Learning Generative Models of Invariant Features

Robert Sim
Department of Computer Science
University of British Columbia
simra@cs.ubc.ca

Gregory Dudek
Department of Computer Science
McGill University
dudek@cim.mcgill.ca

***Abstract*— We present a method for learning a set of models of visual features which are invariant to scale and translation in the image domain. The models are constructed by first applying the Scale-Invariant Feature Transform (SIFT) to a set of training images, and matching the extracted features across the images, followed by learning the pose-dependent behavior of the features. The modeling process avoids assumptions with respect to scene and imaging geometry, but rather learns the direct mapping from camera pose to feature observation. Such models are useful for applications to robotic tasks, such as localization, as well as visualization tasks. We present the model learning framework, and experimental results illustrating the success of the method for learning models that are useful for robot localization.**

I. Introduction

This paper addresses the problem of learning to model image-domain features that are useful for applications to robotic tasks, such as image-based pose estimation. The inferred models capture the relationship between imaging and scene geometry without explicit geometric models, thereby enabling the use of arbitrary imaging devices (such as swapping an omnidirectional camera for a pinhole-like camera), and operation in natural environments, where many scene features might not correspond to localized three-dimensional points (such as salient points caused by occlusion boundaries or specularities). The models are constructed in a generative framework, using feature observations from known poses to compute a generating function that enables the prediction of feature behavior from arbitrary viewpoints. The models are also evaluated in order to estimate model misfit and sensor noise, enabling the probabilistic estimate of the likelihood of a feature observation, given a pose. This likelihood estimate is a useful tool for a variety of inference tasks, such as camera pose estimation. This paper will demonstrate the feature modeling framework and apply it to the problem of global robot pose estimation, providing experimental results that validate the approach.

Recent work in a variety of vision-based inference problems, such as object recognition and robot localization, has demonstrated that inferences based on local image features, as opposed to globally derived features, such as principal components analysis [1], [2] can provide robustness to a variety of factors, such as illumination variation, dynamic environments, and sensor noise [3], [4]. Other work has provided psychophysical evidence for feature-based recognition, as well as support for improvements in computational efficiency [5]. Our work is motivated by these ideas and specifically aims for robust behavior in dynamic environments.

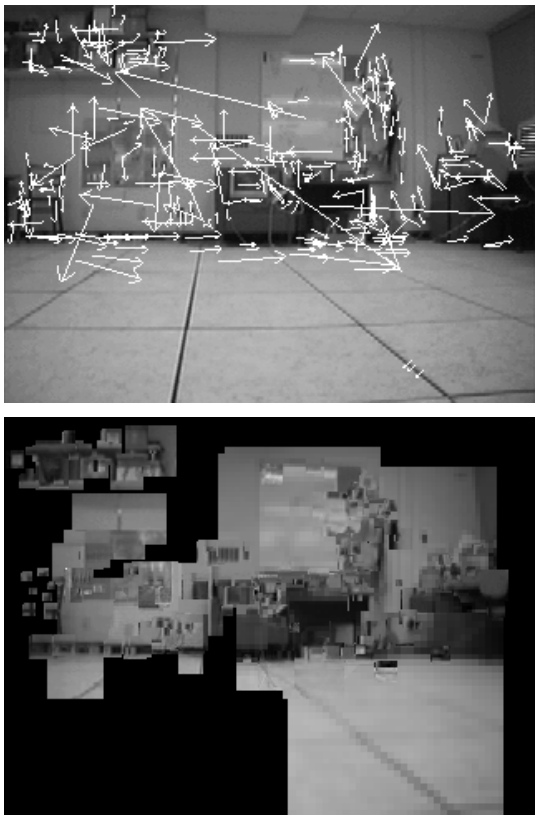


Fig. 1. a) Detected SIFT features in an image. Each arrow corresponds to a detected feature, with scale corresponding to arrow length, and orientation corresponding to arrow direction. b) Modeled SIFT features rendered from a novel viewpoint.

The main contribution of this paper is the application of the Visual Map framework developed in [6] to the problem of learning models of scale and rotationally invariant features. The modeled features demonstrate improved robustness for unambiguous recognition and tracking, and greater versatility for applications to camera pose estimation and visualization tasks. A key component of feature modeling is the reliable acquisition of training observations of individual features. In the previous work, features were initially detected using an edge-density operator. Such features tend to be poorly localized in the image, and subject to instabilities as a function of camera pose. The result is that they are difficult to track reliably and subject to outlier matches, resulting in degradations in the inferred models.

In this paper, the Scale-Invariant Feature Transform (SIFT), developed by Lowe [7] is employed (Figure 1 a). SIFT features provide enhanced stability against variations due to pose and illumination, as well as viewpoint-invariant descriptors for matching in the presence of changes in scale and rotation in the image plane. In addition, SIFT features enable the feature learning framework to model a wider variety of feature properties– specifically the characteristic scale and orientation of the feature– in addition to those that have been previously modeled, such as image position and local appearance (Figure 1b). The result is that enhanced robustness and versatility can be obtained, both in the modeling stage, and when applying the models to inference problems.

The basic problem of feature modeling involves computing a feature observation likelihood, conditioned on camera pose. That is, given a pose $\mathbf{x}$, what is the probability of observation $\mathbf{z}$ occurring, $p(\mathbf{z}|\mathbf{x})$? The ability to answer this question enables an agent to perform a variety of inference problems, such as robot pose estimation. Specifically, given an observation, the probability distribution over robot poses can be constructed from Bayes' Rule:

$$p(\mathbf{x}|\mathbf{z}) = \frac{p(\mathbf{z}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{z})}. \quad (1)$$

where $p(\mathbf{x})$ is the *a priori* distribution over robot poses, and $p(\mathbf{z})$ is independent of $\mathbf{x}$, and hence treated as a normalizing constant.

This paper takes a generative approach to feature modeling. Feature observations are assumed to be the noisy outputs from a generating function $F(\cdot)$ that maps the relationship between pose and feature. The problem of modeling a feature involves constructing a generating function $\hat{F}(\cdot)$ that approximates $F(\cdot)$:

$$\mathbf{z}^* = \hat{F}(\mathbf{x}) \quad (2)$$

and using $\mathbf{z}^*$ as a maximum likelihood estimate of the observation given the pose $\mathbf{x}$. The probability distribution $p(\mathbf{z}|\mathbf{x})$ is approximated as a Gaussian distribution centered at $\mathbf{z}^*$, and with covariance R, determined by a measure of model fit. The details of this model will be described in later sections. Clearly, feature observations are also functions of illumination, scene and camera geometry and sensor characteristics. We assume that these quantities are either static, and hence captured implicitly by the generating function, or the result of noisy real-world processes and hence captured implicitly by the Gaussian noise model. Note that, unlike the approach taken by Se, *et al.*, where SIFT features are modeled geometrically using a stereo camera [8], this work does not compute explicit geometric models of the extracted features. Similarly, this approach differentiates our work from structure from motion (SFM) [9], which imposes assumptions on both the features and the imaging geometry.

The remainder of this paper will address related work, and introduce the feature modeling approach, including implementation details. This will be followed by experimental results illustrating the utility of feature modeling for visualization and robot pose estimation tasks.

II. Related Work

The problem of feature-based localization has been the subject of extensive research. Early work examined triangulation methods for localizing a robot in the plane using 2D point landmarks [10], and culminated in the development of probabilistic approaches to active localization using the Kalman Filter and Markov chains [11], [12]. These principles have also been applied using 3D point features using both stereo, and pinhole camera models [8], [9], [13]. Finally, a variety of linear analysis techniques have resulted in features that are extracted implicitly [1], [2], and localization techniques using linear combinations of views [14]. Our work is similar to the earlier localization techniques in that it applies probabilistic methods to localization from feature observations. However, it is more similar to the latter techniques in that feature and camera geometry are not modeled explicitly but rather the (possibly complex) interaction of feature and sensor is learned as a function of pose.

This paper builds on the Visual Map framework developed in our prior work [6]. In that work, candidate features were extracted as local maxima of edge density, and only their positions and appearance were modeled as functions of pose. This paper employs the SIFT feature detector, enabling robust tracking and the additional modeling of feature scale as a function of pose.

III. Approach: Visual Maps

This paper takes the approach to feature modeling described in [6], adapted somewhat to take advantage of useful properties provided by the particular features that are modeled. Recall that the basic problem of feature modeling is to enable the computation of the observation likelihood function $p(\mathbf{z}|\mathbf{x})$, by first learning a generating function $\hat{F}(\cdot)$ that approximates $F(\cdot)$ for each feature, and subsequently modeling the reliability of the learned functions, and the noise processes that contribute to the observations.

Formally, we address the following problem:
Given:

- I, an ensemble of images of an environment, and
- X, ground truth pose information indicating the pose of the camera from which each image was acquired.

Compute: a feature-based visual representation of the environment by:

1) *Extracting* a set of visual features from I.
2) *Tracking* feature observations across I.
3) *Modeling* the generating function $F_i(\cdot)$ for each tracked feature, using the ground truth pose information X.
4) *Evaluating* the learned feature models for their reliability.

The framework operates by automatically selecting potentially useful features $\{f_i\}$ from a set of training images I of the scene taken from a variety of camera poses X (i.e. samples of the pose-space of a robot). The features are

selected using the SIFT feature detector. Once the features are selected and tracked, using a mechanism described below, the result is a set of observations for each feature, as they are detected from different positions. For a given feature f_i, the modeling task then becomes one of learning the imaging function $F_i(\cdot)$, parameterized by camera pose, that gives rise to the imaged observation $\mathbf{z}_i$ of f_i according to Equation 2. While a variety of alternative modeling approaches exist, this work employs radial basis function (RBF) networks as an interpolation mechanism, followed by cross-validation for model evaluation. The advantage of using an RBF-based approach is that no explicit assumptions about the nature of the features or imaging device are made, thus enabling the modeling of a wide variety of visual phenomena with an arbitrary imaging device.

A key point to note is that we are considering image ensembles for which ground-truth pose information is available. It is assumed that a mechanism is available for accurate pose estimation during the exploratory stage (such as assistance from a second observing robot). This assumption can be relaxed with more sophisticated map-building approaches, such as the utilization of expectation-maximization [15].

IV. Implementation

The feature learning framework is divided into four stages, extraction, tracking, modeling and evaluation. The following sections will describe the details of each stage.

A. Feature Extraction

Potential features are initially extracted from the training images using the SIFT feature detector developed by Lowe [7]. The SIFT feature detector operates by selecting local peaks in a difference-of-Gaussian pyramid computed from an input image. These peaks correspond to image positions and scales which closely meet criteria for scale-space invariance. In addition, the SIFT detector computes a set of dominant orientations for each detected feature point, producing a feature description that includes image position, scale and orientation, all quantities that can vary as a function of pose. Finally, the detector computes an invariant feature descriptor, consisting of a 128-byte vector sampling the local image gradient from a set of local shifts in image position and a set of orientations. The feature descriptor is remarkably stable for matching against changes in orientation and scale, as the following sections will demonstrate.

Once the set of feature points has been computed, the local image neighborhood surrounding each point is presumed to contain useful information, and these feature windows, along with their positions, scales, and invariant descriptors are returned as the output of the operator. Figure 1 depicts the selected features from an image as superimposed arrows over the original. The base of each arrow corresponds to the position of the feature, the direction of the arrow to its orientation and the length of the arrow to its scale.

B. Feature Tracking

Feature tracking is performed incrementally, by starting with the features in an initial training image, and matching those against the features detected in nearby training images. The training images are inserted in order of their distance in pose space from the centroid of the set of training poses. As images are inserted, feature f_i in the the database is only selected for matching if it has been observed from a nearby training pose. As new training images are added, if the number of successfully matched features in an image drops below a threshold of 0.5, the SIFT features in that image are used to initialize new features in the database. It should be noted that conventional feature tracking methods, such as conditional density propagation are dependent on time-series inputs and are not well suited to the requirements of our problem [16].

SIFT features are matched by comparing their invariant descriptors. The invariant descriptor $\mathbf{i}_t$ for feature f_i in the database is defined by the descriptor for the observation that is closest in pose space to the current training pose. The Euclidean distance $D(\mathbf{i}_1, \mathbf{i}_2) = ||\mathbf{i}_1 - \mathbf{i}_2||$ between descriptors in feature space defines the quality of a match, and matches are accepted only when they are unambiguous. Specifically, an optimal match $\mathbf{i}^*$ is accepted if, for the feature template $\mathbf{i}_t$,

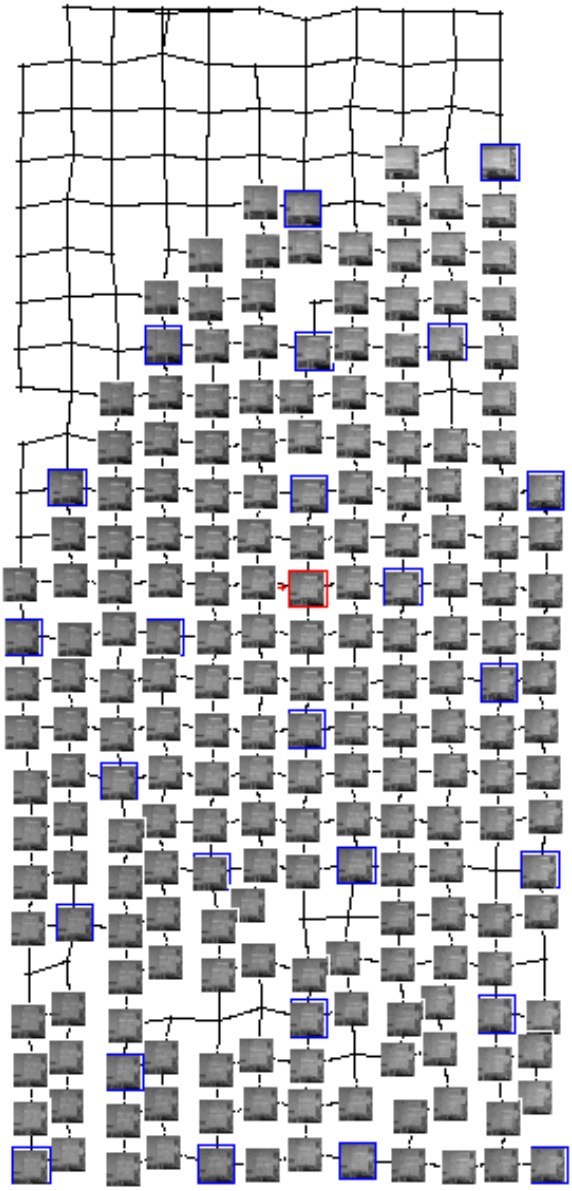
$$\forall_{\mathbf{i}_i \in \mathbf{z}_j} \frac{D(\mathbf{i}_t, \mathbf{i}^*)}{D(\mathbf{i}_t, \mathbf{i}_i)} < \tau \tag{3}$$

Therefore, a match is accepted if it is closer to the template than the next best match by a ratio of τ. For this work, we use $\tau = 0.6$. Note that the adaptive selection of the feature descriptor $\mathbf{i}_t$ as a function of pose enables the SIFT feature to evolve somewhat over the pose space. For tasks involving prior pose information, this descriptor can be modeled by the feature learning framework, whereas when no prior pose information is available, a static exemplar descriptor is selected from the observations.

Figure 2 illustrates a feature that has been tracked using this approach. Each thumbnail corresponds to an observation of the feature from the position represented by the thumbnail's position in the Figure. Here we are assuming a two-dimensional pose space, where the position of the camera varies over a plane and its orientation is fixed. The pose space is approximately 6.0m in the y-axis and 3.0m in the x-axis. The quality of the matching is very robust, and the feature is tracked over a wide region of the pose space.

C. The Generative Feature Model

We now turn our attention to the problem of inferring a generative feature model. We are interested in learning a pose-dependent model of a scene feature, given a set of observations of the feature from known camera positions. We require that the model will be capable of producing maximum-likelihood virtual observations (predictions) of the feature from previously unvisited poses. It will also be capable of estimating the likelihood of a new observation $p(\mathbf{z}_i|\mathbf{x})$, given the pose from which it might be observed.



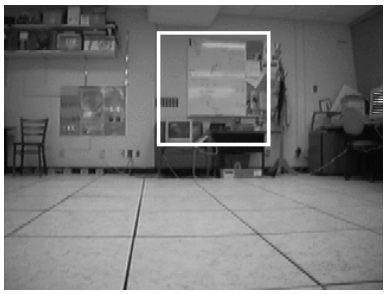
(a)

Fig. 2. A set of observations of an extracted scene feature. The grid represents an overhead view of the pose space of the camera, and feature observations are placed at the grid intersection corresponding to the pose where they were observed. Note that the observations capture variation in feature appearance, scale and position. The corresponding feature in the scene is highlighted in Figure 3.

An observation $\mathbf{z}$ of a feature f will be represented by the vector

$$\mathbf{z} = \begin{bmatrix} s \\ \mathbf{u} \end{bmatrix} \tag{4}$$

where s represents the scale of the feature and $\mathbf{u}$ represents the position of the feature in the image. Feature orientation can also be readily modeled, but is not informative in the context of the experiments presented here. In addition, for tasks involving visualization and reconstruction, the



(a)

Fig. 3. The feature that is depicted in Figures 2 and 4, highlighted in one of the training images.

feature vector $\mathbf{z}$ is augmented by the local intensity image $\mathbf{i}$, whose size is normalized by the scale parameter s. The observation $\mathbf{z}$ is a vector-valued function $F(\cdot)$ of the pose of the camera $\mathbf{x}$. The goal is to learn an approximation $\hat{F}(\cdot)$ of this function. In this paper, robot poses are assumed to be vectors in a two-dimensional space $\mathbf{x} = [x\ y]^T \in \Re^2$, corresponding to a camera moving through a planar environment at a fixed, but arbitrary, orientation.

The approach to learning $F(\cdot)$ is to model each element of the feature vector $\mathbf{z} \in \Re^k$ as a linear combination of radial basis functions (RBFs), each of which is centered at a particular robot pose determined by the set of training poses. Given a set of observations, a set of weights $\mathbf{w}_i \in \Re^k$ can be computed such that a linear combination of RBF's interpolates the observations, approximating the function that generated the observations. Formally, given a set of observations from known poses $(\mathbf{z}_i, \mathbf{x}_i)$, a predicted observation $\mathbf{z}$ from pose $\mathbf{x}$ is expressed as

$$\mathbf{z} = \hat{F}(\mathbf{x}) = \sum_i^n \mathbf{w}_i G(\mathbf{x}, \mathbf{x}_i) \tag{5}$$

where n is the number of training poses, and an exponentially decaying RBF $G(\cdot, \cdot)$ is used:

$$G(\mathbf{x}, \mathbf{x}_c) = \exp(-\frac{||\mathbf{x} - \mathbf{x}_c||^2}{2\sigma^2}) \tag{6}$$

where $\mathbf{x}_c$ represents the center of the RBF, and the response of the RBF is measured as a function of $\mathbf{x}$. The width, or influence, of the RBF is defined by σ.

The computation of the weight vectors $\mathbf{w}_i$ is well understood in the context of regularization and interpolation theory and is described elsewhere [17]. In brief, the optimal weights $W = [w_{ij}]$ are the solution to the linear least squares problem

$$(G + \lambda I)W = Z \tag{7}$$

where the elements G_{ij} of the design matrix G correspond to Equation 6 evaluated at observation pose i and RBF center j, and the rows of matrix Z correspond to the training observations. When λ is 0 and G^{-1} exists, the computed weights result in a network whereby Equation 5 interpolates the observations exactly. However, the

presence of noise and outliers and the complexity of the underlying function being modeled, can result in an interpolation which is highly unstable. The solution can be stabilized by adding a diagonal matrix of regularization parameters λI to the design matrix G. In this work, these regularization parameters and the RBF width σ are set by hand at the outset.[1] While ridge regression can be employed to compute the optimal regularization parameters, empirical experience indicates that this approach is not necessary for the distributions of measurements that are being interpolated.

If the design matrix employs every observation pose as a center for a RBF, the computational cost of computing the weights for n observations is that of an $O(n^3)$ singular values decomposition of an n by n matrix, followed by an $O(n)$ back-substitution for each element of the feature vector $\mathbf{z}$. For computational savings, at the cost of reduced accuracy, the number of RBF centers can be limited to a subset of the observation poses. In practice, we limit the maximum number of centers to 25, and select the subset of observation poses by ensuring that they cover the pose space uniformly. The drawback to this approach is that features that are visible over a large portion of the pose space are limited in terms of accuracy. One subject for further study is the question of whether a well-informed but non-uniform selection of training poses can yield more accurate models.

Figure 4 depicts three generated instances of the same feature from different poses. The predicted feature image $\mathbf{i}$ is plotted at the predicted image location $\mathbf{x}$. Note the variation in both appearance, position and scale of the feature in the image.

In addition to modeling the quantitative properties of a feature, it is also desirable to model its visibility as a function of pose; that is, whether or not a particular feature is visible from a particular location in pose-space. This information aids the task of localization and is important for the problem of reconstructing the scene. The same regularization framework presented in this section is employed to learn a visibility likelihood function $p(visible(f)|\mathbf{x})$, training the function with the binary-valued observability of each feature from each visited pose in the training set.[2] This information is also useful for informing the question of where to collect new training examples.

D. Model Evaluation

Once features have been tracked and modeled, it is desirable to evaluate the quality of the models. Each feature model is evaluated using leave-one-out cross-validation [18], which operates by constructing the model with one data point $\mathbf{z}$ excluded, predicting that data point $\mathbf{z}^*$ using the construction and measuring the difference $\mathbf{z}_e = ||\mathbf{z} - \mathbf{z}^*||$ between the actual point and the prediction.

[1]For the experiments presented here, $\lambda = 0.01$ and $\sigma = 2D/\sqrt{2M}$ where D is the maximal distance between any two poses in the training set and M is the number of training poses.

[2]The computed RBF model could produce likelihood values less than zero or greater than one– these outputs are clamped when they occur.

TABLE I
TRAINING SET STATISTICS FOR SCENES I AND II.

Attribute	Scene	
	I	II
Training images	121	291
Pose space (cm)	200x200	300x600
Ground truth accuracy (cm)	0.5	4.0
Sample spacing (cm)	20	25

By iterating over several (ideally all) of the training data, and computing the covariance R of the resulting error measures, we can build up a measure of how well the model fits the data and, more importantly, how well we might expect it to predict new observations.

The model covariance R is defined as

$$R = \frac{1}{k} \sum_{j=1}^{k} \mathbf{z}_e \mathbf{z}_e^T \tag{8}$$

where k is the number of observations of the feature and $\mathbf{z}_e$ is measured for each removed observation j.

Given R, the observation likelihood function for a feature observation $\mathbf{z}$, conditioned on pose $\mathbf{x}$, $p(\mathbf{z}|\mathbf{x})$ is then expressed as a Gaussian distribution:

$$p(\mathbf{z}_i|\mathbf{x}) = c \exp(-0.5 \mathbf{z}_e^T R^{-1} \mathbf{z}_e) \tag{9}$$

where $c = ((2\pi)^M |R|)^{-1/2}$, M is the dimensionality of the transformed observation space, $|R|$ is the determinant of R, and $\exp(x) = e^x$. In this case, $\mathbf{z}_e$ is computed using the complete feature model with $\mathbf{z}^* = F(\mathbf{x})$.

V. EXPERIMENTAL RESULTS

Fig. 5. Images from Scene I.

Fig. 6. Images from Scene II.

The experimental results presented in this section will be based on training data collected for two scenes (Figures 5 and 6). Table I indicates number of training images, pose-space geometry and approximate accuracy of the ground-truth pose information for each scene. The training pose space was a grid in which the robot collected samples at approximately uniform intervals. In addition, the orientation of the camera was held constant.

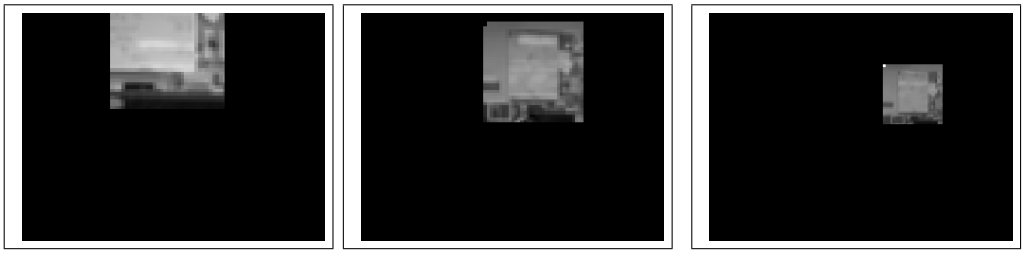


Fig. 4. A single feature as generated from three different camera positions.

Scene I was collected using a camera mounted on a Nomad 200 robot. A laser-pointer was mounted on the robot to point at the floor and the robot's position was measured manually based on the position of the laser point on the floor. It is assumed that the accuracy of these measurements is about 0.5cm.

Scene II was explored by taking 291 training images at uniform intervals of approximately 25cm over a 3.0m by 6.0m pose space. A second observing robot equipped with a laser tracking system was deployed to estimate the ground-truth position of the exploring robot to an accuracy of approximately 4cm. The implementation of the laser tracking system is described in [19]. The observer employed a laser range-finder to accurately determine the camera position from the range and orientation of a three-plane target mounted on the exploring robot. For the purposes of this scene, the robot attempted to take training images at the same global orientation. However, uncertainty in the robot's odometry, as well as the observing robot's estimate, led to some variation in this orientation from pose to pose. While this orientation variation affects the accuracy of the pose estimation process, our experimental results indicated that the reduction in accuracy is small relative to other effects.

A. Scene Reconstruction

Once a set of feature models have been trained, it is possible to apply the generative framework to predict observations from novel viewpoints (Figure 1b). At each position along a virtual trajectory, for each feature, the visibility likelihood of the feature is evaluated, and if it is greater than 0.5, the feature observation is predicted based on the current pose. The feature observation is subsequently rendered into the image using the predicted scale, appearance and position of the observation. Where more than one feature paints the same pixel, a winner-takes-all strategy is employed in which the feature model with smallest covariance determinant $|R|$ is selected. Note that the reconstruction cannot predict pixels for which there is no visible feature model, and as such, the lower edge of the image is left unshaded. It may be that these regions can be shaded by extrapolating from the nearby texture using Markovian reconstruction methods [20], [21].

B. Localization

The goal of the feature learning framework is to achieve accurate localization performance with a mobile robot. Given a set of feature models, the task of robot localization can be performed by applying Bayes' Law, as per Equation 1. When a pose estimate is desired, an observation is obtained and optimal matches $\mathbf{z} = \{\mathbf{z}_i\}$ to the learned features are detected in the image using the method described in Section IV-B. Note that when a prior pose distribution $p(\mathbf{x})$ is available, features can be selected for matching based on the probability that they are visible. When the prior is uniform, that is, no information is available, one can employ the efficient indexing methods described by Se *et al.* to efficiently match features [22]. Such a mechanism is beyond the scope of this paper.

Once matched, each feature observation $\mathbf{z}_i$ contributes a probability density function $p(\mathbf{z}_i|\mathbf{x})$, which is defined as the product of the distribution due to the maximum likelihood prediction of the model $p(\mathbf{z}_i|\mathbf{x}, visible(f))$ (Equation 9) and the feature visibility likelihood $p(visible(f)|\mathbf{x})$:

$$p(\mathbf{z}_i|\mathbf{x}) = p(\mathbf{z}_i|\mathbf{x}, visible(f_i))p(visible(f_i)|\mathbf{x}). \quad (10)$$

Pose inference based on the observation of a set of image features can be accomplished by assuming that the observation model $p(\mathbf{z}|\mathbf{x})$ is approximated by the joint likelihood of the set of feature observations conditioned on pose $\mathbf{x}$:

$$p(\mathbf{z}|\mathbf{x}) \approx p(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n|\mathbf{x}) \quad (11)$$

where we assume the formula is an approximation because we are ignoring any information that might be present in parts of the image other than those occupied by the detected features.

Assuming conditional independence between the individual feature observations, the probability of an observed image is defined to be the joint likelihood of the individual observations:

$$p(\mathbf{z}|\mathbf{x}) = \prod_{i=1}^{n} p(\mathbf{z}_i|\mathbf{x}) \quad (12)$$

In the absence of informative priors, the pose $\mathbf{x}^*$ that maximizes the joint likelihood of the observations is considered to be the maximum likelihood position of the robot. It is not clear, however, that the conditional independence assumption holds for features derived from a single image and, furthermore, outliers can lead to catastrophic cancellation of the joint distribution. Instead, we employ a mixture model defined by

$$p(\mathbf{z}|\mathbf{x}) \approx \sum_{i=1}^{n} p(\mathbf{z}_i|\mathbf{x}) \quad (13)$$

The reader may refer to [23] for a discussion of why this approach is reasonable.

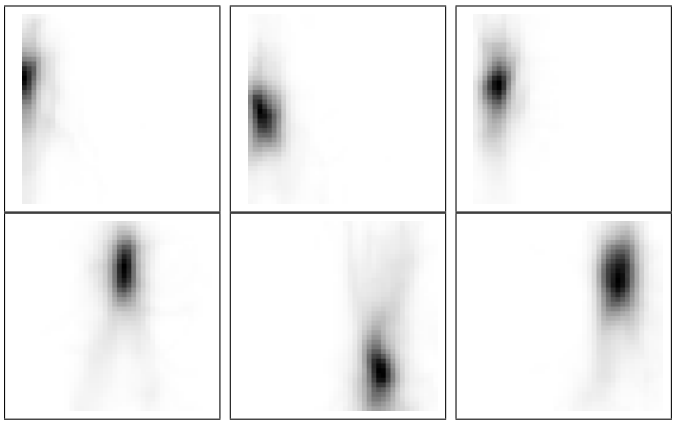


Fig. 7. *A posteriori* pose distributions for a selection of the Scene I validation images. Each image represents an overhead view of the 2m by 2m pose space. Darker regions correspond to more likely poses. Note that several of the distributions are not Gaussian or unimodal.

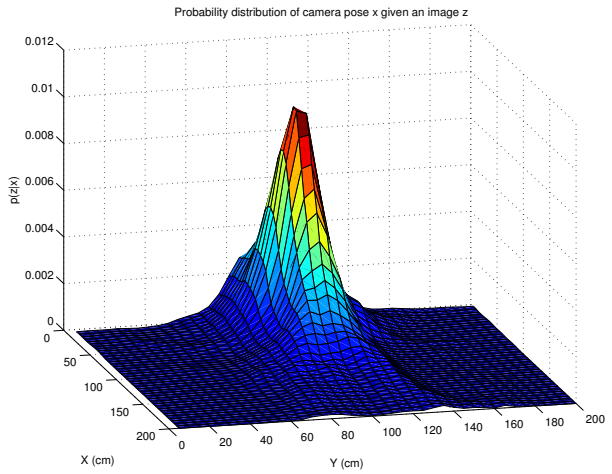


Fig. 8. An example *a posteriori* pose distribution for a validation image from Scene I.

Localization performance was measured using both of the training scenes. For each scene, SIFT features were extracted from the training images and the models were trained as described. Those models with large cross-validation error, or with too few observations to construct a useful model, were removed, resulting in a set of reliable feature models.

To validate localization performance using the learned models, for each scene an additional set of images were collected from random poses, constrained to lie anywhere within the training space. These validation images were used to compute maximum-likelihood (ML) estimates of the camera's position using Equation 13, and these estimates were compared against the ground truth pose information.

The ML estimates themselves were computed by exhaustive search over a multi-resolution discretization of the training space, selecting the $\mathbf{x}$ that maximized Equation 13. In particular, the training space was discretized into a 40 by 40 grid covering the entire training space and Equation 13 was evaluated at each position in the grid. Subsequently, at the maximal grid location a new 10 by 10 grid was instantiated over a neighborhood spanning 7 by 7 grid positions in the larger grid and Equation 13 was evaluated over the new grid. This process iterated recursively to a resolution of 1% of the intervals between training poses, and the maximal grid pose at the highest resolution was returned. Note that a more efficient estimator, such as Monte Carlo sampling, could be easily deployed for applications where computational resources are limited.

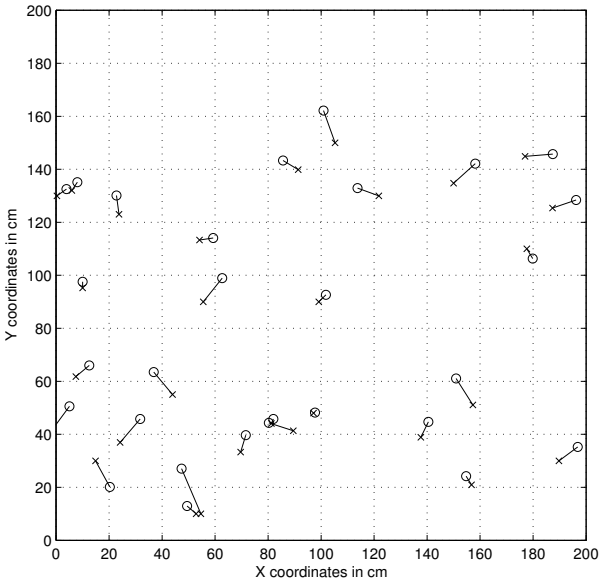


Fig. 9. Localization results for Scene I: the set of maximum-likelihood pose estimates ('x') plotted against their ground truth estimates ('o').

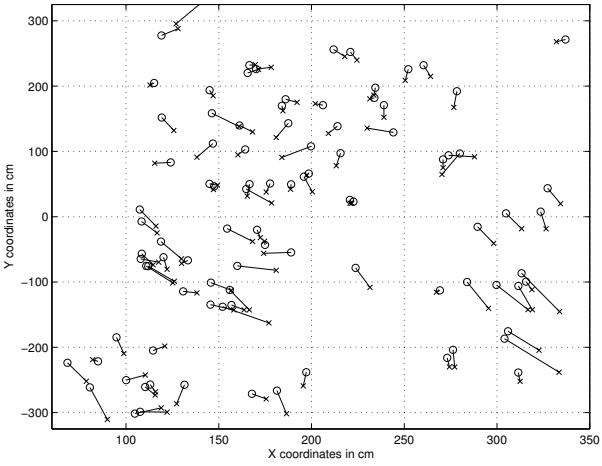


Fig. 10. Localization results for Scene II: the set of maximum-likelihood pose estimates ('x') plotted against their ground truth estimates ('o').

In practical settings, one is not always interested in the ML pose estimate, but sometimes in the entire probability

TABLE II
SUMMARY OF LOCALIZATION RESULTS.

	Scene	
	I	II
Training images	121	291
Pose space (cm)	200x200	300x600
Ground truth accuracy (cm)	0.5	4.0
Sample spacing (cm)	20	25
Validation Images	29	93
Valid Pose Estimates	29	92
Mean Error (cm)	7.7	18.4
Mean x Error (cm)	4.8	7.2
Mean y Error (cm)	5.3	15
Minimum Error (cm)	0.63	2.8
Maximum Error (cm)	18	62

distribution over the pose space, which can provide information about alternative hypotheses in environments which exhibit significant self-similarity. Figure 7 depicts the *a posteriori* pose distributions computed for a selection of the Scene I validation images. Each frame in the Figure represents the evaluation of Equation 13 computed over a uniform discretization of the 2m by 2m pose space, where darker regions correspond to more likely poses. Figure 8 depicts another pose distribution, derived from a Scene I validation image, in greater detail.

Given that each ML estimate has a numerical likelihood, it is possible to reject pose estimates that do not meet a particular confidence threshold. In this way, one estimate was rejected for Scene II, where a lack of successful feature matches reduced the confidence of the estimate.

Figures 9 and 10 plot for each scene the location of the ML estimates for the validation images ('x') against the ground truth camera position ('o') by joining the two points with a line segment for each scene. The length of each line segment corresponds to the magnitude of the error between the corresponding pose estimate and ground truth. The mean absolute error, mean x and y direction errors (corresponding to sideways and looming motion, respectively), minimum and maximum errors and number of retained estimates for each validation set are tabulated in Table II. The larger error in the y (looming) direction corresponds to the fact changes in observations due to forward and backward motion are not as pronounced as changes due to side-to-side motion. Reducing the uncertainty in this direction could be performed using an active localization method, such as Markov Localization [12].

VI. CONCLUSIONS

This paper presented an approach to modeling the pose-dependent characteristics of the scale-invariant feature transform. Such an approach extends our previous work on visual feature modeling and takes advantage of the additional properties of SIFT features, such as optimal feature scale and viewpoint-invariant descriptors, that enable unambiguous matching, and a more versatile feature model. We have applied these models to the problems of scene reconstruction, and robot localization, with successful results.

Our future work involves employing feature models in very large pose spaces. In these domains, SIFT features are well suited as there are extent indexing methods which enable efficient feature matching. We are also investigating on-line map learning approaches, where the ground truth pose information is uncertain, or even unknown.

REFERENCES

[1] F. Pourraz and J. L. Crowley, "Continuity properties of the appearance manifold for mobile robot position estimation," in *Proceedings of the IEEE Computer Society Conf. on Pattern Recognition Workshop on Perception for Mobile Agents*. Ft. Collins, CO: IEEE Press, June 1999.

[2] S. Nayar, H. Murase, and S. Nene, "Learning, positioning, and tracking visual appearance," in *Proc. IEEE Conf on Robotics and Automation*, San Diego, CA, May 1994, pp. 3237–3246.

[3] C. Schmid, "A structured probabilistic model for recognition," in *Proceedings of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, Ft. Collins, CO, June 1999, pp. 485–490, citeseer.nj.nec.com/schmid99structured.html.

[4] A. Shokoufandeh, I. Marsic, and S. Dickinson, "View-based object recognition using saliency maps," *Image and Vision Computing*, vol. 17, pp. 445–460, 1999.

[5] J. K. Tsotsos, "The complexity of perceptual search tasks," in *Intl. Joint Conf. on Artificial Intelligence*, 1989, pp. 1571–1577.

[6] R. Sim and G. Dudek, "Learning generative models of scene features," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 45–61, October 2004.

[7] D. G. Lowe, "Object recognition from local scale-invariant features," in *Int. Conf. on Computer Vision*, Corfu, Greece, September 1999, pp. 1150–1157.

[8] S. Se, D. G. Lowe, and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *Int. J. Robotics Research*, vol. 21, no. 8, pp. 735–758, 2002.

[9] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision.* Cambridge, UK: Cambridge Univ. Pr., 2000.

[10] K. Sugihara, "Some location problems for robot navigation using a single camera," *Computer Vision, Graphics, and Image Processing*, vol. 42, pp. 112–129, 1988.

[11] J. J. Leonard and H. F. Durrant-Whyte, "Mobile robot localization by tracking geometric beacons," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 376–382, 1991.

[12] D. Fox, W. Burgard, and S. Thrun, "Active Markov localization for mobile robots," *Robotics and Autonomous Systems (RAS)*, vol. 25, pp. 195–207, 1998.

[13] A. Davison, "Real-time simultaneous localisation and mapping with a single camera," in *Proceedings of the IEEE Int. Conf. on Computer Vision*, Nice, France, 2003.

[14] R. Basri and E. Rivlin, "Localization using combinations of model views," in *Int. Conf. on Computer Vision*, Berlin, 1993, pp. 226–230.

[15] S. Thrun, D. Fox, and W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Machine Learning*, vol. 31, pp. 29–53, 1998.

[16] M. Isard and A. Blake, "Condensation–conditional density propagation for visual tracking," *Int. Journal of Computer Vision*, vol. 29, no. 1, pp. 2–28, 1998.

[17] S. Haykin, *Neural Networks*. New York, NY: MacMillan College Publishing Company, 1994.

[18] R. R. Picard and R. D. Cook, "Cross-validation of regression models," *Journal of the American Statistical Association*, vol. 79, no. 387, pp. 575–583, 1984.

[19] I. M. Rekleitis, G. Dudek, and E. Milios, "Multi-robot collaboration for robust exploration," in *Proceedings of the IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, April 2000, pp. 3164–3169, http://www.cim.mcgill.ca/~yiannis/Publications/icra00.pdf.

[20] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling," *IEEE Int. Conf. on Computer Vision*, pp. 1033–1038, September 1999.

[21] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 6, pp. 721–741, November 1984.

[22] S. Se, D. Lowe, and J. Little, "Global localization using distinctive visual features," in *Proceedings of the IEEE/RSJ Conf. on Intelligent Robots and Systems (IROS)*, Lausanne, Switzerland, October 2002, pp. 226–231.

[23] T. Minka, "The 'summation hack' as an outlier model," http://www.stat.cmu.edu/~minka/papers/minka-summation.pdf, 2003.