VISION-BASED PLACE CATEGORIZATION

A Thesis
Presented to
The Academic Faculty

by

Richard K. Bormann

In Partial Fulfillment
of the Requirements for the Degree
Master of Computer Science in the
College of Computing

Georgia Institute of Technology
December 2010

VISION-BASED PLACE CATEGORIZATION

Approved by:

Professor Henrik Christensen, Advisor
College of Computing
Georgia Institute of Technology

Professor Frank Dellaert
College of Computing
Georgia Institute of Technology

Professor James Rehg
College of Computing
Georgia Institute of Technology

Date Approved: November 9, 2010

To my Family

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Henrik Christensen, a lot for all the time he took for having interesting discussions and giving invaluable advice. I am grateful that he gave me the opportunity to work in his laboratory with all the great people around. It was a pleasure for me to work together with so many kind and motivated other students who contributed to this positive atmosphere in the lab.

I also want to thank my committee members, Dr. Frank Dellaert and Dr. James Rehg, who broadened my horizon in the field of computer vision, graphical models and 3D reconstruction through their excellent lectures. Their spirit always motivated me to explore these topics more and more.

A special thank you goes to Dr. Mike Stilman who spurred us on writing a paper about his class project and to Dr. Henrik Christensen who made us excited about the topic and who provided guidance when we had hard times.

Furthermore, all my friends here in Atlanta deserve my gratitude since they made me feel home around them and I had really good times with all of them, especially with my old and new roommates as well as with my labmates.

Finally, I like to thank my parents and grandparents so much for all the support and care they provided to me in my life and during my stay in Atlanta. Without their help I would not have had the chance to have these great experiences at Georgia Tech.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

In this thesis we investigate visual place categorization by combining successful global image descriptors with a method of visual attention in order to automatically detect meaningful objects for places. The idea behind this is to incorporate information about typical objects for place categorization without the need for tedious labelling of important objects. Instead, the applied attention mechanism is intended to find the objects a human observer would focus first, so that the algorithm can use their discriminative power to conclude the place category. Besides this object-based place categorization approach we employ the Gist and the Centrist descriptor as holistic image descriptors.

To access the power of all these descriptors we employ SVM-DAS (discriminative accumulation scheme) for cue integration and furthermore smooth the output trajectory with a delayed Hidden Markov Model. For the classification of the variety of descriptors we present and evaluate several classification methods. Among them is a joint probability modelling approach with two approximations as well as a modified KNN classifier, AdaBoost and SVM. The latter two classifiers are enhanced for multi-class use with a probabilistic computation scheme which treats the individual classifiers as peers and not as a hierarchical sequence.

We check and tweak the different descriptors and classifiers in extensive tests mainly with a dataset of six homes. After these experiments we extend the basic algorithm with further filtering and tracking methods and evaluate their influence on the performance. Finally, we also test our algorithm within a university environment and on a real robot within a home environment.

CHAPTER I

INTRODUCTION

One of the main goals in service robotics is to develop robots that can assist humans in a useful and natural way. In order to accomplish this aim robots must be enabled to operate as smoothly as humans in man-made environments. Moreover, not only the behaviour and capabilities of the robot should meet human expectations, also its way of interacting with individuals should be as natural as possible. One step on the long way towards this goal is to enable robots to perceive their environment with more awareness for the semantic contents surrounding them. Part of this problem is the recognition of the kind of place the robot is located at.

This thesis presents a new approach towards enabling robots for the categorization of indoor places into functional units like rooms using vision sensors only. Although this is a very hard problem we feel that place categorization should be possible using vision only as humans exemplify this ability constantly. Even if no stereo vision information is available, for example when watching a photograph, humans can reliably detect a place category. Consequently, a solution for this problem for robots should be possible.

If robots could determine the semantic category of their surroundings this would enhance the applications for robots and improve the interface to humans. First, interacting with the robot in the way of "Please take this document to Steve. His lab is located on the right side after the kitchen." would be possible even in new environments if the robot can understand and detect the semantic categories of places. This means, it would not be necessary to show and map entire Georgia Tech to the robot before it could do useful tasks within the campus. Moreover, maps might be outdated after a while because rooms moved to another place. If a robot only relied on a map, it could end up in a bathroom where it expected a kitchen without knowing that it is in the wrong room. Reliable visual place categorization would provide more robustness for robots in those tasks and help to keep maps up to date.

Moreover, when service robots are eventually mature products they should work out-of-the-box in their destination environment since people do neither want to teach the robot everything, which can be a tedious task, nor do they want to pay a technician who does this job.

Furthermore, if the kind of scene can be detected by the robot, strong priors on the availability or absence of certain objects can be concluded. For example, it is highly likely to find a fridge and dishes in the kitchen while we would expect to find keyboards in an office. A robot which detected an object but cannot decide whether it is a computer monitor or an oven would have an easy decision if it previously knew that it is located in an office. The probability to find an oven in an office is much lower than observing a monitor.

After these illustrations of the place categorization task and the usefulness of solutions to it we now define the problem of this thesis more formally.

1.1 Problem

Within this thesis we adress the visual place categorization problem. We want to understand this problem as finding a label for the kind of place a robot is located at, where place is seen as a functional unit like kitchen, office or living room. Of course, places the robot is intended to classify were never seen before so that the robot is forced to build a category model for the different place classes. The only perceived input into the categorization system is the sequence of images the robot can capture with a standard monocular video camera. This implies that the sequence will naturally contain many meaningless views as well since the robot is assumed to operate autonomously.

1.2 Proposed Method towards a solution

The method we explain in section 3 lies between two previously presented approaches which either categorize a scene by the objects it contains [15, 47, 77] or by some holistic image descriptor which captures the gist of the scene [53, 73, 81]. Our conviction is that humans rely on coarse cues as well as on detail cues when categorizing a scene as indicated in [52, 67]. Therefore we want to combine the evidence provided by holistic image features as well as by object occurences. As previous work demonstrated [15, 47, 77] the object-based

approach requires a lot of labelled and segmented object data. Most of these approaches can therefore only consider a very limited number of objects for place categorization. We want to avoid this problem by enabling the robot to select candidate regions using a visual attention mechanism which should contain objects. Based on the found objects the system builds a place model which can be used for categorization. This idea is inspired by the fact, that humans can recognize scenes very quickly within around 30ms even if only the high-frequency image contents are provided [67] which convey detail information. We suppose that humans use their visual attention mechanism to analyse some parts of the scene first and determine a likely preliminary decision already at this processing stage.

In the next chapter we want to discuss the related work to this topic in detail. Afterwards, we present the applied methods in chapter 3 and the experiments for evaluating the performance of this approach in chapter 4.

CHAPTER II

PREVIOUS WORK

Semantic place categorization using visual features only is a quite new area of research for robotic applications [73, 81]. In the past, many researchers either focused on place recognition tasks [58, 70] and on Simultaneous Localization and Mapping (SLAM) [12, 68] in the area of robotics or on the problem of scene recognition in computer vision [3, 16, 35, 53, 59, 60]. Place categorization is often accomplished in conjunction with a map or laser scanner data [47, 76].

2.1 Scene Recognition in Computer Vision

In the computer vision community a lot of research has been reported on scene recognition, however, this problem is more concerned with categorizing images taken by humans which usually show considerably informative views of the scene. Very popular in this domain is the holistic Gist model of Torralba and Oliva [53] which defines the five perceptual properties naturalness, ruggedness, openness, roughness and expansion for an image that tend to have similar numerical values within one scene category. These properties are also known as the spatial envelope properties and their computation is based on linear combinations of the principal components of the energy spectrum of the whole scene image and of localized parts of it. However, the effort of initial labelling appears very high since every training image has to be manually assigned with the degree of these perceptual properties. This model has shown a strong categorization performance up to 90% for outdoor data [53] but was not that successful on indoor categories [59] because several spatial envelope properties take similar values for indoor places (e.g. naturalness, openness).

Thus, recently Quattoni and Torralba [59] introduced a 67 indoor places dataset for which they obtain quite impressive results with an approach focusing on detecting characteristic objects beside the global Gist image descriptor. For the object detection part, they represent each class with several prototype images in which ten regions of interest

(ROI) were manually labelled in advance. During detection, these ROIs are allowed to move slightly in order to find meaningful objects expected at the trained positions. This idea corresponds with our proposed solution, however, we would prefer not to be forced to tediously label potentially interesting regions in advance and for real robot applications we doubt that ten more or less fixed locations can yield helpful hints from all possible views. Consequently, we propose to employ a visual attention method in order to focus on several interesting regions which are supposed to contain objects.

The approach of Espinace *et al.* [15] is more directed towards the application in robotics because it applies a 3D range sensor attention mechanism in order to filter regions which are likely to contain a known object. Their generative probabilistic hierarchical model decides for a scene category depending on the characteristic objects present in the image. The associations between objects and scenes are learned from a large database of the Flickr website as shown in [32]. The objects themselves are detected with a cascade classifier working on sliding window patches of grayscale, Gabor and HOG [13] descriptors. The 3D attention mechanism helps to avoid unneccessary evaluations of the sliding window areas while contributing some further 3D features. Because of the good object detectors this method yields very high indoor categorization performance on four room classes (office, conference room, hallway, bathroom). The drawbacks are the need for labelled object data (here taken from LabelMe [64]) for the object classifiers, the slow evaluation speed in the order of seconds per image (due to the sliding window detection) and bad scaling properties in the dimension of detectable objects which eventually also limits the number of detectable views. Furthermore, the evaluation was done on single images instead of image sequences which would be natural for a robot application.

Another approach which categorizes places based on detected objects was introduced by Viswanathan *et al.* [77]. They solve the object segmentation and labelling problem for the training set by using already labelled data from the LabelMe databse [64] as well. However, they also state that the provided labels are not always as reliable as desired without any postprocessing because of synonymy and polysemy problems. They describe the object classes with mixtures of multiscale deformable part models [17] and train one

classifier for each model. Since the authors did not report any runtimes, we suspect that it runs in comparable speed as [15] as it depends strongly on good object segmentations. The authors provide the results for distinguishing kitchen images from office images and obtain accuracies around 75%.

Further work on scene recognition using holistic image features was done by Lazebnik *et al.* [35] who use a hierarchical spatial pyramid matching scheme for histograms of densely sampled features. This approach yielded very good results on the 15 scenes problem [16, 35, 53] and outperformed the Bayesian hierarchical method of Fei-Fei and Perona [16] which employs a Latent Dirichlet Allocation model on sparsely sampled interest points that are encoded in a bag-of-words vocabulary of SIFT descriptors [40]. A similarly strong generative approach presented by Bosch *et al.* [3] is based on a bag-of-words vocabulary on color SIFT descriptors but builds intermediate topic representations via pLSA. All methods of this paragraph only require labels for the training images and build intermediate representations on their own. This is a very desirable property which can be found again in our proposed algorithm. A very good overview over scene recognition in the computer vision community until 2006 is collected in [4].

2.2 Place Recognition in Robotics

In contrast to scene recognition, place recognition methods are supposed to recognize previously visited places under varying conditions like illumination changes, the influence of seasons or human activity. Visual cues are either used to improve localization accuracy in SLAM settings [18, 45, 72] by observing certain landmarks which are only visible within a small range of positions [50, 68, 70] or for buildig topological maps [11, 12, 73, 84] of the environment. Often, visual landmarks are employed to improve the loop closing reliability in SLAM applications [7, 11, 12, 25, 49].

Early SLAM methods build their maps using odometry and laser scanner data only [18, 45, 72]. However, location estimates can be improved if further feedback from vision sensors is added especially if the map grows or the environment is ambiguous for laser scanner data. Newman and Ho [50] demonstrate this by finding previously visited places

while the map constructed from odometry already has a significant error. This ability to detect loop-closures independent of the odometry/laser scanner cues is very valuable and also helpful in kidnapped robot scenarios. A vision-only mapping and localization system tested on indoor environments was introduced by Se *et al.* [68]. It estimates the robot pose and builds a 3D map by tracking SIFT features with a stereo vision setup. Zivkovic *et al.* [84] present an algorithm using a graph-clustering technique which can autonomously cluster a sequence of images captured by a robot into convex subspaces which humans would associate with rooms. Although this system assigns very similar images to one topological place, the visual dissimilarity of place categories cannot be mastered with such a concept.

In outdoor environments, visual features are often preferable for mapping and localization tasks since there are less useful laser scanner features. Approaches demonstrated by [11, 12, 25, 49] apply bag-of-words clustering on SIFT features in order to characterize topological locations. If one of the places in the topological map is revisited, these systems can assert a loop-closure. However, all these algorithms have in common that they simply localize the robot within a self-built map. No additional high-level semantic information is collected neither from the laser scanner nor from the visual features. Ranganathan and Dellaert [61] instead use semantic objects and their constellations in order to describe a place which is a more robust landmark for SLAM applications. Similarly, Ekvall *et al.* [14] combine indoor SLAM with the placement of semantic information in terms of integrating the place of detected objects into the created map.

Place recognition systems aim at semantic labelling of whole areas in a map which normally have a meaning to humans in contrast to the normally meaningless topological nodes created in topological SLAM maps (see above). Indoors, areas normally coincide with rooms which fall into a functional category like office, meeting room, bath room, living room, etc. Pronobis *et al.* [56, 57, 58] describe a place recognition system which integrates global and local visual features as well as the laser range features of Mozos *et al.* [43]. As holistic image features they use high dimensional composed receptive field histograms [37] on second order Gaussian derivatives of the intensity image. Local feature points are detected with the Harris-Laplace method [23] and described with SIFT descriptors. The

different features provide single cue decisions before these are fused within a SVM-DAS (discriminative accumulation scheme) framework. The authors test their place recognition system on the KTH-IDOL2 database [42] which consists of a robot trajectory captured many times at differing daytimes and weather conditions (sunny, cloudy, night) and the influence of human activity. The idea of fusing global and local descriptors to lift the classification accuracy appears very sensible to us and is part of the algorithm presented in this thesis (see section 3.3.3).

Outdoor place recognition was demonstrated by Siagian and Itti [69] who developed a biologically-inspired place recognition system. It models a holistic image representation, the gist, which neglects details of the scene. Their gist feature is based on the feature maps computed in their biologically-inspired attention system [28] as well, which are center-surround operations on the intensity image and on different color channels at various scales as well as an oriented pyramid obtained from Gabor filtering. The mean of these feature maps is sampled from a spatial 4x4 grid at all scales and written into a descriptor whose dimensionality is reduced from 544 to 80 features through principal component analysis and independent component analysis. The authors tested this system on three outdoor locations containing ten segments each, which had to be distinguished. The same trajectory was used everytime but recorded at four different daytimes. Leave-out-one cross-validation showed that this setup allowed to distinguish all segments from each other with only 13.5% error. In [70] Siagian and Itti extend this system with their saliency method and furthermore sample SIFT features inside up to five salient regions for place modelling. The SIFT features are used for feature matching when the place is revisited. This enables the system to localize the robot with up to 0.98m accuracy within the place segments. We feel that the biological considerations of this approach make sense and want to use a similar concept of using global image descriptors together with salient region descriptors. However, instead of using the salient regions for matching and localization we intend to apply descriptors enabling the areas to detect typical objects for certain place categories.

2.3 Place Categorization for Robotics Applications

One of the first serious attempts to visual place categorization on image sequences of a moving agent was the wearable test system of Torralba *et al.* [73]. They decribe the scene with a descriptor derived from a wavelet image decomposition into an oriented steerable pyramid. From a spatial 4x4 grid the means at the various scales of the pyramid are concatenated and the resulting descriptor is reduced in its dimension to 80 features from initially 384 via PCA. The place appearance is then modelled as a mixture of Gaussians from the provided prototype views. The system contains subsystems for place recognition, place categorization and indoor/outdoor classification which are implemented as Hidden Markov Models (HMM). While the performance of the place recognition system is very high, the place categorization system only achieves good results on the classes office, conference room and corridor. The indoor/outdoor system works very well, again. The authors also indicate that the place recognition imposes strong priors on objects which are expected in the classified location which is one of the useful applications of a place recognition or a place categorization system.

Further tests with the place recognition system of Pronobis *et al.* [58] were done by Ullah *et al.* [76] using the COLD database [55] on which the performance could be evaluated across three different university environments (Ljubljana, Saarbrücken, Freiburg). These experiments included the assessment of the place categorization abilities when the system is trained on two of the three universities and tested on the remaining set. The results for corridor were good with 76% but the remaining categories could not be categorized well with rates around 10-15%. The three universities dataset is one of the sets we tested our algorithm on.

Mozos *et al.* [43, 46, 47] and Rottmann *et al.* [63] present a system for place categorization along robot trajectories using laser and vision features and for building topological maps with semantic nodes that represent whole rooms using laser features only. The laser features contain 302 simple features described in [43] and the visual cues are the numbers of detected objects from eight classes within a panoramic view using Haar-like features [36] for object detection. The features are written together in one descriptor and classified

with a sequential AdaBoost [19] classifier from which confidence values for the decision are derived. For the topological map building out of the metric map, they can achieve very accurate maps using laser features only when the three classes room, corridor and doorway are to be distinguished. For the classification of places along trajectories they finally apply a HMM for incorporating the sequence information and obtain very good results for six classes (laboratory, office, seminar room, doorway, corridor, kitchen). They also show that the classification error drops by up to 40% through the use of visual features in addition to the laser features. Although the few well-learned objects yield a huge performance gain (also compare with [15]) we doubt that these results are repeatable in a very diverse home environment. We furthermore would like to skip the tedious preparation of labelled object data. Therefore, the algorithm applied in this work tries to capture important objects by itself.

The work done by Wu *et al.* [80, 81, 82] is the most comprehensive towards visual place categorization for robotic applications up to now. They developed a new holistic descriptor CENTRIST [82] based on histograms of census transformed grayscale or Sobel images which is tailored to capture the essential structure of the scene. They achieve very good results on the common test databases of computer vision, e.g. the 15 scenes database [16, 35, 53], by classifying with spatially localized census transform histograms which were reduced in their dimension with principal component analysis (spatial PACT method). Furthermore, the authors contributed the currently most diverse home environment dataset [81] for visual place categorization recording image sequences in six different homes. On this dataset, they demonstrate a system for visual place categorization based on the CENTRIST descriptor which is computed in a 4x4 grid on Sobel images. The obtained histograms are clustered with histogram intersection clustering. Depending on the occurring clusters, a temporal naive Bayes filtering outputs the room categories found along a robot trajectory. The obtained results are with 46,8% significantly higher than if the SIFT descriptor was used (39,8%). We agree with Wu that a strong global classifier like CENTRIST seems to be biologically plausible. However, research of Schyns and Oliva [67] on hybrid images found that human scene recognition mechanisms attend to coarse (low-frequency) image contents if

the allowed processing time is very short (around 30ms) and turn to the fine (high-fequency) contents on longer processing times around 150ms, although both scales are perceptually available after 30ms [52]. This indicates that harder categorization decisions which take more time involve the search for finer-grained objects. The visual attention mechanism used in this work is supposed to attend to smaller objects in the scene, especially those which would draw human attention first.

CHAPTER III

METHODS

As discussed in the previous chapter, Schyns and Oliva [67] have shown that humans can recognize a scene category quite reliably within 150ms. Moreover, their findings suggest that humans can quickly extract the gist of a scene independent of the spatial frequencies contained in the provided image data [52]. They also suggest that in certain cases when a categorization decision might be harder to make because of an ambiguous environment or an unusual view at it, fine resolution image contents contribute to the disambiguation of coarse resolution decisions. We suppose that humans accumulate evidence for a category in those cases through the search for typical objects. The first glances are attracted by objects or parts of them which are salient in the field of view with respect to certain criteria.

We propose to fuse the information obtained from the gist of an image (coarse scale) with the detail information provided by some salient regions in the image in order to categorize a place. The algorithm explained in the following is supposed to learn the room categories and important objects from image sequences whose sole label is the place category for each image. Consequently, it is intended to find the typical objects describing a place on its own. In section 4 we examine whether enough object information for this task can be found within the salient regions. This approach is novel in contrast to several successful former approaches on visual place categorization which relied on object detection for a small collection of discriminating objects which were trained in advance [15, 47, 77]. An overview over the algorithm is provided in Figure 1.

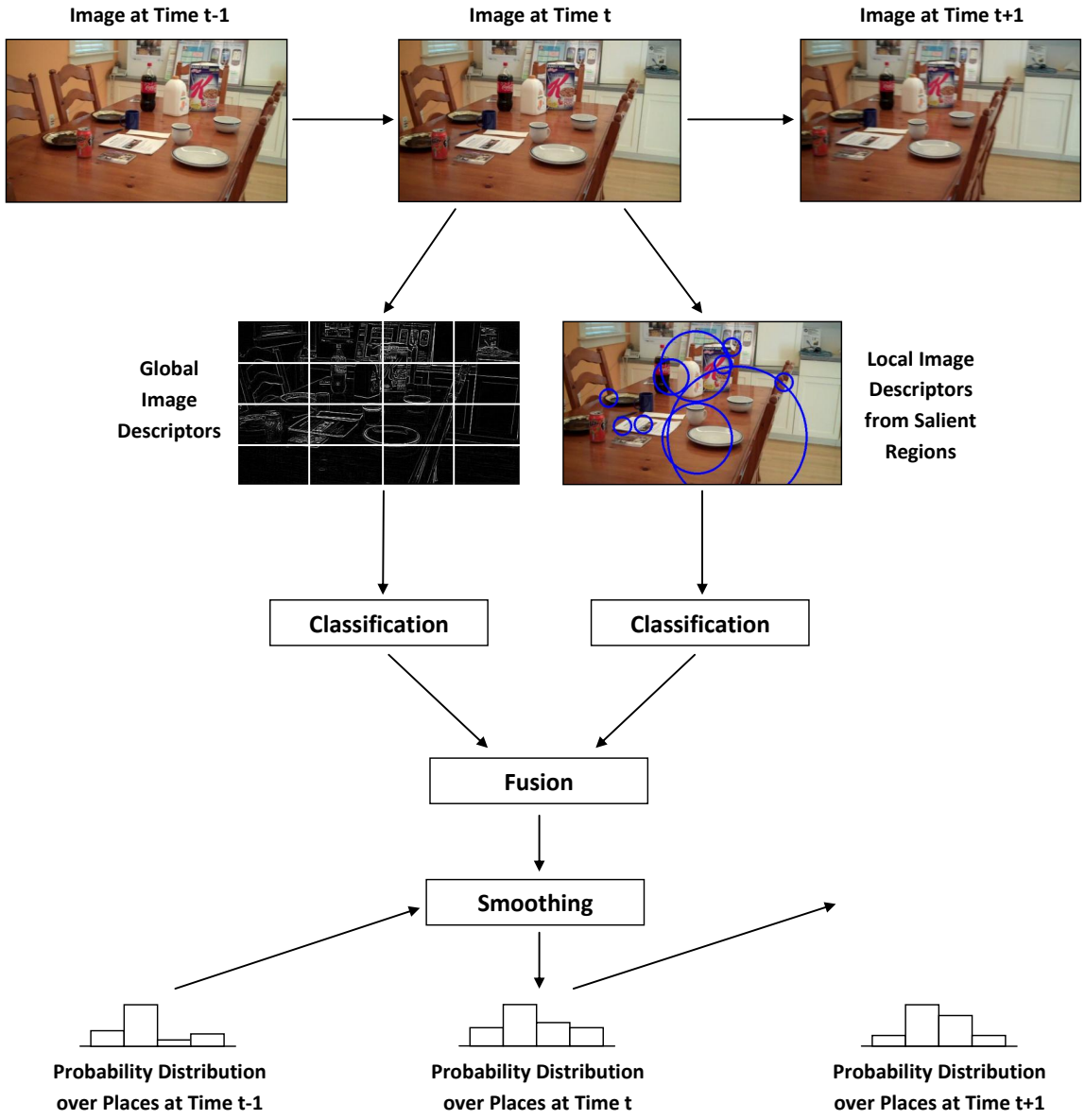


Figure 1: Overview over the algorithm: Images from a sequence are presented to the algorithm which computes some holistic image features and searches for several image regions which would draw human attention first. From those regions further local descriptors are generated which are supposed to contain objects or important scene elements. If the objects are characteristic for a place they should improve the decision for it. Therefore, the classification results of both, local and global features are fused together with the intention that they improve the decision quality. Finally, noise in the decision output is smoothed with respect to the decision sequence in order to avoid unrealistic jumps between room categories.

3.1 Local Scene Descriptors at Salient Regions

This section describes how the salient regions which attract the visual focus are determined and which descriptors could be used to characterize the found objects or object parts.

3.1.1 Visual Attention

A lot of research on computer models for visual attention has been pursued during the recent 15 years. Important models were developed by Itti *et al.* [28], Frintrop [21] and Hou and Zhang [26]. The models of Itti and Frintrop are based on the findings of Treisman [75] about the physiology of the human visual system since they incorporate the canonical stimuli color, luminance and orientation and combine those to a saliency map, which indicates the saliency of individual image regions. Hou's method aims at removing the common information content of the image leaving only the innovation part which indicates the salient areas. This is accomplished by a spectral residual method which eliminates the mean frequency responses in the frequency domain. For a good overview over this topic we refer to [22].

In this algorithm we use a saliency method close to the bottom-up model of Frintrop since it allows to integrate saliency cues from different feature channels. Naturally, it contains a feature channel for luminance, one for color in the CIELAB color space and one for orientation using Gabor filters. On the former two feature channels, center-surround operations at varying scales define salient regions whereas the orientation channel remains unprocessed. Furthermore, we add the saliency map obtained from Hou's method as a fourth channel because of its very different approach. The final saliency map is the weighted sum of the conspicuity maps of each feature channel, which themselves consist of the weighted sum of the feature maps from the respective channel. We select the areas with the most intense response in the saliency map as the desired regions to examine. Their scale is determined from the scale of the feature map which gave the highest response at that position. A detailed description is following in the next section. The displayed images relate to the example image in Figure 2.

Figure 2: The example image for illustrating the process of saliency computation.

3.1.1.1 The Bottom-Up Saliency Model

The bottom-up model is based on three feature channels intensity, color and orientation. For the *intensity channel*, a Gaussian pyramid on the grayscale version of the input image is build initially convolving the grayscale image with a 5x5 Gaussian filter mask and downscaling it by factor 2 each time. As in the original work of Frintrop [21] we begin to use the images from the pyramid at scale 2, this means downscaled two times, in order to avoid too much influence of image noise in the results. A center-surround mechanism is applied to each used image in the pyramid yielding the intensity feature maps I'' at different scales. This center-surround mechanism works on each pixel in the following way: For detecting bright spots within darker areas, the center-surround response is the difference of the center pixel value and the mean of the surrounding pixels in a 3x3 or 7x7 neighborhood. For the detection of dark spots on bright background the operator is used with exactly the opposite difference. The size of the neighborhoods is called radius of the center-surround operator in the work of Frintrop. The obtained intensity feature maps at different scales and radii I'' are then added pixelwise across scales and radii within the bright-spots and the dark-spots subchannel to the two feature maps I'. This procedure is illustrated in Figure 3.

The feature maps from the *color channel* are based on the conversion of the RGB image to the colors of the LAB space which has two color dimensions, one for the opposite color

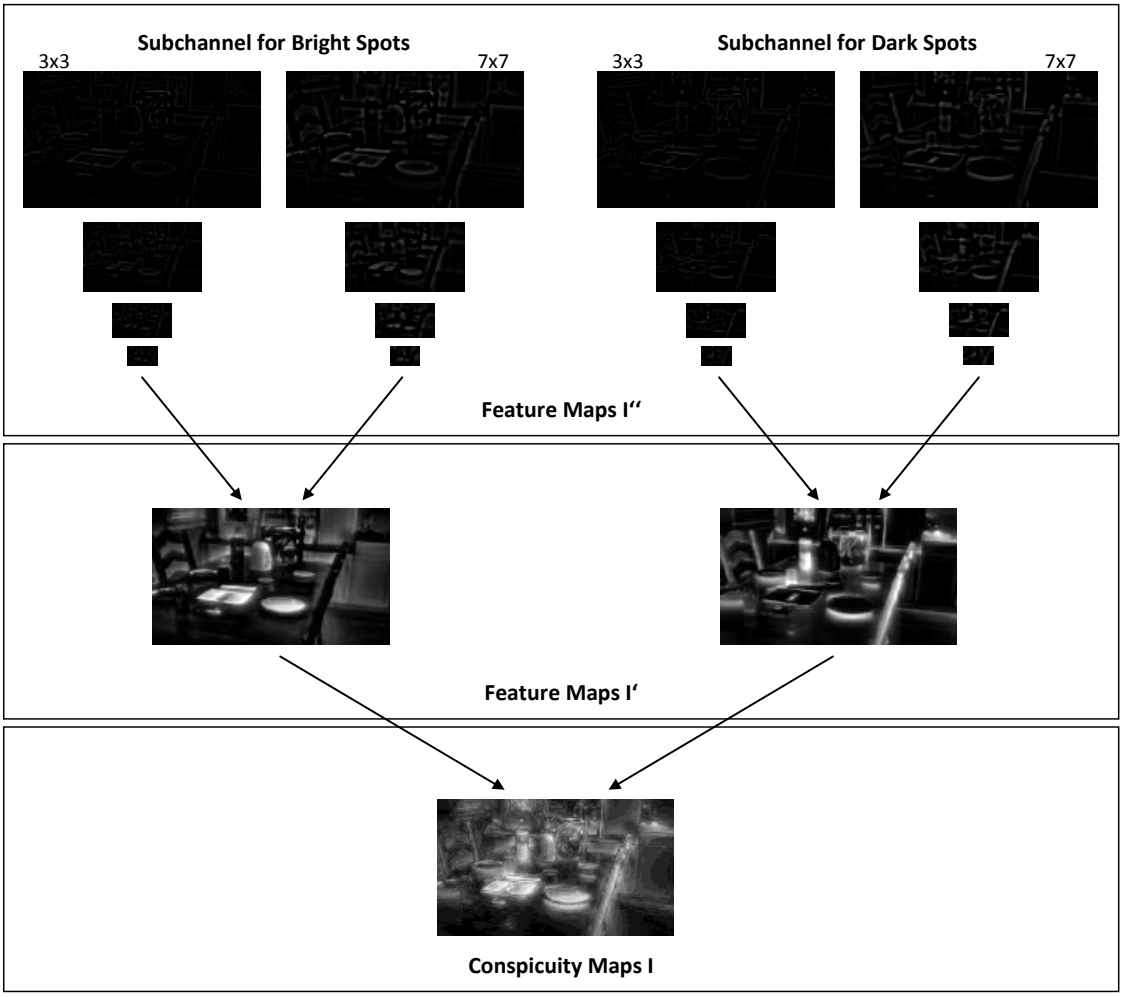


Figure 3: Visualization of the bottom-up visual attention mechanism in the intensity feature channel. The feature maps I'' are shown for different scales and radii for the two subchannels bright and dark spots.

pair red and green as well as one for blue and yellow. However, due to an inconsistency of the OpenCV [5] LAB conversion function on different operating systems we decided to convert the RGB image to the two opposing color pairs based on the simpler way Itti *et al.* [28] introduced in their very similar visual attention system. There, the respective color channels R, G, B, Y are converted from r, g, b as follows after normalizing the RGB channels

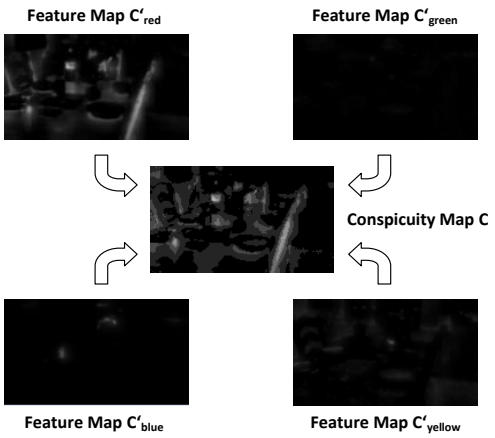


Figure 4: The color feature maps C' for the red, green, blue and yellow subchannel.

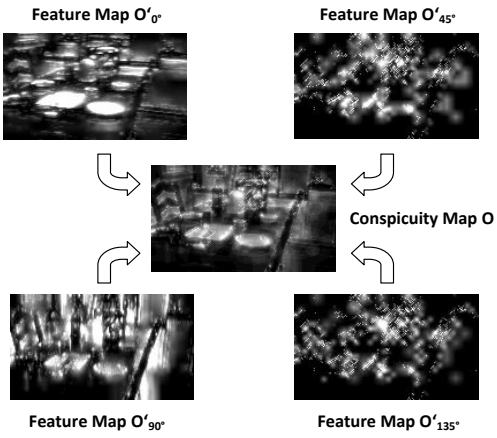


Figure 5: The orientation feature maps O' for the $0°, 45°, 90°$ and $135°$ subchannel.

with the intensity $i = (r + g + b)/3$:

$$R = r - \frac{g + b}{2}$$
$$G = g - \frac{r + b}{2}$$
$$B = b - \frac{r + g}{2}$$
$$Y = r + g - 2(|r - g| + b)$$

In the following, for each of these color channels a pyramid is built in a similar way as for the intensity feature. On the pyramid layers beginning at scale 2, center-surround operations are done yielding the color feature maps C'' at different scales with different radii for the center-surround operator. Across scale and radius addition of the C'' maps finally provides four feature maps C', one for each color channel R, G, B, Y. The four color feature maps C' are displayed in Figure 4.

For the *orientation channel* feature maps we want to detect edges in the image based on their direction. Therefore, we generate an approximation of the Laplacian pyramid by substracting successive images from the Gaussian pyramid and build an oriented pyramid out of the Laplacian pyramid by applying Gabor filters. In detail, we use four different orientations of the Gabor filter corresponding to $0°, 45°, 90°$ and $135°$ and build each pyramid starting at scale 2 as done for the former feature channels. In the end, we obtain the four feature maps O', one for each angle of the edge filter, by across scale addition of the images

within each orientation subchannel of the oriented pyramid. The four orientation feature maps of our example are displayed in Figure 5.

In order to generate one *saliency map* out of the variety of feature maps the latter are fused together in a two-stage process. First, the subchannels of each of the three feature channels intensity, color and orientation are combined to the three corresponding *conspicuity maps* I, C and O. For example, this means that all four color feature maps C' are added together to the color conspicuity map $C = \sum_i C'_i / \sqrt{m_{C'_i}}$. In this summation each subchannel feature map C'_i is weighted by the reciprocal square root of the number $m_{C'_i}$ of local maxima within the map whose strength is above the median strength of these local maxima. The fusion of the feature maps is visualized in Figures 3, 4 and 5.

Before the three conspicuity maps can be combined in the final stage they must be normalized in some way since they consist of different numbers of feature maps. We follow the suggestion of Frintrop [21] and normalize each conspicuity map with the largest local maximum $\hat{m}_i$ from all feature maps of the respective channel such that the range of the conspicuity map is $[0, \hat{m}_i]$. The normalized conspicuity maps are finally weighted and added in the same manner as for their construction yielding the saliency map $S = I/\sqrt{m_I} + C/\sqrt{m_C} + O/\sqrt{m_O}$. The m_X in this computation are again the numbers of local maxima above the median strength within the corresponding conspicuity map X. The final saliency map construction is illustrated in Figure 6.

The procedures described in this section essentially represent the bottom-up saliency computation presented by Frintrop [21]. Because we observed an augmented detection of object regions in some cases we added the saliency map of Hou and Zhang [26] as a fourth conspicuity map to this framework. This method is explained in the next section.

3.1.1.2 The Additional Conspicuity Map Based on Spectral Residua

The saliency map computation of Hou and Zhang [26] works on the grayscale image. First, the grayscale image is transformed into the frequency domain via Discrete Fast Fourier Transform. In the frequency domain the logarithmic magnitude image M_{log} is generated. As described above the idea of this method is to remove the common information from an

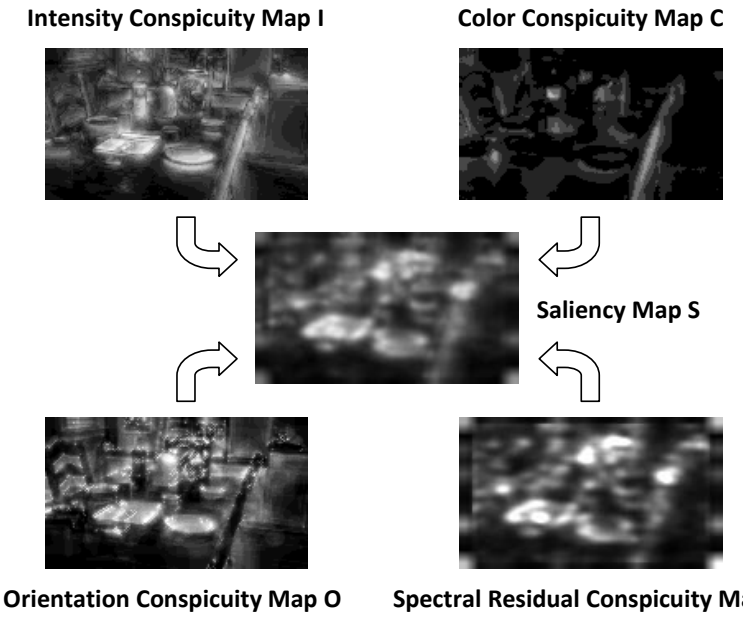


Figure 6: Visualization of the construction of the saliency map out of the conspicuity maps.

image leaving the salient remainder. Hou and Zhang found that the logarithmic magnitude curve is very similar in its general appearance for most of the studied images. However, it has some small peaks on the logarithmic magnitude curve which differ between the images. The reasoning is consequently that the removal of the common logarithmic magnitude part should only leave the innovative parts of the image. Therefore, the obtained logarithmic magnitude image M_{log} is smoothed with a box filter. The difference between M_{log} and the smoothed version $\overline{M}_{log}$ is finally transformed back into the image domain with the inverse Discrete Fourier Transform. The obtained image is the saliency image since it only contains bright areas at the salient regions.

We use this method at different scales of the grayscale image by applying it to the respective images from the Gaussian pyramid. Thus, we obtain a spectral residual saliency pyramid as a fourth feature channel which we can deal with within the framework described in section 3.1.1.1 in the same way as with the other three feature channels. This is also indicated in Figure 6 where the spectral residual saliency map is displayed for the example image.

3.1.1.3 Selection of Salient Regions and Their Scales

After its computation we use the saliency map in order to find salient regions in the field of view. Therefore, we first search for all local maxima in the saliency map and order them with decreasing strength in a priority queue. The queue then allows a fast access to the strongest N local maxima. Sequentially, the strongest of the remaining maxima in the queue is picked and then its scale is computed by determining the feature map in scale space which has the largest response for that location. In order to speed up this process and obey the imposed normalizations, we first select the conspicuity map with the highest response, then the subchannel feature map and finally the scale map within that feature subchannel. For computation speed reasons we decided to describe the salient region as a circular region with a radius r dependent on the found scale s of the salient point.

$$r = \frac{\text{width}(I)}{128} \cdot 2^s$$

The first factor in this formula includes the width of the original image I in order to keep the region size independent of the actual resolution of the input data. The scaling by a factor 1/128 was tuned by hand in order to have a reasonable size which encircles sensible areas like whole objects or prominent object parts. The choice of this definition for the radius also implies to use feature maps of the scales 2 to 5 in order to capture small and huge objects in the scene likewise.

Before a potential new salient region is finally accepted, we furthermore ensure that the distance to former regions is big enough to find another region and not the same again. We obtained reasonable results when the distance of a new point q to each already accepted point p^i satisfies the following inequality in which the components of a point are the image coordinates (p_x, p_y) as well as the scale p_s.

$$\sqrt{(p_x^i - q_x)^2 + (p_y^i - q_y)^2 + \left(\frac{\text{width}(I)}{256}\right)^2 (p_s^i - q_s)^2} \;>\; q_s \frac{\text{width}(I)}{40}$$

This measure is independent of the real image resolution on the one hand and furthermore allows regions of very different scales to be closer to each other than regions of the same scale. This is reasonable since it does not prevent the algorithm from selecting distinctive

Figure 7: The saliency map and the obtained salient regions for the example image.

object parts and the whole object at the same time while the same region is never selected twice in one image. The numbers in this inequality were obtained by hand tuning and manual inspection. For our example image we obtain the result displayed in Figure 7.

3.1.2 Descriptors

Having found a set of interesting regions in the image, it is necessary to describe them in a way that allows matching with similar image patches and distiction from different ones. Therefore, the following set of feature descriptors is studied for the suitability to this task. Several ways how to combine the information from different features in the process of deciding for a place category are discussed in section 3.3.

Obviously, humans can identify objects or scenes quite well even if only a few contour lines are provided. Therefore, we have a closer look at the performance of different structural features.

Mean and Variance of the Orientation Feature Maps A very basic and computationally cheap feature is the *mean* and *variance* of the *orientation feature maps*, which are already computed anyway, in the respective scale of the salient region. We apply the sampling scheme which takes the mean and variance once over the whole salient area and again in nine smaller squares which are aligned in a regular 3x3 grid within the region. Thus, the descriptor has a length of 80 dimensions. The motivation for this feature is the following: Should such an orientation feature map exist in the human brain (c.f. fig. 1 in [70]) then it would likely be used for descriptive tasks besides the search for regions of attention. The

question is whether there are only four kinds of receptors for different angular alignment of lines and whether this data is described by simple means and variances. We examine in section 4.4 whether this feature is too general for the variety of objects or whether it is general enough to leave out those details of objects which might prevent the classifier from proper generalization. The mean should be interpreted as a general amount of edges in the respective directions while the variance is a measure for the clutter of the edges. A high variance indicates many tiny edges whereas a low variance rather describes bold or no edges.

Histogram of Oriented Gradients A more sophisticated descriptor which was successfully applied in pedestrian and object recognition [13] is *Histogram of Oriented Gradients* (HOG). We use the HOG implementation provided by OpenCV [5] on the whole salient region. Therefore we take the grayscale image from the Gaussian pyramid at the respective scale and convert the region of interest to a 64x64 pixel image patch. On this patch we apply the HOG descriptor with the parameters window size 64, block size 64, block stride 32, cell size 16 and number of orientation bins 8. We chose these parameters since they yield a descriptor of 128 dimensions which we do not want to exceed especially for reasons of the curse of dimensionality during the classification stage and because of the amount of available training data. Because of the multi-scale implementation, this HOG descriptor should be able to detect an object at different scales.

SIFT [40] has shown strong results on many different domains like object recognition [39], image stitching [71] and visual SLAM [68]. Similar to the HOG descriptor, SIFT computes the local gradients in a regular 4x4 grid. However, SIFT is rotationally invariant in addition which could be useful for the detection of moveable household objects. Nevertheless, most of the objects in a home environment have a common pose or are fixed completely. Furthermore, SIFT was developed with the goal to detect the same object patches again and discriminate them from other. We therefore examine how suitable the SIFT descriptor is when more generalization to similar objects is necessary.

We compute a SIFT descriptor for the whole salient region on the grayscale image from

the Gaussian pyramid at the respective scale. For computing the descriptor we used parts of the SIFT implementation of Rob Hess [24]. The descriptor is computed on a regular 4x4 sqares grid with 8 orientation bins per square yielding a 128 dimensional descriptor. The same reasoning as for HOG applies to the choice of the descriptor size.

CENTRIST [82] is a new descriptor developed rather for categorization tasks than exact matching. Therefore, we evaluate whether it obtains a better performance especially in comparison to SIFT. It is based on histograms over the census transform [83] of the intensity or Sobel image. In our case, the CENTRIST feature on salient regions provides a better performance if applied to the intensity image from the Gaussian pyramid at the corresponding scale. The census transform generates a 8 bit binary pattern for each image pixel whose ones are set each time when the intensity of the central pixel is higher than the corresponding pixel neighbor in a 3x3 neighborhood. There are 256 possible binary patterns which are just represented by a number. A histogram over the numbers of each pattern forms the CENTRIST descriptor. Due to this computation it has a dimensionality of 256.

Although shape is very important for object recognition, some objects like household sponges generally have strong color constraints as well. Consequently, the combination of the structural descriptors together with color information might improve the overall performance.

Mean and Variance of the Color Feature Maps Taking the *mean* and the *variance* of the four *color channels* red, green, blue and yellow is a very unsophisticated feature which we examine for its descriptive power. The motivation for this choice of feature is similar to the orientation mean and variance: the maps are already computed and their information should be exploited in some way. Therefore, we take the mean and the variance of each of the four color channels once from the whole salient region and nine times from a regular 3x3 grid of squares inside the region. The resulting descriptor has a dimensionality of 80. The means of the channels can be interpreted as the color of an object while the variance can

be seen as a measure for the colorfulness. The lower the variance the more monochromatic is the image patch.

3.2 Global Image Descriptors

As mentioned above, part of the quick human categorization capabilities are due to the rapid extraction of the gist of the scene. Accordingly, it looks fruitful to apply a holistic image descriptor as well which does not care about any scene details.

Oliva and Torralba [53] proposed a scene descriptor which categorized scenes by evaluating several perceptual properties like naturalness and clutter. However, it showed that at least two of these criterions cannot be discriminative in indoor scenes. Siagian and Itti [69] presented a place classification system which uses features obtained from the feature maps constructed for the saliency computation. As well as Oliva and Torralba [53], they call this feature *Gist*. However, they only examined the performance of their system outdoors and focused on localization tasks. Since we already have similar feature maps from the visual attention mechanism, it is tempting to verify the performance of Siagian's and Itti's Gist feature on indoor categorization problems. In detail, it is computed as the mean of each feature map within the 16 cells of a regular 4x4 grid. This means, every color, orientation and intensity feature map from each scale contributes with the mean of its activation within a grid cell. Since there are $2 \cdot 4 \cdot 2$ (subchannels, scales, radii) intensity feature maps, $4 \cdot 4 \cdot 2$ color feature maps and $4 \cdot 4$ orientation maps we obtain a descriptor with $64 \cdot 16 = 1024$ dimensions. This very long descriptor is optionally shortened via principal component analysis depending on the utilized classifier.

We compare the results of the Gist classifier with the results of *CENTRIST* which has already been investigated thoroughly in [80]. Therefore, we employ the same configuration as presented there. This is in detail the computation of the Laplacian edge image which is divided into a regular 4x4 grid. A CENTRIST histogram is then computed for each cell. The idea behind this is to describe the global structures in the image which are mainly represented by the contained edges.

3.3 Classification

At this point we have collected local and global descriptive data from several features. The next task is the classification of this data. There are many possible classification strategies from which we investigate the following ones.

1. Each salient region descriptor and each whole image descriptor can be treated as a separate cue which is directly and independently classified by its own multi-class classifier yielding a set of classification results for the place category.

2. Another approach, which also incorporates the relationships between the salient areas, is to cluster their descriptors and use the cluster representations to parameterize a probability distribution over the place categories.

3. Finally, in both cases there are multiple classification results after the first level of single cue classification which might yield a stronger result if they are combined.

All three cases shall be discussed in more detail below. Please notice that we decided to use the descriptors directly to characterize the places since the building of explicit intermediate representations for object classes is almost impossible from the automatically collected data.

3.3.1 Direct Multi-class Classification of Single Cues

The simple concatenation of all extracted features and their following classification would have three severe drawbacks. First, having such a high-dimensional descriptor requires a tremendously huge dataset for reasonable classifier training and slows down the training even on smaller datasets. Second, there is no way to order the salient regions so that the same object always fits into the same segment of the long descriptor. The same region is rather found as the first region in the one image and as the third region in another. It is unneccessary that the classifier learns this variability. Third, the classifier would learn only very specific constellations of salient region features among each other and in conjuction with the whole image descriptor. Thus, the generalization abilities must be expected low for this procedure even if trained with a huge dataset.

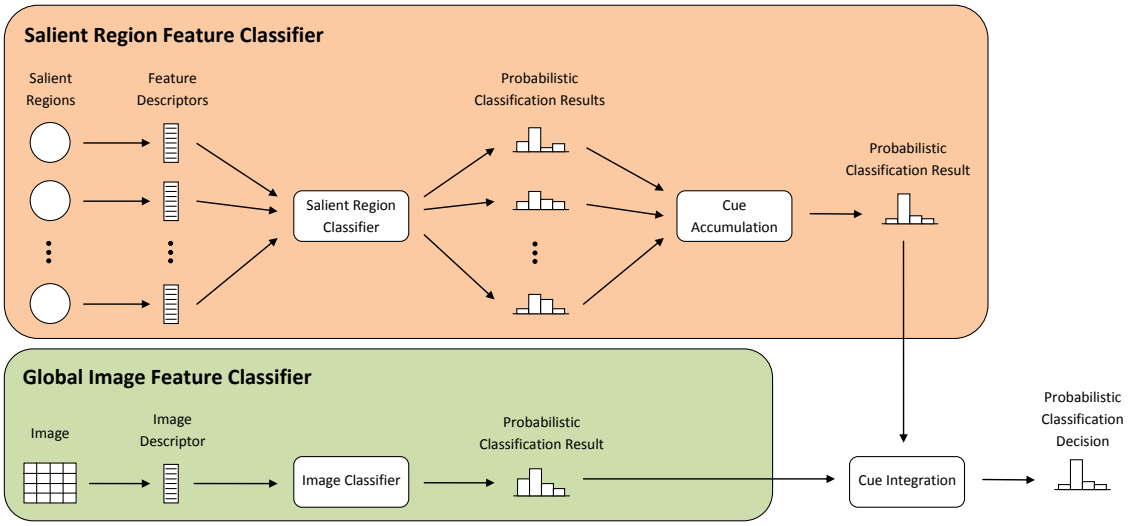


Figure 8: Decision process for direct multi-class classification.

A better alternative is to classify each set of individual short descriptors and fuse the results in one of the ways presented in section 3.3.3. In this approach we collect the descriptors separatedly and train one classifier for the whole image features and one or more for the single cues obtained from the salient regions. Figure 8 illustrates this procedure. In case of the salient regions feature, we first extract a descriptor for each region, classify each descriptor with a probabilistic output and fuse those outputs, for example by a simple voting scheme or the multiplication of all probability distributions. The resulting probability distribution is provided to the cue integration scheme (c.f. section 3.3.3) together with the probabilistic classification output of the whole image feature. The integration scheme finally outputs the categorization decision. Please notice that we could classify the salient regions with different single cue descriptors using one salient region classifier module per descriptor type since the final integration step can handle an arbitrary amount of cues.

For the multi-class classification we enabled the software to use either a modified K-Nearest Neighbor (KNN) classifier or a multi-class Support Vector Machine (SVM), Relevance Vector Machine (RVM) or AdaBoost classifier. We explain both variants in the next sections.

3.3.1.1 Modified K-Nearest Neighbors

One of the simplest classification methods which even provides arbitrary multi-class support is K-Nearest Neighbors. However, we did not apply KNN directly to the place categorization problem for two reasons. First, we have around one million data points within the training data which would slow down the runtime prediction very much if no approximation algorithm is used. However, approximation algorithms like [48] often build a tree for faster access to the nearest points which prevents these methods from easy updating when more data becomes available during the runtime of the robotic system. Second, as we want to find similar but not necessarily the same objects which give hints for the room category the classifier has to be tolerant to noise to some extent. Within a normal K-Nearest Neighbors setting this can only be achieved with fancy distance weighting measures which weight close points almost equally and points far away are discarded, similarly as in mixture of Gaussian models. This would affect the runtime negatively, again.

Since we want to present at least one classifier which can be updated easily during runtime without huge recomputation, which is quick enough during runtime, which can output a good estimate of the probability distribution of the place category and which can still deal with some noise, we decided to employ the following KNN modification. The idea is to cluster the provided data into centers which consist of many data points. Those data points give rise to a probability distribution over the places for each cluster. Because there are many fewer centers than data points, we can achieve good runtime performance with this method. Furthermore, this concept naturally yields a probability distribution without any additional computation during runtime. The noise problem is covered through the averaging within the centroid as well. Finally, an update with new data is easily possible by either adding a new point to a cluster or adding a new cluster if no other center is close to that data.

In detail, the training procedure is the following. The provided training data is initially clustered into m clusters with the hierachical k-means clustering provided in the FLANN library [48]. Then we compute the frequency distributions $f(l_i, c_o)$ of the obtained centers by counting the number of samples for each room class l_i contained in each cluster $c_o, o \in$

$[1, \ldots, m]$. We do not compute the actual probability distribution at this point because this normalization is done at the end of the prediction process anyway and because this approach makes data updates with further data very easy.

The prediction of a probability distribution for the most likely place category is computed for a set of n query points, for example n descriptors from n salient regions of an image, by finding the closest center or the closest K centers for each of the n descriptors. For now let us assume that we only consider one nearest centroid c_k to each descriptor $x_k, k \in [1, \ldots, n]$. With the individual frequency distributions $f(l_i, c_k)$ of the centers we can characterize the likelihoods for each place category l_i. In order to obtain the resulting probability distribution $p(l_i|x_1, \ldots, x_n) = p(l_i|c_1, \ldots, c_n)$ from the n individual distributions $f(l_i, c_k)$ over the places we just multiply these n frequency distributions. This approach preserves uncertainty as well as certainty in the final distribution. In order to compensate for unabalanced training data we divide each entry of the final probability distribution by the number of examples from this class in the training set before it is finally normalized. This method is justified for $K = 1$ nearest neighbor under the naive Bayes assumption that the n salient regions are independent of each other and under the assumption of uniform place category priors $p(l_i)$ during runtime by the following considerations.

$$p(l_i|c_1, \ldots, c_n) = \xi \prod_{k=1}^{n} p(c_k|l_i) \tag{1}$$

ξ is the normalization constant and the terms $p(c_k|l_i)$ are computed from the training data where $f(l_i)$ is the number of samples from place category i and f is the number of samples at all.

$$\begin{aligned} p(c_k|l_i) &= \frac{p(c_k, l_i)}{p(l_i)} \\ &= \frac{f(c_k, l_i)}{f} \frac{f}{f(l_i)} \\ &= \frac{f(c_k, l_i)}{f(l_i)} \end{aligned}$$

If this is put into equation (1) and with $f(c_k)$ denoting the number of points in center k we

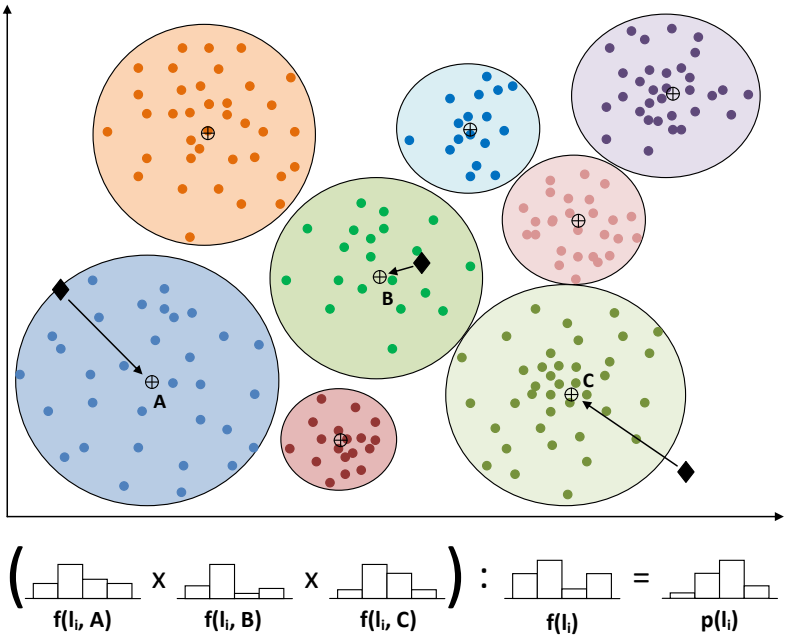


Figure 9: Overview over the KNN classifier with $K = 1$ nearest neighbor. The small dots symbolize the descriptors from the training data. They belong to some cluster which is indicated by a colored circle. Each of the clusters collects the frequency statistics about the class membership of its contained descriptors. A query with three points is indicated by the black rhombuses. In the $K = 1$ nearest neighbor setting these points are associated with the closest centroid and obtain their probability distribution. For the category decision from this query, the three distributions are multiplied element-wise and divided by the category frequencies as shown in equation (2).

obtain

$$p(l_i|c_1,\ldots,c_n) = \xi \frac{1}{f(l_i)} \prod_{k=1}^{n} f(c_k, l_i) \tag{2}$$

$$= \frac{\tilde{\xi}}{\alpha} \prod_{k=1}^{n} \alpha \frac{1}{f(l_i)} \frac{f(l_i, c_k)}{f(c_k)}, \quad \tilde{\xi} = \xi \prod_{k=1}^{n} f(c_k) = \text{const}$$

$$= \frac{\tilde{\xi}}{\alpha} \prod_{k=1}^{n} p(l_i|c_k) \tag{3}$$

Here we can see that it does not matter whether we divide by the class frequencies $f(l_i)$ and normalize the frequency distributions $f(c_k, l_i)$ already in the centers to cluster-related place probability distributions $p(l_i|c_k)$ or whether we do this later when the probability distribution to a set of n query points is searched. The method for $K = 1$ is illustrated in Figure 9.

For the case of multiple nearest neighbors $K > 1$ there is no equality between equations

(2) and (3) because instead of the direct computation from the nearest cluster as seen for $p(l_i|c_k)$ in equation (3) we would have to interpolate the local probability distribution $p(l_i|x_k)$ for each descriptor x_k from the K nearest neighbors (= set $\mathcal{N}_K(x_k)$) as

$$p(l_i|x_k) = \alpha \frac{1}{f(l_i)} \sum_{c_j \in \mathcal{N}_K(x_k)} \frac{f(l_i, c_j)}{f(c_j)} w_{k,j}$$

$$w_{k,j} = \begin{cases} \frac{1}{\|x_k - c_j\|} & , \text{if } c_j \in \mathcal{N}_K(x_k) \\ 0 & , \text{else} \end{cases}$$

In this formula $w_{k,j}$ is a weighting function which returns a value indirect proportional of the distance between x_k and the centroid c_j. We can see that the term $f(c_j)$ cannot be pulled out of the sum so that it cannot be part of the general normalization at the end (i.e. factor ξ) as supposed in equation (2). However, the alternative is to follow the derivation shown above until equation (2) and substitute the term $f(l_i, c_k)$ by a term $f(l_i, x_k)$ which is interpolated from the K nearest centroids to x_k. The set of these clusters is named $\mathcal{N}_K(x_k)$, again.

$$p(l_i|x_1, \ldots, x_n) = \xi \frac{1}{f(l_i)} \prod_{k=1}^{n} f(l_i, x_k) \tag{4}$$

$$f(l_i, x_k) = \sum_{c_j \in \mathcal{N}_K(x_k)} f(l_i, c_j) w_{k,j} \tag{5}$$

This approach has the advantage that it can be derived from $p(l_i|x_1, \ldots, x_n)$ under the given assumptions and furthermore it incorporates not only the distance between the query point and the surrounding centers but also weights those clusters proportionally more which consist of a higher number of sample points. In contrast, the first variant would only weight the cluster's influence by their distance to the query point x_k.

If data should be added during runtime because a human observer has provided labels for an observation or corrected the robots belief, this can be done very easily. First, we have to decide whether the new point should be added to an existing cluster what effectively happens if it is very close to a cluster or when the maximal number of clusters is limited and the query point is closer to some cluster than the distance between any two clusters. Then the respective class counter for the cluster's frequency distribution is incremented

and the center of mass of the cluster is recomputed. In the other case that the query point is far away from all clusters, it is also possible to fuse two close clusters and build a new one for the new point. In all cases the stored distance between all cluster pairs is updated efficiently only for those clusters where an update is necessary.

In general, the choice of the number of clusters for this classifier can be considered as the trade-off between detail information and generalization performance - the more clusters we allow the more details can be distinguished but the less general classes are represented by the centers.

3.3.1.2 AdaBoost, SVM and RVM

Since SVM, RVM and AdaBoost implementations are generally only available as two-class classifiers, we have to extend these basic classifiers for the multi-class case. We decided for a one-against-all scheme which trains one basic classifier for each class discriminating this single class from the remaining classes because this method needs to train less classifiers than an one-against-one scheme. As we obtain a certainty degree (explained below) from each two-class classifier for its decision, the multi-class output could be found by choosing the result of the classifier with the highest certainty. However, this method would fail systematically if one classifier always outputs the wrong decision with the highest certainty. Another approach is the construction of a decision cascade beginning with the strongest classifier. Mozos [46] used such a sequential scheme for place categorization and build some probability-like decision degree histograms from the outputs of the basic classifiers.

We apply a novel decision scheme which computes a real probability distribution for the class to choose. In contrast to the probability-like ouput of [46], the proposed scheme incorporates the different reliabilities of the classifiers in a more principled way. Assume there are N different two-class classifiers, each for one of the N classes. Presented with a data sample x their outputs are the certainties (probabilities) $p(o_1|x) \dots p(o_N|x)$. Let $L = l_1, \dots, l_N$ be the probability variable for the actual class. Using the short form $p(l_i|x) =$

$p(L = l_i|x)$, the probability for class i is then

$$p(l_i|x) = \sum_{k=1}^{N} p(l_i|o_k, x)p(o_k|x) \tag{6}$$

We approximate

$$p(l_i|o_k, x) \approx p(l_i|o_k) = \frac{p(o_k|l_i)p(l_i)}{p(o_k)} \tag{7}$$

The decision reliability term $p(o_k|l_i)$ is determined from statistics from cross-validating the two-class classifiers, the class frequency $p(l_i)$ is obtained from the training dataset. Then we can calculate $p(o_k) = \sum_{i=1}^{N} p(o_k|l_i)p(l_i)$.

For the basic classifiers we used the OpenCV [5] implementation for AdaBoost and the implementations for SVMs and RVMs of the dlib library [31]. Dlib already provides a function for training a probabilistic decision function for the SVM or RVM (see dlib documentation). For AdaBoost we generate a probabilistic decision between the two classes by applying the certainty measure of Friedman *et al.* [20]

$$p(o_k|x) = \frac{\mathrm{e}^{F(x)}}{\mathrm{e}^{-F(x)} + \mathrm{e}^{F(x)}} \tag{8}$$

where $p(o_k|x)$ is the certainty mentioned above for the classifier of class k asserting a positive sample when presented with sample x, that is the probability that x belongs to class k. The weighted sum of the weak classifiers employed in the boosting framework is denoted with $F(x)$ which is negative for the negative class, positive for the positive class and close to zero if the decision is unsure.

For the training of the boosting classifiers we utilize Gentle AdaBoost with the standard parameters suggested by the OpenCV manual because of its numerical stability. The SVM is trained with the ν-SVM algorithm [66].

We added an uncertainty smoothing to the multi-class classifier which can be activated optionally. This function allows to smooth the output probability distribution indirectly proportional to the certainty of the most certain basic classifier. The additional probability mass

$$\beta = \gamma \left(\frac{1}{\max_k p(o_k|x)} - 1 \right) \tag{9}$$

is added to the returned probability distribution before it is normalized. The parameter γ symbolizes the additional probability mass when the most certain basic classifier was certain to 50%. This smoothing operation effectively allows to incorporate uncertainty in the way that a nearly uniform distribution is returned if all two-class classifiers output a low probability.

3.3.2 Clustering and Learning of a Probability Distribution

The multi-class approach presented in the preceding section does not incorporate the potential dependencies between the found salient regions. Imagine that the method found a cup in one region and a keyboard in another. The independent analysis of the regions of the former method might give kitchen a high probability for finding a cup and office a high probability for finding the keyboard. However, the dependency between both objects clearly suggests that a cup might appear close to a keyboard when both are located in an office while it is unlikely to find a keyboard inside the kitchen. The difference is that in the second case the decision for office can be made with much higher certainty.

Therefore, we also evaluate the performance of the following approach for the interpretation of the salient regions. We cluster the region descriptors with the hierachical k-means clustering of the FLANN library [48] in the hope to find meaningful intermediate representations which form a codebook of N objects or object parts. Then we can describe a place by the constellation of found objects and compute the place probabilities $p(l_i|c_1 = q_1, c_2 = q_2, \ldots, c_N = q_N)$ where l_i defines the actual place and $c_k, k \in [1, N]$ denotes the individual clusters from the codebook with $q_k = 1$ if the corresponding object was found in the image and 0 if not.

Of course, for larger codebooks we cannot learn the complete joint probability for the clusters so that we have to make an approximation. Therefore we first use Bayes' rule to make the joint probability accessible.

$$p(l_i|c_1 = q_1, c_2 = q_2, \ldots, c_N = q_N) = \frac{p(c_1 = q_1, c_2 = q_2, \ldots, c_N = q_N|l_i)p(l_i)}{p(c_1 = q_1, c_2 = q_2, \ldots, c_N = q_N)} \tag{10}$$

The simplest approximation which still maintains the dependency information and which can be reliably computed from the number of examples found in common datasets is the

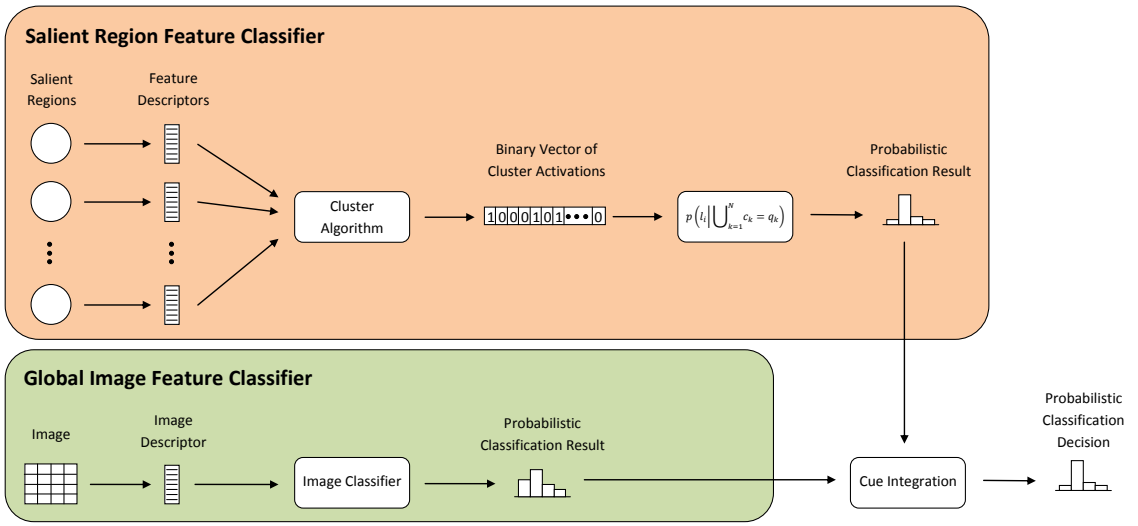


Figure 10: Decision process for classification with cluster configurations occurring in the scene.

first-order dependency approximation which can be computed optimally [9].

$$p(c_1 = q_1, c_2 = q_2, \ldots, c_N = q_N | l_i) \approx p(c_r | l_i) \prod_{k \in [1,N] \setminus r} p(c_k | c_{\Pi(k)}, l_i) \tag{11}$$

In this approximation, r is the root of the optimal dependency tree and $\Pi(k)$ denotes the index of the parent node to node k. All occurring probabilities can then be estimated from the training data. The output of the salient regions classifier is the probability distribution $p(l_i | \bigcup_{k=1}^{N} c_k = q_k)$. A scheme of this method is displayed in Figure 10. Again, the use of several global or salient region descriptors is possible with the integration scheme explained in the next section.

If we would like to use a simpler approximation to the probability distribution $p(c_1 = q_1, c_2 = q_2, \ldots, c_N = q_N | l_i)$, we would have to choose the naive Bayes approach which however asserts independence between the individual codewords.

$$p(c_1 = q_1, c_2 = q_2, \ldots, c_N = q_N | l_i) = \prod_{k=1}^{N} p(c_k = q_k | l_i) \tag{12}$$

This approach is different from the KNN classifier with $K = 1$ nearest neighbors explained in section 3.3.1.1 since the KNN approach only considers the probabilities of the activated subset (size n) of all N clusters for the computation of $p(l_i | c_1, \ldots, c_n)$ whereas the learning of a probability function $p(l_i | c_1 = q_1, c_2 = q_2, \ldots, c_N)$ always considers both, the probabilities

of the activated clusters as well as the probabilities of the not activated clusters. This means, that the latter approach also incorporates probabilities about the objects not found in the scene whereas the KNN method only decides on the basis of found objects.

3.3.3 Feature Integration

Having classified the single cues, the question arises how to make a final place category decision. A simple approach is voting which means that every cue votes for a decision either with its probabilistic weight for the maximum likelihood estimate or with the whole distribution. The final decision consequently falls on the strongest vote. Another more elaborated method would be to develop a probabilistic integration scheme. However, Nilsback and Caputo [51] have shown that a discriminative accumulation scheme (DAS) outperforms probabilistic integration models like [6]. Pronobis *et al.* [58] recently used the strong performance of SVM-DAS in accumulating the results of laser and image features for place classification. The SVM-DAS scheme is essentially the procedure of feeding the outputs of the single cue classifiers into a SVM which outputs the final categorization decision. Because of the good performance and the simple implementation (we have already used the multi-class classifier earlier, see section 3.3.1) we utilize the SVM-DAS algorithm for the integration of the several local and global cues presented before.

3.4 Smoothing Filter

As other work on place recognition and categorization has shown [47], it is helpful to smooth the output by using a Hidden Markov Model (HMM) for modelling the place transitions because of noisy classification results in some intermediate frames. To avoid infeasible immediate jumps between place categories, we apply a HMM for smoothing the decision sequence as presented in [47, 73].

$$p(l_t = q|o_{t:1}) = \gamma \; p(o_t|l_t = q) \sum_{q'=1}^{N} p(l_t = q|l_{t-1} = q', o_{t-1:1}) \; p(l_{t-1} = q'|o_{t-1:1}) \qquad (13)$$

The probability $p(l_t = q|o_{t:1})$ for being in place q at time t can be computed from the transition model $p(l_t = q|l_{t-1} = q', o_{t-1:1})$, the distribution of the former place $p(l_{t-1} = q'|o_{t-1:1})$ and the classifier reliability $p(o_t|l_t = q)$ which can be obtained from the DAS

multi-class classifier statistics. The transition model is estimated from the training image sequences. However, it could also just be set to feasible values manually. The normalization of $p(l_t = q|o_{t:1})$ is represented with the constant γ.

Although the smoothing of a HMM can fix one or two false decisions we observed that sometimes the interrupting sequences of wrong decisions are slightly longer and of course dependent of the framerate of the image sequence stream. In order to influence the update speed of the HMM we additionally model the update process as an asymptotically stable first-order time-delay system with set point input from the HMM. For each category l_q at time t we have the probability $p_q(t)$. The output of the HMM from equation (13) is considered as a set point input into the system $p_q^*(t) = p(l_t = q|o_{t:1})$. Then we have the dynamic system with the input u

$$\dot{p}_q = \alpha p_q + u \tag{14}$$

which we have to convert into the discrete form with the discretizing time step h

$$\frac{p_q(t+1) - p_q(t)}{h} = \alpha p_q(t) + u(t) \tag{15}$$

The update dynamic finally follows the equation

$$p_q(t+1) = p_q(t) + h \cdot (p_q^*(t) - p_q(t)) \tag{16}$$

where we have set $\alpha = -1$ for the stability of the autonomous system and the input represents the output of the HMM $u(t) = p_q^*(t)$. The final classification output of the system is $p_q(t+1)$. The differences between using no smoothing or HMM only or HMM and the additional smoothing is indicated in Figure 11.

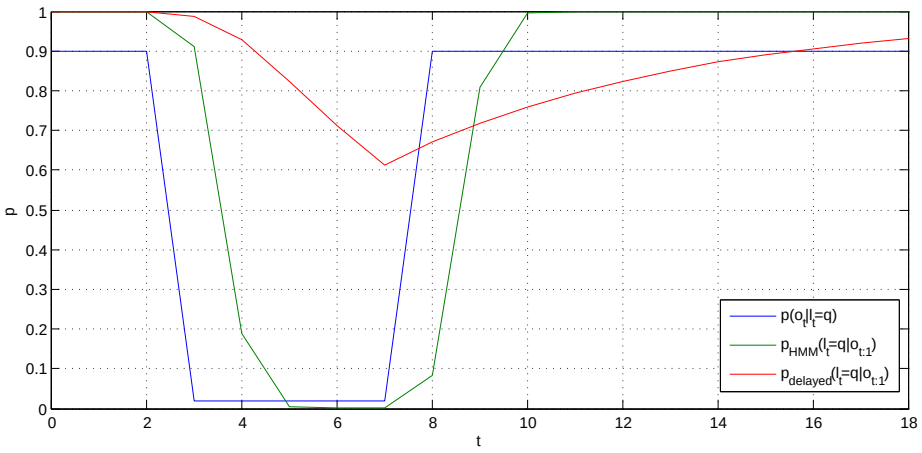


Figure 11: Example showing the outputs for different kinds of smoothing. While the output of a classifier $p(o_t|l_t = q)$ might exhibit arbitrary jumps the HMM $p_{HMM}(l_t = q|o_{t:1})$ can filter out jumps for one time step. When the additional delay system is employed the output $p_{delayed}(l_t = q|o_{t:1})$ can be smoothed further.

CHAPTER IV

EXPERIMENTS

In this section we describe the experiments which assess the performance of the visual place categorization system presented above. After introducing the databases which are used for the evaluation, we discuss the experiments in detail.

4.1 Databases

Currently, there exist only two serious datasets for visual place categorization on the domain of autonomous robots. These are the publicly available COLD dataset [55], which covers university environments, and the home environment dataset of Wu *et al.* [81]. Furthermore, we captured several additional image sequences in home environments using a Segway mounted robot, a HD video camera and a webcam.

Home database The largest dataset captured up to now with respect to the variability of appearance of the recorded places is the home environment dataset of Wu *et al.* [81] which contains image sequences from six very differently furnished homes. The videos were taken with a tripod mounted video camera once in every home. Each sequence consists of 6000 to 10000 images with a resolution of 1280x720. The only modifications to the normal look of the houses are the removal of personal items and the closing of the blinds to avoid external influences. This dataset is well-suited for extensive performance tests in home environments and therefore we do the mainpart of the parameter search and evaluation on this set.

COLD database The publicly available COLD database [55] covers university places across three different universities. It contains a variety of office environment places captured under three different weather conditions (sunny, cloudy, night) by a robot carrying the same camera set up each time (resolution 640x480). At Freiburg and Saarbrücken, sequences were

taken from two parts of a building, at Ljubljana one part was used. For each part, there exists a standard and an extended robot trajectory capturing varying numbers of room categories. Moreover, each scanpath was taken several times in order to record some visual variability originating from normal office activity. This database can be used for systematic visual place categorization experiments when one university is kept for testing each time. However, the quality of the Ljubljana dataset is lower because the camera is mounted very high on the robot.

Aware Home We additionally recorded several videos in the Georgia Tech Aware Home which contains two fully furnished living apartments on two floors. We first used a tripod mounted HD video camera on a chair to capture 6164 images downstairs and 3257 images upstairs. Furthermore, we used our Segway RMP-200 mobile platform robot to capture an image sequence with 5700 images downstairs to be able to verify the performance of the system on a real robot.

Apartment Finally, we used a standard webcam to record 1889 images in an apartment. This dataset is used to verify the robustness of the system when other camera hardware is used and different movements occur.

The COLD database and the home database were downloaded to allow comparisons to other methods presented in the respective papers of the databases whereas the additional self-captured image sequences have the purpose of further evaluation on possibly quite different environments. In the next sections we check the performance of the different approaches with varying descriptor and classifier settings on the Home dataset since it is the most comprehensive of all.

4.2 Experiments on the Gist Feature

The Gist feature of Siagian and Itti [69] proved a good performance for outdoor place recognition tasks. Within this section we want to evaluate the power of the Gist descriptor in indoor environments as this has not been done before. In contrast to other holistic

features like the Gist feature of Torralba *et al.* [73] or the Centrist descriptor [81], which only operate on the intensity information of the grayscale image, the Gist feature of Siagian and Itti [69] actively incorporates the intensity and the color information.

The analysis is based on the Home dataset and is extended to other datasets in sections 4.9 and 4.10. Although the Home dataset contains 12 place categories we only consider the five room classes kitchen, bathroom, living room, dining room and bedroom which can be found in every subset. This corresponds with the procedure in [80] and makes the results comparable. For every test the categorization system is trained with five of the six homes and tested with the remaining one. Due to the runtime of a test cycle, which lasts between 30 and 120 minutes on a laptop[1] depending on the training time of the classifier, only Home 1 and Home 6 were randomly chosen as the test subsets during the basic parameter tweaking and classifier selection process. We furthermore process only every third image of the sequences in training mode because of RAM restrictions and the speedup of the training phase due to less image processing classifier training data. The high frame rate of the video stream and the results obtained for the Centrist descriptor justify this procedure, see section 4.3. Then we have to process around 8000 images for each training cycle instead of ca. 24000. The tests are always carried out with every image of the sequence to preserve the comparability. The processing framerate usually ranges between 3 and 4 Hz.

We begin with the evaluation of the Gist feature in the original formulation which means that every feature map from each scale contributes to the descriptor with its means from the grid cells of a 4x4 grid. Effectively, the 1024-dimensional descriptor consists of intensity feature map means to 25%, of orientation feature map means to 25% and of color feature map means to 50%. If not mentioned explicitly, this descriptor is not shortened via Principal Component Analysis as proposed by Siagian and Itti. The following evaluation compares the performance of the different descriptors explained in section 3.3.

Modified K-Nearest Neighbors The modified K-Nearest Neighbor classifier was applied to the Gist descriptor with varying numbers of clusters. The results for $K = 1$ nearest

[1] Intel Core 2 Duo processor P9500 with 2x2500 MHz, 4 GB RAM

Table 1: Performance of the K-Nearest Neighbor algorithm on the gist descriptor. The influence of varying numbers of clusters is shown.

Number of Clusters	46	91	196	496
Home 1	25.64	24.08	26.29	29.71
Home 5	34.40	31.14	31.76	35.91
Average	30.08	27.61	29.02	32.81

neighbor are shown in Table 1. We always report the percentage of correctly classified images in relation to all images of the respective class and calculate the overall performance as the average of the individual class accuracies since this avoids that big classes can shadow bad results of smaller categories. The range of examined intermediate clusters is justified by the following reasoning. If we want to represent a probability distribution with each cluster, we need to assign at least a small number of data samples to every center. Since the training video sequences consist of around 8000 images, we could assign an average number of eight samples to each cluster when there are 500 clusters. Going beyond this number does not make sense having this consideration in mind. The actual numbers of clusters used in this and the following evaluations is influenced by the hierarchical clustering method of the FLANN library [48] which generates cluster numbers which satisfy the formula $(b-1)\cdot k+1$ where $b=16$ is the user-defined branching factor of the k-means tree and k is an arbitrary number. Because the initial seeding of the clusters is obtained by a randomized algorithm, namely K-Means++ [2], the reported accuracies were obtained as the average of two simulations on different computers with different operating systems. The hierarchical clustering is employed because it regularly outperformed standard k-means clustering by up to 5% during preliminary experiments.

Gentle AdaBoost The test with Gentle AdaBoost required the use of PCA to reduce the gist descriptor size because of the very long training phase which lasted over an hour in some cases. The results for varying dimension reductions and different numbers of weak classifiers of the AdaBoost algorithm are displayed in Table 2. We chose to use Gentle AdaBoost because we observed some numerical issues using Real AdaBoost in preliminary

Table 2: Performance of the Gentle AdaBoost classifier using different degrees of descriptor dimension reduction and varying numbers of weak classifiers for the AdaBoost algorithm. A PCA reduction factor of x indicates that the size of the gist descriptor was reduced to $1/x$ of its original size using principal component analysis.

Parameters	Home 1
PCA reduction factor 2.0, 50 weak classifiers	26.06
PCA reduction factor 4.0, 20 weak classifiers	26.73
PCA reduction factor 4.0, 50 weak classifiers	25.68
PCA reduction factor 8.0, 20 weak classifiers	25.99
PCA reduction factor 8.0, 50 weak classifiers	26.11
PCA reduction factor 8.0, 100 weak classifiers	25.11

Table 3: The performance of the ν-SVM classifier with varying size parameter γ of the radial basis function kernel and different soft margins ν on the gist descriptor.

ν	0.01	0.05	0.1	0.2
$\gamma = 1.0$				
Home 1	33.18	32.76	33.34	n/a
Home 6	33.24	34.40	35.15	37.03
Average	33.21	33.58	34.25	37.03
$\gamma = 2.0$				
Home 1	34.25	34.40	34.34	37.22
Home 6	34.81	36.00	36.92	38.22
Average	34.53	35.20	35.63	37.72

experiments. However, the results indicate that either the AdaBoost algorithm did not have a sufficient number of weak classifiers for a successful classification or is not suited for this descriptor. We did not explore higher numbers of weak classifiers since the relation of accuracy gain to training runtime was bad.

Support Vector Machine For the classification with a Support Vector Machine we decided to use the ν-SVM learning algorithm [66] together with a radial basis function kernel. We varied the kernel size parameter γ and the soft margin parameter ν[2] during our experiments. The results can be seen in Table 3. Although the SVM training cycles lasted pretty long with up to one hour, the obtained results are significantly better than

[2]For a helpful illustration of the ν parameter we refer to the article of Chen *et al.* [8]

those of AdaBoost and the modified KNN classifier. In both subsets we can observe that the optimal parameter setting is $\gamma = 2.0$ and $\nu = 0.2$. With preliminary experiments on these values we found out that the examined parameter ranges are the most promising for the Gist descriptor. The accuracy increases with growing ν which indicates that given the RBF kernel the data can be separated well into more coarse clusters while the soft margin expands. The less cluttered decision boundary shows to generalize better by allowing more samples to move into the soft margin.

The missing value for $gamma = 1.0, \nu = 0.2$ could not be determined because the ν parameter was refused as too high for the provided training data by the algorithm. To avoid this problem in the following experiments we adjusted the ν parameter to the maximal allowed when 0.2 was too high. When applied, the corrected ν ranged always between 0.15 and 0.2.

Learning of a Probability Distribution Finally, we studied the performance of the probabilistic model learning with both approximations of the joint probability distribution $p(l_i|c_1, \ldots, c_N)$ for place categories, the naive Bayes and the first-order dependency approximation. The probability variables $c_1, \ldots, c_N$ with $N = 16$ represent the 16 regions of the regular grid the image is divided into. Their values are cluster indices of clusters obtained from the following processing of the Gist descriptor data. We divide the 1024-dimensional Gist descriptor into 16 region-specific descriptors of length 64. These smaller descriptors are clustered individually for each region into s codewords. Consequently, the joint probability distribution $p(l_i|c_1, \ldots, c_N)$ is computed from the coocurrence of codewords in the 16 image regions. In case of the naive Bayes approximation, this classification method is equivalent to the approach used in [80] for classifying places with the Centrist descriptor.

The results yielded by both approximations in dependence of the number of employed intermediate clusters for each region can be viewed in Table 4. For the naive Bayes approach we demonstrate that a neither too few nor too many clusters can obtain good results since few clusters cannot represent the diversity of the descriptors while too many cluster rather describe details without any generalization. Consequently, the best performance is obtained

Table 4: The performance of the approach in which each place is modelled as the joint probability of coocurring cluster-codewords from the 16 image regions. The probability function was approximated using the naive Bayes assumption as well as the first-order dependency tree. The number of clusters for the 64-dimensional Gist descriptors was varied during this experiment.

Number of Clusters	31	46	91	196	496	991
Naive Bayes						
Home 1	30.63	33.81	33.99	34.90	33.55	32.72
Home 6	36.91	34.53	38.67	35.06	37.18	35.36
Average	33.77	34.17	36.33	34.98	35.36	34.04
First-Order Dependency Tree						
Home 1	29.02	29.60	27.32	30.94	n/a	n/a
Home 6	29.50	30.31	30.78	29.15	n/a	n/a
Average	29.26	29.96	29.05	30.05	n/a	n/a

from intermediate numbers of clusters like 91. Surprisingly, the first-order dependency approximation does not exceed the accuracy of the naive Bayes approximation but performs worse. As we can suppose that there are relations between grid cells imposed by larger objects the only reason for this low performance might be a shortage on training data. The more complex first-order dependency framework has to learn N^2 probabilities per class when there are N different clusters whereas the naive Bayes approach only needs to learn N probabilities per class. Under this consideration we must realize that learning the first-order dependencies from around 8000 images already draws the use of 196 clusters infeasible. On the other hand, using too few clusters results in a poor representation of the variability in the data although the dependencies might be learned better.

Discussion The preceding analysis revealed that the classification results differ by over 10% in dependence of the applied machine learning technique which justifies this in-depth analysis.

Knowing the good classification approaches, we additionally examined further modifications to the Gist descriptor in several smaller experiments. First, we removed the descriptor data of the color channel to show how it contributes to the result. For this experiment we

Table 5: Overview over the two best classifiers on the Gist descriptor: The SVM with $\gamma = 2.0$ and $\nu = 0.2$ as well as the naive Bayes approximation for the joint probability distribution with $K = 91$ clusters. The numbers represent average accuracies over all five room categories obtained from a cross-validation leaving out the respective home subset at each time.

Gist	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
naive Bayes	33.99	35.34	35.65	34.71	38.95	38.67	36.22
SVM	37.22	43.75	50.03	47.79	36.53	38.22	42.26

tested on home 6 using the naive Bayes classifier with $N = 91$ clusters. The average accuracy for this setting was 35.41% which is over 3% lower than the performance of 38.67% of the original Gist descriptor. However, although the additional color information improves the classification result, the improvement is not tremendous compared to the doubling of the descriptor length.

In a second experiment, we checked the performance of the Gist descriptor if we take the means directly from the images of the Gaussian pyramid and the color images of the R, G, B, Y pyramid instead of the intensity and color feature maps. Again, we tested with home 6 and the naive Bayes classifier with $N = 91$ clusters. The average result of 30.96% proves that this approach is definitely worse than the original Gist descriptor and shows that low-level preprocessing like center-surround operations improves the classification performance.

In conclusion, the strongest results were obtained with the Support Vector Machine using the parameters $\gamma = 2.0$ and $\nu = 0.2$ and with the naive Bayes approach using $N = 91$ intermediate cluster codewords. We tested the performance of both classification methods on the remaining subsets of the Home database with the already described leave-out-one cross-validation. The results are displayed in Table 5. As explained before, the SVM sometimes could not handle the ν parameter of 0.2. In those cases it was lowered to the highest possible value which was always above 0.15. The reported accuracies are averaged over all five room categories for each home by taking the average of the single accuracies. This measure also reflects whether there are bad categorization results when the dataset contains room classes with only a few samples in contrast to just dividing the number of

Table 6: Detailed classification accuracies for the best classifier (SVM, $\gamma = 2.0$, $\nu = 0.2$) used in conjunction with the Gist descriptor.

Gist	Bedroom	Bathroom	Kitchen	Living Room	Dining Room	Average
Home 1	67.30	58.90	11.20	46.30	2.30	37.22
Home 2	49.90	48.60	56.70	40.60	23.30	43.75
Home 3	85.10	90.80	26.70	6.20	41.40	50.03
Home 4	46.70	60.60	69.10	44.30	18.30	47.79
Home 5	62.60	78.40	24.10	17.50	0.00	36.53
Home 6	60.50	38.80	72.30	14.90	4.40	38.22
Average	62.02	62.68	43.35	28.30	14.95	42.26

correctly classified images by the total number of images used for testing.

We can see that the performance gain of using SVM compared to using the naive Bayes approximation is bigger than initially expected from the former experiments. A detailed distribution of the accuracies over the individual place categories and test subsets for the SVM classifier is shown in Table 6. In comparison with the performance of the Centrist descriptor as reported by Wu [80] on this dataset and as confirmed in our own experiments (see section 4.3) the overall accuracy is almost identical with 42.26% for Gist and 41.87% for Centrist. However, the distribution of the predictive power differs between both descriptors for the place categories: Gist provides a ca. 14% higher accuracy for bedrooms while there is a small performance drop for the other room classes which is largest for dining rooms with around 5%.

The performance gain with applied delayed HMM smoothing (see section 3.4) is almost 2% smaller than for Centrist (46.78% [80]) as we can see in Table 7. However, in contrast to the observation reported in [80] we encounter a general improvement for all classes by up to 5%, even for the weak categories.

Consequently, Gist appears to be an almost equally good descriptor for the place categorization task as Centrist in the way it was presented in [80]. In the next section, we shortly examine the classification results for the Centrist descriptor when the both most successful classifiers found for the Gist classifier are applied.

Table 7: Detailed classification accuracies when delayed HMM smoothing is applied on the results of the best classifier (SVM, $\gamma = 2.0$, $\nu = 0.2$) used in conjunction with the Gist descriptor.

Gist	Bedroom	Bathroom	Kitchen	Living Room	Dining Room	Average
Home 1	72.90	62.80	8.70	45.10	1.60	38.24
Home 2	50.40	47.80	72.10	42.00	31.40	48.74
Home 3	89.90	94.90	27.50	6.40	56.40	55.04
Home 4	47.50	65.10	72.60	51.80	21.60	51.75
Home 5	62.40	78.10	24.80	13.00	0.00	35.67
Home 6	62.10	36.50	76.60	15.90	8.90	39.99
Average	64.20	64.20	47.05	29.03	19.98	44.90

4.3 Experiments on the Centrist Feature

In this section we analyze our Centrist implementation with the original probabilistic naive Bayes classification framework as presented in [80] as well as with the SVM method which was already most successful on the Gist descriptor.

The Centrist descriptor is computed after the original image is downsized to a width of 320 pixels[3] and convolved with a Laplacian filter[4]. The obtained edge image is transformed with the census transform. The Centrist descriptor contains a 256 bin histogram on the values of the census-transformed image for each of the 16 image cells from the 4x4 grid. Its dimensionality is consequently $16 \cdot 256 = 4096$.

Original Formulation The original classifier for the Centrist descriptor is the joint probability distribution $p(l_i|c_1, \ldots, c_N)$ over the codewords found in the 16 image cells. As already described in section 4.2 we generate 16 codebooks of the 256-dimensional descriptors found in each cell by using a hierarchical clustering mechanism. The naive Bayes assumption is applied as well, resulting in the multiplication of the cell-specific probabilities of the found 16 clusters as indicated in equations (10) and (12).

[3]This implementation detail is not mentioned in [80] but can be found in the accompanying *libhik* library. The downscaling is absolutely necessary to reproduce the reported results since the computation on the originally sized image yields results around 5% worse.

[4]We found in preliminary experiments that the performance of Centrist is lower if it is computed on the original image

Table 8: Overview over the classification results on the Centrist descriptor obtained with the SVM with $\gamma = 2.0$ and $\nu = 0.2$ as well as with the naive Bayes approximation for the joint probability distribution with $K = 91$ clusters. The numbers represent average accuracies over all five room categories obtained from a cross-validation leaving out the respective home subset at each time.

Centrist	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
original	45.53	34.51	42.98	46.03	40.27	48.61	42.99
spatial PACT	43.07	40.05	48.79	51.24	38.60	42.21	43.99

The results of this approach were tested on the whole database with the formerly described leave-out-one subset cross-validation and are displayed in Table 8. It shows that the obtained results correspond very well with the reported results in [80] even on the subset level.

Support Vector Machine For the use with the support vector machine we consider the 4096-dimensional Centrist descriptor which consists of all 16 cell descriptors. Because of its exceptional length we shorten the descriptor to 1024 dimensions, which is the same length as the Gist descriptor. The dimension reduction with PCA is performed on the short descriptor data of each region individually to avoid that whole regions might get filtered out if only one PCA is applied to the whole descriptor. This method is also referred to as spatial PACT in [80]. The obtained descriptor is input into a SVM classifier with the parameters $\gamma = 2.0$ and $\nu = 0.2$, which proved to provide the best results for the Gist descriptor. In those cases when $\nu = 0.2$ was too big for the provided data we decreased it to the largest allowed value.

The classification accuracy for this setup can be found in Table 8. We observe that the SVM can improve the originally reported performance of 41.87% by more than 2%.

Discussion These two evaluations showed that the performance of the original Centrist-based place categorization system can be improved slightly by using a Support Vector Machine on the spatial PACT descriptor. This confirms the trend which could already be observed for the Gist descriptor. To allow a detailed comparison the distribution of

Table 9: Detailed classification accuracies for the SVM classifier with $\gamma = 2.0$ and $\nu = 0.2$ used in conjunction with the spatial PACT Centrist descriptor.

Centrist	Bedroom	Bathroom	Kitchen	Living Room	Dining Room	Average
Home 1	66.50	82.20	11.80	41.50	13.40	43.08
Home 2	55.40	39.50	50.40	23.10	31.80	40.04
Home 3	67.20	94.10	37.00	12.10	33.50	48.78
Home 4	57.90	63.60	62.80	55.50	16.50	51.26
Home 5	90.30	56.10	23.40	19.80	3.40	38.60
Home 6	62.00	50.70	56.80	22.90	18.70	42.22
Average	66.55	64.37	40.37	29.15	19.55	43.99

accuracies for the SVM-based classification on the Centrist descriptor is displayed in Table 9 in dependence of room class and test subset. Compared to the distribution for Centrist using the original system [80] we can see that the accuracy for bedroom increased by over 18% while only kitchen had a decrease of ca. 6%. All in all, there are less very low numbers of accuracies when the SVM is used. Compared with the distribution of accuracies for the Gist descriptor (see Table 6) it shows that the general distribution among the room categories follows a similar pattern: bedroom, bathroom and kitchen reach significantly better results than living room and dining room. Even the percentages are quite identical with the exception of the dining room which is detected 5% less by the Gist descriptor. Despite the similar distributions among the places we can still find substantial differences between Gist and Centrist in the individual room accuracies for certain test subsets, especially among the stronger classes. This finding justifies the attempt to fuse the information of the classifier responses of both descriptors (see section 4.5) to obtain a stronger classifier.

For the sake of completeness we also computed the impact of applying the delayed HMM smoothing operation explained in section 3.4. The detailed overview in Table 10 shows that a 4% improvement on the former results is possible when smoothing is used. This performance is slightly better by 1.26% than the best result (46.78%) obtained with the place categorization system of Wu [80]. In contrast to the effect observed in [80] here the stronger classes remain at a constant level while the accuracies of the weaker classes kitchen, living room and dining room improve visibly by 5%, 18% and 7%, respectively.

Table 10: Detailed classification accuracies when delayed HMM smoothing is applied for the SVM classification with $\gamma = 2.0$ and $\nu = 0.2$ used in conjunction with the spatial PACT Centrist descriptor.

Centrist	Bedroom	Bathroom	Kitchen	Living Room	Dining Room	Average
Home 1	66.40	85.60	8.70	86.60	23.30	54.11
Home 2	52.60	41.90	53.00	24.00	43.80	43.07
Home 3	58.70	95.20	32.60	19.30	42.30	49.63
Home 4	57.50	63.10	83.90	54.70	26.10	57.05
Home 5	91.80	57.40	33.80	8.50	8.80	40.07
Home 6	65.50	50.90	59.80	31.50	13.80	44.31
Average	65.42	65.68	45.30	37.43	26.35	48.04

However, the general trend that living and dining room receive less accurate results still remains.

In conclusion, we decided to use the SVM-based classification for the future experiments because of its higher accuracy and because the trained SVM does not fluctuate in its performance between different training cycles. This instead is the case for the naive Bayes approximation of the joint probability since its training requires to run k-means++ which is a randomized algorithm. These advantages also compensate for the longer classifier training time of close to one hour.

4.4 Experiments on the Salient Region Descriptors

In this section we examine how well the approach works which characterizes a place by the objects or object parts found within salient regions of the scene. The experiments include the assessment of the different descriptors described in section 3.1.2 and combinations of them as well as a study of the best suited classifier with its respective parameters. The first experiments employ the multi-class classifiers introduced in section 3.3.1. Afterwards, the categorization based on the joint probability distribution of objects present in the image is evaluated (see section 3.3.2). Then we test the performance of the SVM-DAS feature integration method [58] and finally, we study the effect of the smoothing filter and several further modifications like an information filter and tracking of the salient regions.

In general, we employ the algorithm sketched in the right track of Figure 1. This

means that we first determine several salient regions of the image by the visual attention mechanism explained in section 3.1.1 which are then described by the local descriptors presented in section 3.1.2. The obtained descriptors are finally input into a classifier which decides for a place category.

Some of the basic parameters for this algorithm were set from the results of preliminary experiments: The number of scales used for the attention computation and the number of salient regions the algorithm is allowed to find. We found that starting the processing of the feature maps with scale 2 as proposed by Frintrop [21] and using the maps up to scale 5, which is one more than proposed in Frintrop's work, yields good results when home 5 of the home database is used for testing. Manual inspection and the improved accuracies suggested using scale 5 in addition. The maximum allowed number of salient regions per image is limited to 25 since initial tests showed that the results are better than if we would only use 12 regions and that the results do not improve if 50 regions are used. After finding the optimal classifier parameters in the following sections, we show again that 25 salient regions is a reasonable number.

4.4.1 Multi-Classifier

The multi-class classifier approach takes the descriptors of the salient regions one at a time and classifies each region individually. The final decision based on all classified regions is obtained from the multiplication of the single regions' probabilities. Although the software framework was initially setup to be used with the modified KNN, AdaBoost, SVM and RVM classifiers we could only examine the performance of KNN and AdaBoost in this section due to the extremely long training times of the SVM and the RVM on the vast amount of data. Their training runtimes are supposed to range in the order of days since we aborted those experiments after one day. Followingly, we discuss the results yielded by the KNN and the AdaBoost classifier.

K-Nearest Neighbors For the first experiment with the modified KNN classifier we have a look at the performance of the single-cue descriptors Orientation Mean and Variance (Ori M/V), Histogram of Oriented Gradients (HOG), SIFT, Centrist and Color Mean

Table 11: Performance evaluation of the KNN classifier with $N = 1$ nearest neighbor on the single-cue features found in salient regions of the image. The influence of varying cluster numbers is examined.

Number of Clusters	91	196	496	991
Ori M/V				
Home 1	29.85	28.27	30.13	28.65
Home 6	33.78	34.03	36.28	31.87
Average	31.81	31.15	33.20	30.26
HOG				
Home 1	36.22	36.07	33.15	30.66
Home 6	38.21	37.92	36.70	28.57
Average	37.22	37.00	34.92	29.62
SIFT				
Home 1	32.34	32.22	31.17	32.51
Home 6	33.50	33.52	32.80	29.34
Average	32.92	32.87	31.99	30.92
Centrist				
Home 1	33.40	30.60	32.04	32.98
Home 6	30.05	35.80	37.93	29.24
Average	31.73	33.20	34.99	31.11
Color M/V				
Home 1	15.36	16.40	16.09	18.01
Home 6	20.54	21.72	22.71	18.81
Average	17.96	19.06	19.40	18.41

and Variance (Color M/V). The results are shown in Table 11. Because of the randomized cluster initialization for the KNN classifier, most of the reported numbers in this and the following tables are averages from two runs on two different computers with different operating systems. As explained before, the cluster numbers are constrained by the branching factor $b = 16$ of the hierarchical clustering of the FLANN library [48] to numbers satisfying the equation $(b - 1) \cdot k + 1$.

Among these results we can see that HOG is the strongest single-cue classifier using the KNN classifier, followed by Centrist. The simple Orientation Mean and Variance descriptor generates a performance that can compete with SIFT. The color descriptor instead shows a significant performance drop compared to the other descriptors. We must conclude from this result that shape information, which is encoded in the other descriptors, is more descriptive than color information. This conclusion makes sense in that way that most objects of a

class rather have a similar shape while the color may vary strongly. The opposite that objects with the same color but different shapes belong to the same class is encountered much more rarely. Furthermore, for all descriptors the classification performance drops if too many clusters are used. We assume that this is due to a too detailed description of the objects which also distinguishes between similar objects instead of putting them into the same cluster.

For the next experiment, we add some spatial information to the salient region descriptors since there was no such information at all in the former experiment. We add one number to each descriptor which represents the *vertical position* (y-Pos) of the salient region within the image. This idea is motivated by the fact that certain objects tend to appear at approximately the same height in human environments like microwaves which are normally placed on a working surface in a kitchen. As long as the camera of the robot does not tilt, we can find those objects at similar y coordinates in the images. We explicitly do not add any horizontal position information since a rotation of the robot is common and allows every object to appear at any x coordinate inside the image. This is a difference to the typical scene recognition problem in computer vision where the conjecture is that objects appear at similar positions in the image because the view at the scene is always a similar one. The results of the localized descriptors are visible in Table 12 where we can observe that the additional information about the y-position of a salient region provides a significant improvement to using the unlocalized descriptors only. This time the Centrist descriptor performs slightly better than HOG and SIFT. The Orientation Mean and Variance descriptor is 2.4% worse than Centrist. Again, the color descriptor is significantly worse than the shape descriptors.

Finally, we evaluate a descriptor which does low-level feature integration by concatenating the localized shape descriptors and the color descriptor. As Table 13 shows the additional color information decreases the performance very much by up to 14% in comparison to the results when using only the localized shape information (see Table 12). We suspect that the additional variety introduced by different colors for similarly shaped objects renders the object categorization problem even harder since less instances for similar

Table 12: Performance evaluation of the KNN classifier with $N = 1$ nearest neighbor on the localized (y-Pos) single-cue features found in salient regions of the image. The influence of varying cluster numbers is examined.

Number of Clusters	46	196	496
Ori M/V + y-Pos			
Home 1	29.23	32.48	29.96
Home 6	44.27	41.14	38.26
Average	36.75	36.81	34.11
HOG + y-Pos			
Home 1	29.94	32.30	29.59
Home 6	47.44	45.20	41.33
Average	38.69	38.75	35.46
SIFT + y-Pos			
Home 1	32.15	37.34	37.61
Home 6	34.61	37.74	39.29
Average	33.38	37.54	38.45
Centrist + y-Pos			
Home 1	29.25	40.01	43.56
Home 6	33.41	38.41	34.09
Average	31.33	39.21	38.82
Color M/V + y-Pos			
Home 1	24.28	17.77	18.97
Home 6	22.44	25.51	20.89
Average	23.36	21.64	19.93

objects are available in the training set which complicates a successful generalization even more. Especially because the first step is an unsupervised clustering we cannot necessarily expect that this procedure can put features with similar shape but strongly differing colors into one cluster. This consideration yields the conclusion that descriptors from different cues should be integrated at a higher level, for example after an initial classification step.

AdaBoost We also examined the performance of AdaBoost on the different sets of descriptors analyzed in the previous section. However, because of the very long training times of the classifier which ranged within several hours and because of the relatively poor results we could only check for some few parameter settings.

The outcome of the first experiment on the single-cue descriptors is displayed in Table 14. We tested once without shortening the descriptors and twice with descriptors of one

Table 13: Performance evaluation of the KNN classifier with $N = 1$ nearest neighbor on the localized (y-Pos) concatenated shape and color features found in salient regions of the image. The influence of varying cluster numbers is examined.

Number of Clusters	46	196	496
Ori M/V + y-Pos + Color M/V			
Home 1	16.13	17.22	22.31
Home 6	25.27	33.41	23.42
Average	20.70	25.31	22.86
HOG + y-Pos + Color M/V			
Home 1	33.22	23.14	30.22
Home 6	38.76	37.68	36.41
Average	35.99	30.41	33.32
SIFT + y-Pos + Color M/V			
Home 1	37.06	33.02	33.62
Home 6	29.20	28.45	29.92
Average	33.13	30.73	31.77
Centrist + y-Pos + Color M/V			
Home 1	24.87	20.91	23.18
Home 6	23.76	25.94	26.19
Average	24.32	23.43	24.69

fourth of the original length. For the reduction we used Principal Component Analysis. The long training times forced us to employ only very few weak classifiers for the AdaBoost algorithm which results in poor categorization performances which are constantly much worse than those from the KNN classifier with the exception of the color mean and variance feature which is slightly better. However, the distribution of the accuracy between the room classes is bad since there are normally only two classes with detection rates greater than zero.

Table 14: Classification accuracy when AdaBoost is applied to the single-cue descriptors obtained from salient regions. Here different numbers of weak classifiers for AdaBoost and different reductions of the data via PCA are examined while the test set is Home 1.

Parameters	No PCA, 20 weak	PCA 4, 20 weak	PCA 4, 10 weak
Ori M/V	24.83	21.33	20.46
HOG	24.48	21.38	21.08
SIFT	26.31	23.06	21.76
Centrist	30.08	26.77	26.64
Color M/V	24.08	23.17	23.22

Table 15: Classification performance of AdaBoost on the composed descriptors obtained from salient regions. The classfier setting is to use 20 weak classifiers after the descriptor data is shortened via PCA by a factor of 4.

Ori M/V + y-Pos	21.84
HOG + y-Pos	22.02
SIFT + y-Pos	21.07
Centrist + y-Pos	23.27
Color M/V + y-Pos	22.97
Ori M/V + y-Pos + Color M/V	22.86
HOG + y-Pos + Color M/V	21.57
SIFT + y-Pos + Color M/V	20.70
Centrist + y-Pos + Color M/V	22.44

Similar observations can be made when AdaBoost is applied to the localized and the composed shape and color descriptors as we can see in Table 15. For reasons of computation time we only checked the four times reduced descriptor with AdaBoost using 20 weak classifiers. The results are comparably poor as in the experiment for the single-cue descriptors. We assume that a lot more weak classifiers would be needed for better results, however, there is no justification for the much longer training times if the same or better results can be obtained with the modified KNN classifier in less time.

As explained before, we cannot provide an analysis of the performance obtained with SVM or RVM classifiers here because of their extremely long training times. Therefore, we proceed directly with the second classification paradigm, the modelling of a joint probability distribution.

4.4.2 Place Modelling with a Probability Distribution

Instead of classifying every salient region of the image by itself and independent of the other regions, this approach clusters the different occuring region descriptors and models place categories as the joint probability of the activation or deactivation of all those clusters. This means, the descriptors from all found salient regions are assigned to clusters q_k and the information which clusters are activated is used to compute the room category probability $p(l_i|c_1 = q_1, c_2 = q_2, \ldots, c_N)$ as explained in section 3.3.2. We consider two approximations of the joint probability $p(c_1 = q_1, c_2 = q_2, \ldots, c_N|l_i)$ which has to be computed within this

Table 16: Performance of the classification approach using a joint probability which is approximated by the naive Bayes assumption. The effect of varying numbers of intermediate clusters is studied when the single-cue descriptors are employed.

Number of Clusters	46	196	496
Ori M/V			
Home 1	29.07	31.08	31.57
Home 6	35.76	34.36	37.20
Average	32.42	32.72	34.39
HOG			
Home 1	31.35	33.72	30.65
Home 6	36.57	38.17	38.93
Average	33.96	35.95	34.79
SIFT			
Home 1	25.74	32.78	30.13
Home 6	37.56	34.76	34.54
Average	31.65	33.77	32.34
Centrist			
Home 1	37.43	33.71	36.25
Home 6	31.05	35.81	36.62
Average	34.24	34.76	36.44
Color M/V			
Home 1	24.49	20.11	20.49
Home 6	20.95	23.86	23.24
Average	22.72	21.99	21.87

approach. The performance evaluation starts with the naive Bayes approximation followed by the optimal first-order dependency approximation.

Naive Bayes Approximation The naive Bayes assumption is the simplest approximation of the joint probability which asserts independence between the activation status of the different clusters. We conducted the same experiments as for the KNN classifier in the preceding section. The first experiment evaluates the performance of the naive Bayes approach when only the single-cue descriptors are used. The results in Table 16 are better than those for the modified KNN classifier in the case of the Centrist descriptor as well as the orientation and the color mean and variance descriptor, almost the same for the SIFT descriptor and worse for the Histogram of Oriented Gradients descriptor. Nevertheless, the general relationships between the accuracies of different descriptors remain similar as

Table 17: Accuracies for the naive Bayes approximation of the joint probability model for room class prediction when the single-cue descriptors are localized by the y-Position of their original salient region. The effect of different cluster numbers of the classifier is studied.

Number of Clusters	46	196	496
Ori M/V + y-Pos			
Home 1	30.05	33.01	33.21
Home 6	40.16	38.91	40.47
Average	35.11	35.96	36.84
HOG + y-Pos			
Home 1	30.51	35.36	30.96
Home 6	40.80	45.51	44.62
Average	35.66	40.44	37.79
SIFT + y-Pos			
Home 1	30.78	36.74	35.34
Home 6	35.14	37.69	39.75
Average	32.96	37.22	37.55
Centrist + y-Pos			
Home 1	29.78	38.07	38.30
Home 6	38.00	38.93	36.43
Average	33.89	38.50	37.37
Color M/V + y-Pos			
Home 1	22.01	21.52	19.20
Home 6	24.97	23.48	24.61
Average	23.49	22.50	21.91

Centrist and HOG yield still the best results closely followed by SIFT. The orientation descriptor is 2% off of the best and the color feature is again very weak.

The evaluation of the naive Bayes approximation of the joint probability distribution for the localized single-cue descriptors can be found in Table 17. We find that the individual results are better for each descriptor compared to the single-cue descriptors without localization information. The comparison to the KNN classifier shows that the HOG and the color descriptor could obtain a better result in this framework while the orientation descriptor remained equally good. SIFT and Centrist yield slightly worse results with this classification approach. Again, the usual ranking of the descriptors remains with HOG and Centrist showing the best results, then SIFT, then orientation mean and variance and finally color mean and variance with a very bad result.

Table 18: Results for the naive Bayes approach on the combined color and shape descriptors for varying numbers of intermediate clusters for the classifier.

Number of Clusters	46	196	496
Ori M/V + y-Pos + Color M/V			
Home 1	17.59	29.32	23.31
Home 6	27.03	28.64	30.11
Average	22.31	28.98	26.71
HOG + y-Pos + Color M/V			
Home 1	25.44	31.85	30.99
Home 6	36.24	36.15	32.98
Average	30.84	34.00	31.99
SIFT + y-Pos + Color M/V			
Home 1	34.44	34.82	29.87
Home 6	33.59	38.39	34.58
Average	34.02	36.61	32.23
Centrist + y-Pos + Color M/V			
Home 1	31.87	25.87	25.36
Home 6	27.17	30.34	32.01
Average	29.52	28.11	28.69

The last experiment on the naive Bayes approach considers the low-level feature integration of shape and color information. The results obtained from the addition of the color descriptor to the localized shape descriptors are displayed in Table 18. It shows again that the addition of color information is very harmful to the categorization accuracy with a drop in performance between 1% for SIFT and 9% for Centrist. However, for this method the performance is still higher than for the KNN classifier for all tested descriptors.

In all three experiments we can observe for almost every descriptor that the accuracy decreases if the number of clusters becomes too big. The same result turned out for the modified KNN classifier and the same reasoning applies here: Too many clusters force to learn too many details of the environment instead of generalizing the visual information.

Having investigated the performance of the naive Bayes approximation we study the more sophisticated optimal first-order dependency approximation in the next section.

Table 19: Performance evaluation of the joint probability distribution modelling with the optimal first-order dependency approximation on the single-cue features found in salient regions of the image. The influence of varying cluster numbers is examined.

Number of Clusters	46	196	496	991
Ori M/V				
Home 1	29.65	30.58	29.38	30.55
Home 6	34.66	34.14	34.96	35.78
Average	32.15	32.36	32.17	33.16
HOG				
Home 1	32.11	33.00	28.39	29.41
Home 6	35.64	41.64	39.25	35.45
Average	33.87	37.32	33.82	32.43
SIFT				
Home 1	29.25	29.63	27.42	27.58
Home 6	32.22	32.32	31.78	29.97
Average	30.73	30.98	29.60	28.78
Centrist				
Home 1	34.73	32.86	32.82	32.44
Home 6	29.80	36.08	35.83	29.45
Average	32.26	34.47	34.33	30.95
Color M/V				
Home 1	20.89	22.36	21.23	22.10
Home 6	29.42	28.39	26.31	31.45
Average	25.15	25.38	23.77	26.77

Optimal First-Order Dependency Approximation The optimal first-order dependency approximation of the joint probability of activated and deactivated clusters can incorporate one dependency for each cluster's probability of activation $p(c_k = q_k | c_{\Pi_k} = q_{\Pi_k}, l_i)$. As explained in section 3.3.2 the probability of the activation status q_k of cluster c_k is not independent of all other clusters as in the naive Bayes case but dependent on the activation of exactly one other cluster c_{Π_k}. Consequently, this method does not only represent the knowledge about which objects are found in the image and which are not but also the knowledge about the coocurrence or the mutual exclusion of objects. The following experiment evaluates whether this more powerful representation can improve the preceding results.

As before, we first examine the classification accuracy for the single-cue descriptors. The outcomes are shown in Table 19. We remark that the first-order dependency approximation only improves the HOG result slightly and the color descriptor accuracy substantially. For

orientation mean and variance, SIFT and Centrist it even yields worse results. We made this surprising observation already for the Gist descriptor for which the provided data was probably not enough to learn the more complex model. For the salient regions there are around 150,000 samples in the training set so that apparently there should be enough data to learn the distribution. However, this reasoning is wrong since there are only video sequences from six homes in the video independent of the actual number of images and objects found within them. Most of the captured regions repeat for many successive frames so that we finally end up with the calculation that we only have a part of the original 8000 training images as really different training images for the salient regions. Consequently, if the training data is not diverse enough then its pure and repeted mass does not help. In fact, much more data is necessary to capture the possible variability correctly which can be represented by the first-order dependency approximation. In general, the better the joint probability is approximated, the more data is needed for learning.

Especially a more sophisticated approximation is able to distinguish more cases which might yield the same outcome with a less sophisticated approximation. We therefore conjecture that under the presentation of so few training data the first-order dependency approximation might learn certain relationships which already encode too much detail information hindering the generalization process. This means, for example, that if there is a microwave and a plate, both would vote individually for kitchen in a naive Bayes framework. However, if this pair is only seen together, a detected microwave without a plate would never vote for kitchen since it was never seen without a plate. The probability would just be 0.5, that is uninformed. We assume that there are a lot of such cases where not all different probabilities could be trained due to the lack of really diverse training data which we would obtain from a dataset with 50 or more houses.

We proceed with the analysis of the localized descriptors whose results are shown in Table 20. As usual, the additional positional information improves the classification rates between 2% and 5%. The comparison to the naive Bayes approximation reveals that the first-order dependency approximation only yields better results for the orientation and color mean and variance descriptors while the performance for HOG is almost identical and for

Table 20: Performance of the joint probability model approximated with the first-order dependencies using the localized descriptors. The analysis indicates the effect of varying numbers of clusters.

Number of Clusters	46	196	496
Ori M/V + y-Pos			
Home 1	32.31	34.78	32.05
Home 6	37.39	40.37	36.68
Average	34.85	37.57	34.37
HOG + y-Pos			
Home 1	35.11	33.65	33.27
Home 6	43.12	46.91	43.04
Average	39.11	40.28	38.15
SIFT + y-Pos			
Home 1	32.09	32.69	34.82
Home 6	33.55	38.55	35.79
Average	32.82	35.62	35.30
Centrist + y-Pos			
Home 1	32.74	35.42	33.90
Home 6	38.51	37.35	37.27
Average	35.62	36.39	35.58
Color M/V + y-Pos			
Home 1	26.37	21.37	22.48
Home 6	31.54	30.34	27.96
Average	28.95	25.85	25.22

SIFT and Centrist worse by ca. 2%.

Finally, we evaluate the categorization accuracies when the concatenated shape and color descriptors are used. The results are displayed in Table 21. The accuracies drop by 2% to 8% for all descriptors, as we have seen with other classifiers before. To verify the general observation that the concatenation of shape and color descriptors provides worse results than using localized shape descriptors only, we repeated the last experiment with a ten times shorter color descriptor which only contained the mean and variance of the whole salient region. However, the results were almost the same if not worse for certain descriptors. We can therefore conclude that the addition of color information to a shape descriptor yields worse results over employing the shape descriptor only. The other way around it showed that an additional shape descriptor can always improve the accuracies obtained from a color descriptor alone.

Table 21: Performance check of the joint probability model with first-order dependency approximation on the composed shape and color features under varying numbers of clusters.

Number of Clusters	46	196	496
Orientation M/V + y-Pos + Color M/V			
Home 1	21.94	26.91	25.53
Home 6	33.22	32.13	30.17
Average	27.58	29.52	27.85
HOG + y-Pos + Color M/V			
Home 1	31.68	29.69	32.03
Home 6	39.15	39.55	37.63
Average	35.41	34.62	34.83
SIFT + y-Pos + Color M/V			
Home 1	32.65	30.87	30.45
Home 6	35.26	38.24	36.26
Average	33.96	34.56	33.36
Centrist + y-Pos + Color M/V			
Home 1	32.08	28.34	24.52
Home 6	32.08	31.42	30.15
Average	32.08	29.88	27.34

In the next section we summarize the findings of the extensive experiments about the descriptor, classifier and parameter optimization performed in this and the preceding section.

4.4.3 Preliminary Summary

The analysis of the preceding experiments provides the following results which represent conclusions drawn on the basis of experiments using home 1 and home 6 as test sets.

- The best descriptor for the salient regions is the combination of a shape descriptor paired with localization information in the y-direction.
- The addition of color information to a shape descriptor regularly decreases the accuracies.
- Speficically, the ranking of the descriptors is the following (with accuracy, classifier and classifier setting mentioned in brackets):
    1. Histogram of Oriented Gradients + y Position Information (40.44% - naive Bayes,

196 clusters)

2. Centrist + y Position Information (39.20% - KNN, 196 clusters)
3. SIFT + y Position Information (38.45% - KNN, 496 clusters)
4. Orientation Mean and Variance + y Position Information (37.57% - first-order dependency approximation, 196 clusters)
5. Color Mean and Variance + y Position Information (28.95% - first-order dependency approximation, 46 clusters)

- There is no best classifier so far as we can see from the ranking above.
- For the SIFT and Centrist, which worked best with the KNN classifier, the best probability model uses a naive Bayes approximation with exactly the same numbers of clusters. For the other three descriptors, the best setting for a KNN classifier are again the same numbers of clusters as they used in their respective probability model. This is an interesting finding which indicates that each descriptor has a different need for the size of the intermediate representation independent of the actual classifier. Regarding HOG and Centrist as well as HOG and SIFT we also observe that this relationship is not connected to the descriptor length.
- The verification for using 25 salient regions as good setting is shown in Table 22. All descriptors are employed with their optimal settings for the classifier based on a probability distribution model. We can observe that 25 regions is the best choice with respect to classification performance except for one exception with Centrist.

4.4.4 Experiments on the Whole Dataset

In this section we verify the performance of the two best classifier setups from the preceding experiments on the whole home dataset since we could not determine one clear winner.

For the first test we use the probability distribution modelling approach with the following settings for the respective descriptors:

- Orientation Mean and Variance: First-order dependency approximation, 196 clusters

Table 22: Accuracies obtained with the best settings for the number of intermediate clusters and the probability distribution model (first-order dependency for color and orientation, naive Bayes for HOG, SIFT, Centrist) using different numbers of salient regions. The test set is home 6.

Number of Salient Regions	12	25	38
Orientation M/V + y-Pos	38.14	40.37	40.05
HOG + y-Pos	39.71	45.51	42.99
SIFT + y-Pos	38.68	39.75	37.53
Centrist + y-Pos	36.23	37.35	40.04
Color M/V + y-Pos	27.30	31.54	29.16

Table 23: Performance analysis of the probability distribution modelling approach with best settings on the whole home dataset. The mentioned homes in the table are the respective test sets.

y-Pos. +	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
Ori M/V	36.55	29.36	34.32	29.77	33.61	40.37	34.00
HOG	35.36	31.76	44.11	39.47	29.25	45.51	37.58
SIFT	35.34	32.58	43.60	33.70	33.12	39.75	36.35
Centrist	38.07	26.45	37.39	30.86	27.66	38.93	33.23
Color M/V	26.46	26.23	39.35	29.65	29.20	31.54	30.41

- Histogram of Oriented Gradients: Naive Bayes approximation, 196 clusters
- SIFT: Naive Bayes approximation, 496 clusters
- Centrist + y-position in image: Naive Bayes approximation, 196 clusters
- ColorMeanVar: First-order dependency approximation, 46 clusters

The classification accuracies are displayed in Table 23. The results reveal that the strongest descriptor is HOG, followed by SIFT, Orientation Mean and Variance and Centrist. Color Mean and Variance has the worst performance as observed before.

For the second test, we use the modified KNN classifier with $N = 1$ nearest neighbor and the following settings for the respective descriptors:

- Orientation Mean and Variance: KNN, 196 clusters
- Histogram of Oriented Gradients: KNN, 196 clusters

Table 24: Classification accuracies obtained with the modified KNN classifier with $K = 1$ nearest neighbor and the best settings for each individual descriptor. The test is done on the whole home database.

y-Pos. +	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
Ori M/V	32.48	29.65	35.95	29.91	27.14	41.14	32.71
HOG	32.30	31.15	43.72	37.56	26.75	45.20	36.11
SIFT	37.34	31.38	39.84	33.61	19.36	37.74	33.21
Centrist	40.01	25.93	36.30	32.56	23.86	38.41	32.85
Color M/V	24.28	34.75	37.15	28.10	35.03	22.44	30.29

Table 25: Classification accuracies obtained with the modified KNN classifier with $K = 3$ nearest neighbors and the best settings for each individual descriptor. The test is done on the whole home database.

y-Pos. +	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
Ori M/V	34.30	28.08	44.53	25.51	27.14	32.26	31.97
HOG	36.13	27.44	36.70	29.73	26.75	36.10	32.14
SIFT	36.80	22.54	35.52	26.72	19.36	27.08	28.00
Centrist	39.31	20.96	27.96	26.65	23.86	22.70	26.91
Color M/V	19.83	35.21	27.42	25.57	35.03	24.98	28.01

- SIFT: KNN, 496 clusters
- Centrist + y-position in image: KNN, 196 clusters
- ColorMeanVar: KNN, 46 clusters

The results for this classifier are shown in Table 24. The ranking of the descriptors is almost the same as in the preceding experiment with the exception that Centrist is now slightly better than the Orientation Mean and Variance descriptor. Furthermore, each descriptor achieves a performance worse up to 3% compared to using the probability distribution modelling.

We also checked the performance of the modified K-Nearest Neighbors algorithm when the $K = 3$ nearest neighbors are considered for a decision. The remaining classifier settings and especially the cluster numbers are identical to the preceding experiment. The results shown in Table 25 indicate that for each single-cue descriptor the classification accuracies decrease when KNN is used with 3 neigbors. Interesting is that the more sophisticated

descriptors HOG, SIFT and Centrist drop in their performance much more with 4% to 6% than the simple mean and variance descriptors which only drop by 1% to 2%.

Analysis of the Salient Region Clusters Nevertheless, the obtained accuracies are not satisfying even with the best settings. In search for reasons of the poor performance we inspect now the contents of the found salient regions as well as the constitution of the clusters computed for the codebook of the probability distribution model with naive Bayes approximation.

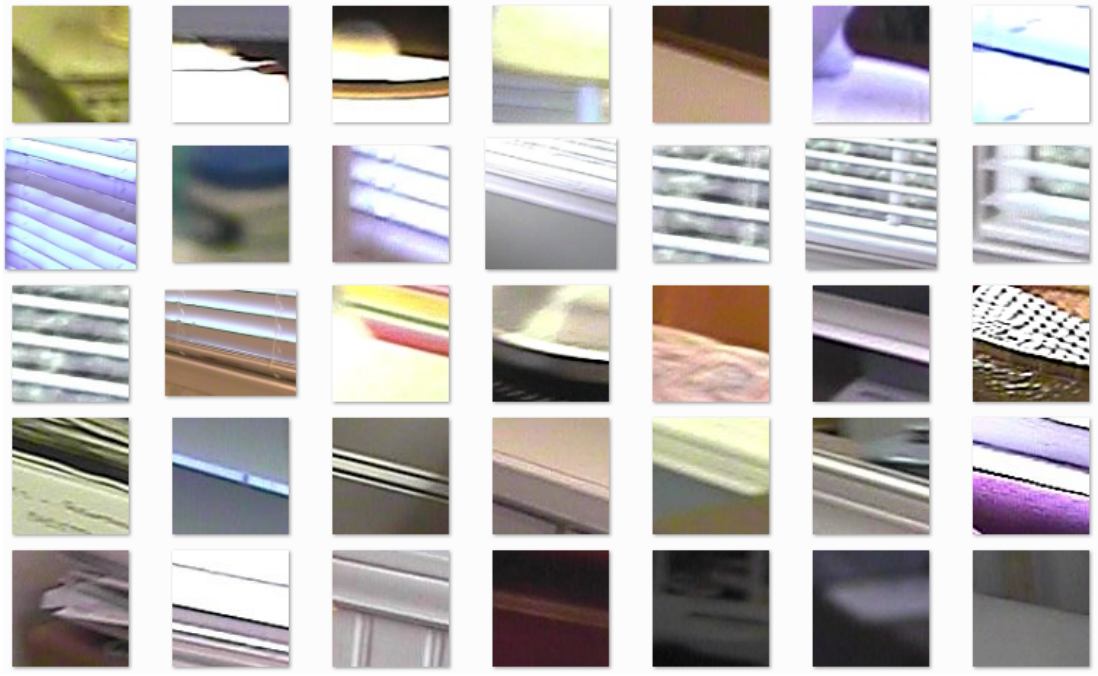
Therefore, we recorded some of the plethora of salient image patches and ordered them by their cluster association when the localized HOG descriptor is employed. Figure 12 shows several examples for cluster 1 which we divided into image patches that have an obvious meaning because they contain typical objects (see Figure 12(a)) and into image areas which do not provide good hints for the place category (see Figure 12(b)). We make three important observations in the display of the salient regions.

First, we see in Figure 12(a) that among the detected objects we can find blankets, a washbasin, a kettle, telephones, tables, newspapers, books or a brush. Although many of these objects are quite common for certain rooms, the whole cluster 1 contains all these objects from very different rooms so that a clear decision based on the cluster association of a salient region is not possible.

Second, we find that only around 10% of the salient regions contain really meaningful contents in the form of whole objects or prominent object parts. This observation is not visible in the displayed selection for cluster 1 but in the visualization of examples from cluster 2 and 3 in Figure 13 and Figure 14. Especially cluster 2 appears to favor mainly any image patch with a strong diagonal edge. The conclusion of this finding is that the bad results obtained in the experiments with the salient regions features are at least partly due to a substantial gap between the original intention of the salient regions detector, namely to find objects, and the actual behavior of the algorithm which often tends to detect other structures.

Besides the detection of non-distinctive image patches like parts of the blinds or the

(a) Detected objects.



(b) Other meaningless image patches.

Figure 12: Examples for salient region patches contained in cluster 1.

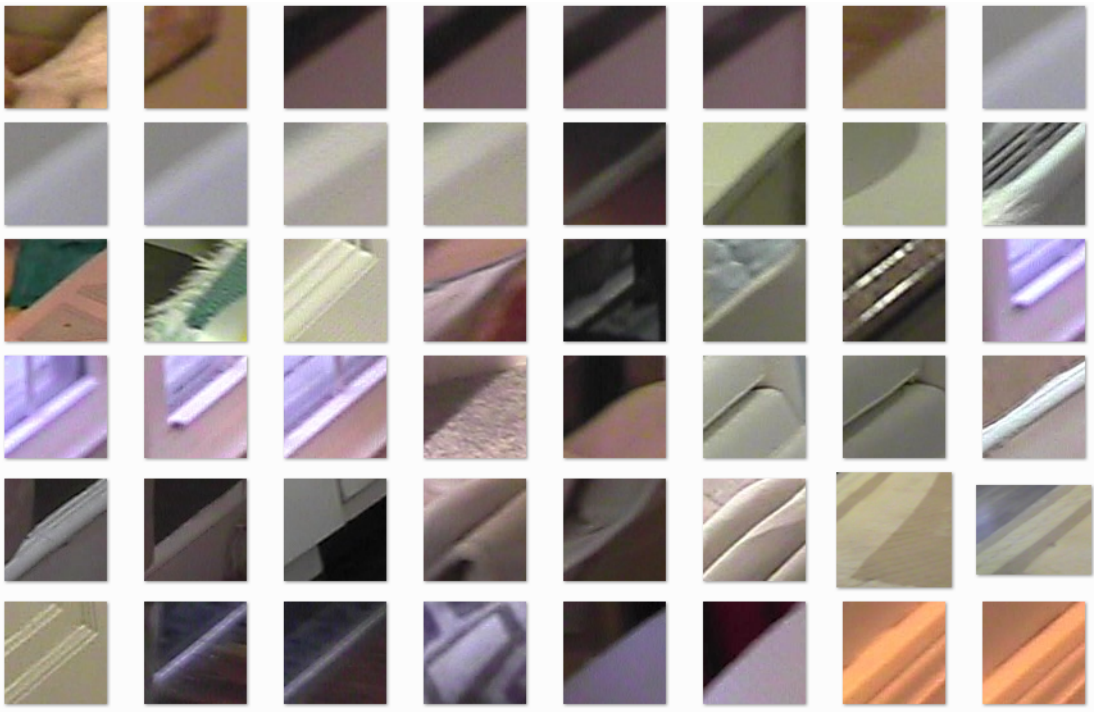


Figure 13: Examples for salient region patches contained in cluster 2.

edges between the floor and a wall we also observe some regions showing a too close or unsharp view of an object. In those cases the image segment has few structure and does not contribute useful information for distinguishing between rooms. This is another source for weak performance of this object detection approach.

Summary In conclusion, we find that the best descriptor for the objetcs found in the salient regions is Histogram of Oriented Gradients which provided the best performance independent of the employed Classifier. The best classification results were obtained with the probabilistic model using the naive Bayes approximation independent of the used descriptor.

In addition to the first finding, the obtained accuracies make SIFT the second choice for a descriptor. It is surprising that SIFT works better for our purposes than Centrist since SIFT is rather known as a feature for matching than for generalizations while Centrist is supposed to do the opposite. Because we are computing the Centrist descriptor for salient regions on the grayscale image and not on the Laplacian edge image, we assumed that the computation on the edge image in the respective scale could improve its performance as

Figure 14: Examples for salient region patches contained in cluster 3.

we observed it for the holistic Centrist descriptor (see section 4.3). However, a quick check showed that for the salient regions the performance decreased when Centrist was computed on the edge image. Consequently, it appears as the Centrist descriptor works better if it is applied to larger image regions than to focused regions. As we have just seen before, a significant part of the salient regions might also be too narrow in its view so that there is not enough gist for Centrist to capture while SIFT usually works better on such lower-level image patches.

We could also observe that the naive Bayes approximation for modelling the joint probability distribution of places works better than the modified K-Nearest Neighbors classifier. This indicates that the knowledge about the presence and absence of *all* objects ever seen provides stronger place information than only the knowledge about the objects present in the observed scene. This result makes sense since the absence of an object can put very strong additional information towards a decision. For example, a cup alone might give a high probability for kitchen and office, however, the additional information that no monitor and no keyboard is present raises the probability for kitchen and lowers it for office.

Finally, we see that even the best performance based on the approach to find objects in salient regions (37.58%) is significantly lower than the performance for a holistic image

descriptor like Gist (42.26%) or Centrist (43.99%). This is likely due to the following two facts: First, the object-based approach must fail if there are no characteristic objects visible or if they are so huge that they are not captured properly by the attention algorithm. Nevertheless the gist of the scene might still be clear so that there are certainly cases in which a holistic gist descriptor has advantages over an object descriptor. Second, as discussed earlier we assume that there is not enough training data to learn such a huge variety of objects which is normally found in household images.

At this point we have analyzed all single-cue classifiers. In the next section we evaluate if the performance of the the place categorization system can be improved if those single-cues are considered together for the place decision. Especially, the impact of additional object information from salient regions is of interest. For these descriptors we always employ the classifiers with the corresponding best settings determined in this section if not stated else.

4.5 Feature Integration

Pronobis *et al.* [58] have shown that the integration of different features via SVM-DAS (SVM-based discriminative accumulation scheme) can yield a better classification result than the single-cues can provide. Therefore, we now test the performance gain, which can be obtained from the combination of the single features investigated in the preceding sections.

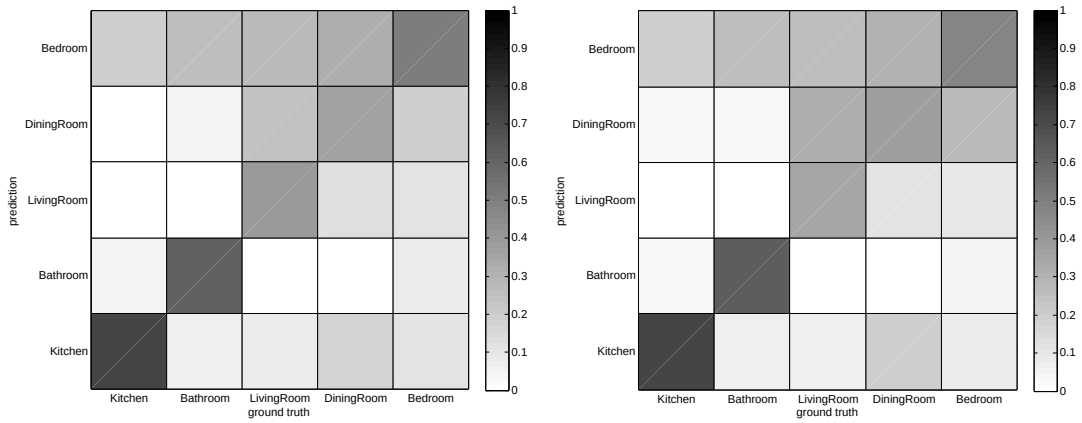
As SVM-DAS works in that way that the outputs of the single-cue classifiers constitute the descriptor and the real label the desired classifier output, we have to deal with the question whether the integrating SVM classifier should be trainind with the same data as the single-cue classifiers. If yes, then the outputs of the single-cue classifiers, which were obtained from the five homes in the training set, are used for SVM-DAS training again. We call this variant the 5-5-1 scheme since five homes are used for the training of the single-cue classifiers, the same five homes are used for the SVM-DAS training and the remaining home is the test dataset. Since we only used every third image for the single-cue classifier training and to introduce some novelty to the SVM-DAS training data, the single-cue responses for SVM-DAS training are obtained from other, hitherto unseen images of the 5 homes. The

Table 26: Categorization accuracies of the single cues, after the cue-integration through SVM-DAS and after the smoothing of the integration results. Results are shown for the 4-5-1 scheme and the 5-5-1 scheme using all available cues and for the 5-5-1 scheme if only a subset of cues is utilized.

	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
all cues used with 4-5-1 scheme							
Ori M/V + y	35.11	28.51	34.81	32.07	32.00	42.33	34.14
HOG + y	28.94	30.94	39.98	40.08	29.27	45.55	35.79
SIFT + y	32.01	28.79	41.77	36.87	32.80	39.70	35.32
Centrist + y	32.64	30.80	36.77	30.44	26.23	43.86	33.46
Color M/V + y	24.27	27.33	37.09	33.49	33.51	28.47	30.69
Gist	35.22	43.52	39.62	44.15	36.98	40.90	40.07
Centrist	43.06	40.37	44.47	48.08	38.97	41.32	42.71
Integration	31.20	43.13	43.08	48.93	40.74	46.67	42.29
HMM	29.59	50.02	44.45	54.01	44.98	50.15	45.53
all cues used with 5-5-1 scheme							
Ori M/V + y	36.55	29.36	34.32	29.77	33.61	40.37	34.00
HOG + y	35.36	31.76	44.11	39.47	29.25	45.51	37.58
SIFT + y	35.34	32.58	43.60	33.70	33.12	39.75	36.35
Centrist + y	38.07	26.45	37.39	30.86	27.66	38.93	33.23
Color M/V + y	26.46	26.23	39.35	29.65	29.20	31.54	30.41
Gist	37.20	43.82	50.04	47.80	36.52	38.18	42.26
Centrist	43.08	40.04	48.78	51.26	38.60	42.22	43.99
Integration	42.38	45.59	50.30	52.23	40.69	43.09	45.71
HMM	42.31	48.45	54.81	52.90	41.84	44.87	47.53
HOG, SIFT, Centrist (salient regions), Gist, Centrist (holistic) used with 5-5-1 scheme							
Integration	43.49	46.13	50.95	52.31	41.16	44.98	46.50
HMM	43.37	47.87	53.68	51.16	42.71	47.29	47.68

other option is to keep one home exclusively for the SVM-DAS training. This method is called the 4-5-1 scheme since only four homes are used for single cue classifier training and five for the training of SVM-DAS.

The results of the integration step are displayed in Table 26 which shows the integration of all available cues using both, the 4-5-1 and the 5-5-1 scheme at first. We learn from these results that the 5-5-1 scheme can obtain an almost 3.5% better performance for the integrated results than the 4-5-1 scheme and a 2% better performance if HMM smoothing is applied to the integration output. While the integration step does not exceed the best single-cue performance with the 4-5-1 scheme, the 5-5-1 scheme clearly manages to improve the best single-cue classifier accurancy on average and even in most of the individual tests. The



(a) Confusion matrix after the integration of all five single cues.

(b) Confusion matrix after the smoothing of the integrated output.

Figure 15: The confusion matrices for the classification performance after integration and after smoothing for home 3.

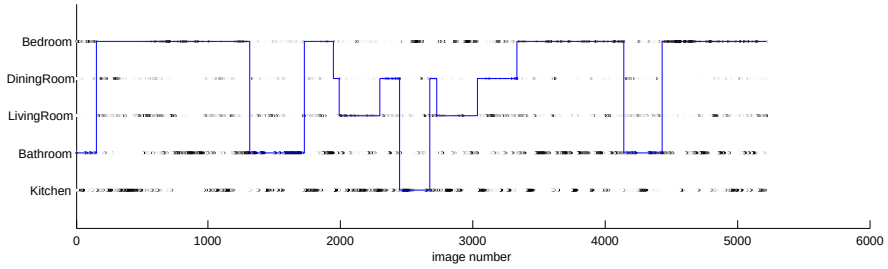
obtained accuracy after integration and HMM smoothing exceeds the 46.78% obtained with the place categorization system of Wu [80] slightly. The corresponding confusion matrices to the results after integration and after HMM smoothing are displayed in Figure 15.

For the following test we had a look at the accuracy rank of each indivudual cue compared to the others. For each test home it showed that the holistic descriptors Gist and Centrist yield the best two accuracies in most cases but the best salient region descriptors were HOG, Centrist and SIFT. HOG was the best classifier on test set home 6 and was ranked third for home 3 and 4. SIFT received the third rank on subsets 2 and 5 and Centrist had rank 2 for home 1. Because the mean and variance features were not ranked high for neither home we also evaluated the results if only HOG, SIFT and Centrist are used as salient region descriptors which are integrated together with the holistic descriptors Gist and Centrist. The outcome is displayed as the last experiment in Table 26. We observe that the accuracy of the integration can be improved furthermore by 0.79% but after applying the HMM the improvement is very small. Apparently, the integration step corrects the same mistakes of the single-cue classifiers systematically which are eliminated through smoothing in the other case.

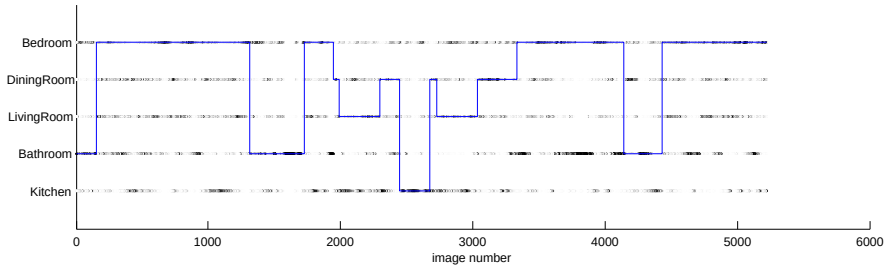
In comparison to the results when smoothing is applied to the holistic Gist and Centrist descriptors we realize that the smoothing on Centrist alone is slightly better with 48.04% than the best result from a smoothed integration output (47.68%). Although this is a discouraging result for the efforts on the other single-cue features and the integration framework, we want to point out that the integration performance, which is not dependent on the arbitrarily effective smoothing step, is still larger by 2.5% compared to using Centrist alone. This shows that the additional object information can improve the single cues at least a little bit.

In order to verify why the improvement is not bigger, we now have a closer look at the output of the single-cue classifiers. In Figure 16 we can see the probability distributions for the HOG, SIFT and Centrist descriptors employed for the description of the salient regions as well as the holistic descriptors Gist and Centrist. The probability for each place is indicated by a white, grey or black dot for each image of the image sequence for home 3. The darker the dot the higher the probability for that place. The blue line always represents the ground truth so that comparisons of the classifier outputs with the real place are easily possible.

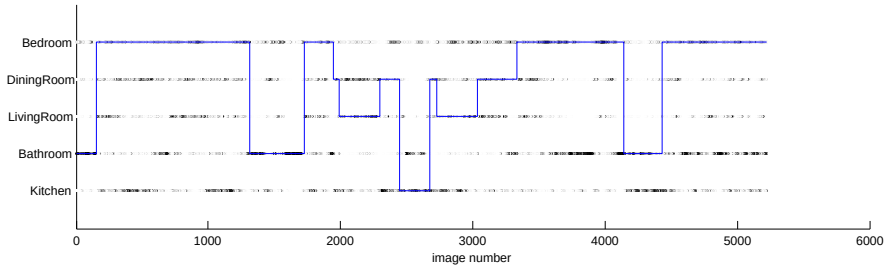
On these diagrams we can compare the outputs of the different single-cue classifiers at a given point of time. We can see that it happens often that the single-cue outputs are different but any of them is correct, for example in several segments of the first, the beginning of the third and in the first third of the last bedroom phase, multiple times in the dining room as well as in the second half of the second living room sequence. The integration scheme can obviously not help in these cases since all classifiers agree in their decision. Furthermore, we observe that sometimes all classifiers vote for the same answer which is the right one in some cases, for example during the second sequence in the bathroom and in the kitchen, and which is wrong in other cases like in the first third of last bedroom sequence when all classifiers decide for bathroom. Specifically, the SVM-DAS integration mechanism cannot decide for a different category although all classifiers agree on a certain class because we also find examples in which all classifiers agree for the correct class in one case, for example during the first bedroom sequence at the end, and for the wrong class (bedroom) in another



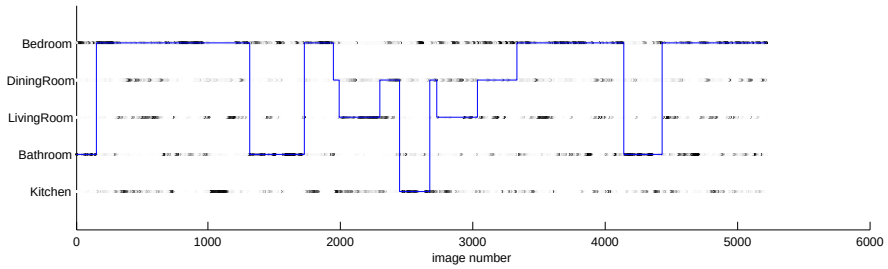
(a) Salient Regions - HOG



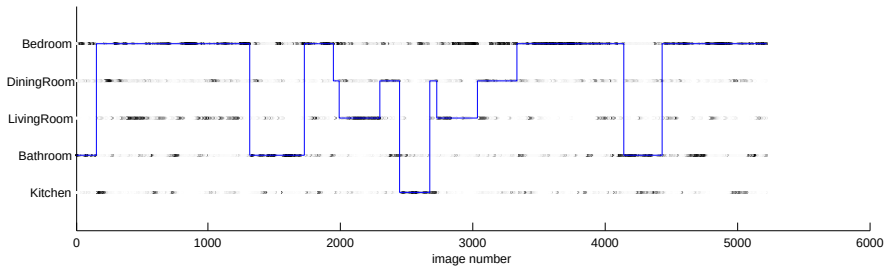
(b) Salient Regions - SIFT



(c) Salient Regions - Centrist

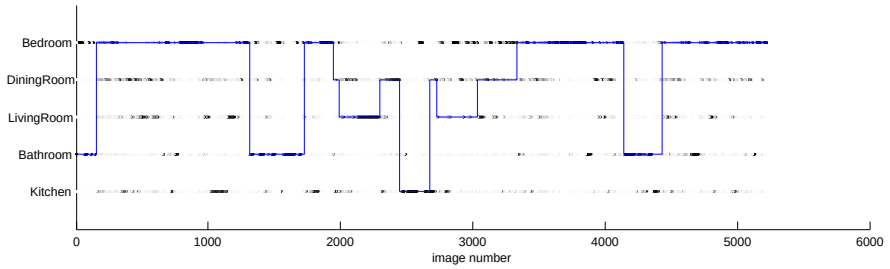


(d) Holistic - Gist

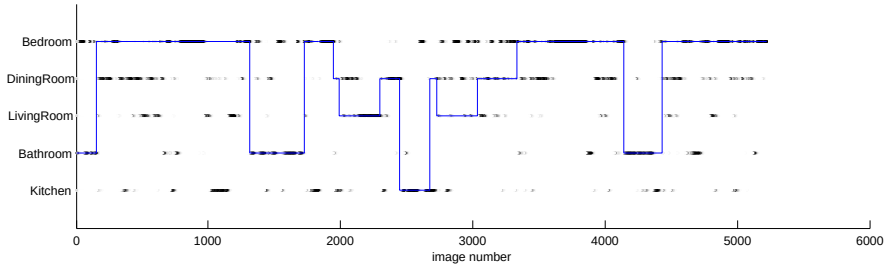


(e) Holistic - Centrist

Figure 16: The probability distributions for each classifier on each image of home 3.



(a) After the integration of all five single cues.



(b) After the smoothing of the integrated output.

Figure 17: The probability distributions after integration and after smoothing for home 3.

(a) Example from the last third of the first bedroom sequence.

(b) Example from the end of the first dining room sequence.

Figure 18: Examples images for which all classifiers output a wrong decision.

case, for example in the middle of the second living room sequence. There is no way that the integration scheme could determine that the first decision for bedroom is right while the second is wrong when all classifiers decide for bedroom in both cases. This is one challenge for the integration scheme which explains the poor improvement through feature integration because there is no solution at the level of classification. Only better descriptors with a more diverse error distribution could help here.

Another challenge and possible source for wrong decisions after integration is the choice of the right cue if the different single-cue classifiers are contradicting. The right behaviour

in those cases should be learned during the training phase of the SVM-DAS, however, there are two reasons why this cannot be done perfectly. First, we trained the integration algorithm with data from the same rooms which were used for the single-cue classifier training. Although the individual images are different, we cannot expect that this approach presents the SVM-DAS integrator with such diverse training data as if a completely unknown data set would be available in addition. Nevertheless, we decided for the 5-5-1 scheme because the overall result was better, probabaly because more training data was available to the single-cue classifiers. We also did a quick check with additional data for the SVM-DAS training from the Aware Home taken downstairs with the Flip camera. We observed almost no difference to the standard procedure and conclude that probably the SVM integration scheme cannot perform much better if the single-cue classifiers do not deliver more accurate decisions. This finding corresponds with the observations made during the comparison of the 4-5-1 and the 5-5-1 scheme.

The second reason for integration errors when at least one cue yields the right answer is that there is generally either not enough training data available to the SVM-DAS to learn all the facets of single-cue outputs or the training data is contradicting. We can observe resulting decisions after the integration and after the smoothing of the integrated results in Figure 17. There we can see that the SVM-DAS cannot handle all of the hard cases, for example when only one of the single cues provides the correct decision which is the case for the first bathroom sequence when only the salient region's Centrist descriptor yields the right result or at the beginning of the first bathroom when only Gist outputs the correct decision. A very hard example can be found at the end of the second bathroom sequence where all descriptors wrongly decide for kitchen except for HOG which yields the correct output. It is unlikely that this case was presented with the training data or if it was, there were probably a lot more cases in which the four other classifiers were right, so that the result after integration is wrong for this special example.

Finally, we have to realize that the integration scheme cannot arbitrarily select the correct class decision whenever at least one of the single-cue outputs provides it. We can rather observe some smaller improvements up to 3% and find that the integrated results

(a) A shower in a bathroom. Information measure = 0.09519.

(b) A wall. Information measure = 0.04046.

Figure 19: Examples for uninformative views.

almost always outperform the best single-cue classifier although this is a different one in dependence of the testing subset. Apparently the bigger problem, which prevents the system from a better performance, is the cluttered output of the single-cue classifiers which often switch between different decisions after a period of time which is too long to be filtered out with the HMM smoothing. More stable and more accurate decisions are the main need after the analysis so far. We therefore try to improve the categorization performance in the next section by removing some of the views which are hard to classify.

4.6 Information Filter

One possible source for the bad performance of all previous approaches might be the uninformative viewing angles, for example when the robot is facing a wall, which occur in the video sequences from time to time. Since most of the employed descriptors describe the structure of the scene it makes sense to declare those views informative which contain a lot of intermediate frequency contents. This corresponds to the presence of objects, which are necessary for the approach using the salient regions to find objects. High frequency contents instead are not useful since they describe patterns and detail information without generalization and very low frequency contents describe intensities and illumination changes which also do not contain object information.

We therefore filter the image with a bandpass filter which is effectively done by a substraction of the last scale of the Gaussian pyramid from the second. The average of pixel intensities of the bandpass-filtered image is considered as a measure for the information

Table 27: Categorization accuracies when the information filter is applied with a threshold of 0.09766.

	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
holistic Centrist descriptor							
Centrist	45.05	41.49	47.04	50.31	39.33	38.75	43.66
HMM	55.78	44.93	53.36	54.85	42.65	42.21	48.96
HOG, SIFT, Centrist (salient regions), Gist, Centrist (holistic) used with 5-5-1 scheme							
HOG + y	33.43	34.70	47.26	36.75	30.68	42.13	37.49
SIFT + y	33.23	31.64	43.66	36.39	33.25	33.05	35.20
Centrist + y	37.36	23.41	38.14	31.53	19.83	32.80	30.51
Gist	37.89	42.30	49.28	46.87	39.48	38.90	42.45
Centrist	45.05	41.49	47.04	50.31	39.33	38.75	43.66
Integration	46.99	44.64	53.02	52.25	41.57	41.38	46.64
HMM	44.70	48.35	54.14	52.67	40.88	40.65	46.90

content of the scene. Manual inspection of the whole dataset showed that a information content threshold of 0.09766 can filter out many images which mainly contain walls while preserving the informative views. Figure 19 shows two examples for views at a wall which have low information content. Unfortunately, some informative views, which only contain very few objects, have a low information content and are also filtered out with this setting, especially for the bathroom class. The classification for the images declined by the filter is the same as for the last accepted image, i.e. if the system knows that an image is not descriptive it outputs the result of the last informative view. The results for using only the holistic Centrist descriptor as well as for integrating the three best salient regions descriptors with Gist and Centrist are shown in Table 27.

We can observe that the HOG descriptor for the salient regions and the holistic Gist and Centrist descriptors reach accuracies which are close to those without information filter. SIFT and Centrist for the salient regions show a drop in categorization performance of 1.15% and 2.95%, respectively. The integration of the five cues provides a performance only slightly better than without information filter. For the HMM we get a 1% improvement for the holistic centrist descriptor but also a 0.78% decrease for the integrated features.

A quick check for the Centrist descriptor on home 2 showed furthermore that a higher

threshold (0.13021) decreases the categorization performance while manual inspection suggests that a lower threshold would almost have no impact at all.

In conclusion, the application of the information filter did not show much effect. Manual inspection showed that the filtered images were classified correctly without filter for several sequences so that the filtering could not remove wrong decisions. Furthermore, if the place decision is wrong exactly before a filtered sequence, the whole filtered sequence inherits the wrong classification. Since the overall accuracy suggests that probably in every second case the classifiers provide a wrong decision before a sequence is filtered out we can understand why the information filter cannot show greater improvements. We expect the filter to be more valuable if the general classification rate can be increased.

4.7 Salient Region Tracking

In this last experiment we check a more biologically inspired approach in connection with the detection of salient regions. Kahneman and Treisman [29, 74] introduced the notion that humans track objects already when they are not yet identified. The accumulation of information about it over time makes the recognition easier. In the meantime all information about the object is stored in a so-called object file.

We adopted this idea in the software framework and tracked salient regions until they leave the image during both, the training as well as the testing phase. In the training phase we manage to collect more views of the found objects which gives hope that those might later be identified better. However, although this functionality apparently makes sense, we must realize in Table 28 that there is a small decrease in performance of around 1% for the salient region classifiers instead of an improvement. We also observe that the performance of the integration step drops a little bit. Nevertheless, the performance of the HMM smoothing is better than for the preceding approaches.

All in all these results show that the biologically more plausible tracking approach cannot improve the performance but also does not crucially hurt it. Again, we could see that there are no guarantees for improvements or their quantity when the HMM is used since its behavior is unpredictable.

Table 28: Effect of the object tracking on the classification accuracy.

	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
HOG + y	28.21	28.68	47.05	39.42	29.00	43.75	36.02
SIFT + y	34.62	28.47	38.38	32.92	31.50	40.41	34.38
Centrist + y	36.14	22.71	37.03	29.47	25.88	38.30	31.59
Gist	37.20	43.82	50.04	47.80	36.52	38.18	42.26
Centrist	43.08	40.04	48.78	51.26	38.60	42.22	43.99
Integration	42.60	44.48	53.41	52.15	41.00	45.24	46.48
HMM	42.49	48.08	54.63	53.18	42.28	47.13	47.97

4.8 Comparison of the Sequential and the Parallel Multiclassifier Scheme

After all the descriptor and classifier tweaking, we finally compared the parallel multi-classifier scheme introduced in section 3.3.1.2 with the sequential scheme of Mozos [46]. The comparison was done on a toy example as well as on the home dataset.

The toy example required the classifiers to distinguish between the inner areas of four disjunct circles in the 2-dimensional Euclidian space and the area around the circles. Both classification schemes solve this simple five-class problem very well when a SVM with $\gamma = 2.0$ and $\nu = 0.2$ is used as base classifier. Specifically, for the sequential classifier the overall classification accuracy is 99.56% whereas for our parallel multi-class scheme the accuracy is 99.74%. It shows clearly that the sequential classifier makes the most errors in the last last decision node of the sequence. This happens when every single-cue classifier outputs a negative decision and the last class is automatically chosen. The sequential classifier shows an error three times bigger than the error for our probabilistic multi-class scheme for this respective class.

The second test on the home dataset confirms the better performance of our parallel multiclass scheme. We checked the classification accuracies for the whole home database using a SVM with $\gamma = 2.0$ and $\nu = 0.2$ as base classifier and the holistic Centrist descriptor as data source. In Table 29 we can perceive that out probabilistic parallel multi-class scheme outperforms the sequential classifier by almost 3.5%. Furthermore, its performance is always better for each testing subset with an exception for home 2.

These results indicate that the more principled probabilistic treatment of the two-class

Table 29: Comparison of the sequential and the parallel multi-class schemes with SVM base classifiers on the home dataset. The applied descriptor is the holistic Centrist descriptor.

Centrist	Home 1	Home 2	Home 3	Home 4	Home 5	Home 6	Average
Sequential	40.06	41.55	44.16	45.92	34.50	37.15	40.56
Probabilistic	43.08	40.04	48.78	51.26	38.60	42.22	43.99

classifier outputs is better-suited at least for our problem than the pseudo-probabilistic output of the sequential classifier.

4.9 Test on the COLD Database

We also tested our algorithm on the COLD database [55] which covers university environments. Because of the bad quality of the Ljubljana subset we decided to use only the Freiburg and the Saarbrücken subsets for our tests. We tried to use the extended sequences when possible since they contain more data. Specifically, we used the sequences Saarbrücken 2, cloudy 2, Freiburg 2, cloudy 2 and the kitchen of Saarbrücken 4, cloudy 3 for the training of the single-cue classifiers and the same sequences with cloudy 3 (and 2 for the kitchen) for the training of the integration scheme. The test set contained of the sequences Freiburg 3, cloudy 2 and Saarbrücken 4, cloudy 3 which contain physically different rooms than the sequences used for training with the exception that the kitchen is the same because there was only one kitchen in the database. All available rooms were mapped to the classes corridor, kitchen, office, printer area and bathroom.

The categorization results obtained with the usual best settings determined earlier can be found in Table 30. We observe a performance around 5% better than for the home environment if all five single-cue classifiers are employed although there was way less training data available in advance. The reason for this result is probably the fact that although the testing was done in other rooms the environment was still quite similar to the training set because of the common architecture within each university.

It shows that in contrast to the results of [76] we do not only achieve good results on the corridor class but also on the office and the bathroom categories. The kitchen result must be interpreted as a recognition result since there was only one kitchen in the dataset.

Table 30: Performance on the COLD database.

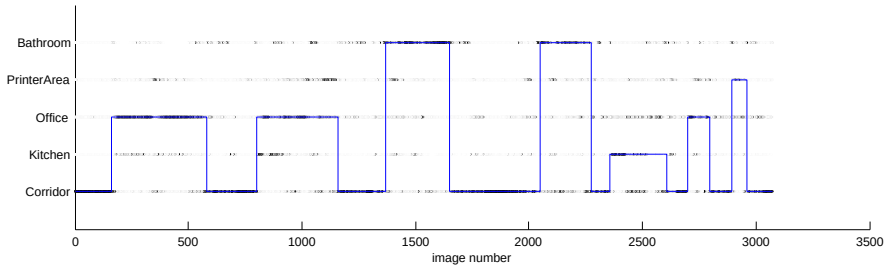
	Corridor	Kitchen	Office	Printer Area	Bathroom	Average
HOG + y	57.80	50.00	36.90	29.20	52.40	45.27
SIFT + y	52.10	22.40	42.00	1.50	47.80	33.17
Centrist + y	38.70	14.00	35.20	1.50	57.30	29.35
Gist	82.90	27.60	53.80	0.00	53.00	43.46
Centrist	89.20	26.80	73.70	16.90	78.60	57.03
Integration	92.80	27.60	64.50	1.50	54.20	48.11
HMM	95.30	29.20	64.50	0.00	76.80	53.15
HOG (salient regions), Gist and Centrist (holistic) only						
Integration	91.40	28.40	65.40	7.70	62.50	51.08
HMM	94.50	33.20	62.00	9.20	78.20	55.41

The detection rate is at 33% because only the first third of the sequence really looks like a kitchen whereas the remainder rather resembles a printer area or an office. In the plots of the probability mass distributions in Figure 20 we can observe that besides a corridor detection sequence, office is detected exactly when the kitchen looks like an office. This shows some generalization abilities of the employed method. It is also nicely visible that the HMM smoothing corrects the clutter after the integration step to some smooth decision bars.

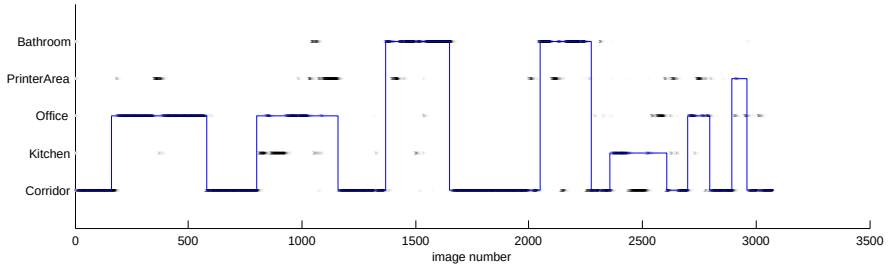
However, we furthermore have to realize that the individual classifiers exhibit very different performances which leads to an integrated categorization accuracy which is 9% below the best single cue Centrist. Apparently, the integration scheme can only work successfully if the single cues do not differ too much in their performance. When we only apply HOG together with Gist and Centrist the integration accuracy increases at least by 3% but is still worse than Centrist alone. Even the HMM smoothing cannot recover from that error in both cases.

4.10 Test with a Real Robot

For the last experiment of this thesis we collected some data downstairs in the Aware Home of Georgia Tech with a robot mounted on a Segway RMP-200 mobile platform. We furthermore captured some own images with a flip camera upstairs in the Aware Home where a second apartment is located as well as in another flat of an apartment complex.



(a) After the integration of HOG, Gist and Centrist.



(b) After the smoothing of the integrated output.

Figure 20: The probability distributions after integration and after smoothing for the COLD dataset.

We used both of the latter datasets for the training of the base classifiers and the SVM-DAS. The test set was the data acquired with the robot.

In Table 31 we can see that although we are distinguishing eight classes this time the algorithm can still classify one third of the images. This is an acceptable result compared with the observed performance so far especially if we consider that the training set was captured with two different cameras, a Flip cam and a webcam, whereas the robot used a professional Prosilica camera. We can also recognize that the cue integration scheme manages to improve the accuracy 3.5% over the best single-cue result. Typically for the

Table 31: Performance on the Aware Home database.

	Corr.	Kit.	Office	Bath	Living	Dining	Bed	Closet	Average
HOG + y	28.90	17.90	25.60	40.90	23.40	31.80	8.20	0.40	22.13
SIFT + y	28.80	13.20	33.30	33.50	30.50	34.80	3.90	0.00	22.25
Centrist + y	0.40	47.70	11.80	53.40	41.20	7.90	4.60	4.70	21.91
Gist	45.40	23.10	33.30	25.40	54.00	32.70	1.60	0.40	27.01
Centrist	34.90	35.50	22.80	74.10	17.00	48.80	0.00	14.00	30.88
Integration	45.50	40.00	33.10	63.60	45.30	46.00	1.20	0.00	34.35
HMM	24.50	27.60	14.60	51.40	44.10	48.60	0.00	46.00	32.90

unpredictable behavior of the HMM smoothing is that this time the performance decreases through the application of smoothing.

CHAPTER V

CONCLUSION

The main goal of this work was the assessment of a new method of analyzing a scene for visual place categorization which tries to imitate human strategies on this task. We used a visual attention mechanism to capture important and typical objects of the scene and collect them in a database. This procedure ideally has the advantage over algorithms relying on object detectors for a small set of supervisedly learned objects that it does not need segmented and labelled object data. Moreover, through the unsupervised method for collecting object information, the presented system is capable of collecting a much bigger set of different objects than previous approaches.

However, in spite of these potential properties we faced some limitations of the employed algorithms, especially for the part of automated object search. The salient regions approach finds way too few meaningful image patches which contain at least distinguishing object parts or even whole objects. Instead a lot of smaller building blocks of the world are found, for example strong edges between walls. Further work on this method should aim at systematically eliminating these uninformative cases in order to assess the performance when characteristic salient region patches are available which could also be identified by a human observer.

The division of the image patches into clusters showed to be another potential source of performance loss since objects from different rooms regularly found together in same clusters. A better separation of those objects would enable the classifier for better performance.

Another finding of this work is that the Gist descriptor, which has already been tested for outdoor place recognition [69], is almost as good for indoor place categorization as the Centrist descriptor. Moreover, the integration of Gist, Centrist and salient region descriptors via SVM-DAS showed that we can reach at least the performance of the best individual cue in each case we tested the algorithm. We observed furthermore that delayed

HMM smoothing can improve the results to some extent but that there is no guarantee for the degree of improvement.

In the end, both, the integration method of all cues and the holistic Centrist descriptor could reach a performance around 48% which is a little improvement to the original Centrist categorization system [80] but not a substantial one. Results like these are still far away from been useful for real robot applications since there is no use for semantic place information which is only correct half of the times.

Nevertheless, we could present a new multi-class classifier framework for SVMs or AdaBoost which deals with probabilistic outputs in a principled way and obtains significant performance improvements compared to the sequential scheme of [43].

Besides the obviously necessary improvements on the automated object detection with salient regions which we already mentioned above, another interesting extension to the system might be the spatial accumulation of cues instead of the temporal accumulation applied within this work. The advantages are clear: While there is no computationally efficient way to determine the robot localization just from the image sequence additional localization information would enable the robot to assign more consistent place labels since it could determine connected areas which should have the same label. Even movement information could already help the robot to decide on a coarse basis when a place label change makes sense.

REFERENCES

[1] Aksoy, S., Koperski, K., Tusk, C., Marchisio, G., and Tilton, J. C., "Learning bayesian classifiers for scene classification with a visual grammar," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 43, pp. 581–589, 2005.

[2] Arthur, D. and Vassilvitskii, S., "K-means++: The advantages of careful seeding," in *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, (Philadelphia, PA, USA), pp. 1027–1035, Society for Industrial and Applied Mathematics, 2007.

[3] Bosch, A., Zisserman, A., and Munoz, X., "Scene classification via plsa," in *Proceedings of the European Conference on Computer Vision*, 2006.

[4] Bosch, A., Munoz, X., and Martí, R., "Review: Which is the best way to organize/classify images by content?," *Image and Vision Computing*, vol. 25, no. 6, pp. 778–791, 2007.

[5] Bradski, G. and Kaehler, A., *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc., 2008.

[6] Caputo, B. and Dorko, G., "How to combine color and shape information for 3d object recognition: Kernels do the trick," in *NIPS'02*, 2002.

[7] Chen, C. and Wang, H., "Appearance-based topological bayesian inference for loop-closing detection in a cross-country environment," *International Journal of Robotics Research*, vol. 25, no. 10, pp. 953–983, 2006.

[8] Chen, P., Lin, C.-J., and Schölkopf, B., "A tutorial on ν-support vector machines," *Applied Stochastic Models in Business and Industry*, vol. 21, p. 111136, 2005.

[9] Chow, C. and Liu, C., "Approximating discrete probability distributions with dependence trees," *IEEE Transactions on Information Theory, 14(3), pp. 462-467*, 1968.

[10] Christensen, H. I., Kruijff, G.-J. M., and Wyatt, J. L., eds., *Cognitive Systems (Cognitive Systems Monographs)*. Springer, 1 ed., 2010.

[11] Cummins, M. and Newman, P., "Fab-map: Probabilistic localization and mapping in the space of appearance," *The International Journal of Robotics Research 2008, 27, p. 647*, 2008.

[12] Cummins, M. and Newman, P., "Highly scalable appearance-only SLAM - FAB-MAP 2.0," in *Proceedings of Robotics: Science and Systems*, (Seattle, USA), June 2009.

[13] Dalal, N. and Triggs, B., "Histograms of oriented gradients for human detection," in *CVPR, vol. II, pages 886-893*, 2005.

[14] Ekvall, S., Kragic, D., and Jensfelt, P., "Object detection and mapping for service robot tasks," *Robotica*, vol. 25, no. 2, pp. 175–187, 2007.

[15] Espinace, P., Kollar, T., Soto, A., and Roy, N., "Indoor scene recognition through object detection," in *ICRA*, pp. 1406–1413, 2010.

[16] Fei-Fei, L. and Perona, P., "A bayesian hierarchical model for learning natural scene categories," in *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, (Washington, DC, USA), pp. 524–531, IEEE Computer Society, 2005.

[17] Felzenszwalb, P., Mcallester, D., and Ramanan, D., "A discriminatively trained, multiscale, deformable part model," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) Anchorage, Alaska*, June 2008.

[18] Folkesson, J. and Christensen, H. I., "Graphical slam - a self-correcting map," in *IEEE International Conference on Robotics and Automation*, pp. 383–390, 2004.

[19] Freund, Y. and Schapire, R. E., "A decision-theoretic generalization of on-line learning and an application to boosting," in *EuroCOLT '95: Proceedings of the Second European Conference on Computational Learning Theory*, (London, UK), pp. 23–37, Springer-Verlag, 1995.

[20] Friedman, J., Hastie, T., and Tibshirani, R., "Additive logistic regression: a statistical view of boosting," *Annals of Statistics*, vol. 28, no. 2, pp. 337–407, 2000.

[21] Frintrop, S., *Vocus: A visual attention system for object detection and goaldirected search*. PhD thesis, Rheinische Friedrich-Wilhelms-Universität Bonn, 2006.

[22] Frintrop, S., Rome, E., and Christensen, H. I., "Computational visual attention systems and their cognitive foundations: A survey," *ACM Transactions on Applied Perception*, vol. 7, no. 1, pp. 1–39, 2010.

[23] Harris, C. and Stephens, M., "A combined corner and edge detector," in *Proceedings of The Fourth Alvey Vision Conference*, pp. 147–151, 1988.

[24] Hess, R., "An open source sift library," in *Proc. ACM Multimedia (MM)*, 2010.

[25] Ho, K. and Newman, P., "Combining visual and spatial appearance for loop closure detection in slam," in *Proceedings of the European Conference on Mobile Robotics*, 2005.

[26] Hou, X. and Zhang, L., "Saliency detection: A spectral residual approach," in *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR07). IEEE Computer Society*, pp. 1–8, 2007.

[27] Itti, L., "Automatic foveation for video compression using a neurobiological model of visual attention," *IEEE Transactions on Image Processing*, vol. 13, pp. 1304–1318, Oct 2004.

[28] Itti, L., Koch, C., and Niebur, E., "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[29] Kahneman, D. and Treisman, A., *Changing views of attention and automaticity*. New York: Academic press, 1984.

[30] Kientz, J. A., Patel, S. N., Jones, B., Price, E., Mynatt, E. D., and Abowd, G. D., "The georgia tech aware home," in *CHI '08: CHI '08 extended abstracts on Human factors in computing systems*, (New York, NY, USA), pp. 3675–3680, ACM, 2008.

[31] King, D. E., "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[32] Kollar, T. and Roy, N., "Utilizing object-object and object-scene context when planning to find things," in *ICRA'09: Proceedings of the 2009 IEEE international conference on Robotics and Automation*, (Piscataway, NJ, USA), pp. 4116–4121, IEEE Press, 2009.

[33] Kroese, B. J. A., Vlassis, N. A., Bunschoten, R., and Motomura, Y., "A probabilistic model for appearance-based robot localization," *Image and Vision Computing*, vol. 19, no. 6, pp. 381–391, 2001.

[34] Kuffler, S. W., "Discharge patterns and functional organization of mammalian retina," *J. Neurophysiol., vol. 16, pp. 3768*, 1953.

[35] Lazebnik, S., Schmid, C., and Ponce, J., "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *In CVPR*, pp. 2169–2178, 2006.

[36] Lienhart, R., Kuranov, A., and Pisarevsky, V., "Empirical analysis of detection cascades of boosted classifiers for rapid object detection," *Pattern Recognition*, pp. 297–304, 2003.

[37] Linde, O. and Lindeberg, T., "Object recognition using composed receptive field histograms of higher dimensionality," in *in Proc. ICPR04*, 2004.

[38] López, D. G., Sjö, K., Paul, C., and Jensfelt, P., "Hybrid laser and vision based object search and localization," in *Proceedings of the International Conference on Robotics and Automation (ICRA'08)*, 2008.

[39] Lowe, D. G., "Object recognition from local scale-invariant features," *IEEE International Conference on Computer Vision*, vol. 2, p. 1150, 1999.

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

[41] Luo, J., Pronobis, A., Caputo, B., and Jensfelt, P., "The KTH-IDOL2 database," Tech. Rep. CVAP304, Kungliga Tekniska Hoegskolan, CVAP/CAS, October 2006.

[42] Luo, J., Pronobis, A., Caputo, B., and Jensfelt, P., "Incremental learning for place recognition in dynamic environments," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS07)*, (San Diego, CA, USA), October 2007.

[43] Martínez Mozos, O., Stachniss, C., and Burgard, W., "Supervised learning of places from range data using adaboost," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1742–1747, 2005.

[44] Mery, D. and Soto, A., "Features: the more the better," in *ISCGAV'08: Proceedings of the 8th conference on Signal processing, computational geometry and artificial vision*, (Stevens Point, Wisconsin, USA), pp. 46–51, World Scientific and Engineering Academy and Society (WSEAS), 2008.

[45] Montemerlo, M., Thrun, S., Koller, D., and Wegbreit, B., "Fastslam: A factored solution to the simultaneous localization and mapping problem," in *In Proceedings of the AAAI National Conference on Artificial Intelligence*, pp. 593–598, AAAI, 2002.

[46] Mozos, Ó. M. and Burgard, W., "Supervised learning of topological maps using semantic information extracted from range data," in *IROS*, pp. 2772–2777, 2006.

[47] Mozos, O. M., Triebel, R., Jensfelt, P., Rottmann, A., and Burgard, W., "Supervised semantic labeling of places using information extracted from laser and vision sensor data," *Robotics and Autonomous Systems Journal*, vol. 55, pp. 391–402, May 2007.

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

[49] Newman, P., Cole, D., and Ho, K., "Outdoor slam using visual appearance and laser ranging," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, (Orlando, Florida), 2006.

[50] Newman, P. and Ho, K., "Slam - loop closing with visually salient features," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2005.

[51] Nilsback, M. E. and Caputo, B., "Cue integration through discriminative accumulation," in *CVPR*, 2004.

[52] Oliva, A. and Schyns, P. G., "Coarse blobs or fine edges? evidence that information diagnosticity changes the perception of complex visual stimuli," *Cognitive Psychology*, vol. 34, no. 1, pp. 72 – 107, 1997.

[53] Oliva, A. and Torralba, A., "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, pp. 145–175, 2001. 10.1023/A:1011139631724.

[54] Perronnin, F., Dance, C., Csurka, G., and Bressan, M., "Adapted vocabularies for generic visual categorization," in *European Conference on Computer Vision*, vol. 4, (Graz, Austria), pp. 464–475, 2006.

[55] Pronobis, A. and Caputo, B., "COLD: COsy Localization Database," *The International Journal of Robotics Research (IJRR)*, vol. 28, May 2009.

[56] Pronobis, A., Caputo, B., Jensfelt, P., and Christensen, H. I., "A discriminative approach to robust visual place recognition," in *Proc. IROS06*, 2006.

[57] Pronobis, A. and Caputo, B., "Confidence-based cue integration for visual place recognition," in *Proceedings IROS07*, 2007.

[58] Pronobis, A., Mozos, Ó. M., and Caputo, B., "Svm-based discriminative accumulation scheme for place recognition," in *ICRA*, pp. 522–529, 2008.

[59] Quattoni, A. and Torralba, A. B., "Recognizing indoor scenes," in *CVPR*, pp. 413–420, 2009.

[60] Quelhas, P., Monay, F., Odobez, J.-M., Gatica-Perez, D., and Tuytelaars, T., "A thousand words in a scene," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 1575–1589, 2007.

[61] Ranganathan, A. and Dellaert, F., "Semantic modeling of places using objects," in *Proceedings of Robotics: Science and Systems*, (Atlanta, GA, USA), June 2007.

[62] Rensink, R. A., "The dynamic representation of scenes," *Visual Cognition, vol. 7, pp. 17 42*, 2000.

[63] Rottmann, A., Mozos, O. M., Stachniss, C., Burgard, W., Martínez, Ó., Cyrill, M., and Burgard, S. W., "Semantic place classification of indoor environments with mobile robots using boosting," in *in Proc. of the National Conference on Artificial Intelligence (AAAI*, pp. 1306–1311, 2005.

[64] Russell, B., Torralba, A., Murphy, K., and Freeman, W., "Labelme: A database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, pp. 157–173, 2008. 10.1007/s11263-007-0090-8.

[65] Sanocki, T. and Epstein, W., "Priming spatial layout of scenes," *Psychol. Sci.*, vol. 8, pp. 374–378, 1997.

[66] Schölkopf, B., Smola, A., Williamson, R., and Bartlett, P. L., "New support vector algorithms," *Neural Computation*, vol. 12, pp. 1207–1245, 2000.

[67] Schyns, P. G. and Oliva, A., "From blobs to boundary edges: Evidence for time and spatial scale dependent scene recognition," *Psychological Science*, vol. 5, pp. 195–200, 1994.

[68] Se, S., Lowe, D., and Little, J., "Vision-based mobile robot localization and mapping using scale-invariant features," in *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA*, pp. 2051–2058, 2001.

[69] Siagian, C. and Itti, L., "Rapid biologically-inspired scene classification using features shared with visual attention," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 300–312, 2007.

[70] Siagian, C. and Itti, L., "Biologically inspired mobile robot vision localization," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 861–873, 2009.

[71] Szeliski, R., "Image alignment and stitching: a tutorial," *Found. Trends. Comput. Graph. Vis.*, vol. 2, no. 1, pp. 1–104, 2006.

[72] Thrun, S., Gutmann, S., Fox, D., Burgard, W., and Kuipers, B. J., "Integrating topological and metric maps for mobile robot navigation: A statistical approach," in *AAAI Conference on Artificial Intelligence*, pp. 989–995, 1998.

[73] Torralba, A., Murphy, K. P., Freeman, W. T., and Rubin, M. A., "Context-based vision system for place and object recognition," *Computer Vision, IEEE International Conference on*, vol. 1, p. 273, 2003.

[74] Treisman, A., "Representing visual objects," 1991.

[75] Treisman, A. M., *The perception of features and objects.* Eds. Clarendon Press, Oxford, 1993.

[76] Ullah, M. M., Pronobis, A., Caputo, B., Luo, J., Jensfelt, P., and Christensen, H. I., "Towards robust place recognition for robot localization," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA08)*, (Pasadena, CA, USA), May 2008.

[77] Viswanathan, P., Southey, T., Little, J., and Mackworth, A., "Automated place classification using object detection," in *Canadian Conference on Computer and Robot Vision (CRV)*, pp. 324 – 330, 2010.

[78] Vogel, J. and Schiele, B., "A semantic typicality measure for natural scene categorization," in *Pattern Recognition Symposium, DAGM*, 2004.

[79] Wolf, J., Burgard, W., and Burkhardt, H., "Robust visionbased localization by combining an image-retrieval system with monte carlo localization," *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 208–216, 2005.

[80] Wu, J., *Visual Place Categorization.* PhD thesis, Georgia Institute of Technology, 2009.

[81] Wu, J., Christensen, H. I., and Rehg, J. M., "Visual place categorization: Problem, dataset, and algorithm," in *IROS*, 2009.

[82] Wu, J. and Rehg, J. M., "Centrist: A visual descriptor for scene categorization," in *submitted to IEEE Trans. PAMI*, 2009.

[83] Zabih, R. and Woodfill, J., "Non-parametric local transforms for computing visual correspondence," in *ECCV '94: Proceedings of the Third European Conference-Volume II on Computer Vision*, (London, UK), pp. 151–158, Springer-Verlag, 1994.

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