Robust and Efficient Simultaneous Localization and Mapping from Bearings Only Sensing

Matthew C. Deans
The Robotics Institute
Carnegie Mellon University

Abstract

The simultaneous localization and mapping (SLAM) problem has been investigated by many researchers, and solutions frequently rely on sensor based approaches, such as very accurate range/bearing sensors (mmw radar, laser rangefinders, imaging sonar) which measure the position of landmarks in the environment relative to the robot and complex kinematic vehicle models or inertial naviation systems which measure the motion of the robot through the environment. Some solutions also require that the robot can measure orientation by some accurate and external means, reducing pose estimation to position estimation. These solutions tend to rely on a Kalman Filter or Extended Kalman Filter to recursively estimate both the map and robot pose in one global, absolute coordinate system.

In the proposed research, I will address the SLAM problem for an autonomous mobile robot in an unstructured environment using landmark bearing measurements and dead reckoning as its only means of sensing. I will investigate the use of a representation of the environment which parameterizes relationships between landmarks rather than their position in a global, absolute coordinate frame. These relationships are invariant to the position from which landmarks are observed, decoupling the uncertainties in landmark positions and robot position. This decoupling has implications on the way in which sensor fusion is done and the uncertainty in the environment map and robot position estimates which result.

The aim of this research is to provide a more general framework for SLAM applicable over a wider range of scenarios, and to provide a solution to the SLAM problem that is efficient, scalable, and robust to sensing and modeling errors.

1 Thesis Statement

I propose to develop a novel approach to the simultaneous localization and mapping (SLAM) problem for a mobile robot which uses information from a bearings only sensor and odometry to produce a map of an unknown environment while localizing the robot within the map. I claim that the algorithm developed will be capable of mapping and localizing without the use of range sensors and without external position or orientation references such as GPS or compass. This will make it possible for an autonomous rover with only a bearing sensor and wheel encoders to navigate an a priori unknown environment, map the locations of landmark features in the environment, and localize itself in a far more precise way than through dead reckoning alone.

The proposed method will construct globally consistent feature based metric maps, and will be more efficient than standard brute force batch estimation procedures but more stable and robust than Kalman Filter based methods which are pervasive in the SLAM literature. Furthermore, I propose to design the method to be robust to outlying sensor measurements and to be capable of recovering bias in process and measurement models. The advances will make it possible to apply SLAM to robots in environments which would present difficulty to current state of the art methods, where motion models are poorly calibrated, noise levels are high, and range measurements and vehicle heading measurements are unavailable.

2 Introduction

Required navigation capabilities for an autonomous mobile robot have been classified as 1. sensing the environment, 2. building its own representation of the environment, 3. locating itself within the environment, and 4. planning and executing efficient routes in this environment [1]. The first of these tasks is referred to as *sensing*, the second *mapping*, and the third *localization*. The three taken together comprise the Simultaneous Localization and Mapping (SLAM) problem. The fourth problem, typically referred to as *path planning* is not considered here.

Suppose that a robot is in a 2D world and is located at $\mathbf{r}_i = (r_x^{(i)}, r_y^{(i)}, r_\theta^{(i)})$. We call $\mathbf{r}_i$ the *pose* of the robot. In the world are landmarks which are distinct points with locations $\mathbf{l}_j = (l_x^{(j)}, l_y^{(j)})$. We define the *map* to be the collection of 2D coordinates of all of the landmarks, $\mathbf{L} = (\mathbf{l}_0, \mathbf{l}_1, ... \mathbf{l}_{N-1})$. The robot has the ability to make stochastic observations of the environment

$$\phi_{ij} = h(\mathbf{r}_i, \mathbf{l}_j) + \nu \tag{1}$$

where $h()$ is called the *measurement equation*, which predicts what the measurement should be given a landmark position and a robot pose, and ν is *observation noise*, frequently modeled by Gaussian noise with zero mean and covariance R, which corrupts the ideal measurement.

In many applications, the use of an *a priori* map enables a mobile robot to estimate its pose within the environment by obtaining a set of measurements $\mathbf{\Phi}_i = \{\phi_{i,0}, \phi_{i,1}, ..., \phi_{i,N}\}$ and then determing a pose which matches the obtained measurements to measurements which are expected given the map. This amounts to finding a robot pose $\mathbf{r}$ which maximizes

$$P(\mathbf{r}|\mathbf{L}, \mathbf{\Phi}_i) = kP(\mathbf{\Phi}_i|\mathbf{r}, \mathbf{L})P(\mathbf{r}|\mathbf{L}) \tag{2}$$

which is a maximum a posteriori (MAP) estimate. This is the *localization* problem. When no prior information on the robot pose is given, $P(\mathbf{r}|\mathbf{L})$ is unknown and we instead maximize

$$P(\mathbf{r}|\mathbf{L}, \mathbf{\Phi}_i) = kP(\mathbf{\Phi}_i|\mathbf{r}, \mathbf{L}) \tag{3}$$

to find the maximum likelihood (ML) estimate. There is a significant amount of research in localization using bearings only measurements of the environment, with either known map-measurement correspondence or unknown map-measurement correspondence [2]. When ν is a Gaussian random variable with covariance R, the distribution over robot pose given the map and observations can be written

$$P(\mathbf{\Phi}_i|\mathbf{r}, \mathbf{L}) = \exp\left(-\left[\mathbf{\Phi}_i - h(\mathbf{r}; \mathbf{L})\right]^T R^{-1} \left[\mathbf{\Phi}_i - h(\mathbf{r}; \mathbf{L})\right]\right) \tag{4}$$

and maximizing (3) is equivalent to minimizing the negative log likelihood

$$\left[\mathbf{\Phi}_i - h(\mathbf{r}; \mathbf{L})\right]^T R^{-1} \left[\mathbf{\Phi}_i - h(\mathbf{r}; \mathbf{L})\right] \tag{5}$$

This is simply a minimum mean square fit between the measurements obtained and those predicted by the model. In some cases, the mean square fit is immediately taken as the goal in the estimation problem, rather than justifying it as a natural result of Bayesian estimation under Gaussian distributed noise.

There may be cases where there exists some prior information on the robot pose, for example if the robot starts from some known location, or there is a pose estimate which is made by predicting the current pose given some knowledge of the robot pose at an earlier time. If this prior can be modelled by a normal distribution with mean $\hat{\mathbf{r}}$ and covariance Σ, (2) is replaced by minimization of

$$[\mathbf{\Phi}_i - h(\mathbf{r};\mathbf{L})]^T R^{-1} [\mathbf{\Phi}_i - h(\mathbf{r};\mathbf{L})] + [\hat{\mathbf{r}} - \mathbf{r}]^T \Sigma^{-1} [\hat{\mathbf{r}} - \mathbf{r}] \tag{6}$$

which is again a minimum mean square fit, which incorporates a fit between the measurements which are observed and those predicted by the model, represented by the first term in (6), and also a fit to the predicted or prior estimate of the robot pose, represented by the second term in (6).

Localization using (3) and (5) is typical in *global localization* where the robot has no prior information on where it is, except perhaps that it is within some region bounded by the known map. This problem is often solved using some iterative technique such as Newton-Raphson search. The type of localization in (2) and (6) is typical in *pose tracking* where the robot pose is updated frequently using old estimates of pose to define a prior on the current pose. This problem is frequently solved using a Kalman filtering type approach.

If no map or external references are available, path integration or dead reckoning offers a means of tracking the pose of the robot, but the accumulation of errors during path integration causes estimates to drift and the precision to degrade without bound. Furthermore, without a map, global localization is impossible since the question of where the robot is within some "map" coordinate frame is meaningless.

In other applications there may be an external reference which provides the robot position throughout its trajectory $\mathbf{R}$, and the robot may then sense the environment and produce a map of the environment using the available pose information and measurements by maximizing

$$P(\mathbf{L}|\mathbf{R},\mathbf{\Phi}_i) = kP(\mathbf{\Phi}_i|\mathbf{L},\mathbf{R})P(\mathbf{L}|\mathbf{R}). \tag{7}$$

This is referred to as *mapping*. There is a significant amount of research in mapping from bearings only sensors as well.

Without a priori map information or external position information, a mobile robot must simultaneously map an area and localize itself within the map it is generating. This is equivalent to maximizing

$$P(\mathbf{L},\mathbf{R}|\mathbf{\Phi}_i) = kP(\mathbf{\Phi}_i|\mathbf{L},\mathbf{R})P(\mathbf{L},\mathbf{R}) \tag{8}$$

This problem is often referred to as Concurrent Localization and Mapping (CLM) or Simultaneous Localization and Mapping (SLAM).

2.1 Assumptions

In order to more completely describe the scope of the work proposed, I will outline a few key assumptions that I will make in formulating the solution to the stated problem.

In particular, it is beyond the scope of this work to solve the visual tracking problem and the landmark identification problem. It will be assumed that there exists a system capable of 1) finding interesting natural terrain features to use as landmarks 2) tracking those features and measuring the bearing to each feature relative to the robot and 3) recognizing old landmark features when they are reacquired after they have been lost from the tracking algorithm. The last of these capabilities is crucial to the SLAM problem, since a robot driving in a large closed path must be able to recognize that it has returned to its initial location or the map estimate will diverge.

Furthermore, it will be assumed that the environment can be reasonably well modeled by a 2D plane, and that the robot pose is fully described by (r_x, r_y, r_θ) which are the horizontal translation and rotation from the initial pose, and (l_x, l_y) the 2D position of each landmark, which can be observed via triangulation using bearing measurements and robot motion estimates. On natural terrain, it may be required to sense the robot pitch and roll using an inclinometer and project the robot pose onto a plane, after which the pose will be represented using the remaining 3 DOF. Solving the SLAM problem in $3D$ requires estimating $(r_x, r_y, r_z, r_\theta, r_\phi, r_\psi)^T$ for the robot and (l_x, l_y, l_z) for each landmark.

Finally, it will be assumed that we can without penalty use every available landmark which is identified by the vision system to produce a map and to localize. It is generally only necessary to use a small number of landmarks for localization and mapping, and very large numbers of landmarks can require large amounts of memory and computing resources, but it will be assumed that there is not such an overabundance of landmarks that the SLAM system cannot handle it. This assumption is justified in that it is likely that a visual tracking module would be overwhelmed before the SLAM system.

2.2 Implications of unstructured environment

- The environment is unstructured and natural terrain features are the only landmarks. Indoor environments offer structure which can be exploited in building maps, such as the fact that walls are planar, etc. The first point here directly influences how landmarks are selected, tracked, and recognized. This problem is not within the scope of this work. However, the second implication of the unstructured nature of natural scenes has a direct impact on map building. We cannot expect to observe geometrically simple objects such as planar walls and corners or hope to recover some parameterized structural model.
- Terrain is 2 1/2D. Indoor environments may often be modelled using a flat ground assumption, but we cannot make this assumption in unstructured outdoor environments. That said, for the purpose of this thesis, we will start with a 2D model of the world and attempt to extend results to 3D afterward, since the complexity of modeling a 3D environment is significantly more difficult than for the 2D case.
- Lighting is uncontrolled and changes during the course of a day. This is again of importance to the vision system since detection, tracking, and most importantly recognition may require insensitivity to lighting conditions.
- Dead reckoning is worse than for indoor robots. This has to do with more complicated wheel/ground interaction, the size of the wheel contact area on the ground, and terrain relief. It will be argued that odometry for robots in unstructured environments should have as little influence on the SLAM estimation as possible due to this problem.

2.3 Implications of "online" requirement

- The online requirement implies that we would like to be able to generate estimates of landmark locations immediately upon observing landmark bearings, and always be able to produce a best estimate given what has been seen so far.

- The map update computation time must not scale with the size of the map, otherwise for any algorithm and minimum execution time there will be a map large enough that it cannot be updated within the required time.
- If the map update is required to take constant time, an update cycle cannot update the position of every landmark in the map, since the best we could hope for in this case would be an O(N) update algorithm. This means that in order to accomodate a sensor measurement that requires an update of all of the landmarks in the map in order to explain the new measurement, the map update will have to happen in an offline manner.

3 Related Work

There are two areas of research which are directly related to the bearings only SLAM problem. The first is of course the community of researchers investigating Simultaneous Localization and Mapping (SLAM), Stochastic Mapping, Concurrent Localization and Mapping, etc. Examples include Durrant-Whyte, Dissanayake, Leonard, Chong and Kleeman, Thrun, Lu and Milios, etc.[3, 4, 5, 6, 7, 8]. The second is the branch of Computer Vision known as Structure from Motion (SFM) and Multi-Frame Structure From Motion (MFSFM). Examples include Hartley, Tomasi, Azarbayejani, Pentland, and many others[9, 10, 11, 12, 13].

3.1 Simultaneous Localization and Mapping

The SLAM problem has been of interest for about a decade. The seminal paper by Smith and Cheeseman on uncertainty representation in robotics presented a framework for a Kalman filter type solution to SLAM [14]. Moutarlier and Chatilla presented the first experiments using a real robot [15], and also presented two different ways of representing relationships between objects, the first they called *relation* and the second *location* [15]. In relation representation, each object has its own reference frame and is related to other objects by uncertain transformations between their reference frames. In location representation, each object uses the same global reference frame, and the object's location in the global coordinate frame is the parameter of interest.

A decade of SLAM research adopted the latter approach, and typically used recursive estimation procedures, almost exclusively Kalman Filters or methods directly related to Kalman Filters. Some methods recursively estimate the entire map and the most current vehicle pose [5, 3, 15], while other methods recursively estimate the history of vehicle poses and the map, which causes a linear growth in the dimension of the state vector but allows smoothing of the trajectory estimate at each step [6]. The implementations often assume that odometry is very accurate, that vehicle motion has Gaussian error, and that bearing and range are measurable.

Recently, there have been several methods which attempt to use the relation approach to SLAM, notably Csorba's Relative Filter [16] which recursively updates relationships between landmark features in a Kalman filtering framework. This approach was extended to enforce global map consistency by Newman with the Geometric Projection Filter [17]. Both methods rely on accurate heading information to keep the problem linear.

The problem of simultaneous localization and mapping has been a topic of interest for over a decade. Most solutions incorporate bearing and range measurements from a mobile robot and recursively estimate the robot position and the locations of landmark features using a Kalman filter or some extension of it. I would like to investigate the use of bearings only measurements, in particular omnidirectional imagery, as a means of sensing for SLAM. Cameras are typically smaller, cheaper, lower power, and more reliable sensors for outdoor mobile robots than laser rangefinders, or millimeter wave radar, and sonar is not applicable to most natural outdoor environments. The use of vision in the the SLAM problem remains relatively untouched within the research community. Monocular vision on a mobile robot calls for a SLAM solution which can deal with bearings only measurements.

Many published SLAM solutions rely on accurate heading information from an external reference such as a compass. However, there are cases in which external heading references do not exist or are unreliable. I would like to develop a SLAM solution that does not rely on external heading measurements, but can incorporate them if available.

I can only speculate as to why bearings only mapping and localization has not been of interest in the past. For one, there are few researchers interested in SLAM for mobile robots who are also interested in computer vision. For another, applications for which SLAM has been identified as useful and actually implemented have largely consisted of AUV's, mining or other heavy machinery, and indoor mobile robots, and these three classes of robots typically use sonar, mmw radar, and sonar, respectively, as sensing modalities which are more conducive to their environments. For small planetary rover type robots, the low power and mass of a camera make it an attractive sensor, and a panoramic camera offers the best sensor field of view in terms of the geometric advantage of being able to observe landmarks anywhere in a 360^0 region around the robot.

3.2 Structure from motion

There is a considerable body of literature on the problem of computing structure from motion, reconstructing an unknown scene from camera images taken when the camera was in unknown positions. Much like the SLAM problem, this seems like an egg and chicken problem. If we knew where the cameras were when the images were taken, then reconstructing the scene would be a matter of triangulation. If we knew the structure of the scene, then we could localize the camera again using triangulation techniques. However, in the general SFM problem there is no a priori knowledge of the scene and no a priori knowledge of the camera positions.

It has been known to photogrammetrists for many decades that for a 3D scene and two views, as few as five corresponding points constrain the structure of the scene and the relative positions of the camera views up to a similitude, e.g. a Euclidean transformation and scale ambiguity. The Euclidean transformation is present because we can not know the absolute positions of the cameras, we can only recover their relative positions. It can be assumed that one of them was at a particular location in order to constrain the solution, i.e. define one camera to be at the origin. The scale ambiguity is introduced due to the fact that scaling the environment and the distance between camera positions by the same factor yields the same images, so from the images one can only hope to recover structure up to this scale factor. In practice, many SFM solutions simply assume that the translation between a pair of cameras is of unit length, or that the distance between two scene points is unity.

For 2D environments and 2D cameras, where the image plane consists of a 1D "image plane", it can be shown that with only two views, no number of point correspondences is sufficient for recovering the scene structure or camera poses. Except in pathological cases, there are an infinite number of solutions which are entirely consistent with the observed data, even under noisy observations. However, with three views, five points will allow Euclidean structure and relative camera pose computation.

For a bearings only sensor such as an omnidirectional camera, there is a nonlinear relationship between a scene point and its projection on the image sphere. A class of SFM solutions exist for orthographic projection which use factorization using SVD [11]. These methods are not applicable to perspective projection or bearings only sensing, however, since the projection is not a linear one. Perspective projection is also nonlinear, and many SFM solutions have used nonlinear optimization techniques, most notably Levenberg-Marquardt, to handle cases where factorization is not applicable [9].

Recursive SFM methods have been reported, following the seminal worka[18] in which an IEKF is used to recursively estimate motion and structure from images. The filter is initialized using a batch processing technique to interpret the first set of frames in the video sequence. The work was later extended to incorporate the estimation of focal length, using a novel camera parameterization to ensure that the problem remain well conditioned[12].

There have also been 3D structure from motion algorithms which compute two-view structure and motion from pairs of images taken sequentially, then recursively filter the resulting structure estimates using an EKF or similar smoothing filter [19, 20]. These methods may be suitable for the current problem if three consecutive views are used.

The general Structure From Motion problem assumes no a priori information about camera motion, which is overly conservative in our case since we may have some motion estimates from odometry, and we may have nonholonomic constraints or other knowledge which can help constrain the solution.

3.3 Bearings only SLAM

The topic of this thesis, bearings only SLAM is closely related to these two topics. The primary difference between the bearings only SLAM problem and the general SLAM problem which is prevalent in the literature is that the range to landmark features from the robot is unknown. The primary difference between bearings only SLAM and the general Structure from Motion problem is that odometry provides some estimate of camera motion between consecutive frames. This information can be used to disambiguate scale and to make otherwise ill-posed problems well-posed, e.g. when only one or two landmarks are visible.

In this thesis, I will bring together and extend ideas from SFM and SLAM to produce a bearings only simultaneous localization and mapping algorithm. I will show that through proper consideration, bearings only measurements and odometry can be used to generate a globally consistent map and vehicle pose estimate. Furthermore, I will develop a method for recursive estimation of the map which has reasonable scaling properties.

4 SLAM as Parameter Estimation

The SLAM problem may be framed as one of parameter estimation. We have an underlying model which we know. The model consists of a 2D plane populated with n_l point features which we call landmarks. Each landmark has two parameters, its (l_x, l_y) coordinates. Assume for now that we have no odometry, and that every landmark is visible from every robot pose. We have a set of n_r images[1] each of which contains n_l bearings, and is taken from a sensor pose which is parameterized by (r_x, r_y, r_θ).

When is this problem solveable? By a counting argument, we can solve for the unknown parameters given the observations when we have as many (or more) constraints as we have unknowns. If the robot takes n_r images and there are n_l landmarks appearing in the images, the landmark positions and sensor poses give us $2n_l + 3n_r$ unknown parameters. If each landmark is visible

[1] in our notation n_r actually refers to the number of distinct robot pose triplets in a trajectory, but for consistency will be used here to denote the number of images

from each image, the measurements give us $n_r n_l$ constraints. Because of the ambiguity in the overall location and orientation of the scene (rigid transformation ambiguity), we can define the position of the camera for the first image to be the origin. To disambiguate scale, we can define the distance between the camera locations for the first two images to be unity. This provides 4 additional constraints. Thus we can solve the problem when

$$\begin{aligned} n_r n_l + 4 &\geq 2n_l + 3n_r \\ n_r n_l - 2n_l - 3n_r + 4 &\geq 0 \\ n_r n_l - 2n_l - 3n_r + 6 - 2 &\geq 0 \\ (n_r - 2)(n_l - 3) &\geq 2 \end{aligned} \tag{9}$$

In other words, we can solve for the structure of the scene and the sensor poses given that we have at least three images and five landmarks, or we have at least four images and four landmarks.

Once we add odometry measurements, we have more constraints to help the optimization. However, one important thing to note from the above analysis is that it is never sufficient to fuse two sets of bearing measurements. If $n_r = 2$ then the problem is ill-posed regardless of how many landmarks there are. This also means that if we fuse two sets of bearing measurements and one odometry measurement, the errors in landmark position estimates are entirely limited by the accuracy of the odometry.

4.1 Batch Estimation

One method by which we can estimate the trajectory of the robot and a map of the environment using all of the available information on odometry and landmark bearings is to do a brute force maximization of the likelihood function using all of the available information. Presume, as before, that the robot can make bearing measurements to landmarks, denoted by $\mathbf{\Phi}_i$, and can also measure its own motion through odometry, obtaining measurements u_i. The bearing measurements and odometry measurements are taken together to form an ensemble of observations $Z^k = \{\mathbf{\Phi}_1 ... \mathbf{\Phi}_k, u_1 ... u_k\}$ which represent everything that the robot knows about its own motion and the relative motion of the landmarks in its coordinate frame. The goal, then, is to find the trajectory $\mathbf{R}$ and map $\mathbf{L}$ which satisfy

$$\{\mathbf{R}, \mathbf{L}\} = argmax_{\mathbf{R},\mathbf{L}} P(\mathbf{R}, \mathbf{L} | Z^k). \tag{10}$$

This requires the optimization of a nonlinear $3n_r + 2n_l$ dimensional function (the likelihood is a function of all robot poses and all landmark coordinates).

4.2 Grid based methods

Grid based methods approximate posterior distributions as piecewise constant functions over the state space and represent the distribution numerically on a grid. In order to propagate the state forward in time, the integration in **??** is carried out numerically.

Thrun's concurrent mapping and localization work uses a type of EM algorithm to probabilistically associate landmark features using a distance metric and maximize the posterior likelihood for the map and trajectory given this soft data association [8]. The procedure operates on all measurements together. Soft assignment can only be done in a batch estimation procedure. Recursive estimation using probabilistic matching would require a set of multiple models to represent samples over sets of assignment probabilities which would grow exponentially through time.

Grid based methods require large amounts of computation and memory, and are not likely to be efficient enough to run on-the-fly.

4.3 Recursive Estimation

The SLAM problem has been considered by many researchers. The seminal paper by Smith, Self, and Cheeseman [14] presented the *stochastic map* framework, which is still widely used. This approach is essentially an Extended Kalman Filter (EKF) where the state variable to be estimated is the robot pose augmented by the positions of all landmarks in the map. For this reason the technique has an alternative name: the Map Augmented Kalman Filter (MAKF).

The class of solutions stemming from the stochastic map can be referred to as Absolute Filters, since these methods attempt to simultaneously and recursively compute the robot pose and landmark locations in a fixed, global coordinate system.

4.3.1 Kalman Filter

Most notable of the absolute filtering techniques is the Kalman Filter and its derivatives. Introduced forty years ago and now ubiquitous in state estimation problems, the Kalman Filter relies on three assumptions which are almost universally violated: independence, linearity, and Gaussian noise. The computational effeciency of Kalman Filters is appealing, however, and many extensions and ad-hoc tricks are used to improve their performance for nonlinear and non-Gaussian problems. Some of these are discussed here.

4.3.2 Extended Kalman Filter (EKF)

The Extended Kalman Filter (EKF) is perhaps the most common of the descendants of the Kalman Filter. With the EKF, the state transition and measurement equations are linearized about a nominal value and then the Kalman Filtering equations are applied. Because the process and observation models are not linear, the Jacobian is not fixed and must be recomputed near the mean value of the state and measurement at each time. The required pdf is still parameterized using only the mean and covariance, which relies on the assumption that it is Gaussian in nature. Linearization may introduce bias or may cause the covariance to be underestimated, and often an EKF must be tuned carefully in order to avoid filter divergence. An ad-hoc introduction of stabilizing noise is commonly used to inflate the covariance of the prediction to ensure that the filter is not overconfident in its prediction which might cause it to unfairly ignore or underemphasize useful observations.

4.3.3 Iterated Extended Kalman Filter (IEKF)

The Iterated Extended Kalman Filter (IEKF) is similar to the EKF except that the state update is done iteratively. With each iteration, the Jacobian of the measurement equation is recomputed to linearize the equations close to the current estimate in order to minimize the effect of linearization on the final solution. The IEKF algorithm is essentially a Newton-Raphson search to maximize the log likelihood of a posterior distribution, where the posterior consists of a Gaussian prior (predictive density) and a Gaussian likelihood (observation density).

The IEKF has been used for solving the SLAM problem using sonar in an indoor environment [21], and has also been used as a means of computing structure from motion, as in the seminal work by Broida, Chandrashekhar, and Chellappa [18]. In the SFM work, the structure and motion are assumed to have a finite number of parameters (i.e. the motion is assumed to have a finite number of derivatives, and the structure is estimated at a finite number of points). The recursive filter is initialized with the result of a batch estimation technique applied to the first several frames and then used to track the structure and motion throughout the rest of the image sequence. When the filter was applied without the initialization using a batch technique, it failed to converge.

4.3.4 Unscented Filter

The Unscented Filter, or Julier-Uhlmann Kalman Filter (JUKF) is an interesting variant of the EKF. The standard EKF uses the state transition equation to compute a predicted value of the state vector using the current state estimate, and then uses the Jacobian of the state transition equation to map the covariance of the state estimate to the covariance of the prediction. Similarly, the mean and covariance of the measurement are projected into state space using the observation equation and its Jacobian, respectively.

In order to deal with nonlinearity, Ullmann[22] proposes to select a set of points which capture the first and second order statistics of the state estimate, project all of these points forward using the nonlinear process model, then compute the sample mean and covariance of the result. This has several advantages over the EKF, including that there is no explicit computation of Jacobians and that the resulting projection has less bias than the linearization in the EKF. The JUKF is essentially equivalent to computing the Jacobian using numerical derivatives, but with step sizes Δx that are comensurate with the actual noise covariance rather than taking $\Delta x \to 0$. By doing so, the resulting prediction covariance is closer to the true covariance under the nonlinear mapping. This method is used with impressive results [5, 21].

4.3.5 Covariance Intersection

Covariance intersection is an attempt to remove the assumption of independence from the Kalman filtering framework. The well-known Kalman Filtering equations are altered by introducing weights to the terms in the update equation. The resulting estimate is guaranteed to be conservative. In other words, the estimated error will always be the same as or larger than the error can possibly be under any unknown correlation between measurements. This technique was developed specifically for dealing with unknown correlations in the SLAM problem and used in a successful implementation using bearing and range sensing [23].

4.3.6 Particle Filters

Sample based methods and Monte Carlo techniques offer a framework for Bayesian estimation without imposing that the posterior distribution be of known parametrized form. Prior distributions are approximated by samples drawn from them, and the samples are propagated, weighted according to their likelihood, and resampled using Importance Resampling [24, 25] to generate a posterior. This allows a filter implementation that is free of the limiting assumptions of previous approaches while still computationally feasible. Sample based methods are not without their own problems, however. Resampling causes the support of the population to decrease, resulting in *sample impoverishment* in which the effectiveness of the particle filter is equivalent to one with far fewer samples. Many heuristics can be employed to mitigate the degree to which sample impoverishment occurs, but the successfulness of these "fixes" seems to be proportional to the computational complexity which they add.

Furthermore, the number of samples needed by a particle filter in order to adequately capture the statistics of the underlying distribution is approximately $O(k^d)$, where d is the number of dimensions in the state space. Because the SLAM problem can potentially require a large number of landmarks or a large number of parameters to describe the map, particle filters become quickly intractable.

Particle filters have been used to localize a robot in a known environment, where samples represent the robot's posterior distribution over pose space [26], where the problem is of low, fixed dimension.

4.3.7 The problem with recursive methods

All of these recursive filtering methods propagate the state estimate by essentially evaluating the predictive state probability density and then marginalizing over the old state estimate. That is

$$P(x_k|Z^{k-1}) = \int P(x_k|x_{k-1})P(x_{k-1}|Z^{k-1}) \tag{11}$$

where the densities are approximated in different ways, for example by mean vector and covariance matrix or by a Monte Carlo sample.

If the old state density $P(x_{k-1}|Z^{k-1})$ is incorrect, there is no recourse to change it. This means that if a new observation becomes available which indicates that the old state estimate was wrong, there is no way to go back and fix it, or to recompute the current state estimate using an estimate for the previous state which incorporates all of the information Z^k.

For linear problems and Gaussian noise, the Kalman filter is optimal, and there is never a need to go back and fix errors in the estimate of previous states $x_{k-1}, x_{k-2}, ...$ if the only desired information is the current state (i.e. where the robot is now, not where is used to be). However, for nonlinear problems, the EKF, Covariance Intersection, Unscented filter, or Particle filter is suboptimal and estimates of $P(x_k|Z^k)$ can be greatly affected by errors in previous state estimates. The strength of recursive filters comes from the computational simplicity in ignoring the errors in old state estimates.

The problem with recursive methods is that they permanently integrate sensor data into a world model which may be inconsistent. Without recourse to update the history of robot poses given observations made at some later time, the recursive approaches lose the ability to guarantee consistent map formation for cases other than linear process and observation models. In order to be able to correct inconsistencies once they are detected, we need to be able to propagate the consistency constraints back through the history of robot poses. Lu and Milios used this argument to justify a batch estimation technique which simply recorded all observations made by the robot during its entire history and building a globally consistent map in one large batch estimation [6].

4.4 Multiple map approaches

There are a number of attempts to break up the world into small local maps, each of which is an absolute map as in the previous section, but with only local information [7, 27]. A full stochastic map is maintained for each such local region, and some less precise or perhaps topological information is stored which relates the local maps. In this way, the robot can leave one local map and enter another, localize precisely within that local environment, and then continue to use existing SLAM methods within the local region.

These techniques require some rules on how the world is tesselated, and must guarantee that if the robot leaves one map and enters a region that was previously mapped that the correct local map is recalled and updated, rather than introducing another new map where a previous map was initialized.

Another heirarchical approach which has been used for indoor environments constructs many small, local, grid based maps, then combines them in one batch step [28]. The maps are constructed by assuming that odometry is perfect over short time intervals and building local occupancy grids. Once a number of submaps are generated, they are combined using an EM optimization step which aligns the maps and generates a global map from them.

5 Relative Filter

Another filtering method has been presented several times under different names, such as *relation based modeling* [15], the *relative filter* [23], and "relation networks" [6], which is not the authors' name for it, but is adopted here, having been used in other references.

The fundamental concept in these approaches is to estimate features of the environment that are invariant to the robot's position, thereby decoupling the uncertainty in the robot's absolute position and the uncertainty in the absolute position of any landmark. In fact, the robot can have no idea where it really is and the relative features can still be estimated as long as the landmarks can be identified. Lu and Milios consider relationships between features which are approximate transforms relating the coordinates of points. Their method computes globally consistent maps by enforcing the constraint that a transformation computed by compounding transformations around a loop of transormations in a relational network must result in a transformation which is the identity operator. They showed results with their relational network representation using horizontal scanning sonar [6]. Csorba's work considers anything invariant to the robot position to be a possible "relative feature," such as the distance between two landmarks or the angle subtended by two landmarks as measured at the location of a third. However, for simplicity their implementation assumes that heading is known and uses the displacement in Δx and Δy as did Lu and Milios. Csorba and Durrant-Whyte showed some results with the relative filter using millimeter wave radar [23]. Their method recursively estimates relationships but does not enforce geometric consistency. Newman later introduced the Geometric Projection Filter (GPF), an extension of Csorba's relative filter which uses geometry to further contrain the absolute map and guarantee that it is geometrically consistent in a manner similar to Lu and Milios [17].

In order to relax the assumption that heading is known, we need to deal with features invariant under any rigid transformation. The fundamental ideas behind the relation based approaches, namely that features invariant to robot position can be estimated and used to reconstruct the absolute map and that overall geometric consistency can be enforced to further reduce error, can be applied in a nonlinear framework.

6 Proposed method

6.1 Integrating recursive and batch techniques

Recursive techniques have convergence problems due to linearization and incorrect approximations of posterior distributions. Batch techniques suffer from heavy computational and memory requirements, since data are all collected up-front, then processed together.

Batch methods and recursive methods can be looked at as two extremes. On the one hand, we have batch estimation algorithms which for N observations will do one estimation step with

N data points. At the other extreme, we have recursive methods which for N data points will do N estimation steps with one data point at a time. In the middle are a class of techniques which perform k estimation steps using N/k data points at each step. Technically, these belong to a class of recursive estimation algorithms which are sometimes referred to as fixed interval smoothing.

The proposed method would integrate batch and recursive methods by implementing a two stage filter. The first stage of the filter takes a batch of odometry and bearing measurements made over one time interval and uses them to optimize a robot trajectory consisting of all of the robot poses from which observations were made and a submap consisting of all landmarks visible in that interval. From the submap, invariant relationships between landmarks are extracted and retained. This is a departure from submap type approaches, in which the resulting submap would be stored intact, and parameterized by one (x, y, θ) triplet which describes how the submap fits into the overall map. By extracting invariant relationships, we eliminate the requirement of having to align maps when they overlap or to ensure that they do not overlap. This stage is designed to run online and have bounded computational and memory requirements.

In the second stage, a global map is maintained and is optimized to best fit the invariant relationships which result from the first stage. This stage of the filter will be designed with efficiency in mind, but its very nature will preclude bounding the computation and memory requirements. The two stages to the filter are explained in more detail below.

6.2 Estimating invariant relationships

After a batch of observations are processed to yield a local submap, relationships between features are computed and the submap is used to initialize new features in the global map. The submap and all observations are then discarded. New relationships are added to a relational graph describing the environment, and new measurements of existing relationships are fused with old estimates to provide more accurate ones. This stage is fast and computationally tractable, and should run online.

Batch processing will be used to compute optimal estimates of vehicle motion and landmark positions over short sequences of measurements. Computing a maximum likelihood trajectory given only odometry measurements is equivalent to finding the trajectory which minimizes the mean square error between actual odometry measurements and odometry measurements which are predicted given the trajectory

$$E_d(\mathbf{R}) = \sum_{i=1}^{n_r} \left(\frac{f_i(\mathbf{R}) - d_i}{\sigma_i^d} \right)^2 . \tag{12}$$

This is equivalent to dead reckoning, which of course does not require a complicated nonlinear optimization to find $\mathbf{R}$. Typically the computation consists of a simple numerical integration which incorporates measurements as they become available. The resulting trajectory estimate is the one which minimizes E_d since each measurement is incorporated into the trajectory estimate exactly, making $E_d = 0$. Figure **??** shows the result of dead reckoning for our example case of a robot with a broken wheel module. The unmodeled wheel failure causes significant bias in the odometry.

Computing a maximum likelihood estimate for the map and trajectory given only the visual tracking output is equivalent to finding a map and trajectory which minimize the error between the real bearing measurements and the bearing measurements predicted given the map and trajectory,

$$E_\phi(\mathbf{L},\mathbf{R}) = \sum_{i=1}^{n_\phi} \left(\frac{h_i(\mathbf{L},\mathbf{R}) - d_i}{\sigma_i^\phi} \right)^2 . \tag{13}$$

This is the familiar Structure from Motion problem when nothing is known about the trajectory (or camera poses). As explained earlier, we require four additional constraints on the problem in order to find a solution, so we set the first robot pose to be at the origin and the second pose to be on the unit circle centered at the first pose. Figure **??** shows the result of optimizing (**??**) for the same example case.

In bearings only SLAM, we wish to incorporate odometry and bearing measurements simultaneously. This can be accomplished by combining the errors with respect to bearing measurements and the errors with respect to odometry measurements into one optimization,

$$E_{d,\phi}(\mathbf{L},\mathbf{R}) = \sum_{i=1}^{n_\phi} \left(\frac{h_i(\mathbf{R},\mathbf{L}) - phi_i}{\sigma_i} \right)^2 + \sum_{i=1}^{n_d} \left(\frac{f_i(\mathbf{R}) - z_i}{\sigma_i} \right)^2 \tag{14}$$

Optimizing (14) will yield a minimum mean squared error (MMSE) estimate for the true robot trajectory $\mathbf{R}$ and map $\mathbf{L}$. With n_u measurements from odometry and n_ϕ bearing measurements there are $n_u + n_\phi$ constraints. With n_r robot poses and n_l landmarks to be estimated, there are $3n_r + 2n_l$ free parameters. We assume that the first robot position is the origin, but unlike the structure from motion problem, there is no scale ambiguity since odometry contains information on robot motion. For Gaussian errors, or for large n_ϕ and n_u, $E_{u,\phi}$ has a χ^2 distribution with $n_u + n_\phi - 3n_r - 2n_l + 3$ degrees of freedom. This means we can use the value of $E_{u,\phi}$ in a chi square test to determine whether the model fits the data adequately, and it may also be possible to use regression diagnostics [29] to detect outlying measurements and exclude them from the loss function (14).

Figure 1 shows histograms of differences between predicted and measured values for landmark bearings, vehicle motion distance, vehicle motion direction, and vehicle rotation. Note that the direction and rotation histograms reveal a significant bias, and that the motion direction histogram reveals one measurement which we can safely label as an outlier.

The size of the batch to be processed can be determined based on memory or processing limitations, but will be required to be well-conditioned. This will be satisfied as long as there are at least three images and at least four landmark points, each of which is observable in every image. Other landmarks that are visible from at least two unique positions may also be included, but landmarks observed only once cannot give any range information and must be discarded.

It may be that sensor measurements are taken at a particular rate and we wish to complete one nonlinear optimization during that cycle. Since the optimization requires a few iterations and each iteration is $O((L+C)^3)$ where L is the number of landmark points and C is the number of camera images in the batch, we can determine how large L and C may become for a given computing configuration and limit the batch size appropriately.

Once a batch of data is processed, we may extract relationships from the estimated submap and trajectory. These relationships will be ones which are invariant to rotation and translation, so that any error in the submap location on the plane do not affect the estimates of relationships. Such relationships will most likely be pairwise distances between landmark points, and angles subtended by two landmarks as measured from the location of a third. These relationships are invariant to the absolute location and orientation of the submap.

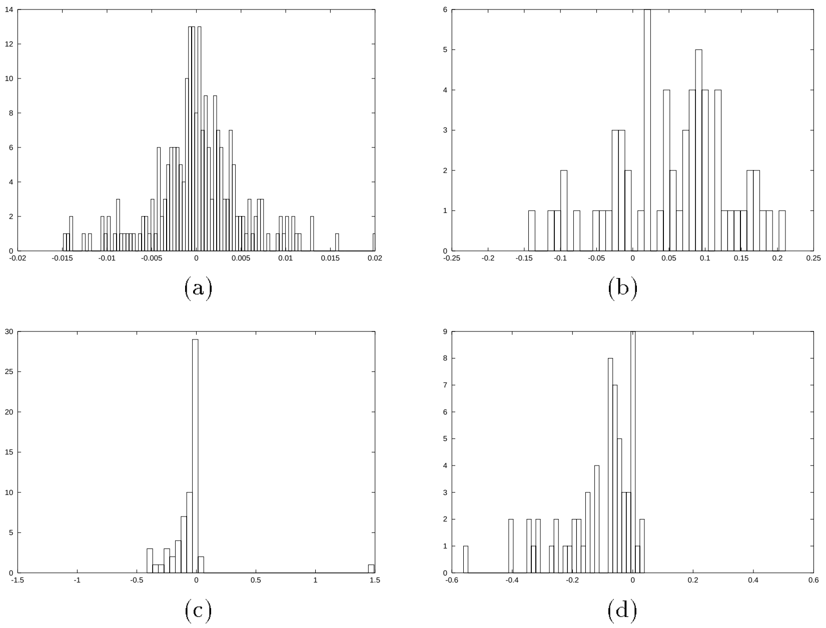

Figure 1: Histogram of errors between predicted and measured values. a) landmark bearings. b) robot motion distance. c) robot motion direction d) robot rotation.

The relationships estimated from a single batch of measurements will of course be highly correlated. In fact, we can expect the covariance matrix to have nonzero entries everywhere. In order to simplify updates in global map estimation using measurements made in separate submap estimates, we marginalize all of the relationships estimated within one submap. The vector of relationships is partitioned into two parts, the first part containing one relationship and the second containing all others. Likewise, the covariance matrix containing information about several relationships is decomposed into four blocks, the upper left block is the proper submatrix containing the covariance of the single relationship, the lower right block is the proper submatrix containing the covariance of all other relationships, and the upper right and lower left contain the cross-covariance between the relationships. Working with a state vector d and the inverse covariance Σ, this gives us

$$D = \begin{pmatrix} d_1 \\ d_{2..N} \end{pmatrix} \tag{15}$$

and

$$\Sigma = \begin{pmatrix} \Sigma_{1,1} & \Sigma_{1,2..N} \\ \Sigma_{1,2..N}^T & \Sigma_{2..N,2..N} \end{pmatrix} \tag{16}$$

Marginalizing (16) gives us (see the appendix for details):

$$\hat{\Sigma} = \begin{pmatrix} \Sigma_{1,1} - \Sigma_{1,2..N}\Sigma_{2..N,2..N}^{-1}\Sigma_{1,2..N}^T & 0 \\ 0 & \Sigma_{2..N,2..N} - \Sigma_{1,2..N}^T\Sigma_{1,1}^{-1}\Sigma_{1,2..N} \end{pmatrix} \tag{17}$$

This marginalization process is repeated until $\hat{\Sigma}$ is diagonal, hence all relationships are independent.

Since the goal is to produce a map and estimate the current robot pose, we can discard all old pose information at the end of one batch estimation. Lu and Milios retained old pose information since the relationship between range scans and the global map were parameterized by the pose from which they were taken[6]. Here we can discard the old pose information since the relationships are parameterized only by relationships between feature points which are invariant to the robot pose.

Note that we do not require that submaps are unique or independent, i.e. the same landmarks may appear in different batch estimations. Estimates of the same relationship coming from different batch estimates will be fused in a later step.

6.3 Finding a globally consistent, accurate map

The second step uses the relational information obtained from the first step to create a globally consistent map. A nonlinear loss function describes how well the current global map fits the observed invariant relationships between landmark features from the first stage

$$E = \sum_{i=1}^{n_D} \left(\frac{g_i(\mathbf{L}) - d_i}{\sigma_i} \right)^2 \tag{18}$$

where d_i is some invariant feature estimated from the first stage, and $g_i(\mathbf{L})$ is a nonlinear measurement equation which predicts what the invariant relationship should be given the current global map $\mathbf{L}$. This loss function is optimized over $\mathbf{L}$ to produce the globally consistent map which most closely fits the observed relationships between features.

This stage must manipulate a data structure which grows potentially without bound, and cannot be expected to run online. However, unlike the Lu and Milios style approach[6], there is an attempt here to simplify the offline step as much as possible by fusing batches of sensor measurements to obtain some map primitives, and by recursively estimating these primitives if they are observed more than once. The result is a minimal number of parameters to be fit and a minimal number of observations of those parameters.

Because of the computational cost, the optimization should be carried out only when necessary. The term "necessary" is intentionally ambiguous: it could be that a precise global metric map is not required for some navigation or other tasks, so the computation can be postponed until required. It may also be that new information about invariant features that are in the map only affect some of the map states and a full optimization can be avoided in favor of one which only updates those states which are affected by the new information.

6.4 Data structure of the proposed filter

Within the invariant filter, there are two important data structures, the relational map in which invariant feature estimates are filtered and stored, and the absolute map, where the invariant feature relationships are used to reconstruct the map in an absolute, global reference frame. The relationship between these two pieces of information is shown in Figure 2.

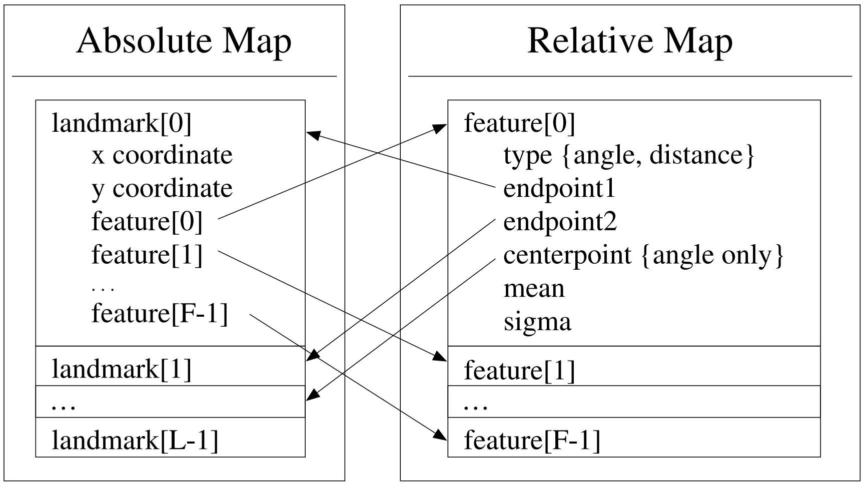

Figure 2: Relationship between Absolute and Relational maps

6.5 Triangulation of the landmark network

Landmarks are assumed to be point features which lie in a plane, or at least the projection of point features in the environment onto a planar surface. The invariant features used in the relative map are pairwise distances between landmarks. This representation leads naturally to a planar triangulation of the landmarks in the absolute map where the edges in the triangulation are distance estimates in the relational map, or where internal angles of triangles in the triangulation are angle estimates in the relational map. A planar triangulation contains on the order of n_L edges, not nearly as many as the $n_L(n_L-1)$ possible edges which can be created.

However, the triangulation is not necessarily best constructed using standard triangulation algorithms. There are several important aspects to the SLAM problem which distinguish it from the usual triangulation problem:

- Landmark positions are not known with precision.
- The landmarks are not all visible at once.
- Edges cannot be created or deleted arbitrarily.

Normally, triangulation is performed on a set of points which are known precisely. In the SLAM problem, we have uncertain knowledge of landmark positions. Second, the landmarks are not all visible at once, so we have to be able to build the triangulation incrementally as landmarks become observed. Third, edges that have been added to the triangulation cannot be removed or inserted arbitrarily if we wish to guarantee independence of the estimates in the relational map. Inserting an edge between two landmarks without making an independent observation about the edge length will create a dependency, i.e. nonzero cross-covariance, between the new edge length and those used to create the estimate.

For this reason, the SLAM algorithm must perform a greedy triangulation, adding new landmarks when visible and linking them to landmarks which are simultaneously visible with the new landmark.

The problem of triangulation of arbitrary 2D point sets has been extensively researched in Computational Geometry. Algorithms exist for incremental construction of Delaunay triangulations, for example. These algorithms generally change the triangulation when a new point is added to the set by deleting edges and adding new ones. However in the context of SLAM using invariant features, we cannot simply remove edges and insert others since we don't have direct measurements of the distances between all landmark pairs. Using existing measurements to infer new relationships would create dependencies between old features and inferred features, unless all edges used in estimating the length of the new edge are then removed. In this context, then, it is advisable to chose edges to insert into the triangulation for a new feature and then leave them in once inserted. This results in a greedy algorithm for triangulation. Furthermore, the resulting triangulation is not guaranteed to be planar, but it is not clear that a planar triangulation is required.

6.6 Data Association

Many feature based mapping strategies model landmarks as indistinguishable point features, and use a generalized Mahalanobis distance criteria for data association [15, 5]. Measurements are grouped together when they fall within each others' high probability density regions, i.e. the Mahalanobis distance falls below some predetermined threshold. Most of these approaches also delete measurements when there are two or more possible landmark correspondences which fall below the threshold (ambiguous match) or when there is no landmark correspondence which falls below the threshold (no match). This is a winner-take-all strategy, or hard assignment method.

Burgard et. al's method uses EM to optimize for a global map given a set of local submaps [28]. The method uses probabilistic data association in the E-step, and maximum likelihood alignment in the M-step. This is a soft assignment method, which makes the technique more robust to data association ambiguities.

Both of these techniques have been widely used in cases where landmarks are indistinguishable. The Mahalanobis distance based winner-take-all strategy has been used in cases where the landmark sensor is a millimeter wave radar or scanning laser rangefinder returning observations of indistinguishable radar or light reflectors. The EM approach has been demonstrated using sonar in indoor environments where matches for planar range measurements can occur along any planar surface, and matches for a corner shaped range measurement can occur at any corner, etc.

In visual SLAM in outdoor environments, however, there is less of a reason to expect measurements to have an ambiguous origin. For a landmark image to have ambiguous origin in an appearance based matching scheme, there would have to be multiple landmarks which look the same. There is a significant body of research devoted to the selection of environmental features based precisely on uniqueness and the ability to distinguish one scene point from all others [30]. Perhaps the harder data association problem in vision is to ensure that two images of the same landmark are not incorrectly interpreted as coming from two distinct landmarks.

However, I would like to investigate soft assignment methods or multiple hypothesis methods for data association in order to provide a solution which is robust to errors in visual feature detection and recognition, and also applicable to sensor modalities other than vision.

I would expect that during one batch estimation step using a sequence of images, visual tracking should be sufficient for data association between frames. However, if the robot leaves an area and some landmarks dissappear from view and later become observable again, there may be some uncertainty as to whether the reobserved landmark is the same as one that is already in the map. I would expect to use hard assignment for processing the batch of new observations, but use soft assignment in relating landmarks in the global map and invariant features measured in old and recent batch estimations.

It is possible that more sophisticated reasoning about data association would simplify the landmark identification problem, essentially reducing it to a landmark detection problem. If a feature in an image passes some criterion then it is labelled a landmark, and there is no search through a memory of the appearance of all previously encountered landmarks to attempt to recognize the new one.

The advantage to using hard assignment for the data association between landmarks in different batch estimations is that if the pairwise distance $d_{i,j}$ is estimated using two different batches of observations, the results may be fused into one estimate for $d_{i,j}$. Each invariant feature therefore appears once in the list of relationships which must be optimized in global map reconstruction. With soft assignment we must maintain both estimates $d_{i,j}$ and $d_{k,l}$ since we must be prepared to accomodate the possibility that $i \neq k$ or $j \neq l$.

7 Proposed Contributions

I plan to develop a nonlinear invariant feature based filtering algorithm that overcomes current limitations for simultaneous localization and mapping problems. In order to advance the capability of autonomous SLAM, I propose to contribute to several aspects of the SLAM solution:

- Dealing explicitly with nonlinear process and observation models.
- Relaxation of assumption that orientation is observable.
- Relaxation of requirement that range is observable.
- Increased robustness to bias and outliers in process or observation.
- Online calibration of motion and measurement models.

Each of these contributions is explained in more detail below.

7.1 Dealing explicitly with nonlinear process and observation models.

The EKF is known to suffer from bias and instability for highly nonlinear problems. Rather than dutifully employing the EKF to the problem of bearings only SLAM, the proposed method will use an iterative nonlinear optimization technique which provides quadratic convergence to the optimal solution in a manner which is robust to ill-conditioned situations. The estimates resulting from the initial batch estimation of the filter will be computed using an ensemble of observations, and will therefore be more justifiably approximated by a normal distribution via use of the Central Limit Theorem. Furthermore, the invariant relationships which result from the initial estimation step will be chosen to be invariant to any rigid transformation rather than only translation, in order to account for the lack of an accurate absolute heading reference.

7.2 Relaxation of assumption that orientation is observable

The assumption that the orientation of the vehicle is observable in a direct, absolute, and independent manner is a strong one in the SLAM problem. It allows a linear treatment of the problem of localization and mapping, reducing the problem of localization to a search over translations in a reference frame with fixed orientation. In situations where the heading cannot be measured with a fixed reference, the uncertainty in orientation introduces a rotation degree of freedom which complicates the SLAM problem. I plan to deal with this problem by using relative features that are invariant to rotation as well as translation. This also complicates the computation of the absolute map from the relative map states, since the relationship between the two is no longer a simple linear one. For a properly constrained absolute map, a system of nonlinear equations must be solved to find the absolute map, and for an overconstrained absolute map, nonlinear optimization must be used to solve for the absolute map with the least reconstruction error.

7.3 Relaxation of requirement that range is observable.

The proposed method will be able to produce globally consistent maps using a bearings only sensor, such as a monocular vision system. Most SLAM work to date has been implemented for sonar, laser rangefinder, or millimeter wave radar applications. Borrowing on results from Structure from Motion, I will develop a system capable of producing a metric map of an outdoor environment and localizing a mobile robot within the map, and that does not require a range sensor.

7.4 Increased robustness to bias and outliers in process and observation.

Catastrophic errors in process or observation can include things such as a stuck or slipping wheel or a spurious bearing measurement from landmark tracking. A successful SLAM algorithm must be robust to these errors. Assuming that errors are Gaussian can cause high sensitivity to such outlying observations.

One method for increased robustness to outliers is to use regression diagnostics[29]. In the LM optimization over small batches of observations, after the optimum χ^2 fit is obtained, the residual is computed for each data point. Those for which the best fit value exceeds a threshold, say 3σ, are thrown away, and the optimization is recomputed without these spurious measurements. If the errors are truly Gaussian, this thresholding will only throw away about 0.3% of the data, and if the outliers are truly outliers then there is no reason to allow them to influence the final estimate.

Another common robust estimation technique is the use of M-estimators. With MMSE as in equation (18), the derivative of the loss function goes to infinity as errors go to infinity, and outlying measurements can have an unbounded effect on the overall estimate. M-estimators replace the square error loss function with one that has a derivative which tends to zero as errors become large, or at least don't grow as quickly as the MMSE loss function. The advantage of M-estimators is that they have a stronger statistical foundation for measurements which may be stochastic with most of the probability mass near the expected parameter value, but are not well modeled by a Normal distribution. The log-loss function for the Cauchy distribution and some other distributions with larger tails than the Normal distribution are M-estimators.

M-estimators are not as easily accomodated in Levenberg-Marquardt optimization as regression diagnostics. Furthermore, there may be little justification for expecting measurements to follow a Cauchy distribution or other heavy tailed unimodal distribution, in which case M-estimators are no better justified than regression diagnostics or MMSE without any robustness considerations.

7.5 Online calibration of motion and measurement models.

Regression diagnostics can also be used to find bias in process or observation models. If every estimate of vehicle rotation that is made via odometry is incorrect by the same scale factor, that may indicate that the incorrect wheelbase is being used in the odometry. By adding the appropriate parameters into the batch estimation, the robot can self calibrate its odometry online while recovering the structure of the local environment and the trajectory over which it travelled. Bias in distance measurements is not as easy to detect due to the inherent scale ambiguity in structure from motion.

8 Experimental Plan and Schedule

Fall 99:

- Estimating local maps and trajectories from short traverses.
- Recursive filtering of existing invariant features.
- Solution of global map from invariant features.

Spring - Summer 00:

- Continue work on solution of global map from invariant features.
- Selection and marginalization of invariant features from local map.
- Accomodating uncertainty in data association.
- Testing in simulation and using Pioneer-2 and/or Bullwinkle.

Fall 00:

- Testing using Marsokhod at NASA Ames.
- Recovery of bias in motion model.
- Compare the proposed method against EKF based system.

Spring 01:

- Demonstrate fully working system.
- Data collection and comparison between proposed method and existing methods.

Summer 01 - Fall 01:

- Write thesis
- Defend

There are several different platforms available for demonstrations, each of which is very different from the others although all are land vehicles. The first of these testbeds is an ActiveMedia Pioneer 2 robot. The second is an RWI ATRV robot named Bullwinkle. The third is the Marsokhod rover at NASA Ames. The fourth is the Nomad robot at CMU. The fifth is "Urbie", the mobile robot designed for DARPA's Tactical Mobile Robot (TMR) demonstration. This is not to say that all of these platforms will be used.

For each robot, a motion model will be developed which maps odometric measurements to translations and rotations of the robot.

For initial experiments, the robot will take images with a small panoramic camera within an artificial environment with easily identified, distinct and unique landmarks in surveyed positions. The appearance of the landmarks will allow simple correspondence of landmark features and the image. The knowledge of the landmark positions will allow comparison of the maps generated to ground truth.

For later experiments, outdoor mobile robots with differential GPS will allow the collection of accurate robot position as well as surveyed landmark positions.

Estimates of the structure of the environment (the map) and the history of robot motion (the trajectory) will be compared to ground truth when available. More important will be a comparison to solutions found using a batch estimation technique and an EKF in order to quantify the differences in computational resource requirements and accuracy of the solution. The expectation is that the proposed method should be much more accurate than recursive methods for scenarios where long times, large distances, unmodeled biases, and outlying measurements play a role, and that the proposed method should be more computationally efficient than current batch estimation procedures.

9 Practical online implementation considerations

For online applications it will be important to characterize the computational complexity of the invariant filtering scheme proposed, as well as the memory requirements. The complexity of two separate steps in the procedure are of interest, as are the memory requirements of two data structures. In the discussion below, let n_l be the number of landmarks, and n_D be the number of pairwise distance estimates in the relative map. Note that in a reasonable triangulation of the absolute map, n_D and n_l will have an approximately linear relationship, i.e. $n_D \approx 2n_l$.

9.1 Computational Complexity

First, the update of the relative map information is computationally trivial if the data association is known. Given an existing (prior) estimate for the pairwise distance between two landmarks, and a new measurement given from a new set of observations, the computation of an updated estimate of the landmark pair distance is $O(1)$. In the worst case, we may be able to see all of the landmarks in the map, which would require $O(n_l)$ for the update.

Because it requires the optimization of a set of nonlinear equations, the update of the absolute map from the relative map is an iterative process, so the important factors to note are the complexity of an iteration and the rate of convergence. Using the Levenberg-Marquardt technique provides quadratic convergence near the solution, so when it converges it converges quickly. The complexity of each LM step is $O(n_l^3)$ where n_l is the number of landmarks in the map. This is because LM involves a matrix inversion.

It is possible that the iteration may be sped up by updating only the landmarks whose constraints have significantly changed, or whose pairwise distances have changed significantly during the previous iterations. For example, if a new distance $d_{i,j}$ is computed, then perhaps landmarks i and j will need to move significantly; however, their immediate neighbors will move less, and their neighbors' neighbors will move still less, and eventually we may be able to ignore the local effect when considering landmarks that are far enough away from the newly updated LPD.

There can be a significant performance improvement over direct application of Levenberg-Marquardt by considering the block structure of the Hessian matrix in the optimization.

9.2 Memory Requirement

For each landmark the absolute map requires memory for the 2D position of the landmark in global coordinates (x, y), and a list of applicable relative features from the relative map. This means that the absolute map requires $O(n_l)$ memory.

For each landmark pair distance the relative map requires memory for the mean and covariance of the estimate, and for the indeces of the pair of landmarks. The relative map therefore requires $O(n_D)$ memory.

10 Convergence of the absolute map optimization

How do we know for sure that the absolute map will converge to the right answer? We must start the optimization with the landmark near the correct answer in order to find the correct answer. Can we guarantee that this will happen?

Appendix

Suppose we have a vector $x = x^*$ which is a solution to a least squares optimization problem with loss function

$$J(x) = (x - x^*)^T C(x - x^*) \tag{19}$$

where the solution has covariance matrix Σ with inverse C, which is the *normal matrix*. We would like to partition the parameter vector into two parts, $x = |x_1 x_2|$ and partition the normal matrix accordingly,

$$C = \begin{vmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{vmatrix} \tag{20}$$

so that the loss function is

$$J(x) = (x_1-x_1^*)^T C_{11}(x_1-x_1^*)+(x_1-x_1^*)^T C_{12}(x_2-x_2^*)+(x_2-x_2^*)^T C_{21}(x_1-x_1^*)+(x_2-x_2^*)^T C_{22}(x_2-x_2^*) \tag{21}$$

Note that because C is symmetric, $C_{21} = C_{12}^T$. The uncertainty on the solution x^* can be visualized as an ellipsoid defined by $J(x) = c$. We would like to know the marginal uncertainty for x_1 independent of the uncertainty in x_2. Unless $C_{12} = 0$, the marginal uncertainty is not simply given by C_{11}.

We start by finding the values of $(x_1 - x_1^*)$ corresponding to points on the ellipsoid where the tangent plane is parallel to the x_2 subspace

$$\frac{\partial}{\partial x_2} J = -(x_1 - x_1^*)^T C_{12} - (x_2 - x_2^*)^T C_{22} = 0. \tag{22}$$

or

$$(x_2 - x_2^*) = -(C_{22}^{-1}) C_{21} (x_1 - x_1^*) \tag{23}$$

If we substitute these points into the equation for the ellipsoid, we find

$$J(x) = (x_1 - x_1^*)^T C_{11} (x_1 - x_1^*) - (x_1 - x_1^*)^T C_{12} (C_{22}^{-1}) C_{21} (x_1 - x_1^*). \tag{24}$$

In other words, the uncertainty ellipse may be described by

$$(x_1 - x_1^*)^T C^{11} (x_1 - x_1^*) = c \tag{25}$$

where

$$C^{11} = C_{11} - C_{12}(C_{22}^{-1}) C_{21}. \tag{26}$$

A similar derivation finds the marginal uncertainty for x_2.

References

[1] C. H. Chen, L. F. Pau, P.S.P. Wang, and (Eds). *Handbook of Pattern Recognition and Computer Vision*. World Scientific Pub Co., 1993.

[2] E. Krotkov. Mobile robot localization from a single image. In *Proceedings IEEE ICRA*, pages 978–983, 1989.

[3] J. J. Leonard and H. F Durrant-Whyte. Simultaneous map building and localization for an autonomous mobile robot. In *IEEE/RSJ International Workshop on Intelligent Robots and Systems IROS '91*, pages 1442–1447, 1991.

[4] H. Feder, J. Leonard, and C. Smith. Adaptive mobile robot navigation and mapping. *International Journal of Robotics Research*, 18(7):650–668, July 1999.

[5] M.W.M.G. Dissanayake and et al. A solution to the simultaneous localization and map building (slam) problem. Technical Report ACFR-TR-01-99, Australian Centre for Field Robotics, University of Sydney, Australia, 1999.

[6] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4(4):333–349, 1997.

[7] K. Chong and L. Kleeman. Large scale sonarray mapping using multiple connected local maps. In *International Conference on Field and Service Robotics*, pages 278–285, 1997.

[8] S. Thrun, W. Burgard, and D. Fox. A probabilistic approach to concurrent mapping and localization for mobile robots. *Machine Learning and Autonomous Robots*, 31(5):1–25, 1998.

[9] Richard I. Hartley. Euclidean reconstruction from uncalibrated views. In Zisserman Mundy and Forsyth, editors, *Applications of Invariance in Computer Vision*, pages 237–256. Springer Verlag, 1994.

[10] R. I. Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997.

[11] C. Tomasi and T. Kanade. Shape and motion from image streams: a factorization method. Technical Report CMU-CS-92-104, Carnegie Mellon University, 1992.

[12] Ali Azarbayejani and Alex P. Pentland. Recursive estimation of motion, structure and focal length. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):562–575, 1995.

[13] C. J. Taylor, D. J. Kriegman, and P. Anandan. Structure and motion in two dimensions from multiple images: A least squares approach. In *Proceedings of the IEEE Workshop on Visual Motion*, pages 242–248, 1991.

[14] Randall Smith, Matthew Self, and Peter Cheeseman. *Estimating Uncertain Spatial Relationships in Robotics*, chapter 3, pages 167–193. Springer-Verlag, 1990.

[15] P. Moutarlier and R. Chatila. Stochastic multisensory data fusion for mobile robot location and environment modelling. In *5th Int. Symposium on Robotics Research*, 1989.

[16] Michael Csorba. *Simultaneous Localization and Map Building*. PhD thesis, Department of Engineering Science, University of Oxford, 1997.

[17] Paul Michael Newman. *On the Structure and Solution of the Simultaneous Localization and Mapping Problem*. PhD thesis, Australian Centre for Field Robotics, University of Sydney, 1999.

[18] T. J. Broida, S. Chandrashekhar, and R. Chellappa. Recursive 3-d motion estimation from a monocular image sequence. *IEEE Trans. on Aerospace and Electronic Systems*, 26(4):639–656, 1990.

[19] J. Oliensis and J. Inigo Thomas. Incorporating motion error in multi-frame structure from motion. In *IEEE Workshop on Visual Motion*, pages 8–13, 1991.

[20] S. Soatto, P. Perona, R. Frezza, and G. Picci. Recursive motion and structure estimation with complete error characterization. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 428–433, 1993.

[21] K.S. Chong and L. Kleeman. Sonar based map building for a mobile robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1700–5, 1997.

[22] Jeffrey K. Uhlmann. *Dynamic Map Building and Localization: New Theoretical Foundations*. PhD thesis, Department of Engineering Science, University of Oxford, 1995.

[23] M. Csorba and H. Durrant-Whyte. A new approach to map building using relative position estimates. *SPIE Vol. 3087*, pages 115–125, 1997.

[24] A.F.M. Smith and A. E. Gelfand. Bayesian statistics without tears: A sampling-resampling perspective. *The American Statistician*, 46(2):84–88, 1992.

[25] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceedings F*, 140(2):107–113, April 1993.

[26] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *Proc. ICRA '99*, 1999.

[27] John J. Leonard and Hans Jabob S. Feder. Decoupled stochastic mapping. Technical Report 99-1, MIT Marine Robotics Laboratory, Cambridge, MA 02139, USA, 1999.

[28] W. Burgard, D. Fox, H. Jans, C. Matenar, and S. Thrun. Sonar-based mapping with mobile robots using EM. In *Proc. of the International Conference on Machine Learning*, 1999.

[29] Zhengyou Zhang. Parameter estimation techniques: A tutorial with application to conic fitting. Technical Report No. 2676, INRIA, 1995.

[30] E. Yeh and D.J. Kriegman. Toward selecting and recognizing natural landmarks. In *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 47–53, 1995.