J Intell Robot Syst (2012) 68:147–163
DOI 10.1007/s10846-012-9671-z

Hierarchical Classifiers for Robust Topological Robot Localization

Ehsan Fazl-Ersi · James H. Elder · John K. Tsotsos

Received: 14 August 2011 / Accepted: 21 March 2012 / Published online: 14 April 2012
© Springer Science+Business Media B.V. 2012

Abstract This paper presents a novel appearance-based technique for topological robot localization and place recognition. A vocabulary of visual words is formed automatically, representing local features that frequently occur in the set of training images. Using the vocabulary, a spatial pyramid representation is built for each image by repeatedly subdividing it and computing histograms of visual words at increasingly fine resolutions. An information maximization technique is then applied to build a hierarchical classifier for each class by learning informative features. While top-level features in the hierarchy are selected from the coarsest resolution of the representation, capturing the holistic statistical properties of the images, child features are selected from finer resolutions, encoding more local characteristics, redundant with the information coded by their parents. Exploiting the redundancy in the data enables the localization system to achieve greater reliability against dynamic variations in the environment. Achieving an average classification accuracy of 88.9% on a challenging topological localization database, consisting of twenty seven outdoor places, demonstrates the advantages of our hierarchical framework for dealing with dynamic variations that cannot be learned during training.

Keywords Topological robot localization · Visual localization · Hierarchical classifiers · Information theory

E. Fazl-Ersi (✉) · J. H. Elder · J. K. Tsotsos
Department of Computer Science and Engineering, York University, Toronto, M3J 1P3, ON, Canada
e-mail: efazl@cse.yorku.ca

J. H. Elder
e-mail: jelder@cse.yorku.ca

J. K. Tsotsos
e-mail: tsotsos@cse.yorku.ca

1 Introduction

One of the fundamental requirements for an autonomous mobile robot is localization, i.e., the capability of knowing where it is located within its world. Robots should be able to localize themselves in order to navigate in the environment, compute a path to a target destination, and recognize that the target destination has been reached. Localization in complex environments usually relies on a map which can be either given to the robot (e.g., topological maps), or learned while the robot discovers its surroundings (e.g., metric maps).

There are two types of localization: qualitative and quantitative. Qualitative localization

(also referred to as topological localization or place recognition) gives the robot the capability of recognizing different places, but not the ability to estimate a precise metrical pose. On the other hand, quantitative localization provides the robot with the capability to estimate its exact pose relative to a metric map. In this paper we focus on the problem of qualitative localization, which can be seen as a starting point for quantitative localization.

The majority of early works on qualitative localization (place recognition) use laser range finders to perceive the environment. These methods usually use shape features (describing the geometric layout of the surrounding) to classify different locations into a set of pre-specified topological places. Examples of such methods are the works of Martínez-Mozos et al. [17, 18] and Friedman et al. [8], where a set of binary classifiers is trained to recognize specific places such as "Room", "Corridor" and "Doorway" in the environment. Binary classifiers are often built by boosting simple geometric features using the AdaBoost algorithm [7], where each simple geometric feature is a numerical value, computed from the observed beams of a laser range scan, or from a polygon representation of the area covered by these observed beams.

Using laser range scans as observations allows these methods to recognize only a certain type of place (e.g., they are not able to distinguish between places with similar geometric structure). Rottmann et al. [24] proposed a method which combines laser range features with visual features to enable the robot to support a greater variety of place categories. Motivated by the fact that typical objects appear at different places with different probabilities, they defined visual features as the number of instances of certain categories of objects (including "Monitor", "Coffee machine", "Office cupboard", "Face" and "Pedestrian") observed in the environment. For this purpose, a fast object detector was built for each of the considered object categories, using the object detection method of Viola and Jones [33]. The visual features along with laser features are used to classify each location in the environment visited by the robot.

Similar to the work of Rottman et al. [24], several other recent approaches to place categorization (e.g., [9, 35] and [31]), are based on the occurrence statistics of different objects in different places. However, these methods often fail to generalize to new environments. This is mainly due to the fact that object detection, for the most part, is still an unsolved problem and cannot reasonably deal with the intra-class variations in the appearance of objects.

Given the difficulties with object based methods, another stream of work suggests that place labels can be estimated from the global configurations in the observed scenes without explicitly detecting and recognizing objects. Such methods can be classified into two general categories: context-based and landmark-based. Amongst context-based methods is that of Oliva and Torralba [20], which uses the Discrete Fourier Transform to encode spectral information of the image. The spectral signals from non-overlapping sub-blocks are then compressed to produce the image representation. Torralba et al. [27] extended this work by using wavelet image decomposition instead of Discrete Fourier Transform to produce more compact and precise image representations. Recently, Wu and Rehg [34] proposed the CENTRIST descriptors which use the Census Transform to capture spatial relations between neighboring pixels. They showed that a spatial hierarchy of such descriptors used with Support Vector Machine (SVM) classifiers performs very well on recognizing a wide variety of places.

In landmark-based approaches (e.g., [12, 21] and [5]), local image features play the main role in scene recognition. Local features characterize a limited area of the image. However, they usually provide more robustness against common image variations (e.g., viewpoint). Among local feature extraction techniques, the Scale Invariant Feature Transform (SIFT) of Lowe [15] has dominated the field. Local image features are usually used for scene recognition within the bag-of-features framework, where only the appearances of features are used and their spatial coordinates are discarded. In this framework, the extracted features from the image are matched to a vocabulary of visual words (each representing a category of

local image features that are visually similar to each other), resulting in a response vector indicating the frequency of each visual word in the image.

Several extensions have been proposed to this basic approach. A group of authors have proposed feature selection techniques to choose the most discriminative visual words for recognition and classification tasks. In [10], three feature selection approaches—namely, the maximization of mutual information [32], odds ratio [1], and linear SVM [16]—have been evaluated for selecting the most discriminative visual words, and the linear SVM is reported as the best one. In [19], visual words that maximally increase recognition performance are iteratively selected. In [11], another information theoretic solution is proposed to address a similar problem through information loss minimization.

Lazebnik et al. [12] proposed a different extension to the bag-of-features method by introducing the spatial pyramid matching method which is based on global geometric correspondence. The method works by partitioning the image into increasingly fine sub-blocks and building a bag-of-feature representation from each sub-block. The local representations are combined in a principled way to produce the image representation.

Although the proposed extensions to the bag-of-features method improve the performance to some extent, they do not address the problem of partial occlusion (i.e., failure to detect some of the expected visual words in the query images) explicitly. This is particularly important in the context of qualitative localization in dynamic environments, where objects (i.e., visual landmarks) could be added to or removed from the environment. In this paper, we present a novel landmark-based algorithm for qualitative localization that explicitly considers the challenges resulting from dynamic changes in the environment (as mentioned above). Unlike the original bag-of-features method, which represents an image by a single global histogram, we use a "spatial pyramid" representation, similar to [12] to also take into account the spatial layout of the image features. The employed pyramid representation subdivides the image into different levels of resolution, and then for each level of resolution and each visual word, aggregates the image features that fall into each spatial bin. These spatial bins are considered as classification features, used within an information theoretic framework to produce hierarchical classifiers for localization.[1] Top-level nodes in the hierarchy are selected from the coarsest level of resolution, based on the *additional* information they can deliver about the class. The child or "backup" nodes, are selected from finer levels of resolution, such that together they can deliver (almost) the same additional information about the class as their parent does. While the higher level nodes encode more holistic statistical properties of the image, the lower level nodes capture more local and detailed statistical characteristics. Parent and child features *together* can then provide a substantial level of robustness against appearance variations resulted from occlusions, variations in viewpoint, lighting, etc. This is because such changes often generate unexpected responses in the higher-level nodes (i.e., present when expected to be absent, and absent when expected to be present), which can be identified and managed when our method seeks clarifying evidence from the "back-up" child nodes.

The remainder of the paper is organized as follows: in Section 2 we describe the different steps of the employed image representation method. Sections 3 and 4 describe our hierarchical learning and classification approaches, and Section 5 presents the implementation details and experimental results. Finally, we conclude the paper and discuss some future work in Section 6.

2 Image Representation

2.1 Feature Extraction

In our method, images are initially represented using the Scale Invariant Feature Transform (SIFT)

[1] Please note that while the hierarchical image representation framework used in our place recognition method is similar to that of Lazebnik et al. [12], our method is different from [12] in that our proposed classification method (which is the main contribution of our work) is also hierarchical, enabling the learned place models to achieve robustness against variations in the dynamic environments. In [12] classification is performed using the standard Support Vector Machine (SVM).

description technique [15]. A 128 dimensional SIFT descriptor represents a local region in the image by a 3D (2 locations and one orientation) histogram of gradient locations and orientations. The contribution of each pixel to the location and orientation bins is weighted by its gradient magnitude. The quantization of gradient locations and orientations makes the descriptor robust to small geometric distortions and certain illumination variations.

In contrast to the traditional use of SIFT descriptors, where local regions detected with some interest point detector are described for image representation, in our method we compute SIFT descriptors on a regular dense grid. As shown in [6], dense features are more suitable for scene classification, particularly because they can capture uniform regions such as walls, road surface, sky, etc. Figure 1 shows the dense SIFT description of a sample image.

2.2 Spatial Pyramid Image Representation

As in the bag-of-features approach, extracted features from images (i.e., the SIFT descriptors of 16×16 pixel patches computed over a grid with spacing of 8 pixels) are quantized into a compact set of *visual words*, built automatically during training. However, unlike the original bag-of-features methods (e.g., [26] and [3]), which represent an image by aggregating the image features into a single global histogram, we use a spatial pyramid approach, similar to that of Lazebnik et al. [12], to also take into account the spatial layout of the image features. This technique works by partitioning the image into increasingly fine sub-regions and computing histograms of local features found inside each sub-region. Therefore, while resolution at which the features are extracted remains fixed, the spatial resolution at which they are aggregated varies at each level. The following subsections describe the different steps of building this spatial pyramid representation.

2.2.1 Visual Words

In our method, visual words are built automatically by grouping visually similar features extracted from the training images, using an agglomerative clustering technique, which guarantees that (i) only visually similar patches are grouped together, and (ii) the resulting clusters are compact [13].

Starting with each feature as a separate cluster, at each iteration, the agglomerative clustering finds the most similar pair of clusters and merges them into one, as long as the average similarity between their constituent members stay above a certain threshold θ:

$$sim(X, Y) = \frac{1}{NM} \sum_{i=1}^{N} \sum_{j=1}^{M} cossim(x^i, y^i) \quad (1)$$

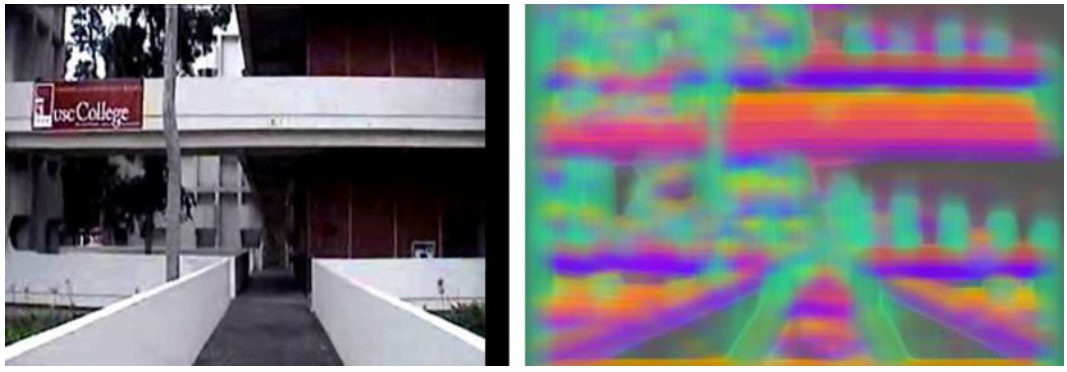

Fig. 1 Visualization of SIFT descriptors of 16×16 patches, computed on a regular dense grid. The visualization is obtained by mapping the first three principal components of each 128 dimension SIFT descriptor into the principal components of the RGB color space [14]. Note that in our experiments SIFT descriptors are computed over a grid with spacing of 8 pixels

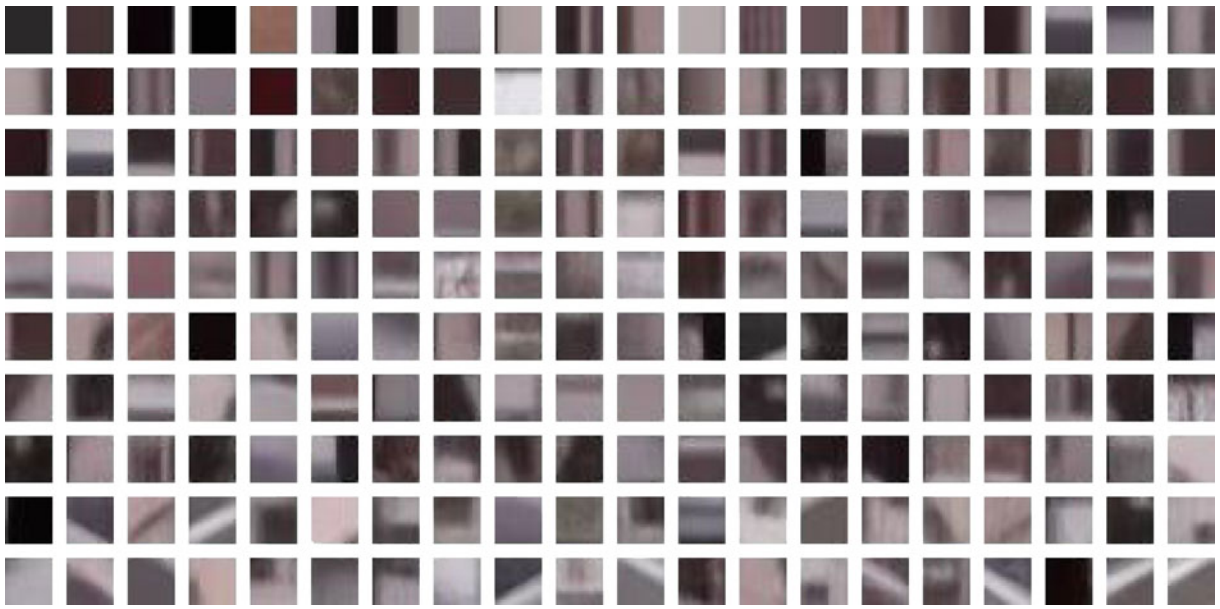

Fig. 2 The vocabulary of visual words resulted from agglomerative clustering of SIFT descriptors of 16×16 patches, randomly extracted from a set of images from the USC-ACB dataset. Each visual word is shown by the average of the image patches, whose SIFT descriptors are grouped together. The average images show the visual compactness of the words

In the above equation, *cossim* (x, y) is the cosine similarity, computed as the cosine of the included angle between x and y, where x and y are the constituent members of clusters X and Y with sizes N and M, respectively.

The main problem with the agglomerative clustering method is its huge memory requirement, as it needs to store an $O(n^2)$ similarity matrix. To solve this problem, we use an efficient implementation for agglomerative clustering similar to [13], based on *reciprocal nearest neighbors*, which reduces the space complexity to $O(n)$.

In our experiments, we experimentally[2] set θ to 0.85 and picked the largest M clusters to form our vocabulary of visual words. Figure 2 shows the vocabulary computed for one of our experiments on an outdoor database (more details in Section 5).

[2]The experiment used to set this parameter (and other parameters that their values are mentioned in the paper) was performed using a portion of the training images used in our reported experiments in Sections 5.3 and 5.4

2.2.2 Spatial Pyramid Construction

Given an image I, X and R are the 2-D location and the 128-D description vectors of the extracted features from the image, respectively. From these data, a spatial pyramid representation is constructed by placing a sequence of increasingly coarser grids over X and taking the histogram of the quantized description vectors, R, corresponding to the points from X that are located at each level of resolution. More specifically, a sequence of grids are constructed at resolution levels $\{0, \ldots, L\}$, such that the grid at level $l \in \{0, \ldots, L\}$ has 2^l cells along each dimension, for a total of 2^{2l} cells. h_c^l denotes the local histogram at level l and grid cell c, where the value of each bin i, i.e., $h_c^l(i)$, is computed by counting all the feature points in the corresponding grid cell, whose descriptors are assigned to the ith visual word. Therefore for L levels and M visual words, the dimensionality of the image representation is $M \sum_{l=0}^{L} 2^{2l}$. In the rest of the paper, for the sake of simplicity, we use h_n for $h_c^l(i)$, where $n = \frac{1}{3}M(4^l - 1) + M(c - 1) + i$.

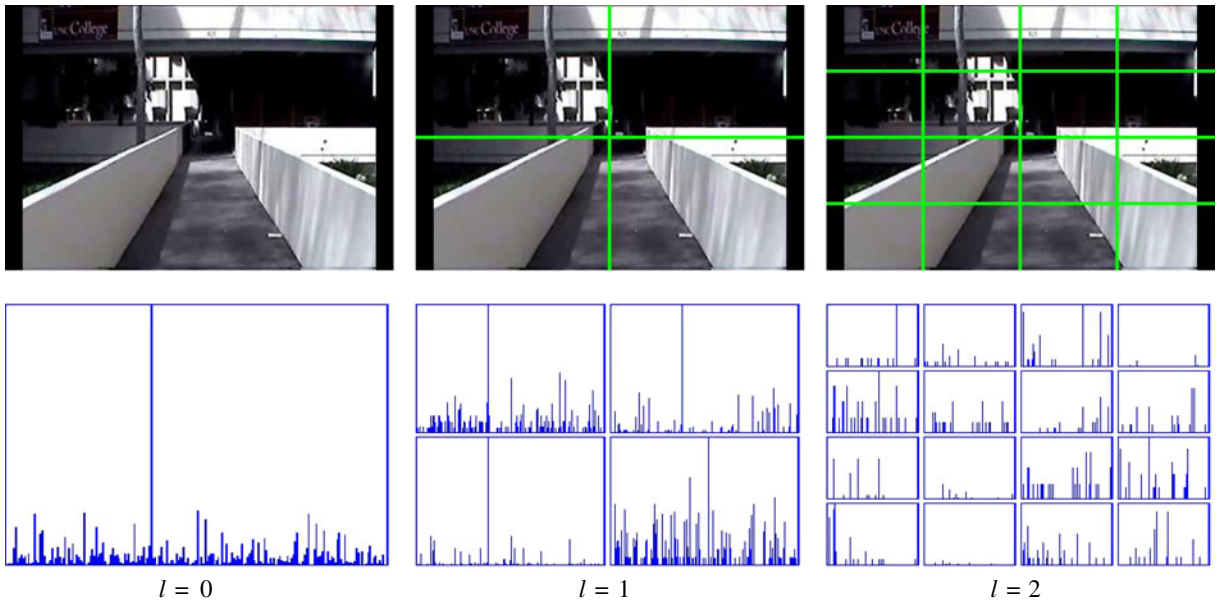

Fig. 3 A schematic illustration of a three-level spatial pyramid representation for a sample image. For $l = 0$, the pyramid consists of just a single global histogram, which is equivalent to the standard bag of features representation. For $l = 1$, and $l = 2$ the image is subdivided into 4 and 16 cells, yielding 4 and 16 histograms, respectively

Figure 3 shows a schematic example for spatial pyramid representation of a sample image.

3 Learning

This section describes our proposed method for learning appearance-based classifiers for different environments.

3.1 Informative Feature Selection

Each bin of the spatial histograms in the pyramid representation can act as a binary classifier, firing when its value (i.e., the number of image features that fall in that spatial bin) is above a threshold, and not firing otherwise:

$$f_n(I, \theta_n) = \begin{cases} 1, & if\ h_n \geq \theta_n \\ 0, & otherwise \end{cases} \tag{2}$$

Here f_n is a binary variable and θ_n is the threshold associated with h_n (in our experiments, $\theta_n = 0$ for all features, since the goal is to use the presence or absence of features in the images of different classes). Given a collection of binary features, the task of learning can be formulated as selecting and combining the appropriate features that can best separate the positive training samples from the negative ones. To this aim, a binary variable $C(I)$ is used to represent the class, where $C(I) = 1$ if the image I belongs to the class, and 0 otherwise.

The discriminative value of each feature is measured by the amount of mutual information it can deliver about the class [4]:

$$I(f_n; C) = H(C) - H(C|f_n) \tag{3}$$

In the above equation, $I(f_n; C)$ is the mutual information between feature f_n and class C, and H denotes entropy. Informative feature selection starts by identifying the feature with the highest mutual information score. It then proceeds by iteratively searching for the next informative feature, f_r, that delivers the maximal amount of additional information with respect to each of the previously selected features:

$$f_r = \arg\max_{f_k \in K_r} \min_{f_j \in S_r} \left(I(f_k, f_j; C) - I(f_j; C) \right) \tag{4}$$

Here K_r and S_r are the set of features not yet selected, and the set of features already selected at iteration r, respectively.

The feature selection process ends when the increment in mutual information gained by selecting a new feature is less than a certain threshold (experimentally set to 0.02), or until the number of selected features reaches a certain limit (experimentally set to 30).

3.2 Feature Hierarchy Construction

Features selected by the above method (called *top-level* features) are often strong enough to discriminate the positive and negative training images of an environment with 100% accuracy. However, it is unrealistic to expect all (or even the majority) of these features to act similarly in the test images. This could be due to the changes in the structure of the environments (e.g., some objects are removed or added), variation in lighting conditions, or substantial view-point changes. To address these difficulties, in [5], we proposed a method which uses the redundancy between the features to select for each top-level feature, a set of *child* features that provide similar information as their parents, complementary to information provided by other top-level features. These child features act as 'back-up' features for their parent, standing in for the parent feature if for some reason it is missing. Iteratively applying this principle, leads to feature hierarchies that are robust to variations that are not present in the training dataset.

In that work, we used the standard bag-of-features method (i.e., single histogram of visual words) for image representation and all parent features and their children were selected from the same level of complexity and resolution (i.e., image frame). While histogram statistics at coarse levels of resolution often provide additional information about the class, they involve increasingly dissimilar features [12].

In the present work, we take advantage of our spatial pyramid representation of images to build the feature hierarchies, selecting the child features from increasingly finer resolutions. Therefore, while the top-level informative features are still selected from the coarsest level of resolution, capturing the *holistic* statistical properties of the images (which has been proven to be surprisingly effective for categorizing the scenes [20]), the children or backup features are selected from finer level of resolution, capturing more local and precise statistical characteristics. Parent and child features *together* can then provide a substantial level of robustness against appearance variations, as is shown in Section 5.

In order to identify the child features of a selected feature, f_m, at level l, the original positive training set is replaced by the training samples for which the selected feature fires (i.e., $f_m = 1$), and the original negative training set is replaced by the training samples for which the selected feature does not fire (i.e., $f_m = 0$). Furthermore, the set of candidate features, K, is initialized by features corresponding to the 2^2 spatial histograms at the resolution level $l + 1$ that overlaps the grid cell of f_m. The goal is then to find a combination of features from K that can (almost) perfectly mimic the action of f_m. This can be done by applying the same information maximization procedure that was used at the higher level,[3] replacing C in Eqs. 3 and 4 with f_m.

This process of hierarchical feature selection continues recursively, until a pre-defined level L (equal to the number of levels in the pyramid representation) is reached. Features with no children are then labeled as *atomic* features.

In the feature hierarchy, the response of each non-atomic node, f_n, indicated by s_n, is computed based on the combination of its children responses, and its own binary response (as computed by Eq. 2):

$$s_n = \left(\left(w_0 + \sum_{i=1}^{m} w_i s_{ni}\right) > 0\right) \wedge f_n(I, \theta_n) \qquad (5)$$

Here, s_{ni} is the binary response of the ith child of the node, m is the number of children, and w_0 and w_i are the bias and weights of the combination, respectively. Once the hierarchy is built, w_0 and w_i

[3] The feature selection process for lower-level nodes ends when the increment in mutual information gained by selecting a new feature is less than 0.02, or until the number of selected features reaches a threshold of 6. These thresholds are determined experimentally.

are computed for every non-atomic parent node, f_n, using the following equations:

$$w_i = \frac{1}{|T_p|}\sum_{j\in T_p} s_{ni}(j) - \frac{1}{|T_n|}\sum_{j\in T_n} s_{ni}(j) \tag{6}$$

$$w_0 = \frac{1}{2}\left(\frac{1}{|T_p|}\sum_{j\in T_p}\sum_{i=1}^{m} w_i s_{ni}(j) + \frac{1}{|T_n|}\sum_{j\in T_n}\sum_{i=1}^{m} w_i s_{ni}(j)\right) \tag{7}$$

In the above equations, T_p and T_n are the positive and negative training sets associated with the parent node, respectively.

To determine the final response of the classifier, a *root* node is assumed for the hierarchy where the top-level features are considered as its children. The bias, w_0, and the weights, w_i, of the top-level features are then computed using Eqs. 6 and 7. The response of the root node, corresponding to the entire class, is then computed using the following equation (which is derived from Eq. 5):

$$s_r = w_0 + \sum_{i=1}^{m} w_i s_{ri} \tag{8}$$

If s_r for a test image is positive, the image is classified as positive.

4 Multi-Class Classification

Our learning algorithm is designed for binary classification. To apply it to multi-class localization, we use pairwise class binarization technique, in which one classifier is learned for each pair of classes, using only the training examples for these two classes and ignoring the training samples of other classes. This leads to $N(N-1)/2$ binary classifiers, where N is the number of classes. To learn a binary classifier for two classes C_i and C_j, the training samples of C_i are considered as the positive training set and the training samples of C_j are considered as the negative training set. Subsequently, if for a test image the learned classifier responds positively, the test image belongs to C_i, otherwise it belongs to C_j.

To decode the predictions of the pairwise classifiers to a final prediction, a simple voting technique is used, where for a given test image, each learned classifier casts a vote for one of its two classes, and in the end, the test image is assigned to the class with the highest number of votes.

The number of votes for the selected class can be used as a measure of certainty (or confidence) for the classification result. When the task is to recognize N places, we expect the selected class to be supported by $(N-1)$ votes from the pairwise binary classifiers. Classification results with less than $(N-1)$ votes are error-prone, or depending on the experimental setup, might correspond to sensory data (i.e., images) acquired from "unknown" places. As discussed in Section 5.3, a large portion of the test cases misclassified by our method, have lower than expected support from the pairwise binary classifiers (i.e., the selected class has less than $(N-1)$ votes).

5 Experiments

5.1 Databases

To evaluate our method, we use two publicly available databases for place recognition and robot localization. The first database, provided by Siagian et al. [25], was created from three outdoor sites on the University of Southern California campus, including the ACB site (a rigid and less spacious man-made environment), the AnF site (comprised of two adjoining parks) and the FDF site (largely open area). Each site is manually divided into 9 continuous segments/places, resulting in 27 topological places in total. We use this database for the majority of our experiments, including the evaluation of our method for topological localization in changing outdoor environments and the examination of the scalability power of our approach.

In order to evaluate the generalization performance of our method, we use a second database, called COLD (CoSy Localization Database) [22], consisting of data sequences acquired in three different indoor laboratory environments with places of common functionality. For each place in each environment, multiple image sequences

were captured over several days, under various illumination conditions. In our experiments, only a subset of this database is used.

5.2 Model Parameters

The accuracy of our proposed model for place recognition and topological localization depends on few major parameters, namely the size of the vocabulary of visual words, the number of layers in spatial pyramid representation and the grid parameters for dense sift computation (i.e., spacing and scale). For our experiments, typical candidate vocabulary sizes are $M = 200$, $M = 300$, and $M = 400$; spatial pyramid representations are computed up to level 4; and possible parameters for computing dense sift descriptors are 8 and 16 pixels for both scale and spacing.

In each experiment two sets of training images are used for each place, one as a *model construction set* and the other one as a *validation set*. A model is learned for every combination of possible parameters using the model construction set, and then the validation set is used to get a performance estimate for the learned models. The model with the highest performance on the validation set (which corresponds to the most optimal tuning of parameters) is selected and applied to the test sequences.

While the selected parameters for each experiment are reported in the corresponding sections, a point worth noting here is that despite dissimilar characteristics of different environments, relatively similar combinations of model parameters are selected in all experiments. SIFT descriptors of 16×16 pixel patches (i.e., scale of 8 pixels) computed over a grid with spacing of 8 pixels provide the best performance in all experiments (these values are consistent with the findings of Fei-Fei et al. [6] and Lazebnik et al. [12]). Similarly, hierarchical classifiers in all experiments show a relatively similar behavior in performance versus the number of levels in the hierarchies (corresponding to the number of levels in spatial pyramid image representation). Figure 4 illustrates this behavior in our experiment on the ACB outdoor database. As can be seen, the performance improves dramatically as we go from $L = 0$ to a multi-level setup, i.e., the addition of even just one layer of spatial processing to the flat classifiers of $L = 0$, provides a great degree of robustness against visual variations. While the performance continues to improve from $L = 1$ to $L = 2$, the performance remains essentially identical for $L = 3$, which might be due to the fact that the highest level (i.e., $l = 3$) in $L = 3$ pyramid is too finely subdivided, with spatial bins yielding no additional information about their parents in the coarser level (i.e., $l = 2$).

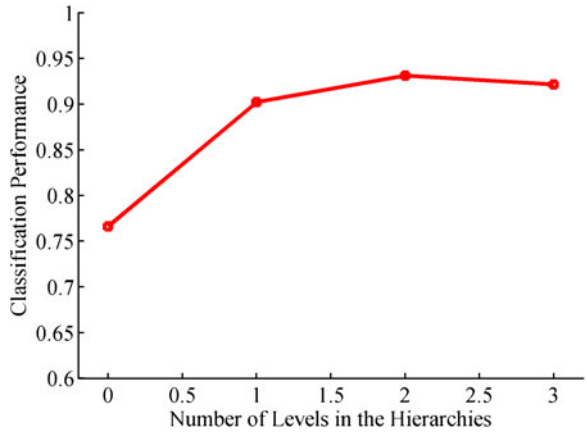

Fig. 4 Classification accuracy versus the number of levels in the hierarchies (corresponding to the number of levels in spatial pyramid image representations). The figure resulted from an experiment on the ACB database

Furthermore, there are several observations in the behavior of our feature selection method that are consistent among all experiments. First, the increment in the mutual information gained by selecting new features (in the top level) decreases as feature selection proceeds (this is shown for an experiment on the ACB database in Fig. 5). Second, similar to the gained mutual information, the increment in the classification performance also decreases as the features selection process selects and adds new top-level features to classifiers (Fig. 6 illustrates this behavior for our experiment on the ACB outdoor database). Third, the feature selection process for top-level nodes often terminates when in average 28 features are selected (after that the increment in mutual information gained by selecting a new feature becomes too small, i.e., bellow the predefined threshold). For each top-level feature, often 3 or 4 backup features (selected from the lower levels of

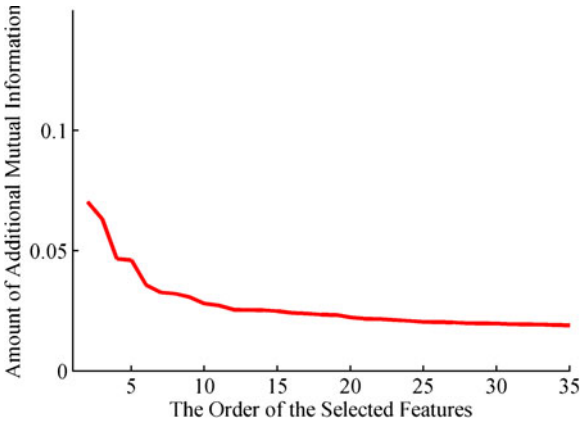

Fig. 5 Shows how the increment in the mutual information gained by selecting new features (in the *top* level) decreases as feature selection proceeds. The figure resulted from an experiment on the ACB database

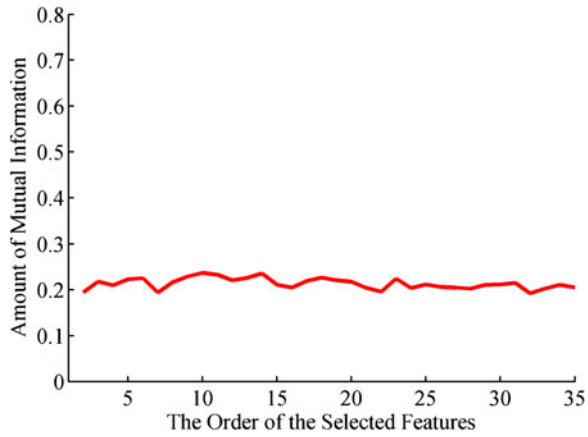

Fig. 7 Shows the amount of mutual information each selected *top-level* node carries about the class in an experiment on the ACB database

representation) are sufficient to provide the same amount of mutual information about the class as the top-level node does (Fig. 7 shows the amount of mutual information each selected top-level node carries about the class in our experiment on the ACB database).

The visual vocabulary size is the only changing parameter in our experiments. While in Experiment 2 (on the combination of three outdoor databases), a vocabulary of 300 visual words yields the highest performance, for the remaining experiments a 200 word vocabulary seems to be discriminative enough. This can be explained by the fact that in Experiment 2, the task is to distinguish between 27 places, while in the other experiments the localization is performed over 5–9 places. Thus, we can conclude that as the number of classes increases, larger visual vocabularies might be required to provide more discriminative power.

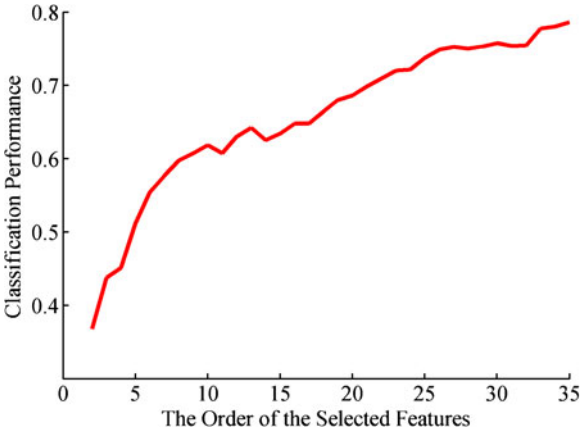

Fig. 6 Shows how the increment in the classification performance also decreases as the features selection process selects and add new *top-level* features to classifiers. The figure resulted from an experiment on the ACB database

5.3 Experiment 1: Localization in Changing Outdoor Environments

We use the three outdoor sites of the USC database (described in Section 5.1) for this experiment, each presenting certain challenges to localization and place recognition. In the ACB site, the places are all part of hallways. Therefore, the surroundings are often flat walls with little texture and solid lines that delineate the walls and different parts of the buildings. The places comprising the AnF site are all part of campus parks, overwhelmingly dominated by vegetations. Large areas of the images acquired from this site are indistinguishable, as leaves overrun most regions. Finally, in the FDF site, the places are part of an open area, where a large portion of the scenes is the sky, mostly textureless, with clouds of random light.

Each site is manually divided into 9 continuous segments/places. Figure 8 shows a sample image from each of the 27 topological places. For each

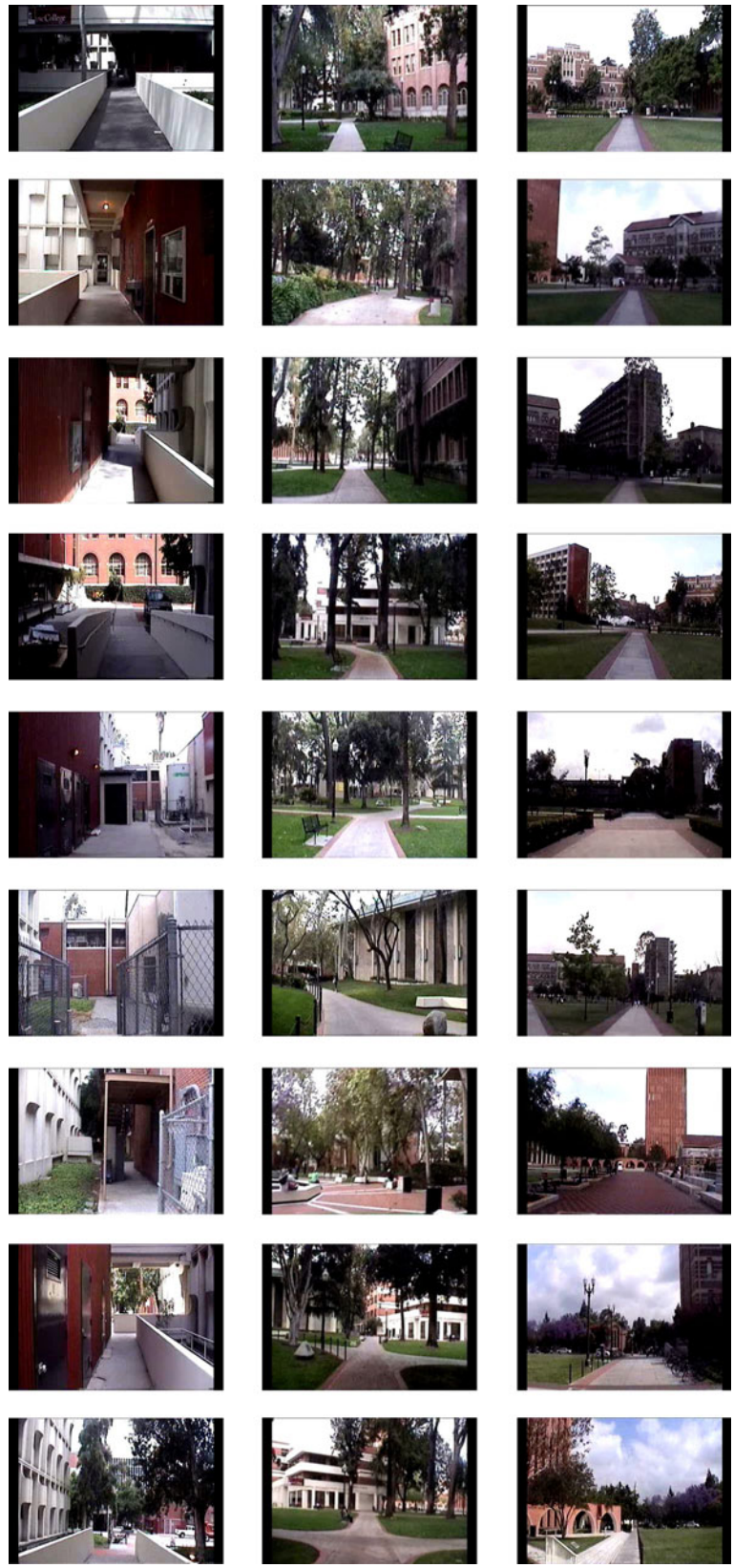

Fig. 8 Sample images from each of the nine segments/places of the three outside sites of the USC database (from *left* to *right*: ACB, AnF, FDF)

Table 1 The performance (%) of our proposed topological localization method, on the USC-ACB database, in comparison to the performance of other methods

Methods	Training sequences	Performance				
		Trial 1 (3583)	Trial 2 (3549)	Trial 3 (3457)	Trial 4 (3377)	Average (13966)
Bag-of-features	2	78.85	76.40	74.26	78.75	77.07
Bag-of-features	8–11	83.02	79.83	78.36	81.11	80.59
Our method	2	93.72	92.14	88.60	92.18	91.68
Our method	8–11	94.83	93.91	90.73	93.03	93.15
Siagian et al. [25]	8–11	91.04	89.18	85.22	86.23	87.96

The number of images in each testing trial is shown in parentheses

place, 12 to 15 image sequences are provided, capturing different lighting conditions, small viewpoint variations and some structural changes (e.g., benches temporarily removed from the parks, service vehicles or cars temporarily parked, storage boxes temporarily placed in different places, etc.). The standard protocol for experimenting on this dataset is to use 8–11 image sequences from each place to train the models and the remaining 4 image sequences (taken on separate days and various lighting conditions) for testing [25]. In addition to an experiment that follows the standard protocol, we perform another experiment with much smaller sets of training data (i.e., two to be particular) to further investigate the extent to which our method is successful in dealing with variations that are present in the test data but not the training data. Note that in both experiments, half of the training data is used for model construction and the other half is used for validation (refer to Section 5.2 for more details). The testing sequences in both experiments are identical to those used by Siagian et al. [25]—the standard protocol.

Tables 1, 2 and 3 and Fig. 9 show the performance of our method on the three outdoor sites of the USC database, in comparison to the performance of an inhouse implementation of the standard bag-of-features technique (similar to the method of Dance et al. [3]) and the results reported in [25].

In our implementation of the standard bag-of-features technique, we represent images by their local (and sparse) distinctive features extracted and described using the Scale Invariant Feature Transform (SIFT) technique of Lowe [15]. Extracted features from training images are quantized with a set of compact visual words, built automatically using the clustering technique described in Section 2.2.1 However, rather than selecting the largest M cluster to build the visual words, here we choose clusters with at least m members (where m is 10 in our experiments).[4] Each image is described by a description vector indicating the frequency of each visual word in the image. Support Vector Machine (SVM) [30] is used for multi-classification using the one-versus-all rule: a classifier is trained to separate each class from the rest and a test image is assigned to the class whose classifier returns the highest response.[5]

In the method of Siagian et al. [25], each place is learned from several image sequences, at least couples per each lighting condition, to assure a wide range of testing conditions (see Fig. 10 for examples of appearance variations caused by changes in the lighting condition). Comparing the performance of our method, when trained with images of a single appearance condition, with that of the method Siagian et al. can indicate to what extent our method is successful in dealing with variations in the test data.

[4]Small values for m result in too many clusters (i.e., visual words) which often are not representatives of distinctive world landmarks. On the other hand, large values for m result in small number of clusters that may not be visually compact. In an initial experiment using a portion of the training images used in our reported experiments in Sections 5.3 and 5.4, we found that $m = 10$ provides a good trade-off.

[5]The LIBSVM tool [2] is used in our experiments. All SVM parameters are set to the default values suggested by the authors of the LIBSVM.

Table 2 The performance (%) of our proposed topological localization method, on the USC-AnF database, in comparison to the performance of other methods

Methods	Training sequences per place	Performance				
		Trial 1 (6006)	Trial 2 (6667)	Trial 3 (7018)	Trial 4 (6706)	Average (26397)
Bag-of-features	2	76.53	75.53	67.51	81.41	75.12
Bag-of-features	8–11	80.14	78.39	71.64	83.15	78.20
Our method	2	90.97	87.33	82.07	94.23	88.51
Our method	8–11	93.09	88.68	84.89	95.61	90.44
Siagian et al. [25]	8–11	86.50	83.20	78.75	88.86	84.21

The number of images in each testing trial is shown in parentheses

The performance of the standard bag-of-features method (reported in Tables 1, 2 and 3), which is the base of our proposed method, is significantly lower than the performance of the method of Siagian et al. [25]. This highlights the major challenges posed by visual variations that are not present in the training dataset and often result in misclassification, due to bad or misleading behavior of learned features.

According to Tables 1, 2 and 3, the performance of our proposed method on all the three outdoor sites is substantially higher than the performance of other evaluated methods. As a matter of fact, our method trained with images from only a single appearance condition still outperforms the method of Siagian et al. [25] and the standard bag-of-features method that were trained using several image sequences for each place. This validates the advantages of our solution in dealing with dynamic changes in the environment. Changes (e.g., objects added to or removed from the environment, changes in lighting conditions, etc.), that often generate unexpected responses in the higher-level nodes, are identified when the algorithm seeks clarifying evidence from the "back-up" child nodes. These detected anomalies in the responses of higher level nodes are taken into account (Eq. 5) for final classification, allowing the system to respond correctly.

As Tables 1, 2 and 3 show, both our method and the traditional bag-of-features method, perform better when they are trained with 8–11 image sequences, compared to being trained with only 2 image sequences. However, the difference in the performance is much smaller for our method. This is mainly because of the fact that our method tries to explicitly account for variations that are not present in the training data (due to the lack of training images that cover a wide range of appearance conditions). In effect, our hierarchical method prevents over-learning of specific features, distributing inference over features that may appear redundant in the training set, but may not be redundant in the test set. As a result, the performance of the proposed method degrades gracefully when faced with smaller set of training images.

Studying the results, we observed that the majority of the classification errors of our method occurred during the transitions between different places/segments, specifically those adjacent places with the possibility of overlapping scenes (e.g., the

Table 3 The performance (%) of our proposed topological localization method, on the USC-FDF database, in comparison to the performance of other methods

Methods	Training sequences per place	Performance				
		Trial 1 (8823)	Trial 2 (8118)	Trial 3 (8815)	Trial 4 (8955)	Average (34711)
Bag-of-features	2	68.47	76.78	70.75	76.08	72.96
Bag-of-features	8–11	74.11	82.13	76.25	79.13	77.82
Our method	2	88.03	94.52	90.05	89.12	90.34
Our method	8–11	90.01	95.72	92.44	91.67	92.39
Siagian et al. [25]	8–11	86.24	92.05	88.22	88.25	88.62

The number of images in each testing trial is shown in parentheses

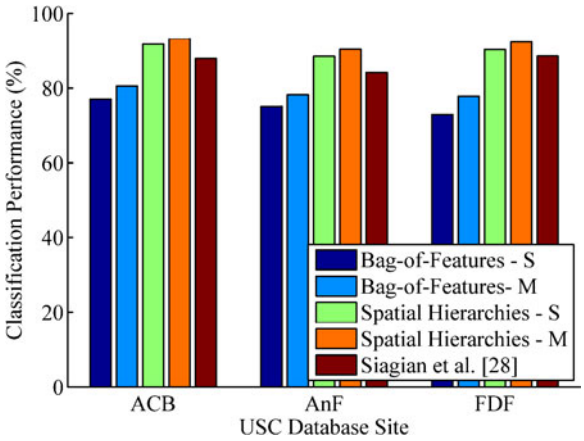

Fig. 9 Average classification performance of our method on different sites of the USC topological place recognition database. 'S' refers to training with a single appearance condition and 'M' refers to training with multiple appearance conditions. The results reported in [25] are based on training with multiple appearance conditions

first and second segments of the ACB site). The remaining errors are mainly due to significant variations in the condition of the environment, especially changes in lighting conditions, which severely influence the SIFT descriptors and subsequent processes, leading to misclassification. Analyzing the number of votes supporting the resulted classification of each test case, we observed that in average, 42.27% of the misclassified test cases have less that 8 votes (which is the expected number of votes for the selected class, when the number of classes in 9). This ratio is only 3.35% for the correctly classified test cases. Therefore, as pointed out in Section 4, the number of votes for the classification of a test case can be used as a confidence factor, to allow the system to predict the certainty in the topological localization of the robot or recognition of a place in the environment.

Our experiments reported in this section, were performed on a PC with a 2.4 GHz CPU. The most time consuming process in our localization system is the image representation (including the extraction of image features, and building the spatial pyramid), which takes around 0.8 s for each image. Given the image representation, recognition is performed extremely fast, in just 15–17 milliseconds, depending on the number of nodes examined in each hierarch classifier.

5.4 Experiment 2: Scalability Performance

Considering that in all the experiments performed in the previous section, the goal of topological localization was to recognize nine different places (with significant similarity in appearance), a natural question is to what extent system performance will degrade and classification time will increase in larger environments with more places. To answer this question we investigate the scalability performance of our method by combining scenes from all the three sites of the USC database and training our method to classify twenty seven different places. Same training and testing image sequences are used as the previous experiments. The selected values for model parameters are the same as the previous experiment, except the vocabulary size which is increased from 200 to 300.

In this experiment, our method achieved a performance of 88.9%, classifying 75,074 scenes belonging to 27 places. The performance of our method, trained on 10,800 images (5400 images for model construction and 5400 images for validation) is superior to the 86.45% classification accuracy of [25] trained on 175,406 images. The successful classification of 88.9% of test cases belonging to 27 places, which is slightly less than the

Fig. 10 Sample scenes from the USC database under different lighting conditions. Each image is taken from one of the four testing sequences

89.95% average classification accuracy achieved for the individual experimental sites with 9 places, indicate that the performance of our method degrades gracefully in larger environments with more places.

The increase in the size of visual word vocabulary affects the computational time of image representation by 0.6 second. The increase in the number of classes, improves the recognition time of each test case to 25.2 milliseconds, which is still neglectable considering the relatively high computational time of image representation.

5.5 Experiment 3: Generalization Performance

As mentioned in Section 5.2, although a separate model parameter tuning was performed for each of the three sites of the USC database, similar combinations of model parameters were selected, which is an initial indication of the promising generalization performance of our method. In this section, we investigate this in more depth by performing some preliminary experiments on an indoor environment, using the same algorithm parameters chosen for the outdoor sites in Experiment 1. To this aim, two image sequences from the Freiburg site of the COLD database, acquired at different lighting conditions, were selected, one for training and one for testing. From the training sequence, 100 images for each of the 5 places were used to compute the vocabulary of visual words and learn the hierarchical classifiers. The learned classifiers were then applied to 1911 images of the testing sequence and achieved a classification performance of 89.1%. Comparing this to the 78.57% $\pm$ 9% classification accuracy achieved by Ullah et al. [29], as a state-of-the-art indoor localization and place recognition method, indicates the capability of our method to be applied to new environments with no calibration of the model parameters.

6 Conclusions

In this paper, a novel technique for topological robot localization was proposed, which combined the advantages of spatial pyramid representation with hierarchical learning to achieve robustness against dynamic changes in the environments. Experiments on a challenging localization database validated the effectiveness of our hierarchical learning in dealing with dynamic variations in the environment. More specifically, even when almost 75% of the testing data were from appearance conditions different from that of the training data used to build our classifiers, our method still managed to outperform other compared methods which were trained on a wide range of appearance conditions (including those of the testing data).

Experiments specifically designed to evaluate the scalability and generalization performance of our method, revealed that the performance of the system degrades gracefully in larger environments with more places, and that with no parameter calibration, the system still performs well in novel environments.

There are a number of potential directions for future work. The temporal continuity of the image frames in topological place recognition imposes the constraint that the computed labels should vary smoothly along the robot's trajectory almost everywhere while preserving discontinuities at the borders between adjacent visited places in the environment. A number of solutions (e.g., [23, 24, 34]) have been proposed to use this as a major source of additional information for improving the initial recognition results in light of contextual cues. For example, Wu et al. [34] and Ranganathan [23] reported improved performance with methods that enforce spatial smoothness. Given these results, we can safely expect that the performance of our method too will be improved by taking advantage of contextual information. While this was out of the scope of this paper, it certainly suggests a potential direction for future work.

Another direction for future work is to extend our place recognition method to place categorization. Place categorization aims at enabling the robot to classify different locations of a new environment into a set of pre-specified categories (e.g., "living room", "kitchen" and "bedroom" for indoor environments) relating them to human-understandable concepts. Place categorization is a more difficult problem than place recognition, in that it requires robustness to intra-class variations as well.

Finally, we would like to experiment with our place recognition method in the context of the Playbot autonomous wheelchair project [28]. Playbot is a visually guided robotic wheelchair, designed in the Laboratory of Active and Attentive Vision at the York University, for children living with mobility impairments to improve the quality of life for them, by allowing them to be more independent. This, as a real-world application, helps us to better understand the strength and weaknesses of our method and facilitates the transfer of technology to industry.

Acknowledgements The authors are grateful for support from the Natural Science and Engineering Council of Canada. The third author also acknowledges support of the Canada Research Chairs Program.

References

1. Brank, J., Grobelnik, M., Milic-Frayling, N., Mladenic, D.: Interaction of feature selection methods and linear classification models. In: Proceedings of the International Conference on Machine Learning, Workshop on Text Learning (2002)
2. Chang, C., Lin, C.: LIBSVM: a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm (2001)
3. Dance, C., Willamowski, J., Fan, L., Bray, C., Csurka, G.: Visual categorization with bags of keypoints. In: Proceedings of the European Conference on Computer Vision, International Workshop on Statistical Learning in Computer Vision (2004)
4. Epshtein, B., Ullman, S.: Feature hierarchies for object classification. In: Proceedings of the International Conference on Computer Vision (2005)
5. Fazl-Ersi, E., Elder, J.H., Tsotsos, J.K.: Hierarchical appearance based classifiers for qualitative spatial localization. In: Proceedings of the International Conference on Intelligent Robots and Systems, pp. 3987–3992 (2009)
6. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 524–531 (2005)
7. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. In: Proceedings of the Second European Conference on Computational Learning Theory, pp. 23–37 (1995)
8. Friedman, S., Hanna, P., Fox, D.: Voronoi random fields: extracting topological structure of indoor environments via place labeling. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 2109–2114 (2007)
9. Galindo, C., Saffiotti, A., Coradeschi, S., Buschka, P., Fernandez-Madrigal, J.: Multi-hierarchical semantic maps for mobile robotics. In: Proceedings of the International Conference on Intelligent Robots and Systems, pp. 2278–2283 (2005)
10. Jurie, F., Triggs, B.: Creating efficient codebooks for visual recognition. In: Proceedings of the International Conference on Computer Vision, pp. 604–610 (2005)
11. Lazebnik, S., Raginsky, M.: Supervised learning of quantizer codebooks by information loss minimization. IEEE Trans. Pattern Anal. Mach. Intell. **31**(7), 1294–1309 (2009)
12. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 2169–2178 (2006)
13. Leibe, B.: Interleaved object categorization and segmentation. PhD Thesis, ETH, Zurich (2004)
14. Liu, C., Yuen, J., Torralba, A., Sivic, J., Freeman, W.: SIFT Flow: dense correspondence across difference scenes. In: European Conference on Computer Vision (2008)
15. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
16. Mladenic, D., Brank, J., Grobelnik, M., Milic-Frayling, N.: Feature selection using linear classifier weights: interaction with classification models. In: Proceedings of the Annual ACM Conference on Research and Development in Information Retrieval, pp. 234–241 (2004)
17. Martínez-Mozos, O., Stachniss, C., Burgard, W.: Supervised learning of places from range data using Adaboost. In: Proceedings of the International Conference on Robotics and Automation (2005)
18. Martínez-Mozos, O., Burgard, W.: Supervised learning of topological maps using semantic information extracted from range data. In: Proceedings of the International Conference on Intelligent Robots and Systems, pp. 2772–2777 (2006)
19. Nowak, E., Jurie, F.: Vehicle categorization: parts for speed and accuracy. In: Proceedings of the International Conference on Computer Vision, VS-PETS workshop (2005)
20. Oliva, A., Torralba, A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. Int. J. Comput. Vis. **42**(3), 145–175 (2001)
21. Pronobis, A., Caputo, B., Jesfelt, P., Christensen, H.I.: A discriminative approach to robust visual place recognition. In: Proceeding of the International Conference on Robots and Systems, pp. 3829–3836 (2006)
22. Pronobis, A., Caputo, B.: COLD: COsy localization database. Int. J. Rob. Res. **28**(5) (2009)
23. Ranganathan, A.: PLISS: detecting and labeling places using online change-point detection. In: Proceedings of the Robotics: Science and Systems (2010)
24. Rottmann, A., Martínez-Mozos, O., Stachniss, C., Burgard, W.: Semantic place classification of indoor environments with mobile robots using boosting. In:

Proceedings of the National Conference on Artificial Intelligence, pp. 1306–1311 (2005)
25. Siagian, C., Itti, L.: Rapid biologically-inspired scene classification using features shared with visual attention. EEE Trans. Pattern Anal. Mach. Intell. **29**(2), 300–312 (2007)
26. Sivic, J., Zisserman, A.: Video Google: a text retrieval approach to object matching in videos. In: Proceedings of the International Conference on Computer Vision (2003)
27. Torralba, A., Murphy, K.P., Freeman, W.T., Rubin, M.A.: Context-based vision system for place and object recognition. In: Proccedings of the IEEE International Conference on Computer Vision, p. 273 (2003)
28. Tsotsos, J., Verghese, G., Dickinson, S., Jenkin, M., Jepson, A., Milios, E., Nu?o, F., Stevenson, S., Black, M., Metaxas, D., Culhane, S., Ye, Y., Mann, R.: PLAYBOT: a visually guided robot to assist physically disabled children in play. Image Vis. Comput.: Special Issue on Vision for the Disabled **16**(4), 275–292 (1998)
29. Ullah, M., Pronobis, A., Caputo, B., Luo, J., Jensfelt, P., Christensen, H.: Towards robust place recognition for robot localization. In: Proceedings of the International Conference on Robotics and Automation (2008)
30. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
31. Vasudevan, S., Gachter, S., Nguyen, V.T., Siegwart, R.: Cognitive maps for mobile robots—an object based approach. Robot. Auton. Syst. **55**(5), 359–371 (2007)
32. Vidal-Naquet, M., Ullman, S.: Object recognition with informative features and linear classification. In: Proceedings of the International Conference on Computer Vision, pp. 281–288 (2003)
33. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, pp. 511–518 (2001)
34. Wu, J., Rehg, J.M.: CENTRIST: a visual descriptor for scene categorization. IEEE Trans. Pattern Anal. Mach. Intell. **33**(8), 1489–1501 (2011)
35. Zender, H., Martínez-Mozos, O., Jensfeltc, P., Kruijffa, G., Burgard, W.: Conceptual spatial representations for indoor mobile robots. Robot. Auton. Syst. **56**(6), 493–502 (2008)