The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

# Hierarchical Appearance-Based Classifiers for Qualitative Spatial Localization

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

*Abstract*—**This paper presents a novel appearance-based technique for qualitative spatial localization. A vocabulary of visual words is built automatically, representing local features that repeatedly occur in the set of training images. An information maximization technique is then applied to build a hierarchical classifier for each environment by learning informative visual words. Child nodes in this hierarchy encode information redundant with information coded by their parents. In localization, hierarchical classifiers are used in a top-down manner, where top-level visual words are examined first, and for each top-level visual word which does not respond as expected, its lower-level visual words are examined. This allows inference to recover from missing features encoded by higher-level visual words. Several experiments on a challenging localization database demonstrate the advantages of our hierarchical framework and show a significant improvement over the traditional bag-of-features approaches.**

## I. Introduction

ONE of the fundamental requirements for an autonomous mobile agent is localization, i.e., the capability of knowing where it is located within its world. Agents 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 agent (e.g., topological maps), or learned while the agent discovers its surroundings (e.g., metric maps).

There are two types of localization: qualitative and quantitative. Qualitative localization gives the agent the capability of recognizing the environments, but not the ability to estimate a precise metrical pose. On the other hand, quantitative localization provides the agent with the capability to estimate its exact pose (i.e., position and heading) 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.

Vision sensors have advantages over laser, ultrasonic and sonar range finders, for the purpose of localization. Cameras are information-rich, relatively inexpensive and easily available. However, visual localization is very challenging, since: (i) the visual appearance of scenes varies significantly with changes in lighting conditions; (ii) objects may be added to or removed from the environment, which can change the visual appearance remarkably from the training time; and (iii) the same scene may look very different from a different viewpoint.

A large number of qualitative localization approaches have been developed and refined in recent years, which can be classified into two categories: context-based (global) techniques and landmark-based approaches.

Amongst context-based methods, which usually use global image features such as color (e.g., [3]), texture (e.g., [4]), or both (e.g., [5]), one influential approach is that of Torralba et al. [6], introducing a low-dimensional global image representation called an *image gist*, encoding the textural properties of the image and their coarse spatial layout. Image gist has been used widely in many localization methods, either solely (e.g., [7]) or together with other techniques (e.g., [8]), and has provided very good localization results.

In landmark-based approaches, such as [10] and [26], local image features play the main role in localization. Unlike global features, local features characterize a limited area of the image. However, they usually provide more robustness against common image variations (e.g., illumination, occlusion, etc.). Among local feature extraction techniques, the Scale Invariant Feature Transform (SIFT) [9] has been used in many localization systems (e.g., [8], [10]). Other examples are Kernel PCA features [11] and MSER features [12].

Local image features are usually used for localization within a bag-of-features framework [2], where only the appearance of features are used and their spatial coordinates are discarded. In this framework, the extracted features from the query image are matched to a vocabulary of visual words (i.e., each representing a category of local image features that are visually similar to each other), resulting in a response vector indicating the presence/absence or the frequency of each visual word in the query image. The response vectors are then used as classification keys for training and recognition.

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 [19], three feature selection approaches - namely, the maximization of mutual information [20], odds ratio [21], and linear SVM [22] - have been evaluated for selecting the most discriminative visual words, and the linear SVM is reported as the best one. In [23] visual words are iteratively selected that maximally increase the recognition performance. More

E. Fazl-Ersi, J.H. Elder and J.K. Tsotsos are with the Department of Computer Science and Engineering, York University, Toronto, ON, M3J 1P3, Canada. E-Mail: {efazl, elder, tsotsos}@cse.yorku.ca

978-1-4244-3804-4/09/$25.00 ©2009 IEEE

recently, Lazebnik et al. [24] proposed another information theoretic solution to address a similar problem through information loss minimization.

Another group of authors, focused on improving the performance of the bag-of-features framework by modeling the statistical relationships between the visual words. Bernoulli mixture model is employed by Ferreira et al. [13] to capture the conditional dependencies between visual words in the vocabulary. In a more recent approach by Cummins et al [18], visual words are used in a probabilistic framework for image matching, where the statistical relationships between the visual word are modeled through Chow Liu trees, capturing the fact that certain combinations of visual words are likely to appear or disappear together in the images of the environment. They experimentally showed that modeling such statistical relationship in a probabilistic framework can improve the result in visual loop closure detection in dynamic environments, allowing more accurate estimation of the probability that two observations come from the same location.

Although the proposed extensions, to some extent, improve the performance of the traditional bag-of-features approach [2], 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 removed from the environment. In this paper, a novel landmark-based algorithm for qualitative localization is presented, which explicitly considers the challenges resulting from dynamic changes in the environment (as mentioned above). Hierarchical classifiers are built for different environments, where each node in a hierarchy is a visual word selected based on the *additional* information it can deliver about an environment, i.e., information not already provided by previously-selected visual words. Hierarchies are built in a top-down manner, where the children of each node in the hierarchy are visual words that together deliver (almost) the same additional information about the class (environment) as their parent does. During classification, if a visual word in a hierarchical classifier does not respond as expected (e.g., it is absent while expected to be present), then the children of that node are examined, and this process successively continues until atomic nodes (i.e., leaves) are reached. Experimental evaluations show that our proposed hierarchical method perform significantly better than competitive approaches, as it survives loss of key features due to changes in viewpoint, dynamic occlusions, shadows, and many other common environmental effects.

The remainder of the paper is organized as follows: in Section II we describe the different steps of our method. Section III presents the implementation details and experimental results. Finally, we conclude the paper and discuss some future work in Section IV.

## II. Method

### A. Image Representation

In our method, images are represented by their local distinctive features. Image features are extracted using the Scale Invariant Feature Transform (SIFT) technique, developed by Lowe [9], which combines a scale invariant feature detector and a gradient distribution based descriptor.

The detector (based on the Difference-of-Gaussians function) identifies locations in image scale space that are invariant with respect to image scaling and rotation, and are minimally affected by noise and small distortions. For each detected feature, a 128 dimensional description vector is created relative to the scale and orientation of the feature [9].

### B. Vocabulary of Visual Words

Similar to the idea of bag-of-features, in our method, extracted features from training images are quantized with a set of compact visual words. Visual words are built automatically by grouping the visually similar features extracted from the training images, using a clustering method. There are many clustering algorithms that can be used, each with certain strengths and weaknesses. The *k-means* algorithm has been widely used for this purpose, since its computational simplicity allows for very large data sets. However, *k-means* clustering requires the user to specify the number of clusters in advance, which is not possible for many applications (including ours).

In our method, we use an agglomerative clustering scheme, which automatically determines the number of clusters. 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 $\theta$:

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

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

For each resulting cluster with more than $m$ members ($m$ is 5 in our experiments), we compute the centre of the cluster and store it as a visual word in the vocabulary.

### C. Hierarchies of Visual Words

Given the set of computed visual words, the next step is to learn a classifier for each environment, by selecting and combining the appropriate visual words. For each environment, a positive and a negative set of images are created. Positive images for the environment are those taken from that environment, and negative images are simply the positive images of other environments. A binary vector, $P_i$, with the length equal to the number of training images is associated with every visual word, $i$. If there is a feature extracted from a training image, $I$, that belongs to visual word $i$, then $P_i(I)$ will be 1, otherwise it will be 0. Every class is also represented by a binary variable $C_j$, where $C_j(I)$ is 1 if $I$ is an image of environment $j$, and 0, otherwise.

The discriminative value of each visual word is measured as the amount of mutual information it can deliver about the class [1]:

$$I(P_i; C_j) = H(C_j) - H(C_j \mid P_i) \qquad (2)$$

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

$$Q_l = \arg\max_{P_k \in K_l} \min_{P_i \in S_l} (I(P_k, P_i; C_j) - I(P_i; C_j)) \qquad (3)$$

Here $K_l$ is the set of candidate visual word, $S_l$ is the set of selected visual words up to iteration $l$, and $Q_l$ is the visual word to be selected in iteration $l$.

This process ends when the increment in mutual information gained by selecting a new visual word is less than a certain threshold (0.05 in our experiments), or until the number of selected visual words reaches a pre-defined limit (15 in our experiments).

In our experiments (Section III), we observed that for most classes, the learnt visual words (called *top-level* visual words) are strong enough to discriminate the positive and negative training images with 100% accuracy. However, it is unrealistic to expect that all (or even the majority) of these visual words will be detected and recovered in the test images. This might be the result of changes in the structure of the environments (e.g., some objects are removed or added), variation in lighting conditions, or substantial viewpoint changes. Therefore, for each of the selected top-level visual words, we search for other visual words that can deliver the same mutual information about the class (relative to the previously-selected visual words), in the absence of that top-level visual word. The selected visual words are then considered as the children of the top-level visual word and are used in a top-down manner during classification (Section II.D).

To identify such lower-level visual words, rather than using all the positive and negative training samples, only those that are successfully classified by the higher-level visual word are used. Therefore, the goal is to find a combination of visual words that can (almost) perfectly mimic the action of the higher-level visual word. This can be done by applying the same information maximization procedure that was used at the higher level.

This process continues recursively, until the children of a selected visual word cannot provide an acceptable discrimination between the corresponding positive and negative samples, or when a pre-defined level (4 in our experiments) is reached. Visual words with no children are then labeled as *atomic* visual words.

Similar to [1], the response of each hierarchical node is computed by linearly combining the responses of its children[1]:

$$r = w_0 + \sum_{i=1}^{n} w_i s_i \qquad (4)$$

In the above equation, $s_i$ is the response of the $i^{th}$ child of the node, normalized by a sigmoid function to the range [-1, 1]. $w_0$ and $w_i$ are the bias and weights of the combination, adjusted iteratively during the training, using the Back-Propagation algorithm (as described in [1]).

### D. Localization

Localization of a query image starts by extracting the local features (Section II.A). The extracted features are then quantized with the set of visual words. Each hierarchical classifier is then examined in a top-down manner: if a visual word in a hierarchy does not respond as expected[2] (e.g., absent while expected to be present), then the children of that visual word are examined, and this process successively continues until atomic visual words (i.e., leaves) are reached. Then the response of a non-atomic visual word is computed using Eq. 4, once the responses of all its children are determined. Similarly, the final response of a classifier is computed when a response for every top-level visual word is available. After applying all the classifiers, localization is determined by the maximal response over hierarchies representing each class.

## III. Experiments

### A. Database

To evaluate our method, we use a publicly available database provided by [15]. The database was created from three outdoor sites on University of Southern California campus. Each site was manually divided into 9 continuous paths (referred to as *segments*). For each segment, 12 to 15 image sequences are provided, capturing different lighting conditions, small viewpoint variations and some structural changes. In [8] and [16], all images from almost half of the database are used for training (at least one run per segment per lighting condition) and the remaining for testing.

[1] If a visual word is an atomic node, its response is either 1 (the visual word is present in the image), or 0 (it is absent).

[2] If the weight between the visual word and its parent is positive, the visual word is expected to be present in the image, otherwise, it is expected to not be present (negative weights are not common and may only occur in low levels of hierarchies).

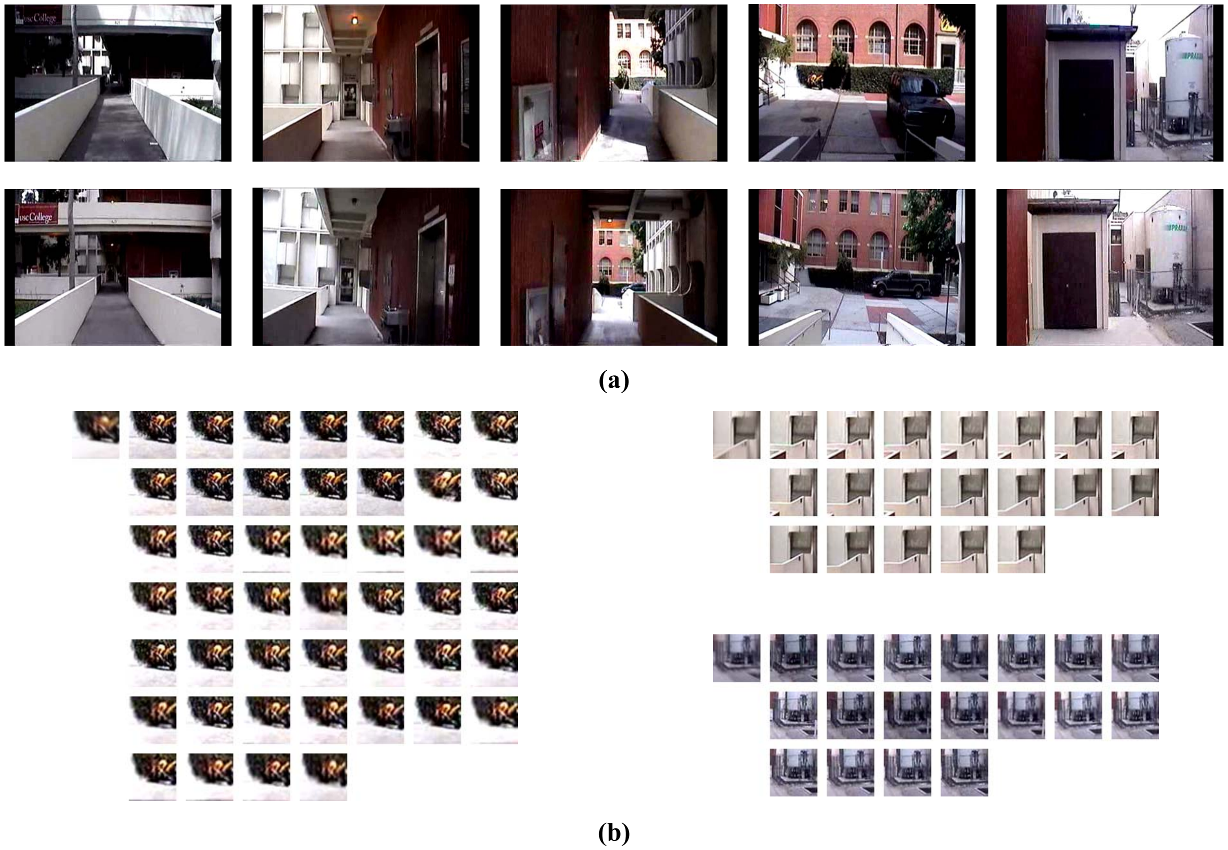

Figure1. (a) Examples from five segments of the ACB site [15] used in our experiments. Top images are samples from test set and bottom images are close samples from the training set. (b) Shows examples of the visual words produced by clustering the SIFT features extracted from the training images. Each example shows a cluster by displaying the image regions corresponding to the SIFT features that belong to that cluster, along with their average (top-left corner) to show the visual-compactness of the cluster.

In our experiments we use only the first five segments of the first site (ACB site). For each segment, two runs are used: a subset (regularly sampled) from one run is used for training, and all the images from the other run are used for testing. Runs were selected manually, such that the difference between the training and testing images (in terms of lighting conditions and structural changes) was maximal. Note that unlike [8] and [16], our system was trained with images of only one lighting condition and was evaluated with images of a different lighting condition. Overall, our training and testing sets consist of 250 and 1960 images, respectively. Figure 1.a shows sample images from training and test sets.

### *A. Results*

Extracting the features from all training images resulted in a pool of 76,965 features, ranging from 233 to 1264 per training image, with more features for highly textured images. Applying the clustering algorithm on the pool of features, a set of 25,284 visually compact clusters were produced. After discarding the clusters with less than 5 features, 3,568 clusters remained to compute the visual words (see Figure 1.b for several examples). For each environment, a separate hierarchical classifier was then constructed. Classifiers were then combined as described in Section II.D, to form the final localization system. Figure 2 shows the top-level visual words selected for some of the classifier.

Our first experiment was designed to evaluate the effectiveness of our hierarchical processing for dealing with dynamic changes in the environment. To this end, we run our localization system on the test images two times, once with full visual word hierarchies and the other time with only top-level visual words. As shown in Table 1, a recognition rate of 89.8% was achieved with full hierarchies, while with only top-level visual words the classification performance was around 77.6%. This confirms the advantages of our hierarchical method in dealing with variations in the test dataset. Variations that are not present in the training dataset may result in missing or misleading features, causing conventional feature-based methods to misclassify scenes. In our hierarchical method, these

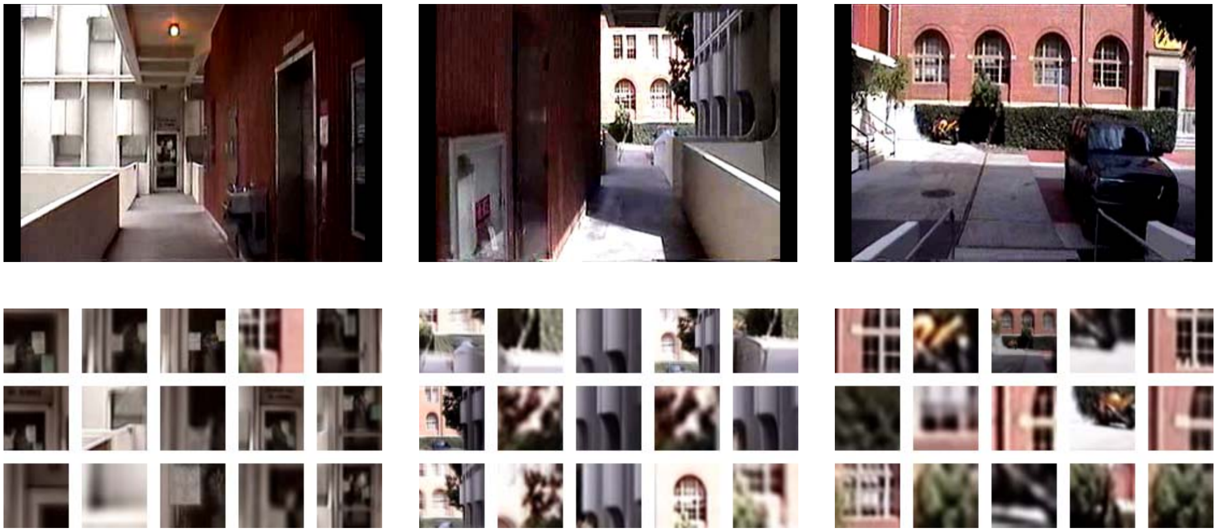

Figure2. Top-level visual words selected during training for three of the five classifiers. Each visual word is represented by the average of the image regions corresponding to the SIFT features that build that visual word. A few of the selected visual words have negative weights (e.g., fourth visual word from top-left in the second environment).

variations generate unexpected responses in higher-level nodes, which lead the algorithm to seek clarifying evidence from "back-up" child nodes. Often, the features coded by these lower-level nodes will have survived the variation and will allow the system to respond correctly. 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.

To put our results in context, we compared them with the performance of three other solutions on the same test set (using the same training set):

**BF1:** an in-house implementation of the traditional bag-of-features technique (similar to [2]). For this, we use exactly the same set of 3,568 visual words used for our hierarchical method. Each image is described by a description vector indicating the frequency of each visual word in the image. A Support Vector Machine (SVM) [17] is then learnt for each class versus the other classes using the description vectors of the training images. To classify a query image, the trained SVM classifiers are applied and the one with maximal score indicates the label of the query.

**BF2:** an implementation of [25], as a recent and well-received extension to the original bag of features method. SIFT descriptors extracted from local regions of the training images are quantized into a tree of visual words, using the hierarchical k-means clustering technique. The tree of visual words is used in a Term Frequency Inverse Document Frequency (TF-IDF) context, scoring the similarity of a training image to the query. Classification of a query image is performed by retrieving *N* (*N* = 6 in our experiments) most similar training images from the database and using their label to cast vote for the label of the query image. In our experiments, we use Andrea Vedaldi's implementation[3] of this technique.

**Global:** the implementation of the global image gist features [6], as described in Section I. As in [7], computed gist features from the training images are used as unique low-dimensional image classification keys to train a SVM classifier for each environment. Similar to BF1, each query image is then classified by applying the trained SVM classifiers and taking the maximal score.

For each of the above methods, we use the parameters proposed by the authors. Table 1 summarizes the results. As can be seen, the classification accuracy achieved from the top-level visual words and *BF1* and *BF2* methods, while close to each other, are substantially lower than the performance of our proposed hierarchical method. This validates the superior robustness of our hierarchical method against dynamic changes in the environments, and supports the idea of explicitly addressing the problem of partial occlusion in learning the classifiers.

All our experiments were performed on a PC with a 2.4 GHz CPU. The most time consuming process in our localization system is the extraction of image features, which takes around 0.7s on average for each 240x352 image. Given the extracted image features, recognition is performed in around 0.1-0.3s, depending on the number of extracted image features and the number of informative visual words contained in each classifier (see Table 1 for the average

[3] http://vision.ucla.edu/~vedaldi/code/bag/bag.html

localization time of other methods used in our evaluations).

## I. CONCLUSIONS

In this paper, a novel technique for qualitative localization was proposed, which combined the advantages of landmark-based approaches with hierarchical learning to achieve robustness against dynamic changes in the environments. Several experiments on a challenging localization database validated the effectiveness of our hierarchical learning and showed a significant improvement over the conventional feature-based and global approaches.

In future work, we intend to evaluate other feature extraction methods and study the scalability of our method. We would also like to investigate the possibility of using a feature tracking technique, rather than a clustering method, to build the vocabulary of visual words from feature points. This could significantly speed up the training process and result in more visually compact clusters.

TABLE I
PERFORMANCE AND COMPUTATIONAL TIME

| Methods | Accuracy | Computational Time |
|---|---|---|
| Full Hierarchy | 89.8% | 0.9 s |
| Top-Level | 77.6% | 0.8 s |
| BF1 | 71.3% | 0.9 s |
| BF2 | 78.6% | 0.8 s |
| Global | 81.2% | 1.7 s |

The performance and average inference time of our hierarchical method on the test database, in comparison with other techniques used in our evaluations.

## REFERENCES

[1] B.Epshtein, S. Ullman, "Feature Hierarchies for object Classification", *International Conference on Computer Vision*, 2005.

[2] C. Dance, J. Willamowski, L. Fan, C. Bray, G. Csurka, "Visual categorization with bags of keypoints", *ECCV International Workshop on Statistical Learning in computer vision*, 2004.

[3] P. Blaer and P. Allen, "Topological mobile robot localization using fast vision techniques", *International Conference on Robotics and Automation*, 2002.

[4] I. Ulrich and I. Nourbakhsh, "Appearance-based place recognition for topological localization", *International Conference on Robotics and Automation*, 2000, pp. 1023 – 1029.

[5] A. Pronobis, B. Caputo, P. Jensfelt, and H. Christensen, "A discriminative approach to robust visual place recognition", *International Conference on Intelligent Robots and Systems*, 2006.

[6] Oliva A, Torralba A, "Building the gist of a scene: the role of global image features in recognition", *Progress in Brain Research*, 2006, pp. 23-36.

[7] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition", *International Conference on Computer Vision,* 2003, pp. 1023 – 1029.

[8] C. Siagian, L. Itti, "Rapid Biologically-Inspired Scene Classification Using Features Shared with Visual Attention", *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 29 (2), 2007, pp. 300-312.

[9] Lowe, D., "Distinctive image features from scale-invariant keypoints", *International Journal of Computer Vision*, 2004, pp. 91–110.

[10] S. Se, D. Lowe and J. Little, "Vision-based Global Localization and Mapping for Mobile Robots", *IEEE Transactions on Robotics*, 21(3), 2005, pp. 364-375.

[11] H. Tamimi and A. Zell, "Vision based localization of mobile robots using kernel approaches", *International Conference on Intelligent Robots and Systems*, 2004.

[12] J. Matas, O. Chum, M. Urba, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions", *British Machine Vision Conference*, 2002, pp. 384-396.

[13] F. Ferreira, V. Santos and J. Dias. "Integration of multiple sensors using binary features in a Bernoulli mixture model", *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2006, pp. 104–109.

[14] S. Agarwal and D. Roth, "Learning a Sparse Representation for Object Detection", *European Conference on Computer Vision*, 2002, pp. 113-130.

[15] C. Siagian, 2007, http://ilab.usc.edu/siagian/Research/data/PAMI07/

[16] C. Siagian and L. Itti, "Biologically-Inspired Robotics Vision Monte-Carlo Localization in the Outdoor Environment", *International Conference on Intelligent Robots and Systems*, 2004.

[17] V. Vapnik. Statistical Learning Theory. Wiley, 1998.

[18] M. Cummins and P. Newman, "FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance", *International Journal of Robotics* Research, 27(6), 2008, pp. 647-665.

[19] F. Jurie and B. Triggs, "Creating efficient codebooks for visual recognition", *International Conference on Computer Vision*, 2005, pp. 604-610.

[20] M. Vidal-Naquet and S. Ullman. "Object recognition with informative features and linear classification", *International Conference on Computer Vision*, 2003, pp. 281-288.

[21] J. Brank, M. Grobelnik, N. Milic-Frayling, and D. Mladenic. "Interaction of feature selection methods and linear classification models", *International Conference on Machine Learning*, *Workshop on Text Learning*, 2002.

[22] D. Mladenic, J. Brank, M. Grobelnik, and N. Milic-Frayling, "Feature selection using linear classifier weights: Interaction with classification models", *Special Interest Group on Information Retrieval*, 2004, pp. 234-241.

[23] E. Nowak and F. Jurie, "Vehicle categorization: Parts for speed and accuracy", *International Conference on Computer Vision, VS-PETS workshop*, 2005.

[24] S. Lazebnik and M. Raginsky, "Supervised Learning of Quantizer Codebooks by Information Loss Minimization", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[25] D. Nister, H. Stewenius, "Scalable recognition with a vocabulary tree". *International Conference on Computer Vision and Pattern Recognition*, 2006, pp. 2161–2168.

[26] M. M. Ullah, A. Pronobis, B. Caputo, J. Luo, P. Jensfelt, H. I. Christensen, "Towards robust place recognition for robot localization", *International Conference on Robotics and Automation*, 2008.