THESIS

# VISUAL LOCATION AWARENESS FOR MOBILE ROBOTS USING FEATURE-BASED VISION

Submitted by

Alexander Kazeka

Department of Computer Science

In partial fulfillment of the requirements

for the Degree of Master of Science

Colorado State University

Fort Collins, Colorado

Spring 2010

COLORADO STATE UNIVERSITY

March 23, 2009

WE HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER OUR SUPERVISION BY ALEXANDER KAZEKA ENTITLED VISUAL LOCATION AWARENESS FOR MOBILE ROBOTS USING FEATURE-BASED VISION BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE.

Committee on Graduate Work

Anthony A. Maciejewski

Charles W. Anderson

Advisor: Bruce A. Draper

Department Chair: L. Darrell Whitley

# ABSTRACT OF THESIS

## VISUAL LOCATION AWARENESS FOR MOBILE ROBOTS USING FEATURE-BASED VISION

This thesis presents an evaluation of feature-based visual recognition paradigm for the task of mobile robot localization. Although many works describe feature-based visual robot localization, they often do so using complex methods for map-building and position estimation which obscure the underlying vision systems' performance. One of the main contributions of this work is the development of an evaluation algorithm employing simple models for location awareness with focus on evaluating the underlying vision system. While SeeAsYou is used as a prototypical vision system for evaluation, the algorithm is designed to allow it to be used with other feature-based vision systems as well. The main result is that feature-based recognition with SeeAsYou provides some information but is not strong enough to reliably achieve location awareness without the temporal context. Adding a simple temporal model, however, suggests a more reliable localization performance.

Alexander Kazeka
Department of Computer Science
Colorado State University
Fort Collins, CO 80523
Spring 2010

# ACKNOWLEDGEMENTS

The development of this thesis would be impossible without the help and guidance of Dr. Bruce A. Draper. The conversations with Dr. Charles W. Anderson and Dr. Anthony A. Maciejewski provided an invaluable source of inspiration and direction for this work.

# DEDICATION

To my family…

## TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# Chapter 1

# Introduction

Location awareness is the result of successful self-localization. In its basic form, self-localization can be formulated as answering the question “where am I?” Successful robot localization accurately establishes the current position of a robot relative to a framework such as a map (“I am in region $x$ of the map”) or previous experience (“the last time I was here was associated with $t$”). Localization is an important ability for mobile robots and supports navigation and other location-dependent activities: knowing where a robot is can provide it with the direction to its destination or with a sense of some other action appropriate at its current location.

The proliferation of mobile robots, seen by many as the next stage in robotics technology, makes solving localization problems in real-life settings very important. Due to its importance, robotic localization is a long-studied problem, and many different localization algorithms have been proposed. Nonetheless a number of open questions remains, including the best means of localization in dynamic environments and efficient representation of location-specific information [Thr02]. These are, in part, the problems addressed by this work.

Fox et al. [FBT99] and Dellaert et al. [DFBT99] describe two related systems evaluated as museum tour guide robots, one in the Deutsches Museum Bonn and the other in the Smithsonian National Museum of American History. In the course of the evaluations, the systems operated during normal working hours and in the presence of museum-goers. As such, they provide examples of real-life applications for mobile robots, and the museum tour guide analogy will be used in the following discussion to illustrate important points.

Traditionally, localization is cast as the problem of finding a robot's position relative to a static Cartesian map, but it is really the target application which determines the optimal framework for expressing a robot's position. In the case of a robotic museum tour, for example, it may be unnecessary to precisely localize the robot's position on a map, since a general position relative to a previous experience—for example confirming that the robot is next to a particular painting—can suffice. Therefore, the exact metric localization is not necessary in more general case.

The localization problem is often solved using multiple sensors, including sonar and laser range finders, GPS systems, and cameras [Thr02]. In contrast to multiple-sensor localization, visual localization relies on camera signals only. This approach is compelling for at least two reasons. First, cameras provide a wealth of information about the environment compared to other sensors. While sonar and laser range finders reveal some information about the environment and are useful for obstacle detection, they are limited in localization applications: since range finders only provide line-of-sight distances, they offer little information for recognizing the objects their signals are incident on. This raises problems in dynamic environments where objects move around and environment changes cause the maps to become outdated. Second, cameras are self-contained and don't rely on external components for successful sensing, as opposed to GPS receivers which rely on satellite signals. This makes cameras usable in environments where such auxiliary components are not available or have limitations (such as indoors).

Although cameras provide a wealth of information without requiring external signals, the best way to use them for robot localization is not clear. One of the obstacles to the ability of cameras to model their environments, reliable object recognition is known to be a difficult problem. This thesis explores the utility for localization of a relatively new approach to object recognition: attentional feature-based vision.

Many contemporary general-purpose computer vision systems employ unsupervised learning of attention-based local image features. The key contribution of this work is the evaluation of an attentional feature-based vision system *SeeAsYou* [Dra07] for the purpose of visual robot localization. Since SeeAsYou is a representative example of the class of unsupervised feature-based vision systems (discussed in the next chapter), the results obtained here can be compared to other feature-based vision systems: both utilizing attention-based local image features [SLL01b], [KBO$^+$05], [OH05], [SI07] as well as global image features [OT01], [SSHW07].

Another contribution of this work is the algorithm developed for evaluation of SeeAsYou—*location awareness through relevance and context* (LA-RC). Since SeeAsYou is a characteristic example from a wider class of feature-based vision systems, the algorithm can also be applied to other systems within this class. LA-RC builds on ideas from information retrieval and statistical modeling: a document retrieval metric *term frequency inverse document frequency* (TFIDF) is applied to establish the most relevant landmarks for each location, and this information is further used in conjunction with a *hidden Markov model* (HMM) to determine the most likely location of a robot given temporal context. Although both TFIDF and HMM are well-known and widely used, to the best of author's knowledge, their combined application to mobile robot localization has not been described before. In addition, LA-RC does not require explicit map construction which makes it easy to use. As will be shown, it presents an intuitive and effective solution to one of the classic robotics problems.

The rest of this thesis is structured as follows: Chapter 2 gives an overview of related approaches to visual robot localization; Chapter 3 describes the evaluation of SeeAsYou for robot localization using TFIDF, a relevance metric; location approximation with TFIDF and HMM is detailed in Chapter 4, combining the relevance metric with a temporal model; Chapter 5 summarizes the results and outlines the directions for future work.

# Chapter 2

# Background

## 2.1 Literature Review: Visual Robot Localization

Robot localization is a long- and well-studied problem [Thr02]. One way to approach the literature is to organize systems according to the framework within which the robot's position is established. Map-based techniques aim to determine location of a robot within a metric or topological map. Experience-based localization methods do not use maps, but solve the problem by matching current locations to previous experience: such as locations or actions.

### 2.1.1 Map-Based Localization

Map-based visual localization is by far the most prevalent approach and can be further subdivided into metric, topological, and hybrid map-based methods. Metric techniques represent the environment as a Cartesian map, with landmarks assigned precise coordinates. Algorithms that learn the map while localizing the robot are called *simultaneous localization and mapping* (SLAM) algorithms [RN03]. SLAM is a well-known problem formulation which entails building a map of the unknown environment and simultaneously estimating the robot's current position within that map. There are many SLAM systems; this review will focus on the ones most relevant to this thesis.

Se, Lowe, and Little provide a visual SLAM solution for unmodified indoor environments [SLL01b], [SLL01a], [SLL02], [SLL05]. They use the *scale invariant feature transform* (SIFT) [Low99] for visual attention-based landmark discovery and matching. SIFT is a localized image feature (keypoint) detection technique which resembles visual attention in primates. To determine such attention-based keypoints in an image, the SIFT operator applies *difference of Gaussians* (DoG) filters at different scales. The DoG filter is defined by:

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) \star I(x,y), \tag{2.1}$$

where $x$ and $y$ are image coordinates, $\sigma$ is standard deviation of the Gaussian, $G$ is the Gaussian function, $k$ is the scale parameter, $I$ is the image, and $\star$ is the convolution operation. The scale parameter $k$ has been shown to have little impact on the results, but for implementation purposes is typically chosen to be $\sqrt{2}$ [Low99]. The pixels in an image which produce the largest DoG responses in their neighborhoods at their respective scales become SIFT keypoints.

Se et al. [SLL01b], [SLL01a], [SLL02], [SLL05] estimate robot movement (egomotion) using least-squares fitting given landmark matching data and odometry. Kalman filtering is used to track the landmarks across frames and assign Cartesian coordinates as the robot moves around to build a 3D landmark map. In particular, Kalman filtering is used to predict which landmarks should be visible given the current estimate of position; the update step involves integrating new images and odometry measurements into the belief state. The authors rely on their stereo camera's intrinsic parameters to position SIFT features in the 3D coordinate system. The map is stored in a database as a set of landmark records. Each landmark record consists of the robot position at the time the landmark was encountered (in robot coordinates), a SIFT feature vector which includes landmark scale and orientation, and the number of consecutive frames the landmark was missed—a parameter aimed to estimate landmark reliability.

In their more recent work Se et al. compare methods for matching locations in a *kidnapping scenario* [SLL05]. In a kidnapping scenario, the robot first learns a representation of an environment. The robot is then "kidnapped" and deployed to an unknown location within that environment. The robot has to figure out which previously visited location matches its current position. The correct solution to the kidnapping problem is the best possible match to a known location. In [SLL05], the authors compare Hough transform on sets of SIFT landmarks and RANSAC (RANdom SAmple Consensus) for determining the robot pose supported by the most landmarks. Overall, the results converge in both methods and the system runs in real time (at 2 Hz). The authors also describe an optimization scheme which splits the map into sub-regions and uses Newton's method to pairwise align sub-regions when a location is re-visited. The test, however, is conducted in a very small 10 by 10 meter lab. Also, as with many SLAM approaches, a static environment is assumed.

The SLAM algorithm of Se et al. has been adopted with small modifications related to thresholding and filtering methods in a commercial application—vSLAM by Evolution Robotics [KBO$^+$05]. The key differences between vSLAM and the work of Se et al. is the application of a pre-filter which sets thresholds for landmark matching (RMS error, number of SIFT keypoints matched, robot slope) and using a particle filter for robot pose update while Kalman filtering is used for landmark pose updates (justified by a simpler dynamics of landmark poses).

Another attention-based visual SLAM process is described by Andreasson et al. [ATD07]. Authors use a modified SIFT algorithm that, in contrast to the original SIFT version by Lowe [Low99], doesn't maintain scale invariance, which is aimed to reduce matches among distant locations. The authors use particle filtering to integrate odometry measurements with matches from a pre-built database of robot positions. The database is built on correlating images with laser scans within a Cartesian map framework: each location is associated with a modified SIFT-based feature vector. Laser scan SLAM map is built on the basis of an earlier work by Frese et al. [FLD05]. The approach achieves localization accuracy to within 2 meters in at least 67% of trials when matched to laser scan data in a 60 by 55 meter indoor area.

An interesting application of attention-based vision to SLAM is shown by Frintrop et al. [FJC08]. VOCUS SLAM combines local feature detection with top-down feedback for active camera tracking. Image features are computed with center-surround filters—technique equivalent to DoG used in SIFT. The top-down behaviors are induced by the weights of a trigonometric function controlling active camera tracking. This system is programmed to exhibit three behavioral scenarios: redetection, tracking, and exploration. Depending on the built-in decision tree, the appropriate behavior is selected which determines the settings for active camera control function. Authors show improved performance of active camera tracking versus passive.

In general, Cartesian localization often suffers from map degradation in changing environments. Compared to SLAM, topological map-based methods overcome some of its limitations by avoiding the assignment of precise coordinates to landmarks. Instead, environments are represented as graphs of adjacent locations, where each location is defined as a set of landmarks. Therefore, when landmarks change their positions the topological map is often still valid and does not need to be recomputed as in SLAM.

An example of the topological map-based approach is the work of Ullrich and Nourbakhsh [UN00]. The authors manually construct a topological map (“an adjacency graph of different locations”) in which each location is assigned a set of representative images during training. Authors use an omnidirectional camera to capture 360° images and then extract hue, luminance, saturation, and RGB histograms. Nearest-neighbor learning trains a classifier for each set of the histograms. A voting scheme is then employed to match locations. The authors compute a confidence measure to help eliminate poor matches. Their experiments take place in 4 settings: a large eight room apartment, two indoor routes, and an outdoor campus route. The localization is approximately 90% correct with no confident bad matches over paths between 131 and 231 images long and tested individually to localize to among 8 to 10 segments. This system also achieves a real-time performance rate of 2 Hz.

Ourehani et al. present another topological robot localization method [OH05], [OHB05]. They use an augmented DoG process (similar to the SIFT features in [SLL01b], [SLL01a], [SLL02], [SLL05], [KBO$^+$05]) to extract attention-based local image features which serve as landmarks in a topological map of the environment. A voting scheme [OH05] or Markov localization framework [FBT99], [OHB05] are then applied to achieve localization. More specifically, a set of four visual cues is extracted from a scene: intensity, red/green, blue/yellow, and corner/edge filter responses. Each cue is then convolved with a DoG filter to produce the so-called conspicuity maps. The third step computes saliency maps such that conspicuity maps with a small number of peaks are promoted while the ones with many low responses are demoted. The last step in feature extraction is peak detection in saliency maps. The topological map is then constructed by assigning top features detected over a significant number of consecutive frames to the corresponding equally spaced path segments. The map contains statistical measurements of the features within each path segment. Robot localization is accomplished using an original matching procedure developed by the authors which produces match scores across possible locations. The match scores are then combined using a voting procedure [OH05] or Markov localization framework [FBT99], [OHB05] to establish the most probable path segment to which the robot is localized. The authors conclude both papers with the descriptions of successful experiments along relatively short paths of approximately 10 meters to support their claims.

Torralba, Murphy, Freeman, and Rubin propose another method of topological map-based visual localization [TMFR03]. The goals of Torralba et al. are (1) to develop an approach to recognize familiar, previously visited locations, (2) break these locations down into broader categories, possibly using that knowledge to classify and learn previously unseen locations, and (3) recognize specific objects present in different locations. To accomplish the first goal, which is the most relevant to this work, they approach the problem by first constructing sets of local and global image features based on textural properties obtained using wavelet decomposition: local features are combined into jets, one for each of 24 sub-bands of an image, while global features are combined into image pyramids with 6 orientations and 4 scales. Then, the features are projected onto 80 principal components to reduce dimensionality. Place recognition is accomplished using a *hidden Markov model* (HMM) to recursively compute the probability of being in a specific location. The transition matrix is obtained by manually counting the transitions between different locations along the path (a uniform Dirichlet prior is added to account for transitions that were not seen in training data). The observation likelihood matrix is a Gaussian mixture, which is experimentally established using cross-validation. The results for known place recognition among 63 possibilities are approximately 85% area under precision-recall curve. To accomplish the other two goals, the authors train a separate HMM on manually added category labels to classify location types and use a Bayesian framework with priming from localization and location type classification to recognize specific objects in scenes.

Hybrid map-based techniques essentially combine metric and topological maps. For example, Siagian and Itti take a biologically inspired approach which employs complimentary saliency (SIFT and auxiliary saliency feature vector generated from the neighborhood of SIFT keypoints) and gist (color, intensity, and orientation metrics generated over whole image) features [SI07], [SI08]. During training, a graph-based topological map augmented with Cartesian coordinates is supplied to the system. In this hybrid map locations are associated with corresponding visual information. Three distinct sites are modeled in this way, with each site being further split into 9 segments. Up to five salient regions are extracted from a frame of video, along with a 4 by 4 grid of the more global gist features. Then, a neural network classifier is trained for segment recognition based solely on gist features. During testing, the neural network is applied to determine which path segment the testing images come from. Then, Monte-Carlo localization is employed for robot position estimation within segments using both gist and, when available, saliency features (a saliency feature may not be available for every image because none were detected or there was no match). The authors successfully localize the robot to within 9.75 meters in large outdoor environments up to 137 by 178 meters, however the error increases with the size of the environment.

In their second paper [SI08], the authors develop an optimization scheme for the landmark database to speed up localization. They propose an experimentally derived cascade of thresholds. First, the similar SIFT keypoints are combined along with their associated saliency feature vectors (a 5 by 5 window centered on the keypoint which includes color, intensity, and orientation histograms) based on a saliency feature score. Another metric is employed to prune weak landmarks. The landmarks are then combined across different training episodes using yet another metric, and a fourth metric is used to prioritize landmarks to speed up search.

A major contribution of Siagian and Itti is the development of a robot platform which extracts gist and saliency in parallel on a 16 core 2.6 GHz machine (each sub-channel has its own thread), which the authors claim is similar to work done in dorsal and ventral visual pathways in the human brain. Operating on images of size 160 by 120, the authors report 50 milliseconds/frame saliency and gist computation time, 1 millisecond segment estimation time, and 2 second landmark search time.

### 2.1.2 Experience-Based Localization

The main drawback of topological map-based localization methods, however, is that they require a significant degree of human participation in creating the maps. Experience-based localization provides an interesting solution to the map generation and maintenance problem characteristic of map-based methods: a robot recognizes previously seen locations without any metric or topological map-building. Giovannangeli, Gaussier, and Desilles present such an approach in [GGD06]. In order to detect the feature points that define a landmark, the authors convolve the gradient image obtained from the camera with a DoG filter, search for extrema, then log-transform the image regions around extrema points to achieve invariance to small changes in rotation and scale. An artificial neuron is then recruited to encode each new landmark using an original activation function developed by the authors; it is trained during one-shot learning to distinguish the landmark from all other landmarks by producing the maximal response to it. The authors call all such neurons landmark neurons. Additionally two other groups of neurons are trained (also using one-shot learning with corresponding activation functions) to produce different levels of activity for different azimuths and elevations of the feature points defining the landmarks. Then, the responses from the three groups of neurons (landmark-azimuth-elevation) are combined (per landmark) into a third-order tensor concisely defining a landmark, the response to which is learned (also using one-shot) by neurons in the merging layer defining place codes. Finally the activity in place cells (another group of artificial neurons) results from the computation of the distance between the learned place codes and current place codes. This setup has some desirable properties: place cells respond continuously to the area around the learned location creating so-called place fields, and learning of new locations can be triggered by low place cell responses, causing the size of representation to grow according to visual, not geometric, properties of the locations. Additionally, the authors develop a method for navigation by associating each place cell with a movement. By implementing soft competition and short term memory, the authors achieve robustness in both place recognition and navigation.

Another research group which developed a map-less approach is Schubert, Spexard, Hanheide, and Wachsmuth. In their 2007 paper [SSHW07], the authors applied an approach similar to one proposed by Oliva and Torralba to represent holistic perceptual gists of scenes [OT01]. Schubert et al. compute 12 different neighborhood filter responses (11 edge filters with different orientations, 1 corner filter) as well as image intensity from 46 differently sized regions of the image. For each patch, they also compute its second (energy) and fourth (kurtosis) statistical moments. The authors then use AdaBoost [SSHW07] to combine weak binary classifiers for each feature. In addition, a rejection scheme and a method to handle potential occlusions by panning the camera or moving the robot are implemented. The results are presented within the framework on a home-tour scenario: the robot is manually shown by panning the camera sets of 90°–120° views from various points in each of the 4 rooms (living room, hallway, dinner room, kitchen). After the classifiers are trained on these views, testing is accomplished by matching similar views collected from different points in corresponding rooms (similarly to kidnapping scenario described above). Two test sets are performed: one with the same furniture arrangement and one with the furniture moved, which also creates occlusion of certain landmarks. Overall the results are good for views with no or little occlusion, but the more difficult locations are correctly classified only at the 75% level.

The methods described above exhibit a wide array of approaches to vision-based localization. Nevertheless, most are based on hand-coded or pre-defined environment maps and system parameters susceptible to environment changes and obfuscating the capabilities of the underlying vision systems with respect to robot self-localization. The contribution of this work is the evaluation of SeeAsYou as a prototypical attentional feature-based vision system for the task of robot localization using intuitive and well-studied models which allow for a closer look at the underlying system's performance. More specifically, the evaluation algorithm LA-RC, compared to [UN00], [SI07], [SI08], [TMFR03], does not require explicit map construction and labeling, nor knowledge of intrinsic camera parameters as in [SLL01b], [SLL01a], [SLL02], [SLL05], nor input from odometry sensors as in [OHB05], [FBT99]. Compared to [OH05], [OHB05], [SI07], [SI08], [ATD07], the presented approach uses a well-known information retrieval metric for landmark importance estimation, which in turn can lead to a more straightforward analysis and optimization of representation for large environments. In contrast to the work by Se et al. [SLL01b], [SLL01a], [SLL02], [SLL05], the proposed evaluation algorithm is not limited to static indoor environments.

## 2.2 SeeAsYou

A distinct quality of *SeeAsYou* is that it implements ideas on regional-functional anatomy of the human vision system: more precisely, it is a biologically-inspired model of *Reverse Hierarchy Theory* (RHT) [HA02]. RHT suggests that human vision is a two-pass process. The first pass extracts the broad categories of images—broad notions of objects present in the view without the specifics of what exactly the objects are or where they are located in an image. For example, after the first pass a person would be able to tell that an image contains an animal, but wouldn't know the exact kind of animal they saw or what its position in the image was [HA02]. This first pass is largely subconscious, we have no control over it. The second pass, on the other hand, is cognition driven and is aimed to corroborate or refute the hypotheses about the exact nature of the objects and their precise locations. At this stage object features are merged into a cohesive whole and final object recognition takes place.

SeeAsYou implements the first pass of RHT to obtain the broad categorization of an

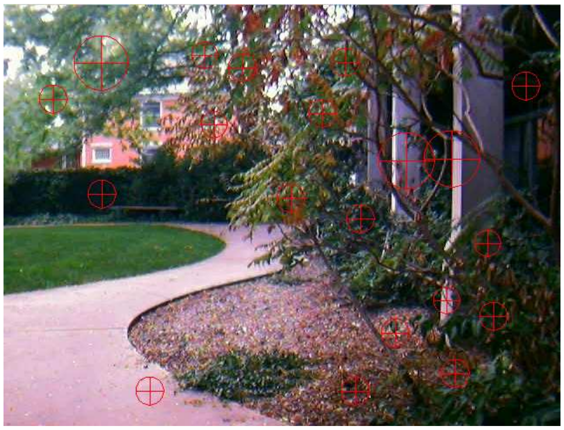

Figure 2.1: Sample image with attention-based keypoints. Keypoints are shown as red circles at corresponding scales.

image. This categorization is represented by image features around particular parts of the image. The parts that describe the image are chosen, similarly to some of the systems described above, using the SIFT operator [Low99]. SIFT has well known parallels to responses in certain neurons in inferior temporal cortex in primates: both respond to regions in an image that are largely invariant to changes in scale, location, and illumination. Additionally, SIFT is a well-known attention-based keypoint detection technique, as detailed in the previous section.

For the purposes of this work, 20 keypoints with highest responses to DoG filters were selected from each image (see Figure 2.1).

Unlike other systems, in SeeAsYou each SIFT keypoint is passed through three separate channels which generate color, edge, and texture histograms and in each of these three channels an unsupervised clustering algorithm is applied to generate hierarchical categories of features. Therefore color, edge, and texture histograms around SIFT keypoints become the image descriptors used by the system to form the broad notion of an image. The rationale for clustering in each channel separately is the belief that there exists a similar separation of these processes in the brain [Dra07]. The clustering algorithm applied is based on the neuro-anatomy of thalamocortical circuits proposed by

Granger et al. [RWG04]. In the end, the resulting hierarchical cluster labels for image descriptors form the notion of an image and can be compared for the purposes of object recognition or image matching. In SeeAsYou, *term frequency inverse document frequency* (TFIDF) (detailed in the next chapter) is applied to compare cluster labels of consecutive images and recognize co-occurrences. The proposed evaluation algorithm also uses TFIDF, albeit at varied levels of granularity. Although TFIDF weighting intuitively corresponds to selecting relevant features for comparison, it has no immediate biological correlates. It is important to note, that while SeeAsYou is a very sophisticated vision system, it is not the only option for input to the evaluation algorithm—any vision system which produces matching sets of image features would suffice.

The next chapter describes the evaluation of information retrieval capabilities of SeeAsYou for location matching and formulates *location awareness through relevance* (LA-R) algorithm. Chapter 4 builds on LA-R and adds temporal context with a *hidden Markov model* (HMM) completing the description of *location awareness through relevance and context* (LA-RC) and concluding the evaluation of SeeAsYou.

# Chapter 3

# Location Awareness through Relevance (LA-R)

Fundamentally, a robotic vision system is responsible for the retrieval of information. For many tasks, a useful perspective to view the vision system from is information retrieval. Many image matching implementations are based on this idea. It is also logical to apply it to match locations for robot localization purposes. The purpose of this chapter is to present the evaluation of SeeAsYou as a prototypical feature-based vision system for the task of localization purely in terms of its information retrieval capabilities. This concept is encapsulated in *location awareness through relevance* (LA-R) algorithm.

LA-R solves a well-known experimental scenario. In this scenario a robot is taken to a number of training locations. At each location it captures a set of images, for instance each set forming a 360° panorama. The robot is then taken to a new location where it also collects images. The robot then has to match this new testing location to the most similar training location.

In robotics literature this experiment is known as the *kidnapping* scenario [FBT99] because it represents a situation when a robot is picked up (or "kidnapped") during its operation and placed in a new position. The robot then has to figure out where it has been placed. This experimental scenario is also applicable during the initial startup of the robot. At the same time, in terms of computer vision, the robot can be abstracted

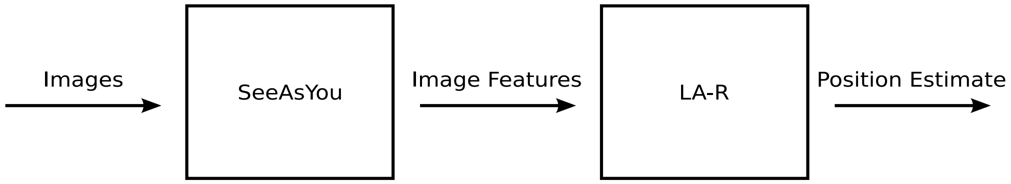

Figure 3.1: LA-R system diagram. System components are shown by square blocks and components' inputs and outputs are shown by arrows.

away and the experiment then becomes one of *forced-choice matching*: given a gallery of image panoramas and a novel set of images, find the most similar gallery panorama.

For the purposes of this chapter, the localization task is defined solely in terms of sets of images; other sensors are not allowed. In addition, no form of reasoning about temporal continuity is allowed (no ordering of images). The latter is an artificial constraint which will be removed in Chapter 4.

To sum up, the goal of this chapter is to use LA-R algorithm to measure how well a specific, feature-based vision system SeeAsYou [Dra07] performs at robot localization based only on its information retrieval abilities. SeeAsYou was designed for object recognition and image matching, not robot localization, and is an example of current trends in computer vision (as discussed in Chapter 2). Specifically, this system requires no a priori knowledge about images or supervised training. Additionally, it does not need to be modified for the localization task with LA-R (see Figure 3.1), the only necessary constraint is that when supplied with input images it outputs sets of matching image feature labels.

## 3.1 Algorithm Description

LA-R solves the problem of robot localization in the context of the kidnapping (or forced-choice matching) scenario. To evaluate the information retrieval capabilities of SeeAsYou as prototypical feature-based vision system for the task of robot self-localization, a well-known information retrieval metric is applied—*term frequency in-*

*verse document frequency* (TFIDF). It uses the hierarchical cluster labels of image descriptors produced by SeeAsYou as *features* and compares them across images taken from training and testing locations. Location in this case can be defined as a particular place in the environment from which one or more images were taken. Although SeeAsYou represents images as labels of hierarchical clusters of image descriptors, to LA-R they are abstract sets of image features. This enables the algorithm to be used with an arbitrary feature-based vision system in which case the features used by LA-R are the ones produced by the underlying vision system.

To accomplish the localization task, the approach is to compare image features based on their relevance to each location. This approach is motivated by the fact that the underlying vision system has no notion about which image features may be most relevant to distinguishing different locations. In other words, image features generated by the vision system may correspond to both very common objects found in every location and very peculiar objects found in only specific locations. The common objects are less relevant to the localization problem than the peculiar objects. For example, if door knobs are the same in every room then finding one in an image provides little information about which particular room the image came from. On the other hand, recognizing a painting that is found in only one specific room will provide a cue for accurate localization.

This approach has a parallel to how a human may attempt to solve this problem. Instead of trying to figure out which location each object they see is likely to belong to, one may look for unique objects that are characteristic to their respective locations. Such characteristic objects are more relevant to the task of self-localization.

The concept of relevance arises naturally in the area of document retrieval when documents are searched for specific terms. TFIDF is a well-known metric that is often used to rank the relevance of documents in a collection to a specific query term [BYRN99]. By analogy with document retrieval, TFIDF can be applied in robot localization tasks to rank the relevance of locations to specific query images. In this case image data from all known locations is analogous to a document collection, images from a specific location are analogous to a document, and corresponding image features are analogous to terms that appear in documents. In such way the TFIDF metric can be used to assign weights to image features such that features which appear in too many locations are discounted compared to rarer features, which presumably have more precise, relevant meanings. Formally, term frequency is computed as:

$$TF_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}}, \tag{3.1}$$

where $i$ and $l$ are the feature indices and $j$ is the location index; $freq_{i,j}$ is the number of times feature $i$ appears in location $j$. The inverse document frequency is computed according to:

$$IDF_i = log \frac{N}{n_i}, \tag{3.2}$$

where $i$ is the feature index, $N$ is the number of known locations, and $n_i$ is the number of locations where feature $i$ appears. The resulting image feature TFIDF weight is then:

$$w_{i,j} = TF_{i,j}\, IDF_i. \tag{3.3}$$

When a test image relevance to the known locations is evaluated, the TFIDF-weighted sum (*TFIDF score*) of features from the test image which match features from the known locations is computed for each known location. The training location that corresponds to the highest TFIDF score is then selected as the localization match. More formally, if $F_q$ is an ordered vector containing counts of how many times each known feature was seen in query location $q$ (essentially a feature histogram) and $W_j$ is the corresponding feature TFIDF weight vector for known location $j$, the resulting LA-R match is the argument of the maximum of a dot product of these two vectors:

$$\underset{\mathbf{j}}{\arg\max}\; L_q W_j. \tag{3.4}$$

As such, the LA-R match selects the training location most *relevant* to the testing location based on image features present in both.

## 3.2 Experimental Methods

As mentioned in the beginning of this chapter, its goal is to use the LA-R algorithm to evaluate how well the SeeAsYou vision system can perform robotic self-localization solely in terms of its information retrieval capabilities. For this purpose the vision system was not modified in any way—the features it extracted from images were input directly into LA-R.

While the "kidnapping" experiment framework is well-known in both robotics and computer vision literature, the setup used to evaluate SeeAsYou was as follows: (1) the robot collects images from several different training locations, the images are then input into the vision system which outputs the features detected in each location; (2) the robot is then "kidnapped" and placed near one of the training locations where it collects additional testing images which are input into the vision system and the corresponding output features are stored; and (3) LA-R algorithm is used to match the testing location to one of the previously visited training locations.

More specifically, a total of 32 360° panoramas each containing 69 images were collected from eight different rooms in the University Services Building at Colorado State University. Data was collected at four different positions in each room. The images were gathered using a 640 by 480 pixel camera mounted on an Evolution Robotics ER1 robot (see Figure 3.2 for sample images). The panoramas were then randomly split into two sets (see Figure 3.3 for their respective positions) such that each set contained panoramas from two distinct and different positions from each room. Images from one set were used to represent the known locations while the images from the other set were used to represent the arbitrary areas the robot was “kidnapped” into. The roles of the sets were then reversed and the experiment repeated.

## 3.3 Experimental Results

Figure 3.4 shows the results of comparing a test panorama from the robot lab to the 16 training panoramas. The vertical axis shows the magnitudes of TFIDF scores. Each bar along the horizontal axis represents one training panorama. The first two training panoramas are from the robot lab, the other 14 are from other locations. Even though the test panorama was taken from a slightly different position, the highest TFIDF score corresponds to one of the training panoramas also taken in the robot lab.

Although Figure 3.4 demonstrates a correct match with LA-R, not every test panorama matches a training panorama in the same location. The top half of Figure 3.5 shows the matching scores between all 16 test panoramas and all 16 training panoramas (two from each location). Swapping the training and testing sets produces the data in the lower half of Figure 3.5. According to a winner-take-all selection, in 19 of 32 trials the closest match is to a training panorama from the same location (only 4 of 32 would have matched randomly). Alternatively, in 13 of 32 trials the closest match is a mismatch.

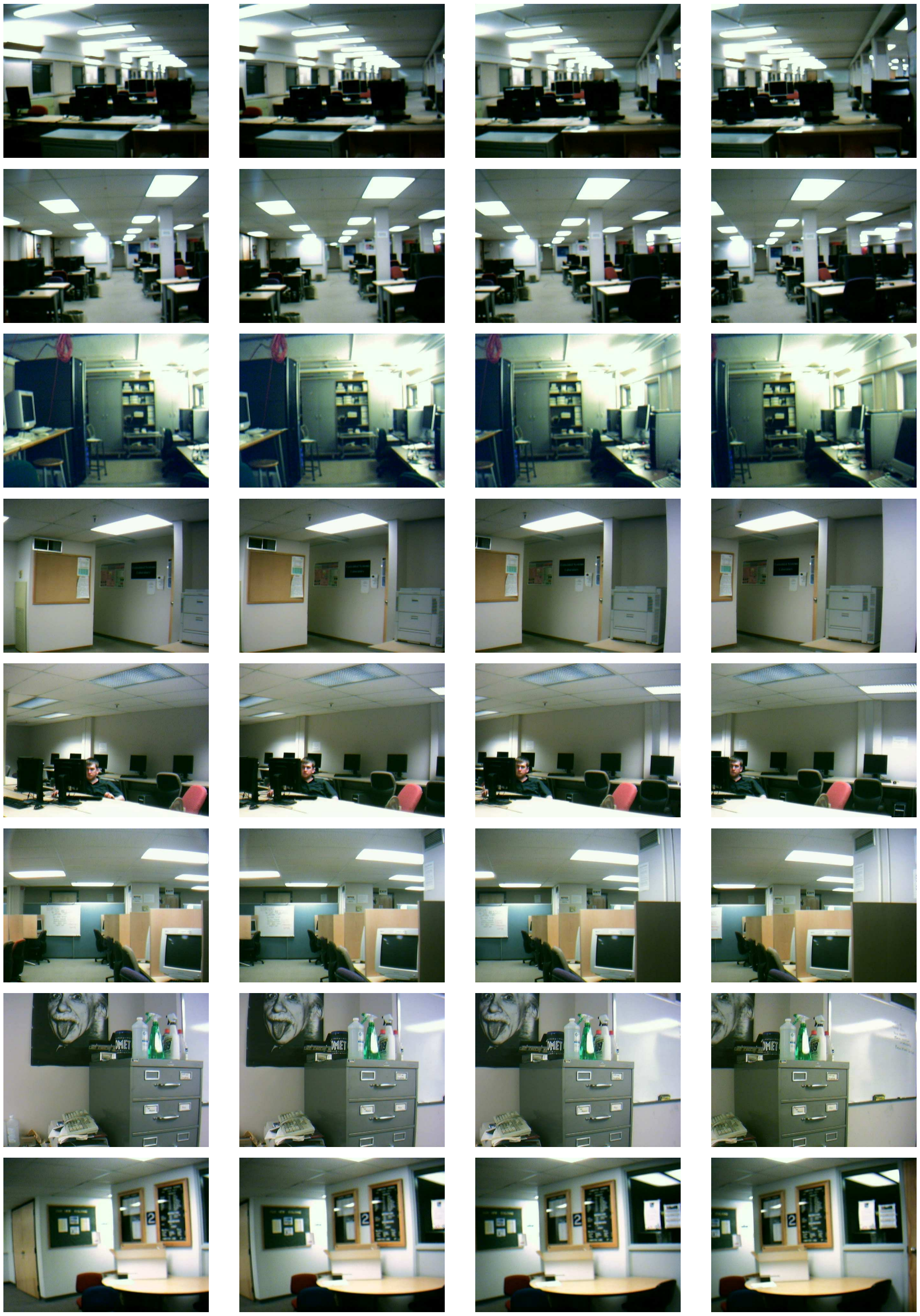

Figure 3.2: Sample images used in the experiments. From top to bottom row images correspond to the following locations: robot lab, north lab, machine room, south lab lobby, HP lab, south lab, systems office, and 2nd floor lobby.

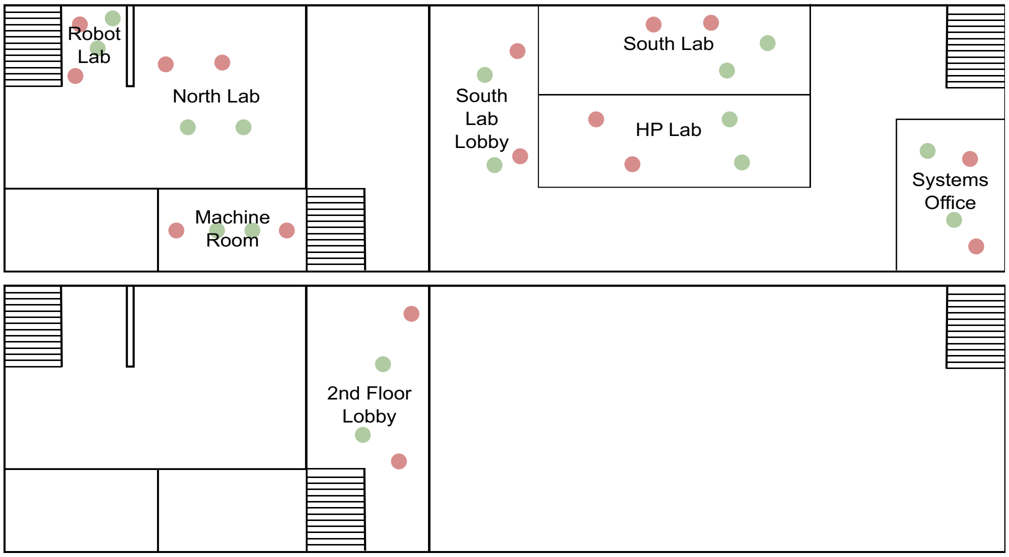

Figure 3.3: Floor plan of locations used in the experiments. Each 360° panorama is denoted by a circle. Training/testing sets of 360° panoramas are shown as circles of the same color respectively.

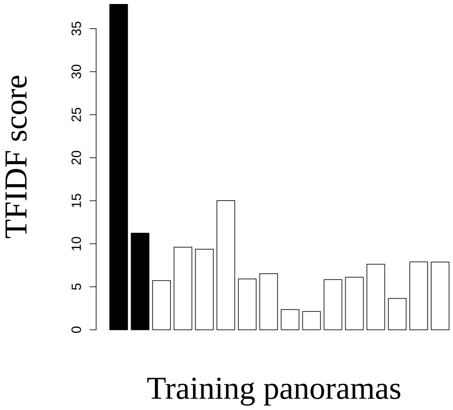

Figure 3.4: Evaluation of a single test 360° panorama against the 16 training panoramas. Vertical axis shows the magnitude of TFIDF scores. Horizontal axis shows TFIDF scores for each known (training) panorama in the database. Left to right the panorama scores correspond to the following locations (two panoramas for each location): robot lab, north lab, machine room, south lab lobby, HP lab, south lab, systems office, and 2nd floor lobby. Scores for training panoramas that are the correct match are shown in black.

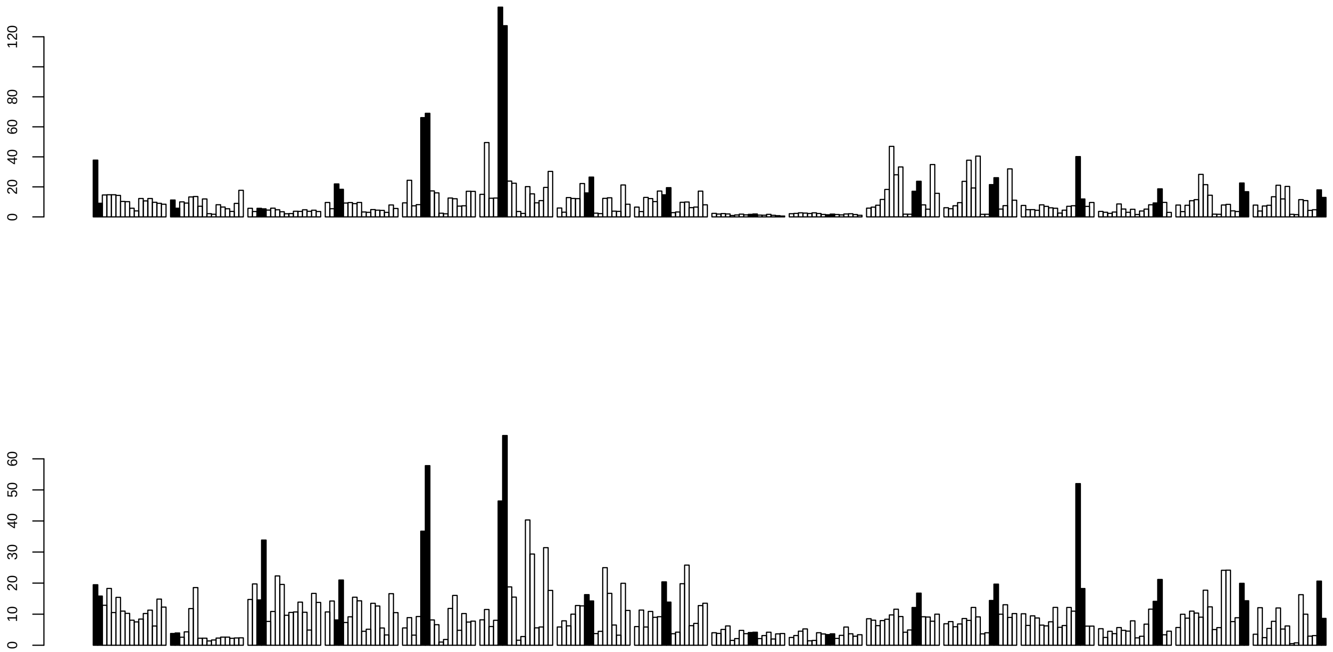

Figure 3.5: Evaluation of 32 test panoramas against 32 training panoramas. Top and bottom histogram sets show results for the two partitions of data. Each sub-histogram is organized as in 3.4. The sub-histograms (two for each testing location) are ordered left to right by their corresponding true matches in the same order as the bars representing TFIDF scores are organized within each sub-histogram (two for each training location): robot lab, north lab, machine room, south lab lobby, HP lab, south lab, systems office, and 2nd floor lobby.

Interestingly, the images collected from the HP lab have lower TFIDF scores in both partitions of data as indicated by the lower overall height of sub-histograms 9 and 10 in both the upper and lower parts of Figure 3.5. This may be attributed to the fact that the HP lab is visually not very distinct from the robot lab or the north lab (see Figure 3.2 rows 5 versus 1 and 2 for sample images from HP lab versus robot lab and north lab respectively). At the same time, the latter two locations possibly contain some distinctive features in addition to the ones common among the three. Therefore, the features found in the HP lab are not sufficient to distinguish it from the robot lab or the north lab but the converse is not the case. In a practical application the event in which some necessary to discern location achieves lower overall TFIDF scores can be used to indicate that additional images need to be collected from that location. This process may need to continue until a sufficient number of relevant landmarks has been discovered as indicated by the TFIDF score the location achieves with itself.

If LA-R results are evaluated not as a forced-choice scenario but rather as a sequence of individual yes/no decisions (binary classification of correct/incorrect localizations), then a look at performance at different acceptance thresholds with respect to TFIDF scores is useful. This idea is encapsulated by a receiver operating characteristic (ROC) curve, as shown in Figure 3.6. The ROC curve shows the true positive rate of the classifier (on a 0 to 1 scale) along the vertical axis and the false positive output along the horizontal axis. Figure 3.6 shows the total scores for a total of 256 (16 by 16) pairs of panoramas. In general, if the match scores are random, the ROC curve will look like a diagonal line; if the match scores are perfect, the ROC curve will touch the upper left corner of the plot.

A useful statistic of ROC representation is the area under the curve (AUC) metric. It shows the probability that a randomly drawn positive sample will be rated higher than a randomly chosen negative sample. For the first partition of data (graph on the left), the

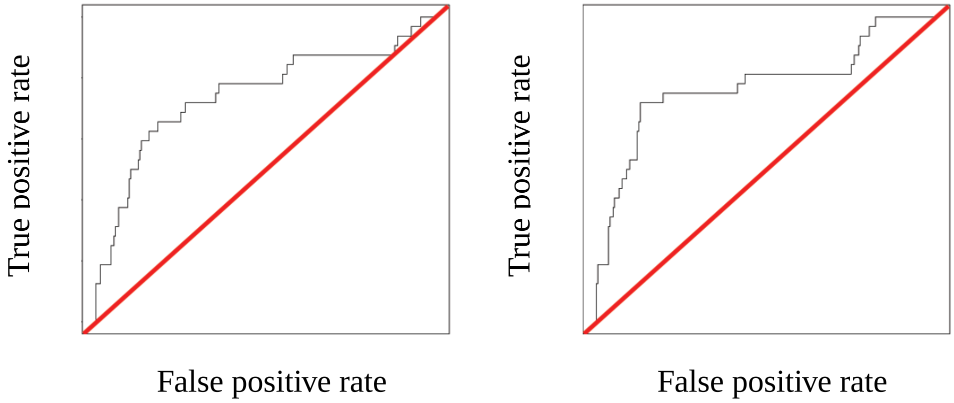

Figure 3.6: ROC curves for each of the two partitions of data. True positive output of the classifier is shown along the vertical axis and the false positive output is shown along the horizontal axis. Random classifier performance is shown by red diagonal lines.

AUC is 74.8%; for the second partition (graph on the right), the AUC is 77.8%.

A way to get a closer look at classification performance of LA-R is shown in Figure 3.7. The figure shows relative magnitudes of TFIDF scores for each training/testing panorama pair in the experiment. The vertical axis shows TFIDF scores for training panoramas as grayscale levels. The largest TFIDF score corresponds to the darkest grayscale level. The horizontal axis shows data for different tests. Correct matches for tests are highlighted in red.

Although in general darker grayscale levels fall within the correct regions, the results clearly show room for improvement. This can be achieved by keeping track of temporal context for localization. One of the constraints imposed in the above kidnapping experiment was that no form of reasoning about temporal continuity was allowed. The next chapter removes this constraint and evaluates SeeAsYou using both image feature relevance and temporal context.

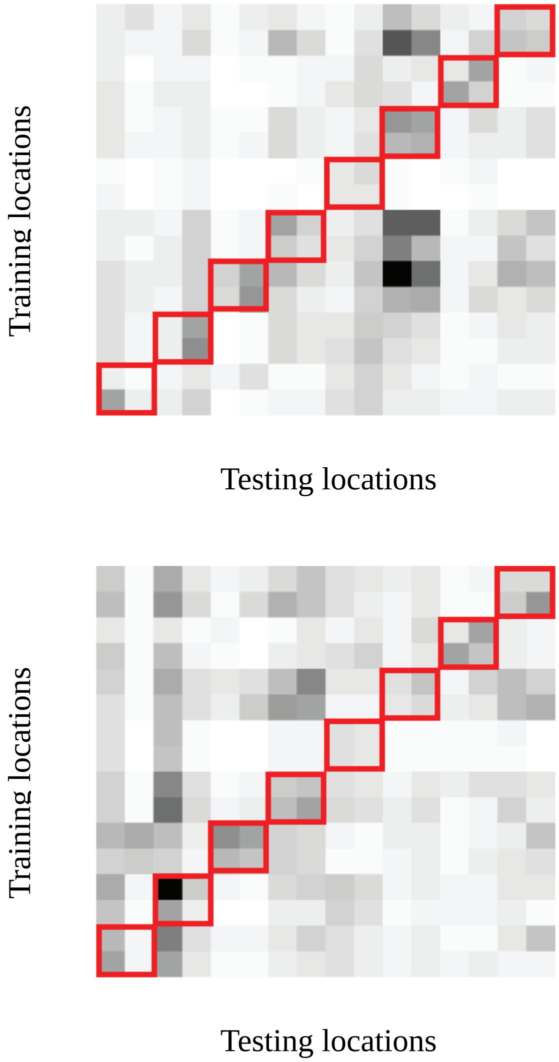

Figure 3.7: Relative magnitudes of TFIDF scores for each of the training/testing panorama pairs. Darker grayscale levels in each column represent higher TFIDF scores for a given training panorama during testing. Horizontal axis represent data for different tests. The order of test sets left to right (testing data) and bottom to top (training data) is the same as in previous figures: robot lab, north lab, machine room, south lab lobby, HP lab, south lab, systems office, and 2nd floor lobby. Each location is represented by two training and two testing panoramas. Correct matches are highlighted in red.

# Chapter 4

# Location Awareness through Relevance and Context (LA-RC)

The omission of temporal context for robot localization is artificial. Indeed, in most real-world applications of mobile robots this information is readily available: at the very least images can be ordered by their acquisition times to provide temporal ordering. Using such localization context in addition to relevance-based location awareness as provided by LA-R forms the crux of LA-RC. In this chapter LA-RC is applied to image features produced by SeeAsYou to further investigate the vision system's localization capabilities, now using both feature relevance and temporal context.

LA-RC solves the problem of robot localization along a path. The algorithm is best understood within the framework of the following experimental scenario: a robot is driven along a path while collecting training images and their temporal ordering; the robot is then re-deployed to a testing location somewhere along the path and starts driving along collecting new images and recording their temporal ordering; as the new images arrive the robot has to figure out where it is located along the path.

Similarly to the approach described in the previous chapter, no input from sensors other than the camera is allowed. This time, however, the localization context is recorded and provided to LA-RC in the form of images ordered by time. The overall system layout is the same as before with the exception that the location awareness module is

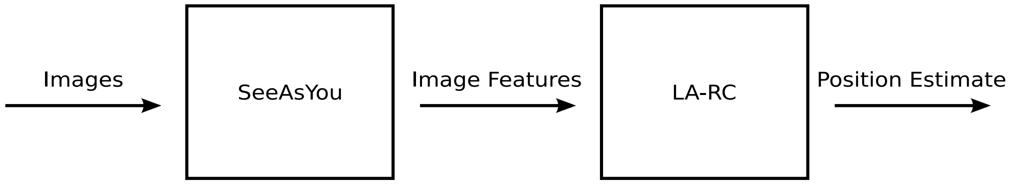

Figure 4.1: LA-RC system diagram. System components are shown by square blocks and components' inputs and outputs are shown by arrows.

now LA-RC instead of LA-R (see Figure 4.1).

As before with LA-R, LA-RC is not confined to SeeAsYou as the underlying vision system—any feature-based vision system can be used. Nevertheless, the goal of this chapter is to evaluate SeeAsYou with LA-RC for the task of localization along a path using both feature relevance and temporal context.

## 4.1 Algorithm Description

LA-RC solves the problem of robot localization along a path by matching features based on their relevance to each path segment and integrating the temporal context of localization. This approach is motivated by the fact that temporal information is readily available in many situations and has the potential to improve on a purely information retrieval based approach. This also has a parallel to how a human may go about solving the problem of localization along a path: instead of trying to match path segments independently, one may consider them in their temporal context (using short-term memory). More specifically, knowing what landmarks a person has recently seen and what was the order the landmarks were seen in can narrow down the possibilities of where the person is currently.

In order to achieve a similar capability, it is necessary for an algorithm to be able to combine previous and current position estimates using their temporal ordering. *Hidden Markov models* (HMMs) provide a simple and natural way to recursively combine previous and current robot observations—in this case relevance-based position estimates:

the TFIDF scores computed similarly to LA-R. The main difference is that in LA-RC TFIDF scores are computed for each image in the training sequence as opposed to one TFIDF score for each training 360° panorama in LA-R. Therefore, in LA-RC localization resolution is to a training image compared to a 360° view in LA-R. This, however, is a parameter that can be adjusted as necessary.

Using HMM formulation, as new position estimates become available the probability distribution across the set of possible locations can be recursively updated as:

$$P(L|O_t) = \alpha \, O_t \, \mathbf{O} \, \mathbf{T} \, P(L|O_{t-1}) \tag{4.1}$$

where $L$ denotes the set of possible locations (here individual training images along a path), $O_t$ is the position estimate at time $t$ which for the initial observation $O_0$ is uniform, $\alpha$ is the normalizing constant to keep probabilities sum up to 1, $\mathbf{O}$ is the observation probability matrix, and $\mathbf{T}$ is the location transition probability matrix. This follows the notation by Russell and Norvig [RN03].

$\mathbf{O}$ and $\mathbf{T}$ define the HMM because they determine how the model evolves over time. Although it is possible to learn both $\mathbf{O}$ and $\mathbf{T}$ using EM (Expectation Maximization) algorithm [Bis06], the current implementation of LA-RC uses pre-set observation and location transition probability matrices. $\mathbf{O}$ is a tridiagonal matrix such that the elements on the main diagonal are set to 0.5 and all other non-zero elements are equal to 0.25 which implies that similar observations come from adjacent locations. $\mathbf{T}$ is designed so that transitions are possible between the state and itself (corresponding to the robot remaining in the same position) with 25% probability, the state and the next state (corresponding to a robot moving ahead one position) with 50% probability, and between the state and the state two positions ahead (corresponding to a robot missing an observation) with 25% probability. The reverse state transitions are not allowed. Although having the location transition matrix configured in such way implies a simple forward location transition model, for the purposes of localization along a path this may be acceptable because during traversal all transitions are gradual and unidirectional (aimed toward destination).

Historically, the idea of integrating the temporal localization context is not novel. This is the approach taken in many SLAM solutions employing Kalman or particle filtering methods [Thr02]. However, path integration in LA-RC relies on HMM for that purpose, which is a simpler model than filtering and doesn't require odometry measurements. Torralba et al. describe a recursive HMM-based place recognition approach which employs a Gaussian mixture model for the observation probability matrix and involves manual construction of the state transition matrix [TMFR03]. As opposed to their method, LA-RC doesn't require manual labeling of transitions among locations.

## 4.2 Experimental Methods

To evaluate SeeAsYou with LA-RC, image sequences were collected in three different settings using the Evolution Robotics ER1 robot equipped with a 640 by 480 pixel camera (same as in the kidnapping experiment in the previous chapter). Out of the three settings, two were inside the CSU University Services Building in computer labs and faculty offices areas, and one was outside in the garden area.

Three image sequences were collected in each of the three settings by manually driving the robot three times along approximately identical routes (see Figure 4.2 for sample trajectories). The routes driven were between 30 and 50 meters long. Due to routes' varying lengths and variations in robot speed, the number of images per sequence ranged between 460 and 1,833: for computer lab setting the sequences were between 1733 and 1833 images long, for faculty offices—between 460 and 474, and for the garden area—between 827 and 879. For each of the settings, one image sequence was used to train hierarchical clusters of image descriptors in SeeAsYou, while the second

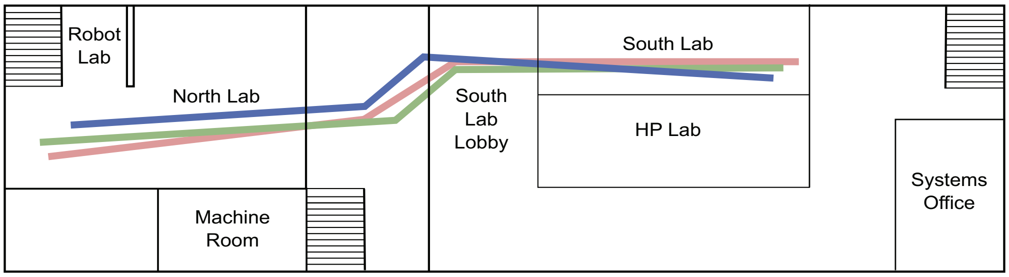

Figure 4.2: Sample routes in one setting (computer labs). Each route is denoted by different color. The image sequences follow similar but not exactly the same trajectories.

sequence was used to generate TFIDF scores for every image along the route. Parts of the third image sequence were used for testing. Since TFIDF scores were generated for every image from the second sequence, each such image represented a distinct location in a setting.

After accuracy of the algorithm was validated and localization with LA-RC and SeeAsYou evaluated within each setting individually, the training and testing sequences were combined across the settings to evaluate localization performance of SeeAsYou for all possible initial testing positions in the combined data set.

## 4.3 Experimental Results

To verify the accuracy of the algorithm, three testing sub-sequences were selected corresponding to arbitrary notable points in each setting. These notable locations were: the entrance to the computer lab in Figure 4.3, the cabinet with stickers in Figure 4.4, and the garden benches in Figure 4.5. For each setting, the goal was to find the training image that most closely matched the probe (testing) image.

The results of the accuracy verification experiment are illustrated in Figures 4.3–4.5. On the left each figure shows a set of histograms corresponding to discrete probability distributions maintained by the algorithm during the first four time steps of exe-

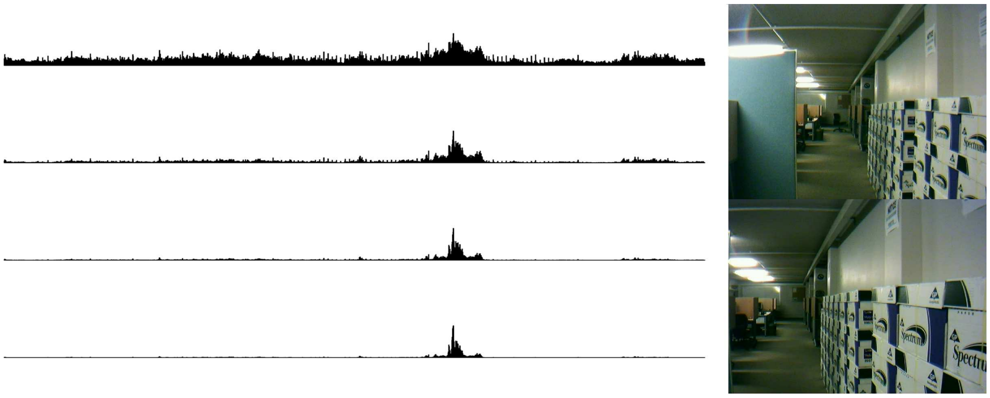

Figure 4.3: Sample convergence and the resulting match for an indoor run (computer lab setting). Histograms on the left demonstrate convergence starting with the first time step of the LA-RC execution at the top and three consecutive time steps below. On each histogram, the probabilities of being in a location are shown along the vertical axis while the horizontal axis shows the training locations (corresponding to individual images from the second image sequence). The histograms represent the discrete probability distributions maintained by LA-RC. The discrete probability distributions result from combining TFIDF scores for new image features (as observations) using HMM according to observation and transition matrices—HMM's $P(L|O_t)$ for $t = 1..4$ where $O_t$ are image features detected in image $t$. The time steps are advanced when features found in a new image are made available by the underlying vision system. The images corresponding to the peak in the bottom histogram are shown on the right: right top is the image from the testing sequence, right bottom is its match in the training sequence based on a winner-take-all selection.

cution (from top to bottom). The discrete probability distributions result from combining TFIDF scores for new and previously seen image features (as observations) using HMM according to observation and transition matrices described above—in other words HMM's $P(L|O_t)$ for $t = 1..4$ where $O_t$ are image features detected in image $t$. Thus, the time steps in HMM are advanced when features found in a new image are made available by the underlying vision system. The top image on the right is the image examined by the vision system at the fourth time step, the image on the bottom is the resultant location match based on a winner-take-all selection from the discrete probability distribution maintained by LA-RC after the fourth time step.

These initial results show fast convergence in each of the testing episodes; all con-

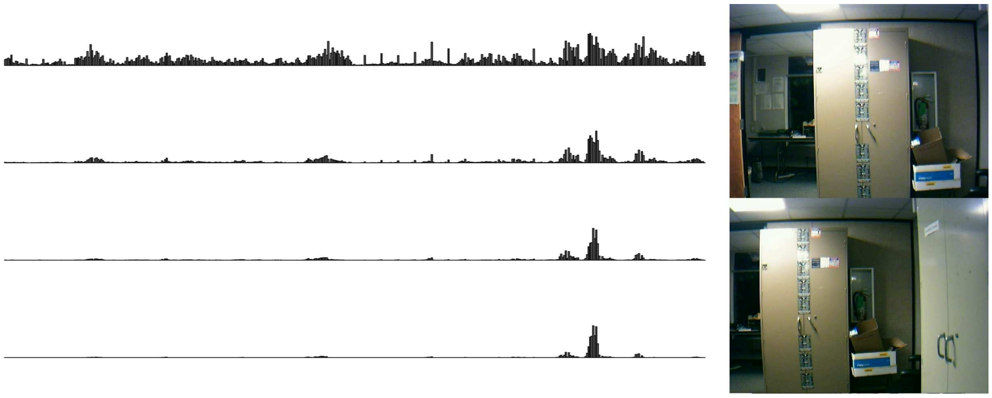

Figure 4.4: Sample convergence and the resulting match for the second indoor run (faculty offices setting). The plot layout is the same as in Figure 4.3.

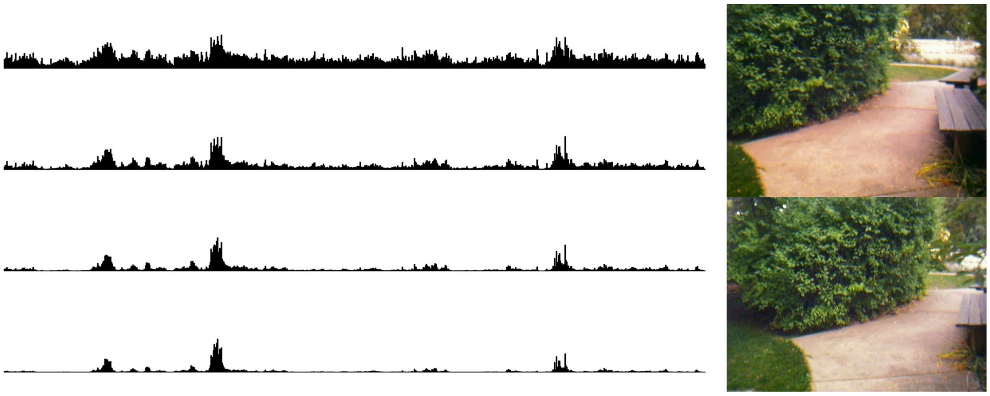

Figure 4.5: Sample convergence and the resulting match for the outdoor run (garden setting). The plot layout is the same as in Figure 4.3.

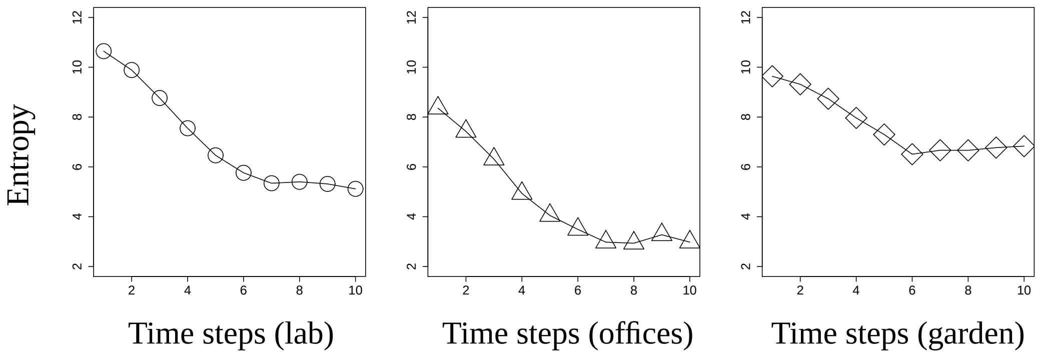

Figure 4.6: Convergence in LA-RC. The vertical axis in each plot shows positional entropy, horizontal—time steps. The graph on the left corresponds to the computer lab route, graph in the middle—to faculty offices, and graph on the right—to outdoor route.

vergences are to near the true location. The qualitative accuracy of localization is good since both the probe (testing) image and the resulting match from the training sequence are visually similar and spatially close to each other, as can be seen on the right-hand side of Figures 4.3– 4.5.

Information entropy [Sha48] can be used to quantify the convergence more precisely. In general, information entropy shows the level of uncertainty in data—lower values correspond to less uncertainty. Convergence in LA-RC also corresponds to reduction of uncertainty—in this case positional uncertainty (entropy). Therefore, convergence in localization with LA-RC can be measured by the loss of positional entropy. Mathematically entropy is defined as:

$$H = -\sum_{i=1}^{n} p_i \log p_i \tag{4.2}$$

where $p_i$ is an element from the set of state probabilities. Their analogues are the position estimates in LA-RC. Positional entropy for the time steps shown on the left-hand sides of Figures 4.3– 4.5 is plotted in Figure 4.6.

Figure 4.6 shows how entropy is decreasing during the first 10 time steps of execution of LA-RC for each setting's respective probe image. Note that after just 4 steps, the

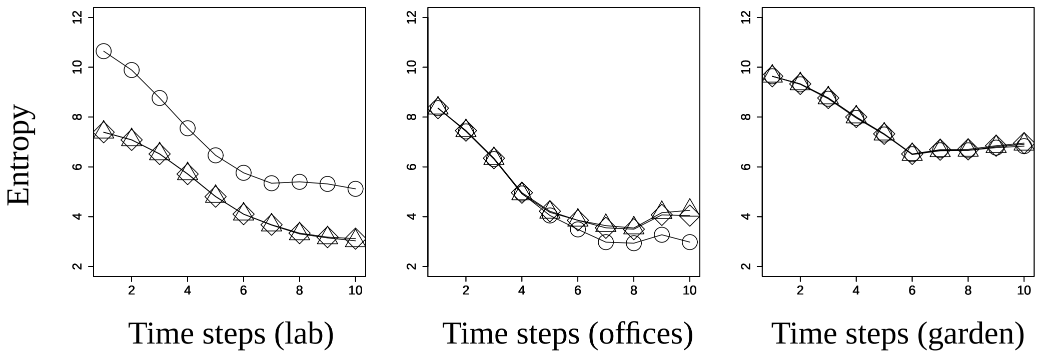

Figure 4.7: Impact of state transition matrices on convergence in LA-RC. The vertical axis in each plot shows positional entropy, horizontal—time steps. The graph on the left corresponds to the computer lab route, graph in the middle—to faculty offices, and graph on the right—to outdoor route. Plots marked by triangles show entropy at each time step when the original 25/50/25 transition matrix is used (25% probability the robot stays in the same location, 50% probability it moves forward one location, and 25% probability it moves forward two locations). Plots marked by circles correspond to 50/25/25 transition matrix, and plots marked by squares correspond to 33.3/33.3/33.3 transition matrix.

level of uncertainty is low enough to pick a close match using a simple winner-take-all strategy.

Entropy plots also allow comparison of alternative state transition matrices used in HMM and their impact on convergence. Figure 4.7 shows how two other state transition matrices compare to the original 25/50/25 matrix (25% probability the robot stays in the same location, 50% probability it moves forward one location, and 25% probability it moves forward two locations). The alternative matrices are 50/25/25 (50% probability the robot stays in the same location, 25% probability it moves forward one location, and 25% probability it moves forward two locations) and 33.3/33.3/33.3 (equal probabilities of robot remaining in the same location, moving forward one location and moving forward two locations). Producing qualitatively comparable results, different matrices have little impact on convergence speed, therefore further results reported below employ the original 25/50/25 transition probability matrix.

Another aspect of convergence which can be observed from the entropy point of view is peak region entropy. Peak region entropy here means how much entropy is concentrated within the vicinity of the most prominent peak in the discrete probability distribution over the set of known locations maintained by LA-RC. For all three testing runs using 25/50/25 state transition matrix, over 50% of positional entropy (3.829983, 3.152670, and 4.094095 for computer lab, faculty offices, and garden settings respectively) is concentrated within a 20-frame window around the peak. This shows that not only localization entropy is decreased during execution of LA-RC, but it is also narrowed down to a small region.

A more comprehensive look at localization performance of SeeAsYou with LA-RC algorithm is provided by Figure 4.8. It shows the maximum magnitudes of localization predictions—maxima in $P(L|O_t)$—after time steps 1, 4, and 10 as black dots for all possible starting locations in each setting. As before, the training and testing were done for each setting individually. From left to right the images show data for computer lab setting, faculty offices setting, and outdoor garden setting. From top to bottom the top images show data after step 1 ($t = 1$), the middle images show data after step 4 ($t = 4$), and the bottom—after step 10 ($t = 10$). In each of the 9 images, the vertical axis shows training locations and the horizontal axis represents the testing locations. The training and testing locations are ordered consecutively from bottom to the top and from left to right, starting from the beginning of the respective training and testing image sequences (a similar ordering as in Figure 3.7). The regions for which the accuracy of localization is improved in the following time steps are indicated by red arrows.

Since the maxima are concentrated along the diagonals, Figure 4.8 shows that SeeAsYou performs localization correctly for the majority of starting points along each path. Also, the presence of the regions for which the accuracy of localization is improved as well as the overall de-noising demonstrates effectiveness of LA-RC. The transition of the maxima closer to the diagonals during the execution of LA-RC also corresponds to

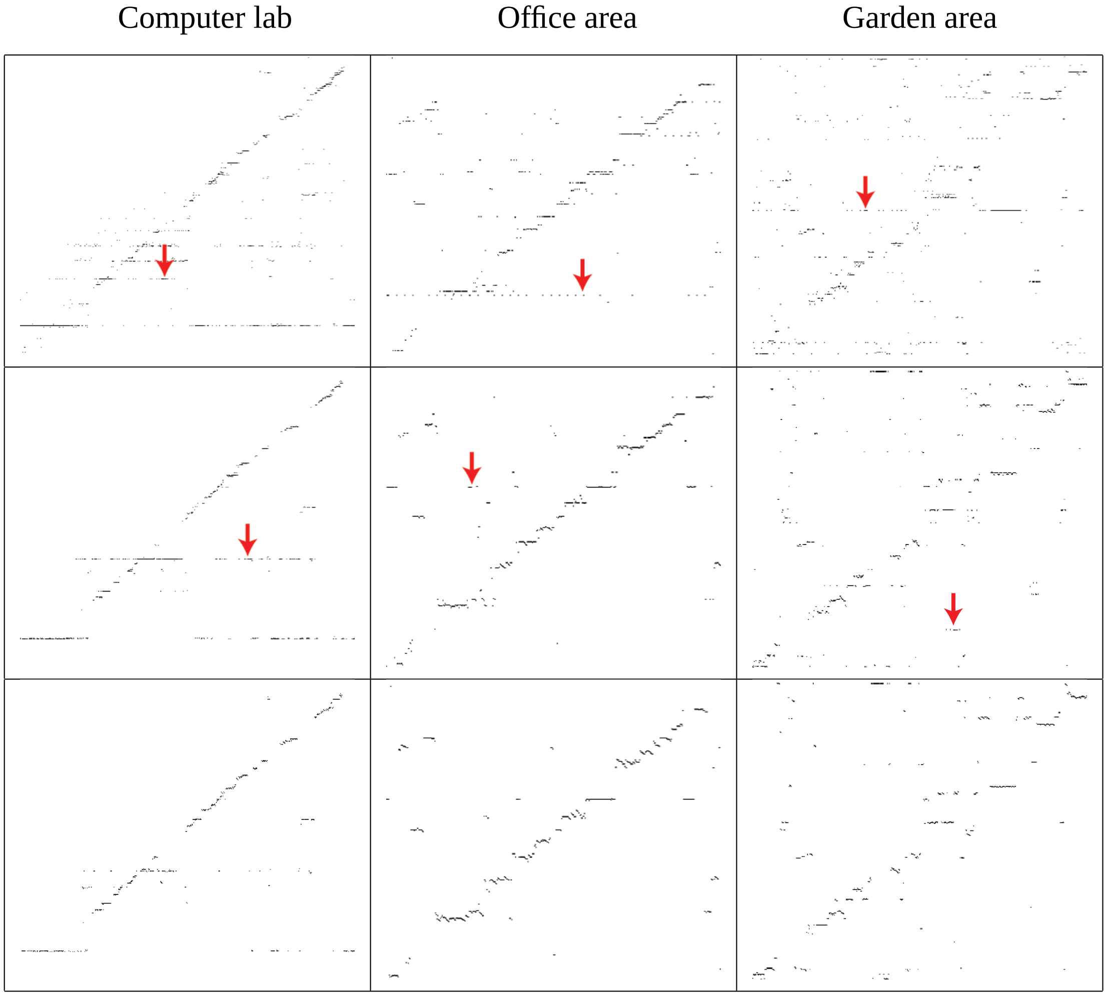

Figure 4.8: Maxima in $P(L|O_t)$ for all possible starting locations within each setting (separate training and testing for each setting). The maxima are shown as black dots. From left to right the images show data for computer lab setting, faculty offices setting, and outdoor garden setting. From top to bottom the top images show data for $t = 1$, the middle images show data for $t = 4$, and the bottom—for $t = 10$. In each of the sub-images, the vertical axis shows training locations and the horizontal axis represents the testing locations. The training and testing locations are ordered consecutively from bottom to the top and from left to right, starting from the beginning of the respective training and testing image sequences. The regions for which the accuracy of localization is improved in the following time steps (shown below) are indicated by red arrows.

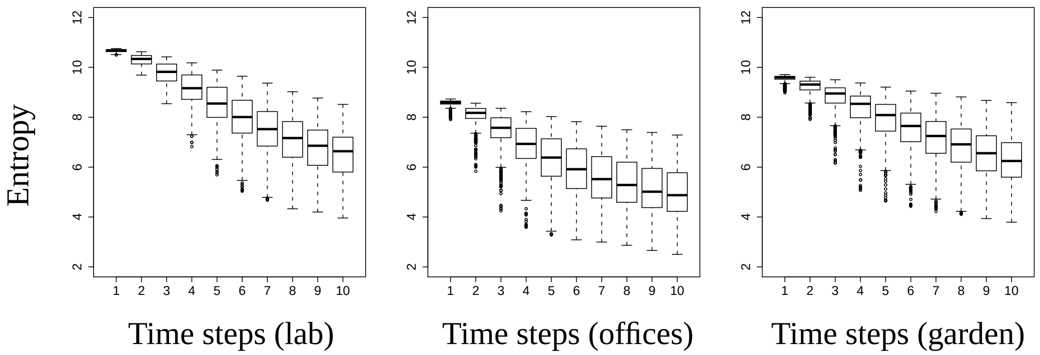

Figure 4.9: Entropy decrease for all possible starting locations within each setting (separate training and testing for each setting). For all possible starting positions, the entropy is plotted along the vertical axis and the time steps are plotted along the horizontal axis. The bottom and top of the box denote 25th and 75th percentiles respectively, line in the middle of the box denotes the median, whiskers denote ± 1.5 times interquartile range from the box, and the circles denote outliers.

the reduction in entropy: for the the computer lab run the mean entropy is 10.66 (standard deviation 0.05) after the first time step and 9.15 (standard deviation 0.59) after the fourth, for the faculty offices run the corresponding values are 8.56 (± 0.14) and 6.78 (± 0.96), and for the garden they are 9.55 (± 0.14) and 8.33 (± 0.73). The summary of entropy decrease in each of the three settings is shown in Figure 4.9. For all possible starting positions in each of the three settings, the entropy is plotted along the vertical axis and the time steps are plotted along the horizontal axis. For each of the time steps, the positional entropy is shown as a box plot: bottom and top of the box denoting 25th and 75th percentiles respectively, line in the middle of the box denoting the median, whiskers denoting ± 1.5 times interquartile range from the box, and the circles denoting outliers. In general, entropy decreases with each time step, at the same time becoming more dispersed. One possible explanation for the entropy dispersion induced by the execution of the algorithm is that for certain locations the uncertainty is not reduced or reduced much better than the average.

Still, two important questions remain: what happens if the image sequences are combined across the three settings? will SeeAsYou be able to localize with LA-RC both across the three settings as well as within each setting? These questions are answered by combining the image data from each setting into three image combined runs from 3034 to 3172 images long. As before, one of the sequences was used to train the hierarchical clustering in SeeAsYou, one—to build the TFDIF database, and parts of the third sequence were used for testing. The individual observation and transition matrices from the previous path experiment were combined along the diagonals to form larger observation and transition matrices. While the observation probability matrix was virtually unchanged (apart from the size), the new transition matrix did not allow transitions from one of the three settings to either of the others.

Figure 4.10 shows the maxima in the resulting $P(L|O_t)$ distributions for all possible starting locations corresponding to each of the three settings. While overall the results are comparable to Figure 4.8, the resulting maxima images are noticeably whiter and noisier (especially for the garden area) which is an indication of decreased performance. Also, as compared to the experiment when each setting was considered separately, the localization across the combined data set gives rise to higher positional entropies as shown in Figure 4.11.

The decrease in performance in this experiment can be easily seen if the fraction of misses—position estimates outside the corresponding training image sequences—is plotted versus time steps in LA-RC execution (see Figure 4.12). While for the computer labs setting the performance is generally positive, for the faculty offices area applying LA-RC actually deteriorates the position estimates after the first step. Also, in the outdoor garden area setting LA-RC fails to appreciably improve the accuracy of localization.

Such variation in performance in the final experiment can be attributed to two factors. First, compared to the computer lab sequence, both faculty offices and garden area

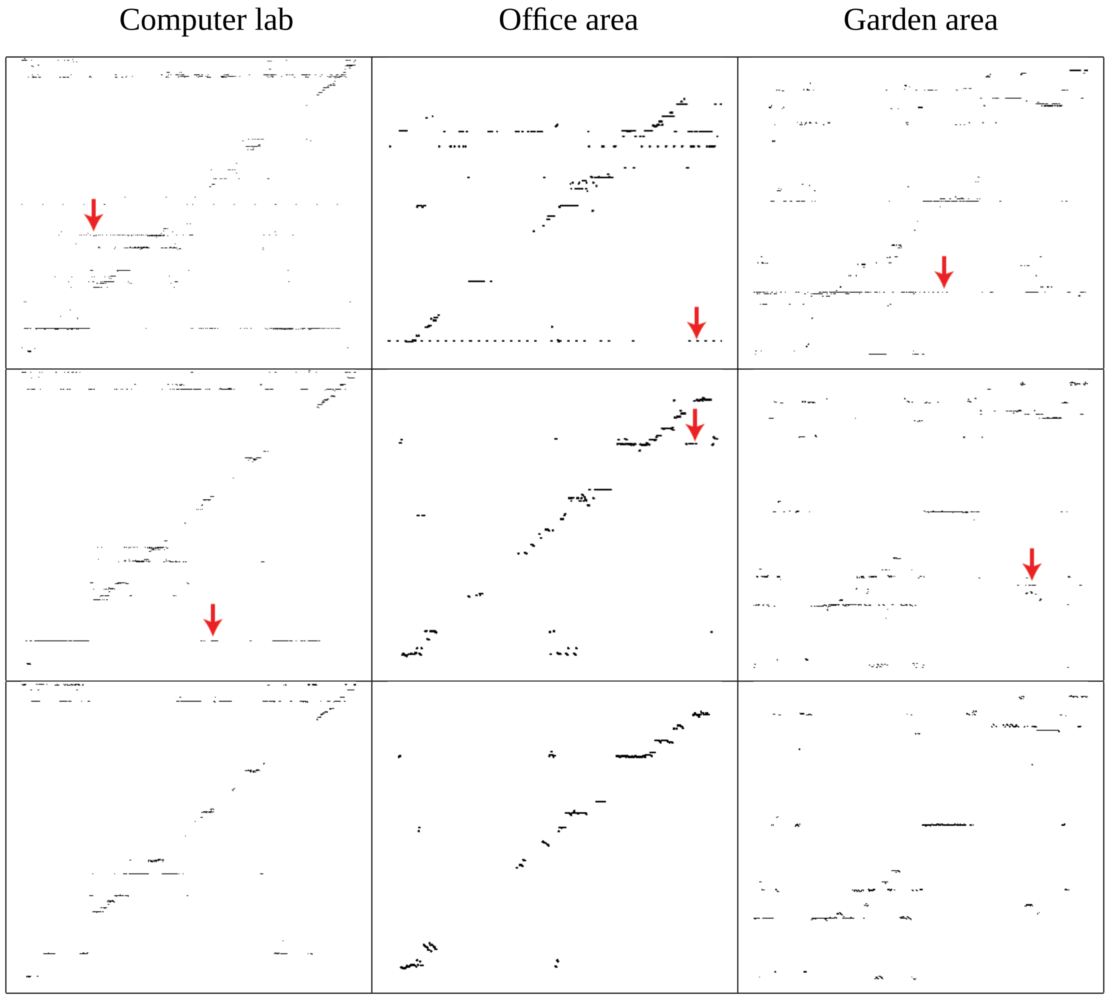

Figure 4.10: Maxima in $P(L|O_t)$ for all possible starting locations within each setting (combined training and testing for all settings). The maxima are shown as black dots. From left to right the images show data for computer lab setting, faculty offices setting, and outdoor garden setting. From top to bottom the top images show data for $t = 1$, the middle images show data for $t = 4$, and the bottom—for $t = 10$. In each of the sub-images, the vertical axis shows training locations and the horizontal axis represents the testing locations. The training and testing locations are ordered consecutively from bottom to the top and from left to right, starting from the beginning of the respective training and testing image sequences. The regions for which the accuracy of localization is improved in the following time steps (shown below) are indicated by red arrows.

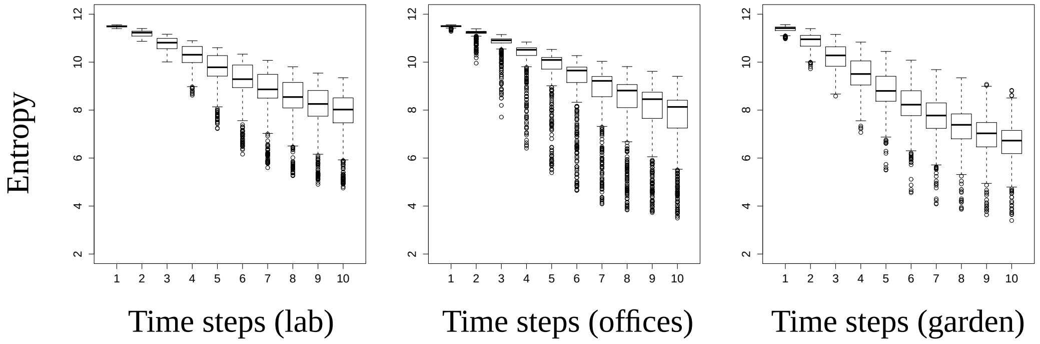

Figure 4.11: Entropy decrease for all possible starting locations within each setting (combined training and testing for all settings). For all possible starting positions, the entropy is plotted along the vertical axis and the time steps are plotted along the horizontal axis. The bottom and top of the box denote 25th and 75th percentiles respectively, line in the middle of the box denotes the median, whiskers denote ± 1.5 times interquartile range from the box, and the circles denote outliers.

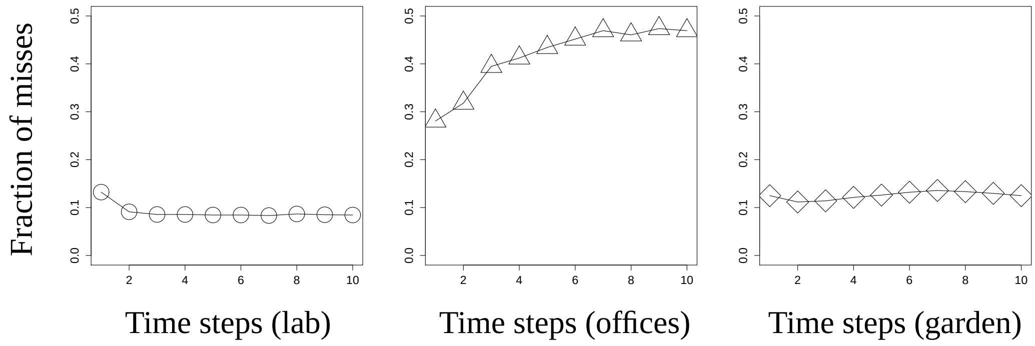

Figure 4.12: Fraction of misses as a function of time steps during execution of LA-RC. In each of the three plots the vertical axis shows the fraction of misses and the horizontal axis shows the timesteps. The plot on the left shows data for the computer labs setting, the plot in the middle—for faculty offices, and the plot on the right—for the garden area.

sequences were underrepresented: the three runs for the faculty offices contained from 460 to 474 images and the garden sequences contained from 827 to 879 images; on the other hand, the computer lab sequences contained between 1733 and 1833 images. Within the scope of the performed experiment, there exists a correlation between the number of images in the sequence and the quality of localization with LA-RC. This may also be a related issue to one described in the previous chapter when one of the locations (namely the HP lab) seemed to lack enough distinctive features compared to other settings which inhibited localization. The second factor contributing to the negative performance can be the simplicity of the observation and transition matrices used in HMM. While sufficient for a more homogenous experimental setup when each setting was tested independently of others, the transition and observation matrices employed may have been inadequate for the more heterogeneous conditions in the final experiment. The investigation of these factors as well as other issues encountered in heterogeneous settings with complex transitions is a possible subject for future work.

Although the experiments with SeeAsYou using LA-RC suggest reasonable convergence and accuracy of localization, a number of open questions remain such as what exactly is the accuracy of the algorithm and the underlying vision system, what is their resilience to errors, as well as what are the underlying effecting factors. These are some of the issues discussed in the next chapter.

# Chapter 5

# Conclusions & Future Work

Overall the experimental results show that SeeAsYou in conjunction with the LA-R/LA-RC algorithms can be used to localize a robot based exclusively on camera signal. Since SeeAsYou is a prototypical implementation of a feature-based vision system, the results are applicable to other vision systems adhering to that paradigm.

More specifically, using SeeAsYou with LA-R achieves 74–78% AUC in a kidnapping (forced-choice) experiment with eight different locations. This shows that the underlying feature-based vision system is capable but not sufficiently robust to accurately localize a mobile robot without the temporal context. Adding the temporal context with LA-RC, on the other hand, allows the vision system to achieve low-entropy location awareness after just four time steps in homogeneous settings, which, in terms of the performed experiments, is equivalent to localization after traveling 20 centimeters on a 30 meter route. At the same time, given a heterogeneous experimental setup with non-trivial transitions, SeeAsYou with LA-RC fail to achieve definitive location awareness. Concerning LA-RC, two possible issues may contribute to adverse performance: underrepresented locations and oversimplified HMM observation and transition matrices. The underrepresentation problem can be addressed by monitoring the number of relevant features detected in each location based on the TFIDF score of that location with itself. When this score reaches an acceptable threshold, accumulation of training images for that location may stop. In addition, the simple observation and transition matrices employed in the experiments carried out in this work can be augmented using EM (Expectation Maximization) [Bis06] to achieve better performance. Addressing these issues is a possible future research direction.

Another important issue with the presented results is that they are largely qualitative: at this point it is unclear what exactly is the accuracy of the algorithm and the underlying vision system, and what is their resilience to errors. To derive quantitative conclusions and explore the effecting factors, a more controlled set of experiments may be conducted in a simulated environment. USARSim [WLH$^+$05] provides a high-fidelity 3D platform for such simulations and exploring the localization capabilities of SeeAsYou using LA-RC within that framework is another potential future research direction.

Employing controlled simulated environments can also reveal important characteristics of LA-RC algorithm. As shown in the previous chapter, the algorithm is capable of localizing a robot along a path using SeeAsYou as the underlying vision system. However, due to time constraints and the limitations of the robot platform, exploring performance in large dynamic environments was not carried out. Intuitively, LA-RC complexity grows with the number of relevant landmarks acquired by the underlying feature-based vision system. The number of landmarks critical to achieve accurate location awareness is limited by the number of locations and their visual, but not geometric, complexity. Therefore it may be possible to concisely represent even very large environments. Since LA-RC does not require explicit maps and represents locations by simply associating relevant landmarks with them, minor changes in environment should not affect the performance. At this point, applying LA-RC algorithm in large dynamic environments is another future work possibility.

Finally, although the presented results suggest viability of the localization approach derived in this thesis for more than just vision system evaluation, the definitive answer to the question of the practicality of the algorithm can only be made by employing LA-RC with a vision system on an actual robotic platform coupled with a navigation system to do useful tasks. At the very basic level, navigation can be accomplished by associating the desired movement with each location. The robot can then use simpler sensors (for example laser or sonar range finders) for collision avoidance while following the main movement vector based on the visual location awareness provided by LA-RC. The implementation of such end-to-end system, however, remains a subject of future work.

To sum up, this thesis presented an evaluation of SeeAsYou as a prototypical feature-based vision system for the task of robot localization. It was shown that SeeAsYou was capable but not sufficiently robust to localize a robot without the temporal context, while adding such context suggested improved performance. LA-RC (Location Awareness through Relevance and Context), a simple general-purpose mobile robot localization algorithm using feature-based vision, was developed and tested with SeeAsYou. It was shown that with SeeAsYou, LA-RC works both indoors and outdoors, does not require manual map construction or transition labeling, input from odometry sensors, nor knowledge of intrinsic camera parameters. This favorably sets it apart from some of the other visual localization algorithms. Moreover, it is was designed to not not be limited to a specific vision system. As such it can be added as a component to existing mobile robot systems to do useful tasks. This, however, remains the subject of future work.

# REFERENCES

[ATD07] H. Andreasson, A. Treptow, and T. Duckett. Self-localization in non-stationary environments using omni-directional vision. *Robotics and Autonomous Systems*, 55(7):541–551, 2007.

[Bis06] C. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*, pages 615–618. Springer, 1st edition edition, 2006.

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*, pages 29–30. Addison Wesley, 1999.

[DFBT99] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *IEEE International Conference on Robotics and Automation*, page 7, May 1999.

[Dra07] B. Draper. A biomimetic vision architecture. In *International Conference on Computer Vision Systems*, page 10, 2007.

[FBT99] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11:391–427, 1999.

[FJC08] S. Frintrop, P. Jensfelt, and H. Christensen. Simultaneous robot localization and mapping based on a visual attention system. In *Attention in Cognitive Systems. Theories and Systems from an Interdisciplinary Viewpoint: 4th International Workshop on Attention in Cognitive Systems, WAPCV 2007 Hyderabad, India, January 8, 2007 Revised Selected Papers*, pages 417–430, Berlin, Heidelberg, 2008. Springer-Verlag.

[FLD05] U. Frese, P. Larsson, and T. Duckett. A multilevel relaxation algorithm for simultaneous localisation and mapping. *IEEE Transactions on Robotics*, 21(2):196–207, April 2005.

[GGD06] C. Giovannangeli, Ph. Gaussier, and G. Dsilles. Robust mapless outdoor vision-based navigation. In *IEEE/RSJ International Conference on Intelligent Robots and systems*, pages 3293–3300, Beijing, China, 2006.

[HA02] S. Hochstein and M. Ahissar. View from the top: Hierarchies and reverse hierarchies in the visual system. *Neuron*, 36:791–804, 2002.

[KBO$^+$05] N. Karlsson, E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. Munich. The vslam algorithm for robust localization and mapping. In *IEEE International Conference on Robotics and Automation*, pages 24–29, 2005.

[Low99] D. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, pages 1150–1157, 1999.

[OH05] N. Ouerhani and H. Hgli. Robot self-localization using visual attention. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, page 6, 2005.

[OHB05] N. Ouerhani, H. Hgli, and A. Bur. Visual attention-based robot self-localization. In *European Conference on Mobile Robotics*, page 6, 2005.

[OT01] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42:145–175, 2001.

[RN03] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*, pages 549–551. Prentice-Hall, Englewood Cliffs, NJ, 2nd edition edition, 2003.

[RWG04] A. Rodriguez, J. Whitson, and R. Granger. Derivation and analysis of basic computational operations of thalamocortical circuits. *Journal of Cognitive Neuroscience*, 16(5):856–877, June 2004.

[Sha48] C. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27(3):379–423, July 1948.

[SI07] C. Siagian and L. Itti. Biologically-inspired robotics vision monte-carlo localization in the outdoor environment. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 861–873, Oct 2007.

[SI08] C. Siagian and L. Itti. Storing and recalling information for vision localization. In *IEEE International Conference on Robotics and Automation*, pages 1848–1855, May 2008.

[SLL01a] S. Se, D. Lowe, and J. Little. Local and global localization for mobile robots using visual landmarks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 414–420, 2001.

[SLL01b] S. Se, D. Lowe, and J. Little. Vision-based mobile robot localization and mapping using scale-invariant features. In *IEEE International Conference on Robotics and Automation*, pages 2051–2058, 2001.

[SLL02] S. Se, D. Lowe, and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *International Journal of Robotics Research*, 21:735–758, 2002.

[SLL05] S. Se, D. Lowe, and J. Little. Vision-based global localization and mapping for mobile robots. *IEEE Transactions on Robotics*, 21:364–375, 2005.

[SSHW07] F. Schubert, T. Spexard, M. Hanheide, and S. Wachsmuth. Active vision-based localization for robots in a home-tour scenario. In *International Conference on Computer Vision Systems*, page 10, Bielefeld, Germany, March 2007.

[Thr02] S. Thrun. Robotic mapping: A survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*, pages 1–35. Morgan Kaufmann, 2002.

[TMFR03] A. Torralba, K. Murphy, W. Freeman, and M. Rubin. Context-based vision system for place and object recognition. In *International Conference on Computer Vision*, pages 273–280, 2003.

[UN00] I. Ulrich and I. Nourbakhsh. Appearance-based place recognition for topological localization. In *IEEE International Conference on Robotics and Automation*, pages 1023–1029, 2000.

[WLH$^+$05] J. Wang, M. Lewis, S. Hughes, M. Koes, and S. Carpin. Validating usarsim for use in hri research. In *Proceedings of the 49th Annual Meeting of the Human Factors and Ergonomics Society*, page 5, September 2005.