for solving open problems [25, 26], the release of pre-trained models allows for a direct application of the deep learning systems generated [27]. This is possible thanks to the existence of modular deep learning frameworks such as Caffe [28]. The direct application of pre-trained models avoids the computational requirements for learning them: long learning/training time even using GPU processing, and massive data storage for training data. From the existing deep learning models, we should point out those generated from images categorized with generalist and heterogeneous lexical labels [10, 29]. The use of these models lets any computer vision system annotate input images with a set of lexical labels describing their content, as it has been recently shown in [27, 30, 31].

3. Lexical-based Image Descriptors

In contrast to most of the topological mapping proposals, we describe or represent images by means of a set of predefined lexical labels. The rationale behind this representation is to describe the content of the image by means of a set of semantic concepts that can be automatically attributed to this image. For example if we describe an image saying that the appearance in it of concepts such as fridge, table, chair, cabinet, cup, and pan, is much more likely than other different concepts in the predefined set, then we can say that that image represents a kitchen with a high degree of confidence. The use of lexical labels may result into a loss of resolution suitable for increasing the perceptual aliasing problem [32]. Besides fine grain representations, by means of large sets of labels

in our proposal, the perceptual aliasing problem is reduced by taking into account the temporal continuity of the sequence. This is expected to associate different locations to different nodes, even when both are translated into similar descriptors.

To implement the lexical annotation process we make use of existing deep learning annotation tools. Deep learning techniques, and more specifically Convolutional Neural Networks (CNN [33]), allow the generation of discriminant models while discovering the proper image features in a totally unsupervised way, once the network architecture has been defined. This is possible nowadays thanks to the availability of huge image datasets annotated with large and miscellaneous set of lexical labels, which efficiently permits the training of these discriminative classification models. In this work, we focus on the application of existing CNN models. The definition and building of these CNN models is beyond the scope of this paper, so we refer the reader to [34] for a more detailed view of deep learning in general and, to [28] for a better understanding of these CNN models.

Once every image is represented by a set of lexical labels, we need to define a similarity measure between two image descriptors or between an image descriptor and a node descriptor. A node on the topological map is composed of a set of images representing that node/location.

The complete process of annotating images using CNNs and the similarity computation details are described below.

3.1. *Image annotation using CNN*

Let $L = \{l_1, \ldots, l_{|L|}\}$ be the set of $|L|$ predefined lexical labels, I an image, and N a node of the topological map formed of $|N|$ images. The direct application of the existing CNN models on an input image I generates a descriptor $d_{CNN}(I) = ([p_I(l_1), \ldots, p_I(l_{|L|})])$, where $p_I(l_i)$ denotes the probability of describing the image I using the i-th label in L. This obtains a representation similar to the Bag of Visual Words (BoVW [35, 36]) approach, which generates a descriptor vector $d_{BoW}(I) = [n_I(w_1), \ldots, n_I(w_k)]$ of k visual words, where $n(w_i)$ denotes the number of occurrences of word w_i in image I. Despite the fact that spatial relation between words is completely removed, we decide not follow any of the proposed techniques, like the spatial pyramid [37], to solve this drawback. In addition to avoid the higher processing requirements this technique requires, our selection relies in the assumption that the presence of lexical labels is much more important than their position to describe any input image.

We use a similar notation to represent the descriptor of a node N of the topological map, which is composed of a set of $|N|$ images ($N = \{I_1, \ldots I_{|N|}\}$). The descriptor of N is defined as the vector of the average label probability of its $|N|$ images, and the corresponding vector of standard deviations. More formally:

$$d_{CNN}(N) = ([\bar{p}_N(l_1), \ldots, \bar{p}_N(l_{|L|})], [\sigma_N(l_1), \ldots, \sigma_N(l_{|L|})]) \tag{1}$$

where:

$$\bar{p}_N(l_i) = \frac{1}{|N|} \sum_{j=1}^{|N|} p_j(l_i) \tag{2}$$

and

$$\sigma_N^2(l_i) = \frac{1}{|N|} \sum_{j=1}^{|N|} (p_j(l_i) - \bar{p}_j(l_i))^2 \tag{3}$$

This average computation is actually the aggregation of all image descriptors that form the node.

In Fig. 2 a visual interpretation of this aggregation process is shown.

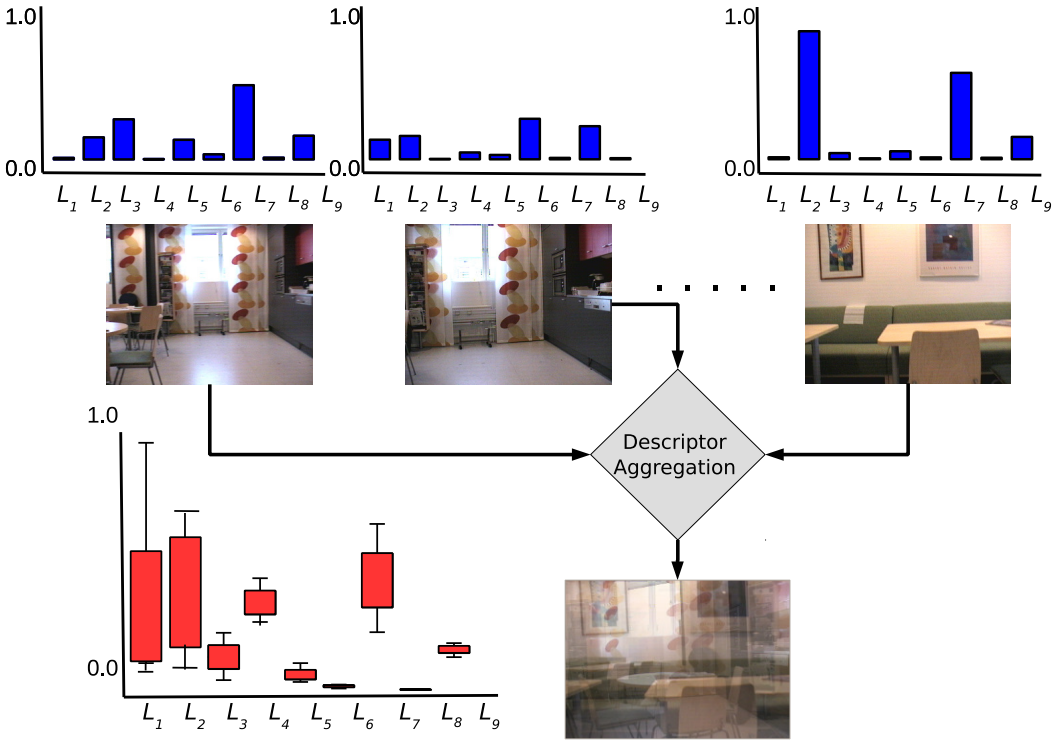


Figure 2.: An example of the aggregation process of n different images to define the descriptor of a node/location of only nine lexical labels

While the i-th average values encode the probability of describing the location using the lexical label i (e.g. the probability of describing the location as "table" is 75%), the standard deviation indicates whether this label is representative of the location. That is, large deviations denote lexical labels whose likelihood is not constant in the images from the same location. Therefore, we propose to integrate this information into the descriptor definition of the node in order to reduce the importance of lexical labels with large standard deviations, which are eventually considered not representative of the node/location.

In order to train a CNN model, we need to provide both the architecture of the network and the database to be used as training set. The architecture refers to internal details such as the number of convolutional or fully connected layers, or the spatial operations used in the pooling stages. On the other hand, the training set determines the number of lexical labels used to describe the input image. In this proposal, we take advantage of Caffe [28], a fast, modular and well documented deep learning framework that is widely used by researchers. We opted for this framework because of the large community of contributors providing pre-trained models that are ready to be used in any deployment of the framework.

3.2. *Image/Node Descriptor Similarity Computation*

On the one hand, the similarity between two images, I_a and I_b, whose representation has been obtained from a CNN model, can be estimated using histogram intersection techniques [38]. In this case, we need to compare two descriptors, $d(I_a)$ and $d(I_b)$, encoding the set of likelihoods describing images I_a and I_b using the set of pre-defined lexical labels. We can adopt well-known similarity measures such as p-norm based distances (i.e. Manhattan or Euclidean distance), the

Bhattacharyya or the χ^2 distance, among others, to compare them.

On the other hand, the similarity between an image I and a node N is defined in this proposal as a weighted similarity using the standard deviation of the labels ($\sigma_N(l_i)$) within the node. This is done to explicitly reduce the importance of labels presenting large variance in the node, which are considered non-representative ones, as well as to increase the relevance of those labels with low variance.

Based on the weighted euclidean distance formulation, the distance between a node/location N and an image I is computed according to:

$$D(N, I) = \frac{1}{\sum_{i=1}^{|L|}(w_i)} \sum_{i=1}^{|L|}(w_i \cdot (\bar{p}_N(l_i) - p_I(l_i))^2) \tag{4}$$

where w_i has been defined to be inversely proportional to the standard deviation and normalized in the range $[0, 1]$.

4. LexToMap: Lexical-based Topological Mapping

From the image descriptors obtained from CNNs, and using the distance functions described above to estimate the distance between an image and a location, we define the lexical-based topological mapping using the pseudo-code in Algorithm 1.

In this process, we can find the starting situation where a new node (representing a location) is created from the first image. From there, we firstly try to add the images to the current node in order to take advantage of the temporal continuity of the sequence. If this is not possible, due to a big difference between the image and the current node (using threshold T_1), we search in the node list for the most similar node. If this node exists, and it is similar enough to the image (using threshold T_2), we mark it as the current one, we add the image to it, and create a transition (edge) from the former node to the current one, if it does not already exist. Otherwise, we create a new node on the map, which is established as the current one, and then the transition from the past node to the new one is created.

Each node or location consists of a set of image descriptors encoded as vectors representing lexical label probabilities. For evaluation and visualization purposes, we can also identify a node by its $<x, y>$ position in the environment by taking advantage of the ground truth. The coordinates of a node are represented by the average values of x and y computed from the position coordinates of all the images included in the node.

The topological maps generated with our proposal would be trajectory dependent, as the first image acquired with the robot plays a very important role in the process. The temporal continuity is also exploited to reduce the perceptual aliasing problem. Nevertheless, this dependency also allows the mapping procedure to generate maps in an online fashion. This avoids waiting for further acquisitions for making decisions about nodes and transition generation, which is undesired for any robotic system. Moreover, the online generation of topological maps permits the robot to return to intermediate previous locations. This situation is commonly faced due to battery problems, when the robot should come back as soon as possible to the charging area. Rescue robots may also cope with similar scenarios, where the riskiness of a discovered area encourage the robot to return to previous safe locations.

Algorithm 1 LexToMap: Lexical-based Topological Mapping

NodeList = $\emptyset$
CurrentNode = None
for each image I_j acquired from the robot **do**
  if length(NodeList) == 0 **then**
    Create a new Node N_{new} from I_j
    Add N_{new} to NodeList
    CurrentNode = N_{new}
  else
    if $D(\text{CurrentNode}, I_j) < T_1$ **then**
      CurrentNode = CurrentNode $\bigcup I_j$
    else
      $N_{sim} = None$
      $Min_{dist} = \infty$
      for each node N_z in NodeList **do**
        $d_z = D(N_z, I_j)$
        if $d_z < Min_{dist}$ && $N_z \neq CurrentNode$ **then**
          $N_{sim} = N_z$
        end if
      end for
      if $N_{sim} \neq$ None && $D(N_{sim}, I_j) < T_2$ **then**
        Create a transition from CurrentNode to N_{sim}
        CurrentNode = N_{sim}
        CurrentNode = CurrentNode $\bigcup I_j$
      else
        Create a new Node N_{new} from I_j
        Add N_{new} to NodeList
        Create a transition from CurrentNode to N_{new}
        CurrentNode = N_{new}
      end if
    end if
  end if
end for

5. Experimental Results

The LexToMap topological map generation approach was evaluated under the three different lighting conditions proposed in the KTH-IDOL2 dataset. We followed the procedure explained in Algorithm 1, which starts from the descriptor generation procedure. This steps relies on the use of a CNN using a pre-trained model, and we evaluated seven different alternatives. The procedure depends on two different thresholds, T_1 and T_2, which determine the generation of new nodes and the transitions between them. All these steps are detailed in the following subsections.

5.1. *Dataset*

We opted for the KTH-IDOL 2 dataset [39] for the evaluation of our proposal. Image Database for rObot Localization (IDOL) is an indoor dataset that provides sequences of perspective images acquired under three different lighting conditions: sunny, cloudy and night. These sequences were generated using two different robot platforms, namely Minnie (PeopleBot) and Dumbo (PowerBot),

controlled by a human operator. The ground truth in the dataset includes the following information by image: the semantic category of the room where the image was acquired, the timestamp, and the pose of the robot ($<x, y, \theta>$) during the acquisition. There are 5 different room categories: corridor (CR), kitchen (KT), one-person office (1-PO), two-persons office (2-PO), and printer area (PA). The dataset includes four different sequences for each combination of robot and lighting conditions. From all these sequences, we selected the twelve ones acquired with Minnie, whose camera position (around one meter above the floor) is more similar to most of the current mobile robot platforms.

Lighting Cond.	**Cloudy**	**Night**	**Sunny**
Sequences	1,2,3,4	1,2,3,4	1,2,3,4
#Images CR	1632	1704	1582
#Images 1-PO	458	522	468
#Images 2-PO	518	633	518
#Images KT	656	611	556
#Images PA	488	534	482
#Images	3752	4004	3606

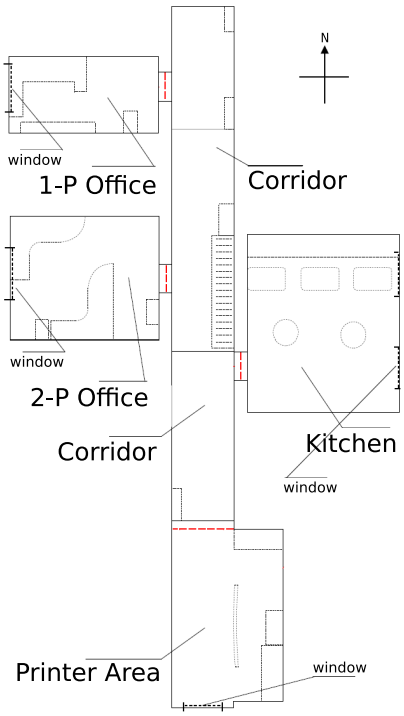


Figure 3.: KTH-IDOL 2 information: Image distribution (left) and environment (right).

The number of images in the dataset by lighting conditions and semantic category, as well as the map used for the acquisition, are shown in Fig 3. The image distribution is clearly unbalanced as most of the images belong to the Corridor category. Sequences 3-4 were acquired six months later than the acquisition of sequences 1-2, which introduces small environment variations due to human activity. Fig. 4 presents 15 exemplar images from the dataset. From these examples, it can be observed how the visual representations are affected by the lighting conditions. Moreover, Fig. 5 illustrates the effect of human activity over the same locations of the environment.

5.2. *Model Selection*

From the whole set of available trained models, we selected the 7 different candidates that are summarized in Table 1. These models differ in the architecture of the CNN used, the dataset used for training them, and the set of predefined lexical labels used by the model. We opted for these models because they were all trained over datasets that consist of images annotated with a large set of generalist lexical labels. In this experimentation, we are interested in the categorization capabilities of the lexical labels generated through the CNN models. Therefore, we firstly propose an unsupervised learning procedure using the dataset sequences as input. This was carried out by using a k-means clustering algorithm considering different values of k in the range $[1, 50]$. Each cluster represents a location (topological node) computed using only image similarity information. That is, the temporal continuity of the sequence is not taken into account. Then, we evaluate the

Figure 4.: Exemplar images from the KTH-IDOL 2 dataset acquired under three lighting conditions (rows) within five different room categories (columns).

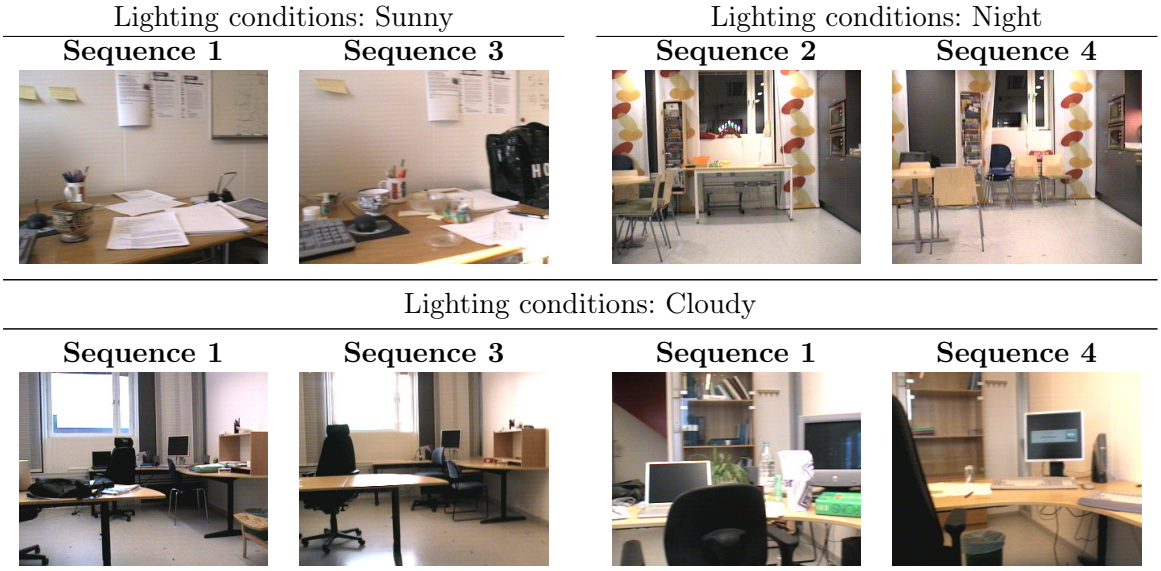


Figure 5.: Images illustrating changes produced by human activity on the environment.

average spatial intra-cluster variance by using the dataset ground truth $<x, y>$ location of the input images. Fig. 6 graphically presents the evolution of the intra-cluster variance when using different values of k in the k-means clustering algorithm. This figure introduces a bias–variance trade-off where larger k values result in less generalist but more accurate clusters.

Table 2 shows a subset of the results obtained using four representative values of k. In this table, each column presents the average spatial intra-cluster variance of the whole combination of lighting conditions and sequences for a specific value of k. Lower intra-cluster variances are desirable as they denote a more precise representation of the data. Indeed, low variances are obtained with clusters that consist of images acquired from nearby environment positions. From the complete set of results, we computed the average ranking for all values of k in the range [1,50]. That resulted in 50 different test scenarios where each model was ranked between the first and the seventh position, depending on its intra-cluster variance.

Table 1.: Details of the seven CNN models evaluated in the proposal

Model Name	**CNN Architecture**	**CL**[a]	**FCL**[b]	**Training Datasets**	**#Labels**
ImageNet-AlexNet	AlexNet [29]	5	3	ImageNet2012 [8, 9]	1000
ImageNet-CaffeNet	AlexNet	5	3	ImageNet2012	1000
ImageNet-GoogLeNet	GoogLeNet [40]	11	3	ImageNet2012	1000
ImageNet-VGG	VGG CNN-s [41]	5	3	ImageNet2012	1000
Hybrid-AlexNet	AlexNet	5	3	Hybrid MIT [10]	1183
Places-AlexNet	AlexNet	5	3	Places205 MIT [10]	205
Places-GoogLeNet	GoogLeNet	11	3	Places205 MIT	205

[a] Convolution Layers
[b] Fully Connected Layers

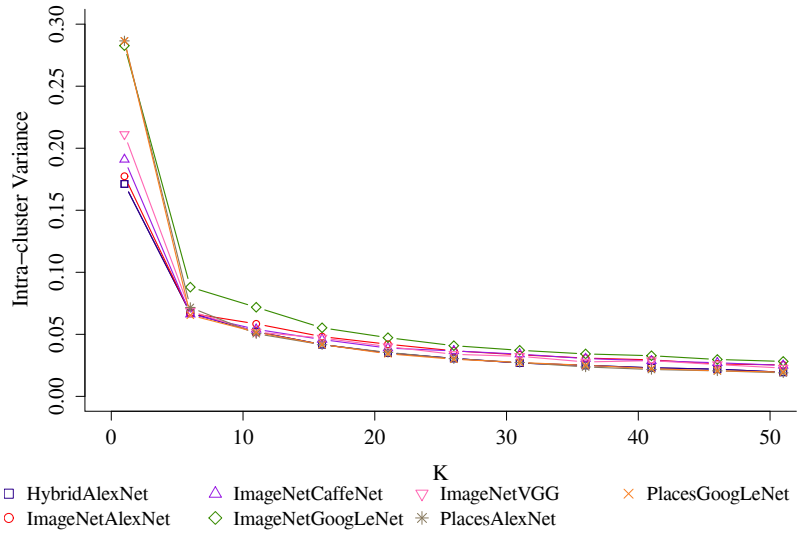


Figure 6.: Intra-cluster spatial evolution using different values of k for the 7 CNN models studied

Table 2.: Intra-cluster variance ($\cdot 10^{-2}$) for 7 CNN models and four representative k values. Lowest values per column are in bold.

CNN Model	k=7	k=15	k=30	k=50
HybridAlexNet	**5.49**	**3.99**	**2.50**	**1.85**
ImageNetAlexNet	7.53	4.19	3.04	2.42
ImageNetCaffeNet	6.24	5.34	3.23	2.43
ImageNetGoogLeNet	7.35	6.02	3.97	2.90
ImageNetVGG	5.69	4.44	3.17	2.39
PlacesAlexNet	6.64	4.37	2.90	1.99
PlacesGoogLeNet	6.91	4.59	2.85	1.97

The ranking comparison summary is presented in Fig. 7, where it can be observed how Hybrid-AlexNet clearly outperforms the rest of the evaluated models. Therefore, we selected this model as optimal (among those used in this study) for topological mapping, and it was used for the rest of the experimentation. The proper behavior of the Hybrid dataset comes from the fact that it has been generated from a combination of both Places and ImageNET datasets, once the overlapping scene categories were removed [10]. The ranking comparison also pointed out the appropriateness of using the Places dataset in contrast to ImageNET, which is explained due to the nature of the

annotations, more suitable for discriminating between indoor locations. With regard to the network architecture, those with lower number of convolution layers presented the best behaviors.

method	rank	win	tie	loss
HybridAlexNet	1.40	-	-	-
PlacesAlexNet	3.00	46	0	4
PlacesGoogLeNet	3.50	48	0	2
ImageNetVGG	4.04	46	0	4
ImageNetAlexNet	4.28	45	0	5
ImageNetCaffeNet	4.88	45	0	5
ImageNetGoogLeNet	6.90	50	0	0

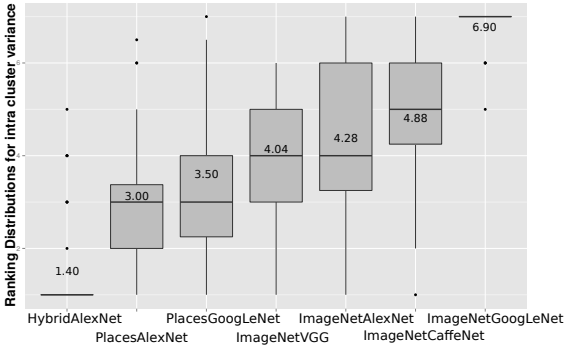


Figure 7.: Ranking comparison of 7 CNN models. Win, tie and loss results (left) and average ranking visualization (right)

5.3. *Topological Map Generation*

Once the CNN model has been chosen, we should establish the values for the two thresholds used in the topological mapping: T_1 and T_2. These threshold values can be selected to generate different types of maps based on the end requirements of the topological maps. For example, we can prevent the algorithm from generating a large set of nodes/locations by selecting large T_1 values. And, large T_2 values facilitate the generation of connections between existing nodes. This increases the average number of connections by node. The automatic selection of these two thresholds would require the availability of a quantitative metric to evaluate the goodness of any topological map. Unfortunately, we could not find any proven metric in the literature and its generation is not trivial. In order to establish a trade-off between specificity and generality, we empirically selected $15 \cdot 10^{-2}$ and $15 \cdot 10^{-3}$ for T_1 and T_2 thresholds, respectively. Using the Hybrid-AlexNet CNN model, we generated a set of twelve topological maps for a more detailed evaluation and discussion from all the dataset sequences. All these maps were generated using the same internal parameters (CNN model and thresholds).

The maps generated are shown in Fig. 8 for three lighting conditions. It can be observed how valuable topological maps can be generated thanks to the use of the LexToMap proposal without the need for any other additional or complementary visual information. Although lighting variations within indoor environments are not so challenging as for outdoor ones, we opted for an indoor dataset incorporating some lighting changes (see Fig. 4). The maps generated are not drastically affected by these changes thanks to the use of the lexical labels to compute the similarities between images, which are proposed instead of standard visual features.

In Fig. 9, we can observe two different types of transitions, which correspond to the generation of the map from sequence 3 acquired with cloudy lighting conditions. Concretely, we illustrate the timestamp when the robot backs to the corridor after visiting the one-person office. During previous tours along the corridor, the algorithm created different nodes and transitions between them. Before leaving the one-person office, the mapping algorithm has Node 10 as current node. When the robot acquires an image different from previous ones (Fig. 9 bottom right), Node 4 is discovered as an aggregation of images similar to the last robot perception. This is translated into a new transition between nodes 10 and 4. After a certain number of acquisitions, a new transition is requested due to the contrast between the new image (Fig. 9 top left) and the current node. However, no similar past nodes are detected, and then a new node (Node 11) is generated and established as current node.

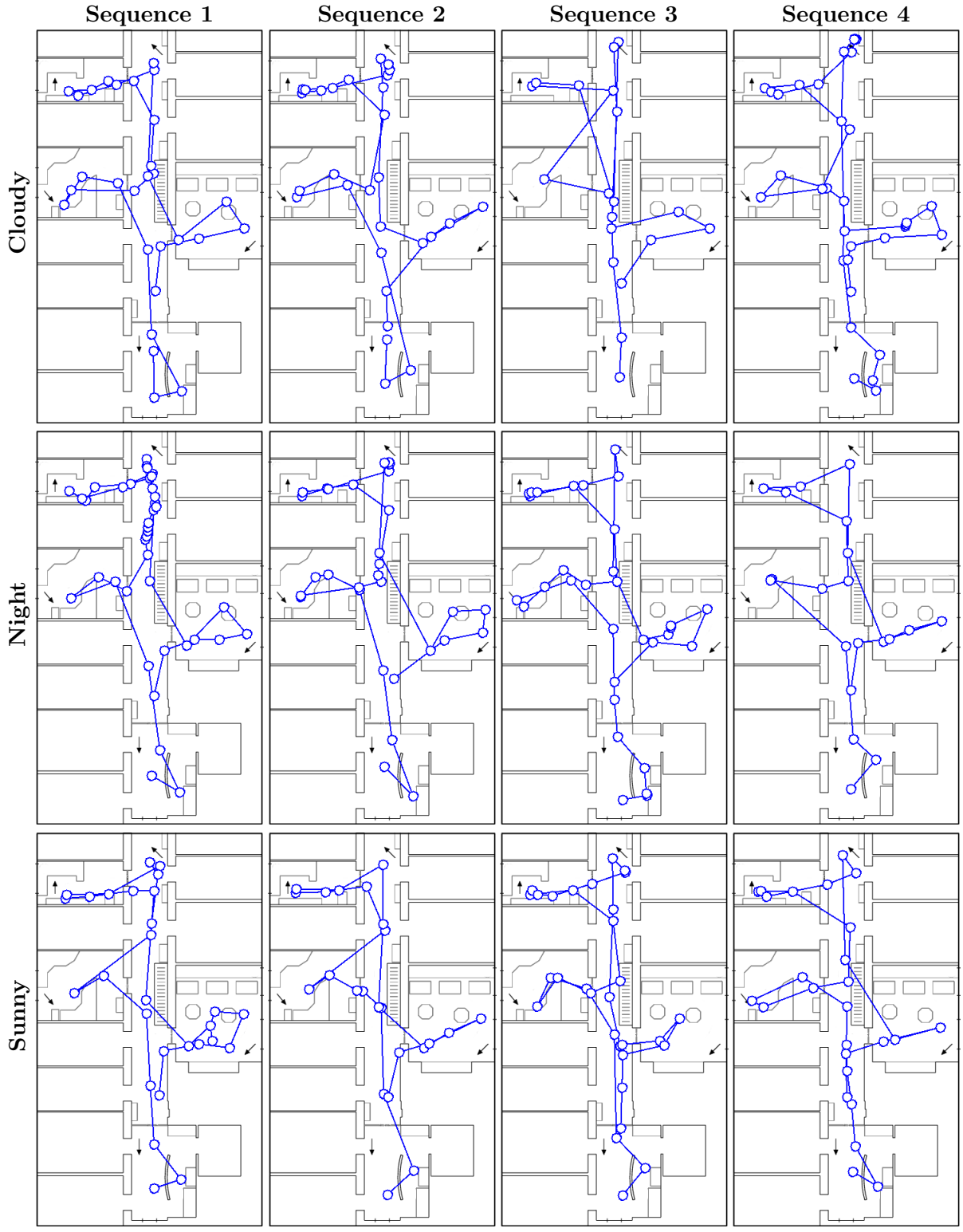


Figure 8.: Topological maps generated for three different lighting conditions: cloudy (top), night (middle) and sunny (bottom)

Figure 9.: Transition generation during a LexToMap mapping procedure.

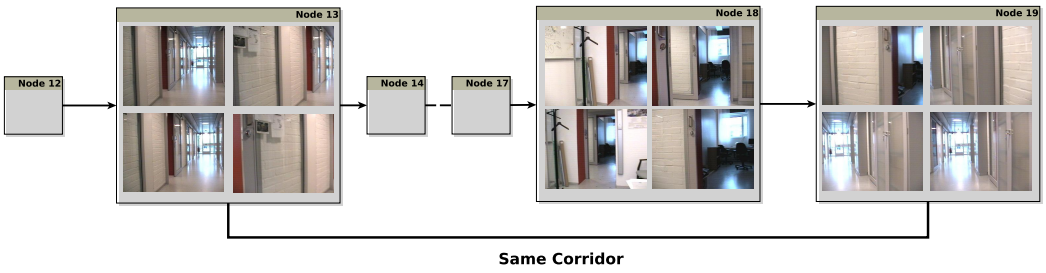


Figure 10.: Transition to an already visited location not detected.

Despite the promising results obtained with the LexToMap technique, there are some failure cases as the one illustrated in Fig. 10. It corresponds to the generation of the topological map from Sequence 1 Cloudy, and presents a node generation (Node 19) that should not have been performed, as there was a previous node (Node 13) created from images acquired in the same location. This failure may come from the threshold selection, which compromises a trade-off between specificity and generality, and aims to generate valid maps for all the sequences in the dataset. Another point to be taken into account is the difference of the corridor with respect to the rest of room categories. Namely, the corridor imaged in the dataset is unobstructed without the presence of objects. This avoids detecting some loop closures due to the lack of discriminating objects, as that shown in Fig. 10.

The characteristics of the corridor also help us discover a proper behavior of the proposal: its adaptability to cope with heterogeneous rooms. Concretely, we can observe how largest transitions in the topological maps appear in the corridor area. This involves a lower density of nodes in this room category. This is desirable because the rest of rooms, especially the offices and the kitchen, are more suitable for incorporating relevant sub regions due to the presence of specific objects, like fridges or computers.

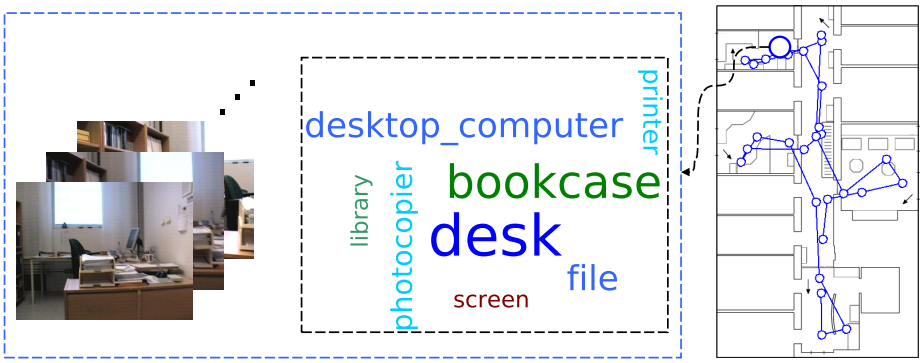
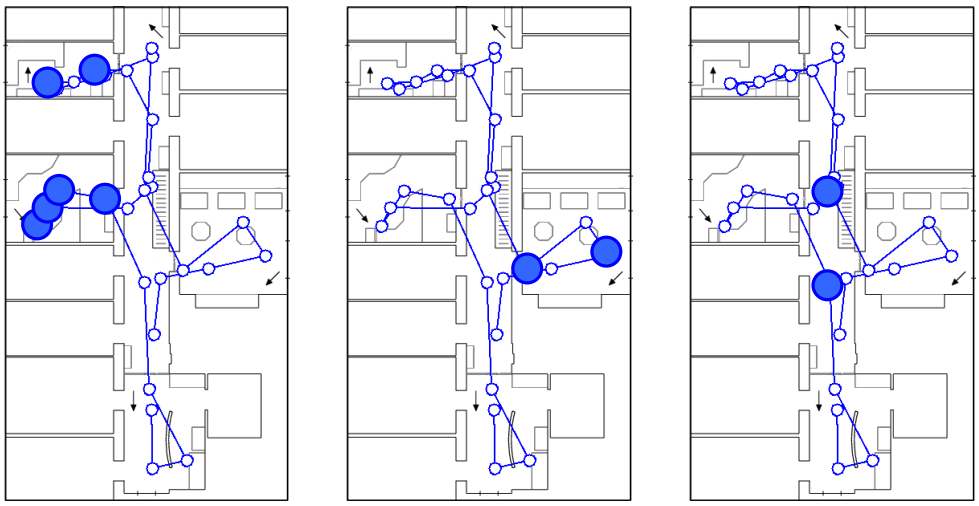


Figure 11.: Location description by means of the cloud of representative lexical labels.

5.4. *Description Capabilities*

The topological maps generated with our proposal present a clear advantage when compared with those generated with state-of-the-art approaches, namely their descriptive capabilities. This comes to the fact that, each topological node is generated from a set of lexical labels that can be used to describe its content. Fig. 11 shows an example of a topological map generated from sequence 1 under cloudy conditions (Fig. 8 top left). In this figure, we highlight a location (which belongs to the one-person office room category in the dataset), along with some of the images this location consists of, and the lexical labels word-cloud. This cloud is computed from the set of most-representative lexical labels, where font sizes denote the likelihood of the label in this location.

In addition to the descriptive capabilities, the lexical labels are amazingly useful for goal-driven navigation tasks. That is, the labels associated to a topological location refer to the content of the scenes, and therefore can determine the type of actions the robot would perform. In order to illustrate this capability, we remarked the locations on the same topological map including three different labels in their top-five most representative (higher likelihood) ones: desktop computer, refrigerator and banister.



Desktop Computer Refrigerator Banister

Figure 12.: Locations representative of the lexical labels “desktop computer”, “refrigerator” and “banister”.

This is shown in Fig. 12, and it can be observed how all the locations selected with the label

"desktop computer" belong either to a one-person or two-person office, which are the semantic categories (in comparison with kitchen, corridor or printer area) more likely to contain a desktop computer. A similar scenario was obtained with labels "refrigerator" and "banister", which select locations that belong to kitchen and corridor categories respectively.

In addition, we are also interested in knowing how a lexical label is distributed over a topological map. For a better understanding we illustrated two examples in Fig. 13 using the lexical labels "sliding door" and "photocopier". In this figure, we use a heat color coding to represent the probability of describing each location using the provided lexical labels.

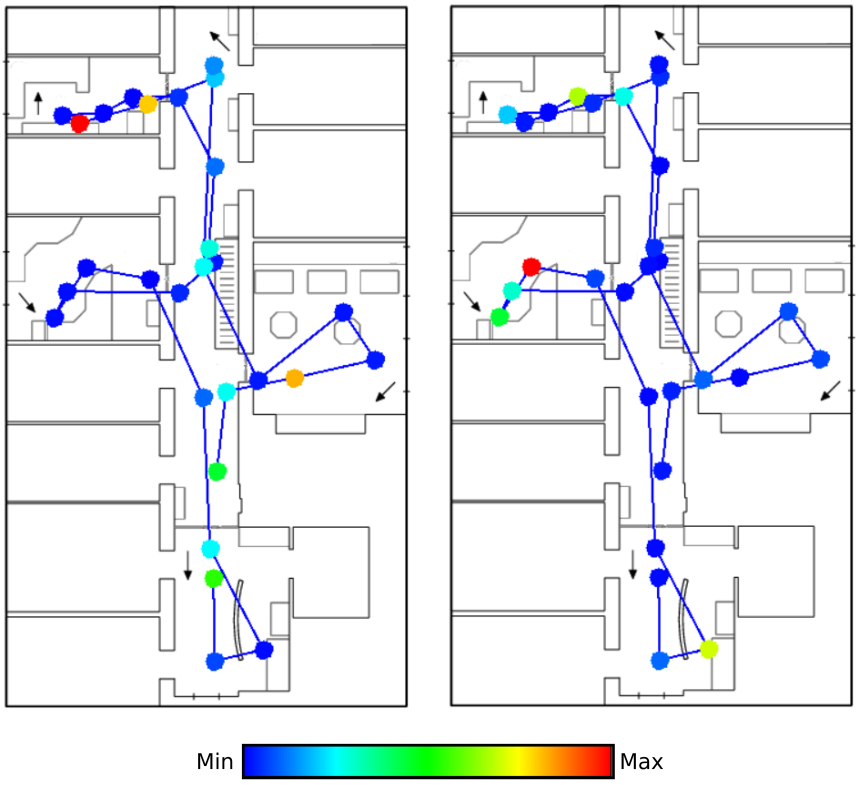


Figure 13.: Topological maps where color codes represent the probability of describing each location with the lexical labels "sliding door" (left) and "photocopier" (right).

6. Conclusions and future work

We have presented a novel approach for lexical-based topological mapping in this article. The proposal relies on the annotation capabilities of the available CNN pre-trained models, and takes advantage of them to compute the similarity between input images. This strategy presents two main benefits. Firstly, the similarity between map locations is computed from a lexical point of view and not only from visual feature similitude. This point increases the robustness of the method to challenging environments with small lighting variations or changes in the viewpoint. Secondly, the integration of annotations with lexical labels in the map generation procedure increases the representational capabilities of the maps, as locations can be described using a set of lexical labels. Moreover, these labels could be extremely useful for future navigation purposes. Based on the results obtained under different lighting conditions, we can conclude that valuable topological maps can be obtained by following a standard approach without the need for selecting and tuning computer

vision (for feature extraction) or machine learning (for matching and classification) algorithms.

We have presented a qualitative evaluation of our method. We know that a quantitative metric must be provided for a better evaluation. However, for the best of our knowledge, there is a lack of such quantitative metric and we plan to develop it in future work.

As future work, we plain to integrate state-of-the-art strategies for loop-closure detection. We also have in mind the comparison of the maps generated from different proposals, including traditional approaches using visual features. To this end, as well as to automatically select the optimal values for the thresholds included in the algorithm, we are additionally working on the proposal of a metric suitable for evaluating the goodness of any topological map.

Acknowledgments

This work was supported by grants DPI2013-40534-R and TIN2015-66972-C5-2-R of the Ministerio de Economia y Competitividad of the Spanish Government, supported with Feder funds, and by Consejería de Educación, Cultura y Deportes of the JCCM regional government through project PPII-2014-015-P. José Carlos Rangel is funded by the IFARHU grant 8-2014-166 of the Republic of Panamá.

References

[1] Thrun S. Exploring artificial intelligence in the new millennium. Chapter Robotic Mapping: A Survey; San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 2003. p. 1–35.
[2] Booij O, Terwijn B, Zivkovic Z, et al. Navigation using an appearance based topological map. In: International Conference on Robotics and Automation. IEEE; 2007. p. 3927–3932.
[3] Bhattacharya P, Gavrilova ML. Roadmap-based path planning - using the voronoi diagram for a clearance-based shortest path. IEEE Robotics Automation Magazine. 2008;15:58–66.
[4] Pronobis A, Martínez Mozos O, Caputo B, et al. Multi-modal semantic place classification. The International Journal of Robotics Research. 2010;29:298–320.
[5] Fraundorfer F, Engels C, Nistér D. Topological mapping, localization and navigation using image collections. In: International Conference on Intelligent Robots and Systems. IEEE; 2007. p. 3872–3877.
[6] Valgren C, Lilienthal A, Duckett T. Incremental topological mapping using omnidirectional vision. In: International Conference on Intelligent Robots and Systems. IEEE; 2006. p. 3441–3447.
[7] Angeli A, Doncieux S, Meyer JA, et al. Visual topological SLAM and global localization. In: International Conference on Robotics and Automation. IEEE; 2009. p. 4300–4305.
[8] Deng J, Dong W, Socher R, et al. Imagenet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition. IEEE; 2009. p. 248–255.
[9] Russakovsky O, Deng J, Su H, et al. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision. 2015;115:211–252.
[10] Zhou B, Lapedriza A, Xiao J, et al. Learning deep features for scene recognition using places database. In: Advances in Neural Information Processing Systems; 2014. p. 487–495.
[11] Lowe DG. Object recognition from local scale-invariant features. In: The proceedings of the seventh IEEE international conference on Computer vision; Vol. 2. IEEE; 1999. p. 1150–1157.
[12] Bay H, Ess A, Tuytelaars T, et al. Speeded-up robust features (SURF). Computer Vision and Image Understanding. 2008;110:346–359.
[13] Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: Conference on Computer Vision and Pattern Recognition; Vol. 1. IEEE; 2005. p. 886–893.
[14] Cyr CM, Kimia BB. 3D object recognition using shape similiarity-based aspect graph. In: International Conference on Computer Vision; Vol. 1. IEEE; 2001. p. 254–261.
[15] Tudhope D, Taylor C. Navigation via similarity: automatic linking based on semantic closeness. Information Processing & Management. 1997;33:233–242.
[16] Martínez-Gómez J, Jiménez-Picazo A, Gamez JA, et al. Combining invariant features and localiza-

tion techniques for visual place classification: successful experiences in the RobotVision@ImageCLEF competition. Journal of Physical Agents. 2011;5:45–54.
[17] Goedemé T, Tuytelaars T, Van Gool L, et al. Feature based omnidirectional sparse visual path following. In: International Conference on Intelligent Robots and Systems. IEEE; 2005. p. 1806–1811.
[18] Koseck J, Li F. Vision based topological markov localization. In: International Conference on Robotics and Automation; Vol. 2. IEEE; 2004. p. 1481–1486.
[19] Liu M, Scaramuzza D, Pradalier C, et al. Scene recognition with omnidirectional vision for topological map using lightweight adaptive descriptors. In: International Conference on Intelligent Robots and Systems. IEEE; 2009. p. 116–121.
[20] Filliat D. A visual bag of words method for interactive qualitative localization and mapping. In: International Conference on Robotics and Automation. IEEE; 2007. p. 3921–3926.
[21] Cummins M, Newman P. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. The International Journal of Robotics Research. 2008;27:647–665.
[22] Kuipers B, Modayil J, Beeson P, et al. Local metrical and global topological maps in the hybrid spatial semantic hierarchy. In: International Conference on Robotics and Automation; Vol. 5. IEEE; 2004. p. 4845–4851.
[23] Ball D, Heath S, Wiles J, et al. Openratslam: an open source brain-based slam system. Autonomous Robots. 2013;34:149–176.
[24] Mur-Artal R, Montiel JMM, Tardós JD. Orb-slam: A versatile and accurate monocular slam system. IEEE Transactions on Robotics. 2015 Oct;31:1147–1163.
[25] Bo L, Ren X, Fox D. Unsupervised feature learning for rgb-d based object recognition. In: Experimental Robotics. Springer; 2013. p. 387–402.
[26] Neverova N, Wolf C, Taylor GW, et al. Multi-scale deep learning for gesture detection and localization. In: Computer Vision-ECCV 2014 Workshops. Springer; 2014. p. 474–490.
[27] Rangel JC, Cazorla M, García-Varea I, et al. Scene classification based on semantic labeling. Advanced Robotics. 2016;30:758–769; Available from: http://dx.doi.org/10.1080/01691864.2016.1164621.
[28] Jia Y, Shelhamer E, Donahue J, et al. Caffe: Convolutional architecture for fast feature embedding. In: nternational Conference on Multimedia. New York, NY, USA: ACM; 2014. p. 675–678.
[29] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems; 2012. p. 1097–1105.
[30] Carneiro G, Nascimento J, Bradley AP. Unregistered multiview mammogram analysis with pre-trained deep learning models. In: Medical image computing and computer-assisted intervention. Springer; 2015. p. 652–660.
[31] Murthy VN, Maji S, Manmatha R. Automatic image annotation using deep learning representations. In: International Conference on Multimedia Retrieval. ACM; 2015. p. 603–606.
[32] Chrisman L. Reinforcement learning with perceptual aliasing: the perceptual distinctions approach. In: Proceedings of the tenth national conference on Artificial intelligence. AAAI Press; 1992. p. 183–188.
[33] Lee H, Grosse R, Ranganath R, et al. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: International Conference on Machine Learning. ACM; 2009. p. 609–616.
[34] Bengio Y. Learning deep architectures for AI. Foundations and trends® in Machine Learning. 2009; 2:1–127.
[35] Sivic J, Zisserman A. Video Google: A text retrieval approach to object matching in videos. In: International Conference on Computer Vision. IEEE; 2003. p. 1470–1477.
[36] Martínez-Gómez J, Morell V, Cazorla M, et al. Semantic localization in the PCL library. Robotics and Autonomous Systems. 2016;75, Part B:641 – 648.
[37] Lazebnik S, Schmid C, Ponce J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2; CVPR '06. Washington, DC, USA: IEEE Computer Society; 2006. p. 2169–2178; Available from: http://dx.doi.org/10.1109/CVPR.2006.68.
[38] Lee S, Xin J, Westland S. Evaluation of image similarity by histogram intersection. Color Research & Application. 2005;30:265–274.
[39] Luo J, Pronobis A, Caputo B, et al. Incremental learning for place recognition in dynamic environments. In: International Conference on Intelligent Robots and Systems. IEEE; 2007. p. 721–728.
[40] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. In: Conference on Computer Vision

and Pattern Recognition. IEEE; 2015. p. 1–9.
[41] Chatfield K, Simonyan K, Vedaldi A, et al. Return of the devil in the details: Delving deep into convolutional nets. In: Proceedings of the British Machine Vision Conference. BMVA Press; 2014.