Modeling a dynamic environment using a Bayesian multiple hypothesis approach

Ingemar J. Cox

NEC Research Institute, 4 Independence Way, Princeton, NJ 08540, USA

John J. Leonard

MIT Sea Grant College Program, E38-308A, 292 Main Street, Cambridge, MA 02139, USA

Received January 1992
Revised November 1992

Abstract


Dynamic world modeling requires the integration of multiple sensor observations obtained from multiple vehicle locations at different times. A crucial problem in this interpretation task is the presence of uncertainty in the origins of measurements (data association or correspondence uncertainty) as well as in the values of measurements (noise uncertainty). Almost all previous work in robotics has not distinguished between these two very different forms of uncertainty. In this paper we propose to model the uncertainty due to noise, e.g. the error in an object's position, by conventional covariance matrices. To represent the data association uncertainty, an hypothesis tree is constructed, the branches at any node representing different possible assignments of measurements to features. A rigorous Bayesian data association framework is then introduced that allows the probability of each hypothesis to be calculated. These probabilities can be used to guide an intelligent pruning strategy.

The multiple hypothesis tree allows decisions concerning the assignment of measurements to be postponed. Instead, many different hypotheses are considered. Expected observations are predicted for each hypothesis and these are compared with actual measurements. Hypotheses that have their predictions supported by measurements increase in probability compared with hypotheses whose predictions are unsupported. By looking ahead" two or three time steps and examining the probabilities at the leaves of the tree, very accurate assignment decisions can be made.

Correspondence to: I.J. Cox, NEC Research Institute, 4 Independence Way, Princeton, NJ 08540, USA. Telephone: (609) 520-1555. Fax: (609) 951-2481. E-mail: ingemar@research.nj.nec.com.
For dynamic world modeling, the approach results in multiple world models at a
given time step, each one representing a possible interpretation of all past and current
measurements and each having an associated probability. In addition, each geometric
feature has an associated covariance that models the uncertainty due to noise.
This framework is independent of the sensing modality, being applicable to most
temporal data association problems. It is therefore appropriate for the broad class of
vision, acoustic and range sensors currently used on existing mobile robots. Preliminary
results using ultrasonic range data demonstrate the feasibility of the approach.

1. Introduction

In this paper, we examine the data association problem as it applies to
a mobile robot building and maintaining a map of its environment. Au-
tonomous vehicles need to automatically build and maintain a map of their
environment. A coherent interpretation of a robot's external environment
is necessary in order for it to reason about such things as its position, mo-
tion planning, and obstacle avoidance. Maintenance of the map is needed
because even if a map is provided a priori, few robot environments are com-
pletely static. Consequently, as the robot's environment alters, so must its
world model. Map building is a dynamic process that involves providing an
interpretation of observed sensor information in terms of physical features
in the environment. The prediction of expected events relies on the fact
that a correct interpretation has been found for geometric features that are
repeatably observed. Further, geometric features that are initiated to explain
unexpected events are an attempt to develop such an interpretation over
time. Thus the map is built up as new events are observed and explained,
and is refined by reobserving events that have been correctly interpreted.

Map building and maintenance is problematic because of the difficulty
in interpreting sensor observations. Deciding where measurements originate
from is surprisingly difficult, because the input data is noisy and ambigu-
ous. There may be false and/or missing measurements and the number of
perceptually relevant groupings is unknown before hand. Further, in dy-
namic environments, the number of perceptually relevant groupings changes
with time, some groups unexpectedly terminating and others being cre-
ated. Section 2 discusses earlier work on data association in the context of
constructing a map (or world model) for a mobile robot.

Dynamic world modeling for a mobile robot can be cast as a (target)
tracking problem [23]. Usually one is concerned with a stationary observer
and moving targets, but one can just as well have a moving observer and
stationary targets (a mobile robot and a static environment) or both a
moving observer and stationary and moving targets (a mobile robot in a
dynamic environment) [17]. In Section 3 we illustrate how the multiple
hypothesis algorithm first proposed by Reid [32] in the context of multi-
target tracking can be used to build and maintain a dynamic model of
a robot's environment. This technique provides a powerful formalism for addressing the correspondence problem and ambiguities that may arise at any time in the interpretation of sensor measurements. The appeal of the technique is that it can refrain from making irreversible decisions too early in the interpretation process.

Section 4 discusses the implementation of the algorithm. It describes the choice of geometric features used to describe the environment and how these features can be represented efficiently using Kalman filtering. Techniques for containing the growth of the hypothesis tree are also described, based on a combination of screening prior to hypothesis generation and pruning after hypothesis generation. Section 5 describes experimental results on both real and synthetic data. The synthetic example serves to illustrate the functioning of the hypothesis tree, demonstrating, for example, the ability to learn a feature and then later to forget this feature when the environment changes. The real data is from an ultrasonic rangefinder aboard a mobile robot. The example demonstrates the ability of the vehicle to learn a very precise geometric map, even though the individual sensor readings can be notoriously difficult to evaluate.

Finally, Section 6 concludes with a discussion of the advantages and disadvantages of the multiple hypothesis approach and some areas of possible future work.

2. Earlier work

Almost all previous attempts at map building for autonomous robot vehicles have not discriminated between spatial uncertainty due to noise and uncertainty in the origin of sensor measurements [10]. Significant progress has been made towards dealing with noisy measurements and the corresponding spatial uncertainty that is introduced. Impressive results are reported by Ayache and Faugeras [3] for a mobile vehicle using trinocular stereo to determine depth. Kriegman et al. [18] also describe how a map can be built using binocular stereo. Crowley [12] describes a similar approach using ultrasonic sensing. Common to all these approaches is the use of the Kalman filter to model and propagate uncertainty in both the position of the robot and the geometric features of the world model using covariance matrices. The Kalman filter is a powerful tool for dealing with noise, but is of little help in modeling the uncertainty in the origin of sensor measurements. The experimental results presented by these groups are all for static scenes, i.e. the environment is unchanging. It is unlikely that these approaches would be successful in a dynamic environment without extension, perhaps along the lines of the work described here.
In an earlier paper [22], we examined the problem of constructing and maintaining a map, i.e. a world model, from which the vehicle's position could be accurately computed. This paper addressed the central issue of uncertainty management, pointing out the need for two independent measures of uncertainty. This proposal is similar to earlier work by Crowley [11]. However, the data association problem was not explicitly addressed. In this paper, we expand on our earlier work in two ways. First, we provide the capability to model dynamic as well as static features in our model. Second, we replace the heuristic-based "credibility" measure with a rigorous Bayesian formalism that estimates the probability of a hypothesized world model.

Early work by Ayache and Faugeras [3] reduced the data association problem to the much simpler one of validating a measurement to an existing geometric feature. A measurement is validated to a feature by examining the statistical Mahalanobis distance [16] between the predicted and actual measurements. If this distance is less than some constant determined by chi-squared distribution tables, then the measurement (very) probably originated from the feature. If only one unique measurement is validated to each feature then the data association problem vanishes. However, if a measurement is validated to two or more features or multiple measurements are validated to a single feature, then the correspondence problem must be addressed. And, in fact, this is commonly the case.

A simple and common solution to the data association problem is the "nearest neighbor" algorithm which assigns to each track the measurement (statistically) closest to it (for examples see Singer [34], Crowley et al. [13], and Deriche and Faugeras [14]). Unfortunately, this approach may lead to unexpected results as noted by Bar-Shalom and Fortmann [4] and Zhang and Faugeras [41] and more sophisticated schemes must be used.

The target tracking community has extensively investigated correspondence problems. Most of their solutions involve some form of Bayesian probabilistic association. Bar-Shalom and Fortmann [4] provide an excellent overview of the topic as applied to surveillance and military target tracking. These algorithms are increasingly finding their way into the robotics and vision communities. Representative techniques can be broadly classified into two categories based on their computational complexity: finite fixed versus exponential time and memory requirements. Another important consideration is whether the number of targets is known a priori or not.

The suboptimal probabilistic data association filter (PDAF) [5] is restricted to the case of tracking only a single target (geometric feature). It provides a state estimate based on the weighted sum of all the measurements in the validation region of the track. Obviously, this algorithm is suboptimal because measurements that should not have been associated with the track
are used to update its state estimate. Consequently, there is a risk that the state estimate can diverge from the actual state under certain circumstances. The advantage is that it requires fixed and finite computational resources. The joint probabilistic data association filter (JPDAF) extends the PDAF to a fixed known number of targets. This algorithm has been used by Chang and Aggarwal [7] in the context of computer vision and structure from motion. One disadvantage of the PDAF is that track initiation is not directly incorporated in the algorithm. More importantly, however, we believe that dynamic world modeling requires an approach that can evaluate and resolve alternative hypotheses regarding the origins of measurements.

Zhang and Faugeras [41] use a track splitting filter [35], but restrict the number of simultaneous measurements validated to a track to the two closest measurements, presumably to constrain the potentially exponential growth of the hypothesized track trees. The track splitting filter is very close in spirit to the multiple hypothesis algorithm proposed here. The primary disadvantages of track splitting in comparison with the Bayesian multiple hypothesis approach are:

1. There is no explicit modeling of the probability of detection of a track, i.e. it is implicitly assumed that the track is seen at every time interval.
2. Although new tracks may be initialized by various schemes, the algorithm itself does not explicitly model this function.
3. The algorithm does not yield a probability that a sequence is correct.
4. Most seriously, potential tracks are considered on an individual basis.

This ignores "competition" for the same measurement between several tracks, and therefore implicitly allows tracks to share measurements. This is usually physically unrealistic. More realistic is the "no split–no merge" scenario which assumes that a measurement originated from a single source and that a track has only one associated measurement per time interval. The track splitting algorithm can be modified to accommodate this assumption. The resulting joint likelihood method [27] reduces to an integer programming problem. However, this is a batch rather than recursive procedure. The severity of these disadvantages is, however, hard to assess, and cannot be evaluated without reference to the expected type of environment.

The problem of map building and maintenance may also be viewed as an unsupervised learning problem [9]. Lumelsky et al. [26] are concerned with determining a path for the robot, the traversing of which would guarantee that the entire environment is viewed. Navigation is not an issue here, but rather the development of a sensing strategy for "terrain acquisition". Rivest and Schapire [33] examined the problem of unsupervised learning for deterministic environments that can be adequately described by a finite state automaton (FSA). While their results are impressive, it is not apparent
how their approach can be extended beyond the domain of FSAs to the application environment of an actual robot. Mozer and Bachrach [30] have presented an alternative connectionist implementation to address the same class of environments addressed by Rivest and Schapire. They report potentially faster performance at the expense of sometimes unpredictable behavior, such as lack of network convergence for some sets of random initial weights. Extension of either of these approaches to real-world domains is made difficult by several factors:

(1) any actions the robot might perform are inherently uncertain, as are the observations its sensors provide;
(2) because typical application environments are not FSAs, the robot needs to learn a representation composed of geometric constructs;
(3) further, in dynamic environments, these (geometric) concepts are time-varying. Kuh et al. [20] have suggested an extension of computational learning theory [40] for situations in which concepts vary over time and addressed associated complexity issues.

Finally, we emphasize that this paper assumes that the vehicle's position is precisely known. This is a limited form of the general autonomous navigation problem, which requires that world modeling and vehicle localization be undertaken simultaneously [24]. With precise knowledge of the vehicle's position and motion, it is reasonable to assume that measurement errors between features are uncorrelated. This allows each geometric feature's state, i.e. position and velocity, to be independently estimated. In contrast, uncertainty in the vehicle's position leads to correlated measurement errors between features. As a consequence, any optimum estimation algorithm cannot be decoupled. Some notable previous research in mobile robot map making [29,36], while not explicitly addressing the data association problem, has incorporated motion and noise uncertainty. Smith, Self, and Cheeseman presented a fully-coupled, Kalman filter-based stochastic map of spatial relationships that addressed vehicle motion error and sensor noise, but not data association uncertainty [36]. Moutardier and Chatila implemented a framework similar to the stochastic map using real laser rangefinder data, relying on the Mahalanobis distance and a nearest neighbor strategy to overcome the data association issue for their experiments [29]. Serious computational complexity issues arise if one wants to join the stochastic map to the multiple hypothesis framework presented here.

3. Multiple hypothesis framework for dynamic world modeling

In this section we describe the multiple hypothesis algorithm of Reid [32] and demonstrate how such a framework is applicable to the modeling of
dynamic environments for autonomous robot vehicles.

We envision a world model consisting of simple two- or three-dimensional geometric primitives, such as line segments and points, together with associated information describing the temporal dynamics of each feature, e.g. velocity and acceleration. A track is defined to be a sequence of measurements that are assumed to originate from the same geometric feature. A track may be static, representing a stationary geometric feature such as a wall or corner, or dynamic, representing a moving geometric feature such as a person. A global hypothesis is one possible world model of the robot's environment together with an associated probability of the hypothesis given all past and current measurements. An hypothesis or world model is a collection of tracks that partition the measurement data into sets, each set representing measurements assumed to originate from the same geometric feature. The tracks modeling geometric features in a world model have spatial (position) and dynamic (velocity) attributes with corresponding uncertainties represented by traditional covariances. The grouping together of geometric features into higher level objects is not currently considered. Fig. 1 shows the structure of our current proposal.

At the beginning of each cycle of the algorithm, we have a set of hypotheses, each hypothesis denoting a different set of assignments of measurements to features. Different sets of assignments expect to see different sets of measurements. Thus, each hypothesis predicts a set of expected sensor measurements and these are compared with actual measurements recorded from sensors on board the vehicle. These comparisons are represented in the form of an hypothesis matrix, defined in Section 3.1, which concisely models the ambiguities present in assigning measurements to features. Each hypothesis in the tree has an associated matrix from which it is possible to generate a set of children (see Section 3.1), each child representing one possible interpretation of the new set of measurements. Finally, in order to contain the growth of the tree, it is necessary to prune unlikely branches (see Section 4). However, before this can be accomplished, we need to evaluate the likelihood of each hypothesis. Section 3.2 provides the mathematical framework for estimating the probability of each leaf in the tree.

3.1. Hypothesis generation

In our earlier work [22] a credibility measure was assigned to an individual feature in the single world model of the robot. In contrast, the algorithm described here generates multiple world models, i.e. multiple interpretation hypotheses, and assigns a probability to each. Within each world model, we do not explicitly represent the probability of a model feature. However,

\[1\text{In fact, in the current context, the terms geometric feature and object are interchangeable.}\]
Fig. 1. A multiple hypothesis framework for dynamic world modeling.

probabilities for individual model features can be determined. One method for doing so is to use the sum of the probabilities of the hypotheses which include the feature [28].

At time $k$ we have a set of association hypotheses (world models) $\Omega^k$ obtained from the set of hypotheses $\Omega^{k-1}$ at time $k - 1$ and the latest set of measurements

$$Z(k) \triangleq \{z_i(k)\}_{i=1}^{m_k},$$

where $m_k$ is the number of measurements $z_i(k)$ at time interval $k$. An association hypothesis is one possible interpretation of the vehicle’s sensor observations. Since at any one time it is very likely that there will be some ambiguity to the interpretation of sensor observations, there can be many association hypotheses. The multiple hypothesis filter allows the assignment of a probability value to each of these hypotheses.
More formally, an association hypothesis groups the history of measurements into partitions such that $Z^{k,t} \triangleq \{z_{i_{1},(1)}, z_{i_{2},(2)}, \ldots, z_{i_{n}(k)}\}$ is the set of all measurements originating from geometric feature $t$. The individual measurements $z_{i}(k)$ might be obtained from a variety of sensors including video cameras and ultrasonic or optical rangefinders.

New hypotheses are formed by associating each measurement as

1. belonging to a previously known geometric feature.

That is, the measurement $z_{i}(k)$ is determined to correspond to a known geometric feature in our current world model. Determination of correspondences is achieved using the validation gate procedure outlined in [22].

If a measurement falls outside of the validation gates of all known/modelled geometric features, then it is either

2. a new geometric feature

or

3. a false alarm.

In addition, for geometric features that are not assigned measurements, we also have the possibility of

4. deletion of geometric feature.

This situation may arise when a learned feature, such as a stationary desk, is moved to a new position.

We define a particular global hypothesis at time $k$ by $\Theta^{k}_{m}$. Let $\Theta_{l(m)}^{k-1}$ denote the parent hypothesis from which $\Theta^{k}_{m}$ is derived, and $\theta_{m}(k)$ denote the event that indicates the specific status of all geometric features postulated by $\Theta_{l(m)}^{k-1}$ at time $k$ and the specific origin of all measurements received at time $k$.

Let $T$ denote the total number of geometric features postulated by the parent hypothesis $\Theta_{l(m)}^{k-1}$. Next, we define the event $\theta_{m}(k)$ based on the current measurements to consist of $\tau$ measurements from known geometric features, $\nu$ measurements from new geometric features, $\phi$ false alarms, and $\chi$ deleted (or obsolete) geometric features from the parent hypothesis.

---

2 The reader is directed to Uhlmann [8,37-39] for a description of efficient validation gate algorithms.

3 In general, there may be a problem initiating a new geometric feature from a single measurement. Section 4.1.1 discusses this in more detail.
For all the current measurements, $z_i(k), i = 1, \ldots, m_k$, we define the indicator variables

$$\tau_i \triangleq \begin{cases} 1, & z_i(k) \text{ came from a known geometric feature,} \\ 0, & \text{otherwise;} \end{cases}$$  \hspace{1cm} (2)$$

$$\nu_i \triangleq \begin{cases} 1, & z_i(k) \text{ is a new geometric feature,} \\ 0, & \text{otherwise;} \end{cases}$$  \hspace{1cm} (3)$$

$$\delta_i \triangleq \begin{cases} 1, & \text{if geometric feature } t \text{ (in } \Theta^k_{t(m)}) \text{ is detected at time } k, \\ 0, & \text{otherwise;} \end{cases}$$  \hspace{1cm} (4)$$

$$\chi_i \triangleq \begin{cases} 1, & \text{if geometric feature } t \text{ (in } \Theta^k_{t(m)}) \text{ is deleted at time } k, \\ 0, & \text{otherwise.} \end{cases}$$  \hspace{1cm} (5)$$

These variables will be used in the development of the probability calculations below.

We can construct a set of events $\theta_m(k)$ by first creating a hypothesis matrix in which known geometric features are represented by the columns of the matrix and the current measurements by the rows. A nonzero element at matrix position $c_{i,j}$ denotes that measurement $z_i(k)$ is contained in the validation region of geometric feature $t$. In addition to the $T$ known geometric features in the world model the hypothesis matrix has appended to it a column 0 denoting false alarms and $m$ columns $N + 1, \ldots, N + m$, each denoting one of $m$ possible new geometric features. Fig. 2 depicts a situation in which we have two known geometric features ($T_1$ and $T_2$) and three new measurements ($z_1(k), z_2(k)$, and $z_3(k)$). This situation is represented by the hypothesis matrix.
\[ \Omega = \begin{pmatrix} T_F & T_1 & T_2 & T_{N+1} \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & z_2(k) \\ 1 & 0 & 1 & 1 \end{pmatrix} \]

where, for simplicity we have assumed only a single possible type of new target. Hypothesis generation is then performed by picking one unit per row and one unit per column except for columns \( T_F \) and \( T_{N+1} \), where the number of false alarms and new targets is not restricted. This procedure assumes that a measurement has only one source and a geometric feature has at most one associated measurement per iteration, i.e. a feature gives rise to only a single measurement per scan of the sensors [28]).

### 3.2. Probability calculations

This section derives the probability equations needed to evaluate the different possible sets of assignments of measurements to features. These probabilities are then used to guide a pruning strategy applied to the hypothesis tree. The notation used here is derived from Bar-Shalom and Fortmann [4] and is based on Reid’s original multiple hypothesis proposal [32] and Kurien’s extension to allow for the termination of features [21].

The new hypothesis at time \( k \), \( \Theta^k_m \), is made up of the current event and a previous hypothesis based on measurements up to and including time \( k - 1 \), i.e.

\[ \Theta^k_m = \left\{ \Theta^{k-1}_{l(m)}, \theta^k_m \right\}. \]  

We need to calculate the probability of such a hypothesis, i.e.

\[ P \left\{ \Theta^k_m \mid Z^k \right\}, \]  

where \( Z^k \) denotes all measurements up to and including time \( k \). Using Bayes’ rule we have

\[ P \left\{ \Theta^k_m \mid Z^k \right\} = P \left\{ \theta^k_m \mid \Theta^{k-1}_{l(m)}, Z^k \right\} \]

\[ = \frac{1}{c} P \left[ Z(k) \mid \theta^k_m, \Theta^{k-1}_{l(m)}, Z^{k-1} \right] \]

\[ \cdot P \left\{ \theta^k_m \mid \Theta^{k-1}_{l(m)}, Z^{k-1} \right\} P \left( \Theta^{k-1}_{l(m)} \mid Z^{k-1} \right), \]

where \( c \) is a normalization constant. The last term of this equation, \( P \left( \Theta^{k-1}_{l(m)} \mid Z^{k-1} \right) \), represents the probability of the parent global hypothesis and is therefore available from the previous iteration. The remaining two terms may be evaluated as follows.
The number of false alarms and new features are assumed to be Poisson distributed\(^4\) with densities \(\lambda_F\) and \(\lambda_N\), respectively. The second factor of (9) is then obtained by combining results from [4] and [21] to yield

\[
P\{\Theta_m(k) \mid \Theta_{i(m)}^{k-1}, Z^{k-1}\} = \frac{1}{m_k!} \exp\left(\left(\lambda_F V + \lambda_N V\right) \phi_\gamma \left(\lambda_N V\right)^\nu\right)
\prod_i \left(P_D^i\right)^{\lambda_i} \left(1 - P_D^i\right)^{1 - \lambda_i} \left(P_T^i\right)^{\lambda_i} \left(1 - P_T^i\right)^{1 - \lambda_i},
\]

(10)

where \(V\) is the observation volume, and \(P_D^i\) and \(P_T^i\) are the probabilities of detection and termination (deletion) of track \(i\).

To determine the first term on the right-hand side of (9) we assume that a measurement \(z_i(k)\) has a Gaussian probability density function (PDF)

\[
N_i = N[z_i(k)] \\
\triangleq N[z_i(k); \hat{z}_i(k \mid k - 1), S^i(k)] \\
= \left[2\pi S^i(k)\right]^{-1/2} \exp\left(-\frac{1}{2} ((z(k) - \hat{z}(k \mid k - 1))^T \left(S^i(k)\right)^{-1} (z(k) - \hat{z}(k \mid k - 1)))\right),
\]

(11)

if it is associated with geometric feature \(t_i\), where \(\hat{z}_i(k \mid k - 1)\) denotes the predicted measurement for geometric feature \(t_i\) and \(S^i(k)\) is the associated innovation covariance. The prediction \(\hat{z}_i(k \mid k - 1)\) and innovation covariance \(S^i\) are precisely what is calculated using the (extended) Kalman filter. Thus, a Kalman filter may be associated with each geometric feature; each state vector represents the position and velocity of the associated feature. If the measurement is a false alarm, then its PDF is assumed uniform in the observation volume, i.e. \(V^{-1}\). The probability of a new geometric feature is also taken to be uniform\(^5\) with PDF \(V^{-1}\). Under these assumptions, we have that

\[
p[Z(k) \mid \Theta_m(k), \Theta_{i(m)}^{k-1}, Z^{k-1}] = \prod_{i=1}^{m_k} \left[N_i[z_i(k)]\right] V^{-1} \left(1 - \tau_i\right) \\
= V^{-\phi+\nu} \prod_{i=1}^{m_k} \left[N_i[z_i(k)]\right].
\]

(13)

\(^4\)In general, almost any distribution can be accommodated. However, the Poisson distribution conveniently eliminates certain combinatorial factors from the equation. Uniform distributions also allow for a concise mathematical representation.

\(^5\)Intuitively, the choice of uniform PDFs for false alarms and new features seems less justifiable for robotic applications than for traditional radar and underwater sonar tracking applications. The impact of these assumptions is a topic we intend to investigate.
Substituting (13) and (10) into (9) yields the final expression for the conditional probability of an association hypothesis

\[ P\{\Theta_m^k \mid Z^k\} = \frac{1}{c} \phi_{m-k}^{\lambda} N \prod_{i=1}^{m_k} [N_{i} [z_i(k)]]^{x_i} \]

\[ \cdot \left\{ \prod_{i} (P_{D}^i)^{x_i} (1 - P_{D}^i)^{1-x_i} (P_{X}^i)^{x_i} (1 - P_{X}^i)^{1-x_i} \right\} \]

\[ \cdot P\{\Theta_{l(m)}^{k-1} \mid Z^{k-1}\}. \] (14)

4. Implementation

In this section we describe design decisions regarding the choice of geometric feature models, the structure of a measurement vector and several techniques for efficient implementation of the Bayesian hypothesis tree.

4.1. Geometric feature models

The types of geometric features used to describe the world are heavily influenced by the sensing modality used. In the experimental section to follow, we use ultrasonic rangefinders to sense the environment. Our sonar sensor model follows in part from [19], and is described in detail in [25]. We assume that the actual 3-D environment geometry is orthogonal to the horizontal plane, so that the world can be adequately represented by a 2-D model. In a single sonar scan, it is impossible to differentiate the returns of walls, corners, and multiple reflections. For this reason, each new observation is used to initialize two new features: one planar surface and one corner. Multiple reflections are regarded as false alarms.

4.1.1. The Kalman filter for walls and corners

The parameters of each wall or corner form the state vector of an associated Kalman filter. There are several advantages to using a Kalman filter to track each feature, including optimum estimation (in a MMSE sense) of the geometric feature parameters. Thus, each time an hypothesis assigns a measurement to a track, this measurement is used to improve the parameter estimates for the associated geometric feature. Second, the Kalman filter stages of prediction, measurement and update fit conveniently into the multiple hypothesis framework depicted in Fig. 1. Third the Kalman filter is a recursive estimator and so there is no need to keep previous measurements, only the state and covariance matrices need to be stored at the end of each cycle. Finally, part of the Kalman filter computations are also needed to calculate the probabilities of a hypothesis, i.e. equation (12).
Table 1
Kalman filter models for plane and corner features. The sensing location at time $k$ is given by $x_R(k) = (x(k), y(k), \theta(k))^T$.

<table>
<thead>
<tr>
<th></th>
<th>Planes</th>
<th>Corners</th>
</tr>
</thead>
<tbody>
<tr>
<td>State vector $x_i(k)$</td>
<td>$x_i(k) = (R_i, \theta_i)^T$</td>
<td>$x_i(k) = (x_i, y_i)^T$</td>
</tr>
<tr>
<td>Plant model $x_i(k+1)$</td>
<td>$x_i(k+1) = x_i(k)$</td>
<td>$x_i(k+1) = x_i(k)$</td>
</tr>
<tr>
<td>Plant noise stddev</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Measurement vector $z(k)$</td>
<td>$(r, \alpha)^T$</td>
<td>$(r, \alpha)^T$</td>
</tr>
<tr>
<td>Measurement model $r$</td>
<td>$</td>
<td>R_i - x(k)\cos(\theta_i)</td>
</tr>
<tr>
<td>$z(k)$</td>
<td>$\alpha = \theta_i(k) - \theta(k)$</td>
<td>$(r, \alpha)^T$</td>
</tr>
<tr>
<td>Range error $\sigma_r$</td>
<td>1 cm</td>
<td>1 cm</td>
</tr>
<tr>
<td>Angle error $\sigma_\alpha$</td>
<td>5 deg</td>
<td>5 deg</td>
</tr>
<tr>
<td>Validation size $\gamma$</td>
<td>2.0</td>
<td>2.0</td>
</tr>
</tbody>
</table>

At time $k$, a 2-D line is represented by the geometric feature state vector $x_i(k) = [R_i(k), \theta_i(k)]^T$, where $R_i(k)$ is the perpendicular distance from the origin of the global coordinate frame to the line and $\theta_i(k)$ is the orientation of a perpendicular drawn from the origin to the line. In addition, each line has two endpoints associated with it, to facilitate visibility prediction. At this stage, endpoint parameters are estimated independently from the Kalman filter by projecting new observations onto the infinite line defined by the feature state vector. Of course, the endpoints could also be included in the state vector (e.g., see Deriche and Faugeras [14]). A corner is represented as a point in 2-D with a feature state vector of the form $x_i(k) = [x_i(k), y_i(k)]^T$. The location of the vehicle is specified by the vector $x_R(k) = [x(k), y(k), \theta(k)]^T$.

The polar measurements $z_j(k) = [r_j, \alpha_j]^T$ are a nonlinear function of the vehicle location $x_R(k)$ and the location of the geometric feature from which the measurement originated, subject to a noise disturbance $w_k(k)$, as given by the measurement model

$$z_j(k) = h_i(x_R(k), x_i(k)) + w_j(k),$$

where the measurement function $h_i(\cdot, \cdot)$ takes a different form depending on the type of feature (plane or corner), as shown in Table 1.

This measurement model provides the basis for the use of a Kalman filter to recursively compute the optimal MMSE estimate for the location of the geometric feature as new measurements are validated (hypothesized to originate from) the feature. As the Kalman equations are well known (for example, see [4]), we do not restate them here. Features that move can be accommodated by extending the state vector to incorporate velocity information.
4.1.2. Filter initialization

With densely sampled sonar data, as in the experiments below, new geometric features can be effectively initialized with a single observation, assuming the features are stationary. However, in general it may not be possible to initialize a new geometric feature based on a single measurement. This would be the case, for example, for moving features in which a single measurement provides no information regarding the velocity of the feature. In fact, track initiation is a problem whenever a sensor provides insufficient information to allow all parameters of a geometric feature, i.e. the Kalman state vector, to be given initial estimates. Another example is the case of monocular computer vision, in which range information is not directly measured. Monocular vision systems would not be able to initialize a stationary geometric feature from only a single measurement (see Aidala and Hammel [2] for a 1-D analysis of this problem).

There are two main solutions to the problem of track initiation from a single measurement. One method waits for the next set of measurements in order to estimate temporal parameters, for example. The other method provides default values to the unknown state vectors, but with very large associated covariances. The reader is directed to [1,4] for a detailed discussion of filter initialization.

4.2. Hypothesis management

Hypothesis management plays a crucial role due to the potentially overwhelming growth in the number of hypotheses. We eliminate unlikely hypotheses by a combination of both screening (prior to the generation of hypotheses) and pruning (after the generation of hypotheses).

Screening refers to the selective rather than exhaustive generation of hypotheses. Very unlikely sets of assignments are not examined since these are very likely to be pruned from the hypothesis tree later. One simple screening strategy that is very effective is to ignore the possibility of false alarms or new targets if the corresponding measurement is matched to only one target and this target has no other measurements matched to it. Similarly, we only consider the possibility of track termination if the corresponding track has no measurements associated with it at any given time step.

Screening slows the exponential growth of the hypothesis tree but is not sufficient alone to contain the tree to a manageable size. Pruning strategies are therefore needed to delete unlikely branches of the tree. Pruning strategies

---

6Initializing the state estimate is also more difficult for single, isolated range measurements from a wide angle sonar, such as the Polaroid, because of the high angular uncertainty associated with a single return.
are based on examining the probabilities of leaf nodes of the hypothesis tree.\footnote{Note that the incorporation of multiple feature types (corners and planes) requires the use of two terms for the density of new features, $\lambda_{NC}$ and $\lambda_{NP}$, in the probability calculations described above.}

4.2.1. Spatially disjoint clusters

Since it is not necessary to form a single global hypothesis tree containing tracks that do not have any common measurements, geometric features can be partitioned into separate clusters as proposed by Reid [32]. Tracks within each cluster share common measurements whereas tracks in different clusters do not. A separate hypothesis tree is grown for each spatially disjoint cluster and consequently, the combinatorial problem associated with forming global hypotheses is significantly reduced.

Of course, with each new set of measurements, we must check whether a measurement is shared (falls in the validation region) between two or more clusters. If so, these clusters must be merged. Similarly, a cluster containing two or more geometric features that do not share common measurements may be split.

4.2.2. Pruning

Pruning is essential to any practical implementation of this algorithm. Pruning is based on a combination of the "$N$-scan-back" algorithm [21] and a simple lower limit probability threshold.

The $N$-scan-back pruning algorithm assumes that any ambiguity at iteration $k$ is resolved by iteration $k + N$. Then, if hypothesis $\Theta^k_m$ at iteration $k$ has $q$ children, the sum of the probabilities of the leaf nodes is calculated for each of the $q$ branches. Whichever branch has the greatest probability is retained and all other branches are pruned. The result is an irrevocable decision regarding the assignment of measurements to geometric features based on looking ahead $N$ iterations. Consequently, below the decision node there is a tree of depth $N$ while above the decision node the tree has degenerated into a simple list of assignments, as illustrated in Fig. 3. It is clearly computationally advantageous to set $N$ as small as possible. In our experiments, the depth $N$ is set to 3. While this may seem quite small, previous work in tracking [21,32] suggests that even $N = 2$ can provide near-optimum solutions.

Note that the $N$-scan-back pruning algorithm may not always prune the expected leaves. In particular, consider a node at time $k$ with two branches. Further, assume that one branch, denoted branch $A$, has 51 leaves, each with a probability of 0.01 and the other branch, denoted branch $B$, has two leaves of probability 0.245, as illustrated in Fig. 4. Since the probabilities of
branches A and B are 0.51 and 0.49 respectively, branch B will be pruned. However, branch B contains the two most likely individual hypotheses, while branch A contains many unlikely hypotheses. Intuitively, this seems incorrect. Although we did not witness this behavior in our experiments, we believe this problem should be examined.

After N-scan-back pruning, the number of leaf nodes can still be very high. A second phase of pruning removes all nodes whose probability is less than a lower limit (currently set to 0.01) so that at the end of each iteration there are no more than 100 hypotheses remaining. At first glance,
this procedure, sometimes referred to as "beam search", might be considered to replace the $N$-scan-back pruning procedure altogether. However, if this procedure is applied without $N$-scan-back pruning, then trees can develop in which the leaf hypotheses are essentially identical, differing only by a single assignment that occurred very early on in the growth of the tree. Fig. 5 illustrates this problem. Here, a single measurement at time $k = 1$ causes two hypotheses to be generated; one a false alarm $\theta^1_1$, the other a new target, $\theta^1_2$. At time $k = 2$ another single measurement is obtained which validates to the feature of hypothesis $\theta^1_2$. This feature is tracked to form the child hypothesis, $\theta^2_2$. Meanwhile hypothesis $\theta^1_1$ forms two new child hypotheses; the second measurement is a false alarm, $\theta^2_3$, or the second measurement is a new track, $\theta^2_4$. At time $k = 3$ a single measurement validates to both features in hypotheses $\theta^2_2$ and $\theta^2_4$. Both hypotheses in turn form child hypotheses $\theta^3_2$ and $\theta^3_4$ which track the feature. These two hypotheses only differ in the assignment of the first measurement. As time progresses, two almost identical hypotheses develop. Clearly, one could perform some test to check that hypotheses were sufficiently different, but this may be computationally expensive. $N$-scan-back pruning makes this unnecessary.
Table 2
Algorithm parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probability of detection $P_D$</td>
<td>0.85</td>
</tr>
<tr>
<td>Probability of termination $P_z$</td>
<td>0.01</td>
</tr>
<tr>
<td>False alarm density $\lambda_F$</td>
<td>0.5</td>
</tr>
<tr>
<td>New plane feature density $\lambda_{NP}$</td>
<td>0.5</td>
</tr>
<tr>
<td>New corner feature density $\lambda_{NC}$</td>
<td>0.5</td>
</tr>
<tr>
<td>$N$-scan-back pruning depth N_SCAN</td>
<td>3</td>
</tr>
</tbody>
</table>

Finally, it may be interesting to reverse the order of the two pruning strategies, first removing all hypotheses with probabilities less than a given threshold and then performing $N$-scan-back pruning. The threshold pruning would eliminate the many leaf nodes with very low probabilities thereby preventing the peculiar pruning solutions that are possible with the $N$-scan-back strategy. Meanwhile, the $N$-scan-back pruning prevents almost identical hypothesis being continued indefinitely.

5. Experimental results

The above framework has been implemented with a mobile robot equipped with ultrasonic range sensors [25]. In this arrangement the multiple reflections common to ultrasonic imaging are considered false alarms. The final output of the algorithm is the classification of the origin of each sensor observation as corner, wall, or multiple reflection, and the grouping of observations of common origin into corner or planar geometric features.

We present results from the use of the algorithm with both simulated and real sonar data. In a single sonar scan, it is impossible to differentiate the returns of walls, corners, and multiple reflections. For this reason, each new observation is used to initialize two new features: one planar surface and one corner. Multiple reflections are considered false alarms. Feature state estimates are produced by Kalman filters which use different plant and measurement models for each type of feature, as shown in Table 1. The incorporation of multiple feature types (corners and planes) requires the use of two terms for the density of new features, $\lambda_{NC}$ and $\lambda_{NP}$, in the probability calculations described above.

Table 2 shows the algorithm parameter values used for the subsequent experiments. The values of Table 2 were arrived at through a combination of estimation and guess work. However, in theory at least, each of the parameters represents a statistical property of either the vehicle's environment or its sensors and can therefore be measured. Our limited experience suggests that the algorithm is not overly sensitive to the values of parameters and this is supported by previous work [6,28,32].
5.1. Simulated sonar data

Fig. 6 shows a sequence of simulated measurements taken from twelve accurately known sensing locations. During the five first time steps (Fig. 7), both a wall and corner are simultaneously visible. At time 6, the wall "disappears", and at time 9 (Fig. 8) a spurious measurement is detected. Since the wall, corner, and spurious measurement are spatially disjoint, their hypothesis trees can be viewed independently.

First, consider the hypothesis tree for the wall feature, illustrated by Table 3. The first measurement for this tree has three interpretations, false alarm, corner or planar wall, each with a probability of 0.33 because $\lambda_{FA} = 1.0 = \lambda_{NF}$. At the next time step, only the third hypothesis (the planar wall) receives a valid measurement, and hence its probability increases to 0.978. After updating the hypothesis tree at time 3, the probability of the correct hypothesis increases to 0.995, and the feature is correctly identified as a wall when $N$-scan-back pruning is first executed. The hypothesis tree then reduces to a single hypothesis, of probability 1.0, until time 6, when the predicted measurement is not detected. At this point, two alternatives are hypothesized: either the wall is no longer there, and the geometric feature should be terminated, or the sensor failed to detect the existing wall, and the geometric feature should be continued to the next time frame. Subsequent missed observations result in the correct deletion of the wall feature. This
"learning" and "forgetting" demonstrates the algorithm's ability to deal with changing environments. Tables 4 and 5 show the hypothesis trees for the corner feature and false alarm respectively.

Fig. 7. Sequence of simulated sonar observations (part 1).
5.2. Results with real sonar data

Fig. 9 shows the processing sequence for fourteen sonar scans that were taken in a small clutter-free room with the standard Polaroid ranging system.
Table 3
Hypothesis tree for wall cluster.

<table>
<thead>
<tr>
<th>Time</th>
<th>Hypothesis probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>False alarm</td>
</tr>
<tr>
<td>1</td>
<td>0.333</td>
</tr>
<tr>
<td>2</td>
<td>F 0.006</td>
</tr>
<tr>
<td></td>
<td>Delete corner</td>
</tr>
<tr>
<td>3</td>
<td>3 £’s 0.005</td>
</tr>
<tr>
<td></td>
<td>Wall 0.005</td>
</tr>
<tr>
<td></td>
<td>2 £’s 0.001</td>
</tr>
</tbody>
</table>

| 4    | Wall 1.000  |
| 5    | Wall 1.000  |
| 6    | Delete wall 0.063 | Continue wall 0.937 |
| 7    | Delete wall 0.298 | Delete wall 0.044 | Continue wall 0.658 |
| 8    | Delete wall 0.668 | Delete wall 0.100 | Delete wall 0.015 | Continue wall 0.217 |
| 9    | Wall deleted 1.000 |

Table 4
Hypothesis tree for corner cluster.

<table>
<thead>
<tr>
<th>Time</th>
<th>Hypothesis probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>False alarm</td>
</tr>
<tr>
<td>1</td>
<td>0.333</td>
</tr>
<tr>
<td>2</td>
<td>F 0.004</td>
</tr>
<tr>
<td></td>
<td>Corner 0.004</td>
</tr>
<tr>
<td></td>
<td>0.004</td>
</tr>
<tr>
<td>3</td>
<td>F 0.00005</td>
</tr>
<tr>
<td></td>
<td>Corner 0.00003</td>
</tr>
<tr>
<td></td>
<td>0.00004</td>
</tr>
<tr>
<td>4</td>
<td>Corner 1.000</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Corner 1.000</td>
</tr>
</tbody>
</table>

[31]. Fig. 10 shows input sensor observations, in the form of “regions of constant depth” [25], for six of the input scans.

The algorithm partitions the input measurements into four wall and ten corner features, whose line and point state estimates are shown in Fig. 11. Fig. 12 superimposes the estimated geometric features over the hand-measured room model to reveal the high accuracy of the learned map. Tables 6 and 7 provide a quantitative comparison of estimated and hand-measured feature locations. Twelve input measurements were classi-
Table 5
Hypothesis tree for false alarm.

<table>
<thead>
<tr>
<th>Time</th>
<th>Hypothesis probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>False alarm 0.333</td>
</tr>
<tr>
<td></td>
<td>Corner 0.333</td>
</tr>
<tr>
<td></td>
<td>Wall 0.333</td>
</tr>
<tr>
<td>10</td>
<td>False alarm 0.760</td>
</tr>
<tr>
<td></td>
<td>Delete corner 0.008</td>
</tr>
<tr>
<td></td>
<td>Continue corner 0.113</td>
</tr>
<tr>
<td></td>
<td>Delete wall 0.008</td>
</tr>
<tr>
<td></td>
<td>Continue wall 0.113</td>
</tr>
<tr>
<td>11</td>
<td>False alarm 0.537</td>
</tr>
<tr>
<td></td>
<td>Delete corner 0.009</td>
</tr>
<tr>
<td></td>
<td>Delete corner 0.001</td>
</tr>
<tr>
<td></td>
<td>Cont. corner 0.021</td>
</tr>
<tr>
<td></td>
<td>Delete wall 0.009</td>
</tr>
<tr>
<td></td>
<td>Delete wall 0.001</td>
</tr>
<tr>
<td></td>
<td>Cont. wall 0.021</td>
</tr>
<tr>
<td>12</td>
<td>False alarm 1.000</td>
</tr>
</tbody>
</table>

Fig. 9. Hand-measured map of the room, with triangles at each of fourteen locations where sonar scans were taken. The room is 3 meters wide, with a closed door in the upper right-hand region of the figure.

Table 6
Comparison of learned wall feature parameters with actual values, hand-measured to a few millimeters of accuracy. Range is given in meters, orientation in degrees.

<table>
<thead>
<tr>
<th>Geometric feature</th>
<th>Estimated type</th>
<th>Actual type</th>
<th>Estimated $R$</th>
<th>Actual $R$</th>
<th>Estimated $\theta$</th>
<th>Actual $\theta$</th>
<th>Difference $R$</th>
<th>Difference $\theta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Wall</td>
<td>Wall</td>
<td>1.709</td>
<td>1.712</td>
<td>0.4</td>
<td>0.0</td>
<td>0.003</td>
<td>0.4</td>
</tr>
<tr>
<td>2</td>
<td>Wall</td>
<td>Wall</td>
<td>1.407 $-89$</td>
<td>1.402 $-89.8$</td>
<td>0.005</td>
<td>0.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Wall</td>
<td>Wall</td>
<td>0.507 $-89.8$</td>
<td>0.500 $-90.0$</td>
<td>0.007</td>
<td>0.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Wall</td>
<td>Wall</td>
<td>$-1.009$ $-0.1$</td>
<td>$-1.000$ $0.0$</td>
<td>0.009</td>
<td>0.1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Fig. 10. RCDs extracted from six of the sonar scans.

ified as false alarms; these are shown in Fig. 13. Figs. 14–16 show the input measurements assigned to each geometric feature. Shaded triangles indicate sensing locations from which the feature was not observed. All but one of these features correspond to actual walls or corners in the room, with the
exception of geometric feature 9, which is the result of multiple reflections off the top wall onto the convex corner in the lower right-hand part of the room (feature 8) and back.

Finally, Fig. 17 illustrates how the number of measurements assigned as false alarms varies as a function of $\lambda_f$, the mean rate of false alarms. At the two extremes, of either very low or very high false alarm rates, we see all
Table 7
Comparison of learned corner feature locations with hand-measured values. Positions are given in meters, with respect to the 2nd vehicle position. Track 9 is the result of false multiple reflections.

<table>
<thead>
<tr>
<th>Geometric feature</th>
<th>Estimated type</th>
<th>Actual type</th>
<th>Estimated X</th>
<th>Estimated Y</th>
<th>Actual X</th>
<th>Actual Y</th>
<th>Difference X</th>
<th>Difference Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Corner</td>
<td>Corner</td>
<td>-1.002</td>
<td>-1.410</td>
<td>-1.002</td>
<td>-1.406</td>
<td>0.002</td>
<td>0.004</td>
</tr>
<tr>
<td>5</td>
<td>Corner</td>
<td>Corner</td>
<td>0.822</td>
<td>0.495</td>
<td>0.826</td>
<td>0.500</td>
<td>0.004</td>
<td>0.005</td>
</tr>
<tr>
<td>7</td>
<td>Corner</td>
<td>Corner</td>
<td>-1.013</td>
<td>0.509</td>
<td>-1.000</td>
<td>0.500</td>
<td>0.013</td>
<td>0.009</td>
</tr>
<tr>
<td>8</td>
<td>Corner</td>
<td>Corner</td>
<td>1.232</td>
<td>-0.997</td>
<td>1.230</td>
<td>-0.992</td>
<td>0.002</td>
<td>0.005</td>
</tr>
<tr>
<td>9</td>
<td>Corner</td>
<td>MR</td>
<td>1.268</td>
<td>1.980</td>
<td>No target</td>
<td>Multiple reflection</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Corner</td>
<td>Corner</td>
<td>1.642</td>
<td>0.503</td>
<td>1.654</td>
<td>0.500</td>
<td>0.012</td>
<td>0.003</td>
</tr>
<tr>
<td>11</td>
<td>Corner</td>
<td>Corner</td>
<td>0.278</td>
<td>-1.400</td>
<td>Unrecorded</td>
<td>Unrecorded</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Corner</td>
<td>Corner</td>
<td>0.161</td>
<td>-1.404</td>
<td>Unrecorded</td>
<td>Unrecorded</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Corner</td>
<td>Corner</td>
<td>-1.230</td>
<td>-1.392</td>
<td>-1.243</td>
<td>-1.398</td>
<td>0.013</td>
<td>0.006</td>
</tr>
<tr>
<td>14</td>
<td>Corner</td>
<td>Corner</td>
<td>1.713</td>
<td>-0.978</td>
<td>1.712</td>
<td>-0.970</td>
<td>0.001</td>
<td>0.008</td>
</tr>
</tbody>
</table>

Fig. 13. Observations that were classified as false alarms.

measurements being hypothesized as either all targets or all false alarms, as expected. More importantly, we see a reasonably stable region within the range $\lambda_f = 0.1$ and $\lambda_f = 10$ in which good results are obtained. We saw similar characteristics with other parameter variations, all of which support our belief that the algorithm is not overly sensitive to parameter variations.

While multiple reflections are to some extent unavoidable when using sonar, their effect is less severe with our approach in comparison with most earlier work in sonar map building. To further eliminate any stubborn multiple reflection artifacts, a number of strategies might be employed. For example, geometric feature 9 is clearly inconsistent with the top wall, fea-
Fig. 14. Corner and wall tracks produced by the algorithm (part 1). Shaded triangles indicate sensing locations from which the feature was not observed.

ture 4. One could apply a simple test, similar to Drumheller's sonar barrier test [15], to each possible wall-corner pairing to reveal this inconsistency. Feature 4, observed at fourteen of fourteen sensing locations, would be the
clear winner in a contest with feature 9, observed from just two locations. However, as an alternative to implementing a sensor-specific post-processing stage, the parameter variation study used to create Fig. 17 indicated that multipath rejection can be improved by varying the parameters of the basic
6. Discussion

We have demonstrated how a Bayesian multiple hypothesis framework can be used to build and maintain a world model of a dynamic environment. Dynamic environments are challenging problems because the number of perceptually relevant features is a priori unknown and, moreover, changes with time. The primary advantage of the approach is that it postpones
making irrevocable decisions. Instead, when ambiguities arise, each possible interpretation or hypothesis is generated and its associated probability calculated. Subsequent measurements are then applied to each hypothesis. Temporal integration of measurements resolves many (hopefully all) past ambiguities. At any time then, there are multiple world models each one representing a possible interpretation of all past and current measurements and each having an associated probability.

The uncertainty in assignment is compounded by measurement noise in the sensors. The decoupling of uncertainty due to noise (modeled by the Kalman filter state covariance matrix) and uncertainty in the data association or grouping process (modeled by an associated probability) is a distinguishing feature of this approach that should allow for more robust interpretation and integration by later processing modules.

The algorithm possesses both top-down and bottom-up control competences; the Kalman filter providing a tool for looking for the expected and the hypothesis generation and associated Bayesian probabilities providing a mechanism for explaining the unexpected. As such, the algorithm seems very suitable for extension to the problem of active perception, whereby the algorithm directs sensing operations to reduce ambiguities. The algorithm also possesses a limited recognition capability, being able to distinguish between corner and wall features from successive measurement sequences.

The main disadvantage of the multiple hypothesis approach is the very large number of hypotheses that may be generated. However, the hypothesis management techniques of pruning and clustering appear to constrain the hypothesis trees to manageable sizes. A secondary issue is that of choosing values for the algorithm parameters, both for the probability calculations of Section 3.2 and for the underlying Kalman filters. However, we emphasize that all such parameters are measurable statistical quantities of either the environment or the sensor. The algorithm also appears not to be too sensitive to parameter values.

Perceptually relevant geometric features are likely to vary depending on higher level task objectives, e.g. searching for an object. Future work should examine the possibility of incorporating higher level knowledge into the priors of the probability estimates. Further work should also explore extending the algorithm to provide an active sensing capability, incorporation of multiple sensors and modeling the effects of occlusion.

While this paper has focused on the uncertainties in the values and the origins of sensor measurements, a general solution to the mobile robot navigation problem requires the incorporation of a third source of error, namely, uncertainty in the motion of the vehicle. The subsequent correlation of measurement errors may significantly complicate the data association process. For example, in a fully coupled approach, one could not perform the cluster partitioning described in Section 4.2.1. In addition, it seems
difficult to incorporate a track deletion capability into the fully coupled approach in order to deal with dynamic environments. For these reasons, we believe that a suboptimal decoupled approach will be necessary. Our future research will be directed toward the incorporation of this issue into the framework.

Acknowledgements

Thanks to Y. Bar-Shalom and H. Durrant-Whyte for valuable discussions on issues related to target tracking, and to several anonymous reviewers for their helpful comments. Also thanks to S. Hingorani for her excellent software implementation.

References


