Dynamic Bayes Net Approach to Multimodal Sensor Fusion

Amit Singhal Chris Brown

Department of Computer Science
University of Rochester, Rochester, NY 14627

ABSTRACT

Autonomous mobile robots rely on multiple sensors to perform a varied number of tasks in a given environment. Different tasks may need different sensors to estimate different subsets of world state. Also, different sensors can cooperate in discovering common subsets of world state. This paper presents a new approach to multimodal sensor fusion using dynamic Bayesian networks and an occupancy grid.

The environment in which the robot operates is represented with an occupancy grid. This occupancy grid is asynchronously updated using probabilistic data obtained from multiple sensors and combined using Bayesian networks. Each cell in the occupancy grid stores multiple probability density functions representing combined evidence for the identity, location and properties of objects in the world. The occupancy grid also contains probabilistic representations for moving objects. Bayes nets allow information from one modality to provide cues for interpreting the output of sensors in other modalities. Establishing correlations or associations between sensor readings or interpretations leads to learning the conditional relationships between them. Thus bottoms-up, reflexive, or even accidentally-obtained information can provide tops-down cues for other sensing strategies. We present early results obtained for a mobile robot navigation task.

Keywords: dynamic Bayes nets, multimodal sensor fusion, mobile robot navigation, occupancy grids

1. INTRODUCTION

Autonomous robots rely on numerous sensors to obtain a consistent and coherent view of the current world state. This inherently introduces uncertainties as different sensors may react differently to the same stimuli, or may provide incorrect or inconsistent data. These sensor discrepancies have to be handled in some framework to allow the robot to visualize a unified view of its environment. Strictly speaking, *multisensor fusion* and *multisensor integration* refer to two different processes. Multisensor integration refers to the synergistic use of the information provided by multiple sensory devices to assist in the accomplishment of a task by a system. Multisensor fusion refers to any stage in the integration process where there is actual combination (fusion) of different sources of sensory information into one representational format.[9] Although the above distinction is not standard in the literature, it allows us to concentrate an a particular aspect of the general problem of combining multiple sensors.

Early work characterized the sensor fusion problem as one of incremental combination of geometric information. Most of these techniques were *ad-hoc* and did not take the uncertainties in the system into consideration. The first work on incorporating uncertainty in an explicit manner in sensor fusion was performed by Smith and Cheeseman.[12] They proposed the use of *Bayesian estimation theory* and derived a combination function that was an equivalent form of *Kalman Filter*. This caused a rapid paradigm shift towards probabilistic estimation theories closely related to *Bayesian estimation, maximum likelihood estimation* and *least squares methods*.[3] While these methods have been successful in combining sensor data that determine a common subset of the robot state vector (eg. robot location derived from GPS, odometry and gyroscope readings), there seems to be no obvious extension to *multi-modal* sensor fusion processes. The advantage of multi-modal approaches is that one can combine data from dissimilar sensors (eg. decibel-meter providing noise levels and camera providing visual imagery) to mimic human-like reasoning and guidance strategies (eg. hearing a car implies a higher likelihood of seeing a car).

Further author information -
A.S.(correspondence): Email: singhal@cs.rochester.edu; WWW: http://www.cs.rochester.edu/u/singhal; Telephone: (716)275-8479; Fax: (716)461-2018
C.B.: Email: brown@cs.rochester.edu; WWW: http://www.cs.rochester.edu./u/brown; Telephone: (716)275-7852; Fax: (716)461-2018

2. RELEVANT LITERATURE

Probability theory, with its inherent notions of uncertainty and confidence, has found widespread popularity in the multisensor fusion community. The probabilistic models proposed by various researchers can be classified into four broad categories : Bayesian reasoning,[4] evidence theory,[5] robust statistics,[7] and recursive operators.[2]

Bayesian reasoning and inference procedures have been widely used in other areas of sciences for a long time but have only recently gained popularity in multisensor fusion. Kortenkamp[8] first proposed the use of a *Bayesian network* (as opposed to statistical Bayesian methods) for performing multisensor fusion for *topological map* building. Brooks[1] had earlier argued very persuasively for using topological maps as a means of dealing with uncertainty in mobile robot navigation. Elfes[4] later proposed another framework called *occupancy grids* which was also shown to be very efficient in dealing with the above problem. Martin and Moravec[10] proposed the *evidence grid* representation for mobile robot navigation.

Our research extends the work performed by the above researchers towards developing a general purpose navigational and sensor fusion framework. In the rest of the paper, we present an approach called *dynamic Bayesian networks in a dynamic occupancy grid framework* for multisensor fusion and mobile robot navigation.

3. BACKGROUND INFORMATION

3.1. Bayesian networks

Bayesian networks use directed graphs to represent the dependencies and relationships between various entities in the system. The connections (the directed arrows) represent the conditional probabilities (or likelihoods) of inferencing the existence of one entity (which is being pointed to) given the existence of the other entity. Each entity can have many such directed inputs, each specifying its dependence relationship to the entities the inputs originate from. Given this interpretation, a Bayesian network can be thought of as a knowledge base. It explicitly represents our *beliefs* about the system and the relationships between the various entities of the system. Pearl[11] presents a detailed description of Bayesian networks and associated theoretical proofs.

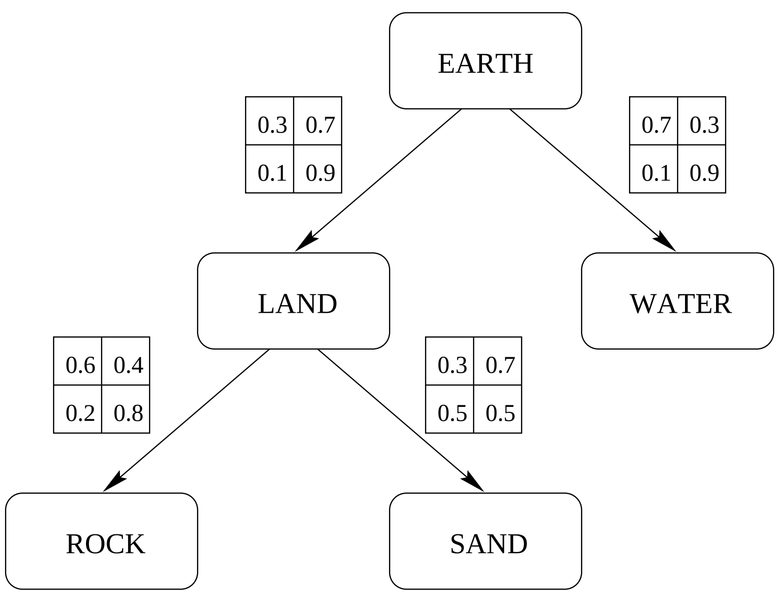

Figure 1. An example of a Bayesian Network

Bayesian networks operate by propagating beliefs through the network once some evidence about the existence of certain entities can be asserted. When we assert the existence of an entity, we can propagate this belief *upwards* in the network by calculating posterior probabilities of the existence of all other entities connected to the asserted entity. This allows us to compute our current belief in the existence of all the entities in the network, given our *knowledge* of the existence of a few of the entities and the relationships between them.

Consider the Bayesian network described in Figure 1. Various dependencies in the system are depicted via the directed arrows. Our belief in the existence of earth depends upon the existence of land and water. Further, our belief in the existence of land depends upon the existence of rock and sand. These beliefs are discretized/quantified and presented as conditional probability tables and are associated with the directional arrows. For example, the probability of the existence of rock given the existence of land is 0.6 and the probability of existence of rock given absence of land is 0.4 (0.2 and 0.8 are the corresponding probabilities for absence of rock given presence or absence or land respectively). The probabilities reflect our knowledge of the system and can be derived by making observations and gathering sample data. In the above case, if we had ten samples where we saw land, then rock was also present in six of them. We can then use *Bayes rule* to calculate the probability of being on Earth whenever we make an observation that supplies evidence for the existence of rock, sand or water.

A simple sensor fusion Bayesian network can be easily designed by representing the sensor data from the individual sensors as the *inputs* and a unified map representation as the single *output*. The degree of confidence in each sensor's measurements can be represented using the conditional probability matrices associated with the causal relationship linking the sensors to the unified map. These conditional probabilities do not have to be known a-priori and can be *learned* using statistical sampling techniques or supervised learning approaches.

One of the most popular approaches to sampling techniques uses Gibbs sampling for generation of training data.[11] In this approach, a small set of complete training data (sensor observations and actual environment state) serves as a base for a sample generator to generate a large set of training data from incomplete samples. The incomplete samples can have missing sensor observations or missing environment observations. Supervised learning is done by placing the mobile robot at various locations in the environment and polling the sensors. The sensor observations and actual observations are compiled in contingency tables. These contingency tables encode empirical expert knowledge about the reliability of each sensor in correctly observing the environment. When a sufficiently large amount of sample data is gathered, this conditional probability table can be calculated from the contingency tables.

Kortenkamp[8] first proved the usefulness of Bayesian networks in performing multisensor fusion for robot localization. He used two sensors, vision and sonar, and showed that while neither of the two could independently recognize every location in the world unambiguously, a Bayesian network with the two sensors as inputs and a single output could unambiguously perform robot localization at every location. The Bayesian network used by Kortenkamp encoded expert knowledge as conditional probability tables and was limited in its application. There is no obvious extension of this to a generalized navigation problem where the environment may be unstructured or changing constantly.

3.2. Occupancy grids

Occupancy grids are a stochastic tessellated representation of perceived spatial information about the robot's operating environment. The environment is explicitly divided into small square spatial areas called *cells*. Each cell stores a probabilistic estimate of the occupancy of that cell in the form of a *state variable*. The representation implicitly captures spatial adjacency information.

In a single dimension occupancy grid, each binary state variable, $s(C_i)$, predicts the presence or absence of an object in a cell C_i. Elfes uses the labels OCC and EMP to identify the two possible states for each cell, *occupied* and *empty*. Since the states are exclusive and exhaustive, $P[s(C_i) = OCC] + P[s(C_i) = EMP] = 1$. Bayesian estimation rules are used to update the values associated with the state variables.

The major advantage of this approach is that the navigation system only needs to look at the occupancy grid to see what cells around the robot's location are probabilistically free of obstacles to move the robot towards a coordinate specified goal. If all the sensor models, and the prior probability distributions of the configuration space are learned or known a-priori, then the occupancy grid can be updated very fast using the sensory data.

However, the above approach also has some drawbacks. Firstly, sensor models need to be known accurately to do updating of the occupancy estimates in the occupancy grids. This is usually not possible due to two reasons : (1) weak, ill-designed sensors and (2) multiple dependent sensors. Multiple independent sensors whose individual sensor models are known can be handled at some computation cost by serializing the updating operation for all the sensors.

4. THE PROPOSED FRAMEWORK

We propose a probabilistic framework that combines and extends Bayesian networks and occupancy grids for performing mobile robot navigation in a *generalized environment*. A generalized environment can include :

- outdoor and indoor locations
- rough and smooth terrain
- stationary and moving obstacles
- structured and unstructured landmarks
- single or multiple robots
- varying sensors and sensor models

Uncertainties can arise due to lack of a-priori knowledge about the environment (ie. no map or inaccuracies in map), presence of dynamic obstacles, and limitations of the sensor systems among other reasons. The representation chosen for modeling the environment is a modification of the occupancy grids proposed by Elfes. Section 4.1 presents the details of this representation. We use a modification of Bayesian networks, termed *dynamic Bayesian networks* as our tool for updating the dynamic occupancy grid representation. Details of dynamic Bayesian networks are presented in section 4.2.

The last component of our proposed framework is a methodology for updating the dynamic occupancy grid using dynamic Bayesian networks. To relax synchronization issues and constraints, we employ an asynchronous update policy that uses dynamic Bayesian networks to create new probability density functions (PDF). These PDFs are then used to update the information stored in the dynamic occupancy grids. The navigation control system uses this information to infer occupancy characteristics of the robot's neighborhood when planning robot motion.

4.1. Dynamic occupancy grids

A representation used for modeling the physical attributes of the environment and the entities present in it must possess certain qualities. It must be powerful enough to express all the entities that need to represented. At the same time, it must be adaptive in that only entities that effect the navigation algorithm at any given stage need to be represented during that stage. Thus, if there is a road present in the environment, but the robot is nowhere near it, we may not need to store information about it in the representation. Such an approach restricts the representation space from growing exponentially, allowing for implementation of tractable algorithms for motion planning. As an aside, note that this does not preclude the use of maps when the environment is known. A map can still be maintained in parallel with the proposed framework and the occupancy grids can derive information about static objects from them.

Another desired characteristic of any representation is that it must be able to quickly update its knowledge about the current state of the environment without too much computational effort. This is especially important in our approach since we rely purely on the information present in the dynamic occupancy grids for performing navigation.

We use the occupancy grid approach developed by Elfes[4] and modify it to represent a more generalized environment as follows:

1. We associate a set of *state vectors* (instead of a single state variable) with each cell. There is a state vector associated with each entity currently represented by the occupancy grids. Each state vector contains probabilistic estimates of an object's identity, location and characteristics (such as velocity, acceleration, or other behavior). Thus, each cell of our modified occupancy grid captures a lot more information about the environment.

2. We make our modified occupancy grids *dynamic* by allowing the set of represented entities to change over time. Only those entities currently affecting the navigation system are included in the representation. The set of entities represented at any given time is obtained by using an independent process that polls the sensors (and available maps) and constructs a set of detected entities. This provides us with major space reduction as the navigation system is only concerned about events occurring in some pre-defined neighborhood around the robot. This adaptive representation seems especially well suited for entities that only have a limited presence in the environment (eg. cars driving in a direction orthogonal to the robot).

3. A decay function is associated with the PDFs that increases the variance of the beliefs when they have not been updated for a period of time.

4. We change the updating procedure by using these PDFs and discretizing their values in each cell to obtain the information for the state vectors directly. These PDFs are generated using dynamic Bayesian networks described in section 4.2 instead of a Bayesian estimation rule. The advantages of using dynamic Bayesian networks are analyzed in the same section.

4.2. Dynamic Bayesian networks

We need a method to generate the PDFs used in the dynamic occupancy grids for estimation of entity characteristics. Such a method must possess certain qualities to be applicable to any general environment. Firstly, it must explicitly deal with uncertainties as the outputs of this method are PDFs. Secondly, it must be easy to construct or build the method for producing these PDFs based on the sensory observations (our only source of information about the environment). Thirdly, the time required to construct a new set of PDFs based on sensor data received must be minimal to allow the dynamic occupancy grids to quickly reflect changes in the environment. Lastly, since our dynamic occupancy grids can represent different entities (and this requires different PDFs) at different times, the method used to generate these PDFs must also be adaptive in the same manner.

Pearl[11] describes Bayesian networks in great detail and motivates their use for tasks that require causal relationships to be modeled. The PDFs required by the dynamic occupancy grids are definitely causally dependent on the sensory inputs. Hence, we choose to explore and develop a Bayesian network based tool for updating the PDFs used by the dynamic occupancy grids. Our approach only modifies what our network's structure looks like over time and does not change their functioning in any manner. One can refer to Pearl's book[11] for a detailed description of the working of these Bayesian networks.

We propose the following extensions to the Bayesian network approach described by Pearl[11] and used by Kortenkamp[8]:

1. Addition of a history node as shown in figure 2 linked to the corresponding node in the Bayesian network to explicitly encode a temporal aspect into the Bayesian network.

2. Dynamic structure changes in the Bayesian network to represent human-like reasoning strategies. These structural changes are triggered by the beliefs crossing a threshold value. One such example is shown in figure 3.

3. Run time selection of Bayesian networks (from a pre-developed or learned library of networks) to be used at any given time based on the entities represented in the dynamic occupancy grids at that same time.

These dynamic Bayesian networks can then be used for updating the dynamic occupancy grids using the methodology described in the next section.

4.3. The methodology for navigation

The navigation system consists of two parts. The first is a motion planner that looks at the dynamic occupancy grids and determines the current state of the robot's neighborhood and then plans the next motion to execute. This motion planner can be dependent on the environment in which the robot is operating. A fuzzy rule based system can also be used to select from a set of motion planners depending on the characteristics of the environment, thus achieving generalization in robot navigation.

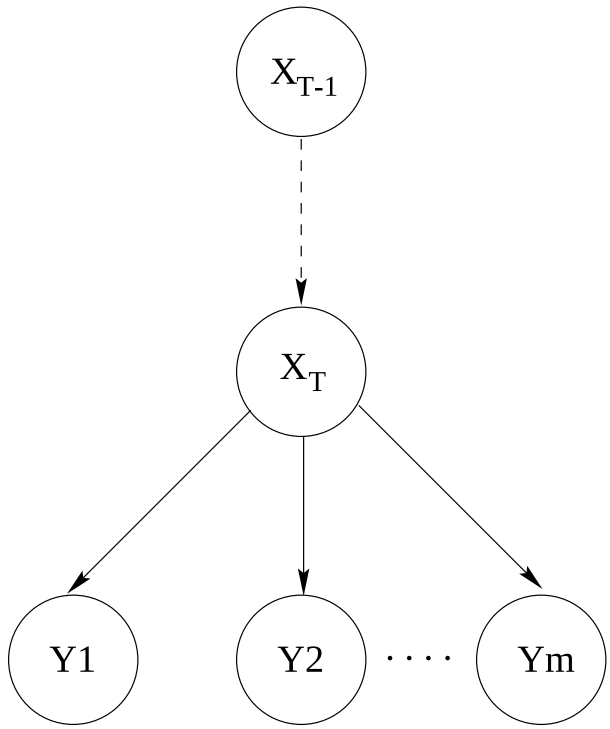

Figure 2. Incorporating past information into a Bayesian network using a history node.

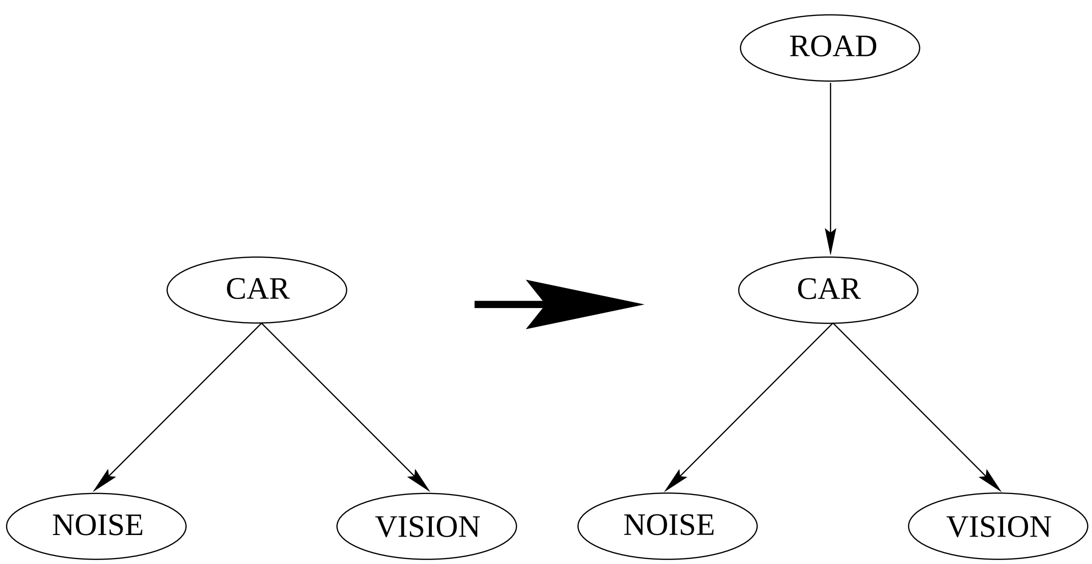

Figure 3. A dynamic Bayesian network for car and road detection from vision and noise sensors. The road detection node is adaptively added to the original car detection Bayesian network when the probability of the presence of car gets more than 0.5.

The second part of the navigation system encodes the methodology used in the framework to update the PDFs associated with the dynamic occupancy grids, based on the results of the dynamic Bayesian networks. This component system continuously polls the sensors for any new evidential information. Whenever new evidence is received from a sensor, corresponding Bayesian networks (that include the sensor as one of their nodes) are started up in parallel. After belief propagation, the new PDFs generated are used to update the corresponding PDFs in the dynamic occupancy grids. The Bayesian networks are fired up asynchronously, whenever any of the sensors generates some new information. The occupancy grids are also updated in an asynchronous manner. This is especially useful since the motion planner does not have to wait for the occupancy grid to be updated but simply uses the current PDFs to produce an estimate of the environment.

5. SYSTEM IMPLEMENTATION

The above framework was implemented using C++ on an UltraSPARC station. A library of classes was developed for representing occupancy grids and storing state vector information associated with objects in the occupancy grid. COBRA,[6] a locally developed Bayesian network package was used to create the necessary Bayes nets and perform belief propagation. The current system has three distinct parts.

The first part, the Bayes net package, encodes all the causal relationships in the system in the form of multiple Bayes nets. These Bayes nets are read in from text description files and stay loaded in the computer's memory through the execution of the system. At any given time, the system selects a subset of these Bayes nets corresponding to its current beliefs about the identities of objects in the world. While the system constantly changes the subset of nets that is active at any given time, no new nets can be learned by it. All the Bayes nets are created by the human supervisor and encode their knowledge about the causality relationships in the robot's operating environment. One major feature of the system is that the Bayes nets themselves can change their structure depending on the current beliefs. An example of this is shown in figure 3.

The second part of the system creates the occupancy grid representation. This part is closely interlinked with the first part in that the active subset of Bayes nets is chosen here. Only those objects in the robot's neighborhood are deemed of interest and studied further. The occupancy grid framework is updated constantly using new PDFs received from the Bayes nets package via a connecting stream. Another feature of this component system is the decay function that reduces the beliefs about the presence and characteristics of dynamic objects in the environment if no new information is received from the Bayes nets.

The last part of the system is responsible for generating the motor drive commands for moving the robot. This is a pure robot motion control system that uses the occupancy grid as its neighborhood map and (possibly in conjunction with a global map) performs robot navigation. The control package does not wait at any time period for the grids to be updated, rather it takes the current view and generates the next commands.

6. EXPERIMENTAL WORK

Experiments were performed using simulated sensors to test the performance of Bayes nets as a means for multimodal sensor fusion and of the system as a whole. This involved simulating streams of input data from sensors such as sonars, infra-red proximity sensors, bump sensors, cameras (image processing and object recognition), audio receivers, speech analyzers, GPS, and wheel encoders. The simulated data represented sensor readings that were observed when moving the wheelchair from one end of the university campus to another multiple times.

Early results show that using Bayesian networks for mulitmodal sensor fusion is very useful. We were able to identify and predict obstacles with greater probability than that obtained from any of the sensors individually. The system was also able to correctly predict the presence of cars and people, even when none of the sensors by themselves were able to do it. Figure 4 shows the performance of the car finding Bayes net against the audio and vision sensors individually. The Bayes net was able to predict the absence or presence of a car with greater certainty than the two sensors alone, as well as handle the cases where the two sensors gave conflicting results. Also, the Bayes net successfully predicted the presence of roads, even though no information about roads was received from any sensor.

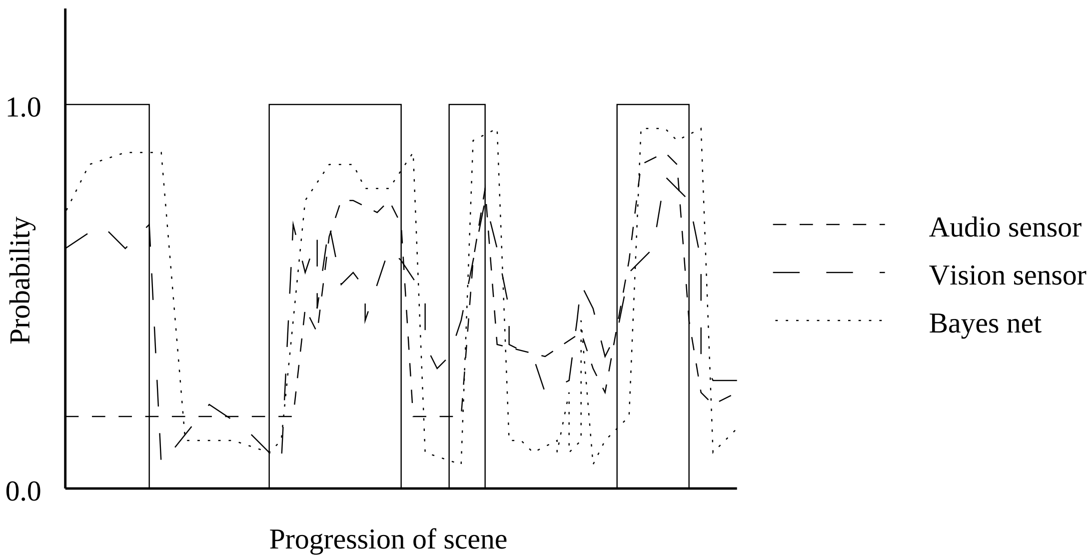

Figure 4. Performance of various sensors and the car-finding Bayes Net. The solid line represents when a car was actually present or absent in the scene.

7. FUTURE WORK AND CONCLUSIONS

We have proposed a probabilistic Bayesian network based method for performing multimodal sensor fusion. Preliminary experimental results prove the soundness and correctness of the proposed method. The system is able to predict the presence or absence of objects in the environment with higher degree of certainty. It is also able to employ limited reasoning abilities encoded in the Bayesian networks to predict the presence or absence of related objects that are not sensed by the various sensors. Another advantage of the system is that it is able to fuse sensor data from differing sensor modalities, eg. audio and vision sensors for predicting cars.

The Bayesian networks form an integral part of our proposed framework for mobile robot navigation. In the future, we plan to implement the complete system on the wheelchair robot and analyze its performance in real time. We also plan to test the system in different environments to determine its generalizability.

REFERENCES

1. R.A. Brooks and J.H. Connell. "Asynchronous Distributed Control System for a Mobile Robot," *Proceedings of SPIE*, vol. 727, Oct 1986, pp 77-84.
2. Y. Cheng, R.C. Kashyap. "A Study of Associative Evidential Reasoning," *IEEE Transactions Pattern Analysis and Machine Intelligence*, 1989, pp 623-631.
3. J.L. Crowley. "Mathematical Foundations of Navigation and Perception For an Autonomous Mobile Robot," *Reasoning With Uncertainty in Robotics 1995*, Amsterdam, The Netherlands, 4-6 Dec, 1995, pp 9-51.
4. A. Elfes. "Occupancy Grids : A Probabilistic Framework for Robot Perception and Navigation," *Ph.D. thesis*, Carnegie Mellon University, Pittsburgh, PA, May 1989.
5. G. Hager, M. Mintz. "Searching for Information," *Proceedings of the AAAI Workshop on Spatial Reasoning and Multisensor Fusion*, Los Altos, CA, 5-7 Oct, 1987, pp 313-22.
6. P.V. Kaenel. "Cobra: A General Bayesian Network Class in C++," Technical Report DRAFT, University of Rochester, Rochester, NY, May 1994.
7. G. Kamberova, M. Mintz. "Robust Multisensor Fusion : A Decision Theoretic Approach," *Proceedings of the 1990 DARPA Image Understanding Workshop*, 1990, pp 867-73.
8. D. Kortenkamp et. al. "Mobile Robot Exploration and Navigation of Indoor Spaces Using Sonar and Vision," *Proceedings of CIRFFSS'94*, Houston, TX, 21-24 Mar, 1994, pp 509-19.

9. R.C. Luo and M.G. Kay. "Data Fusion and Sensor Integration : State-of-the-Art 1990s," *Data Fusion in Robotics and Machine Intelligence*, eds. M.A. Abidi and R.C. Gonzalez, Academic Press Limited, San Diego, CA, 1992, pp 7-136.
10. M.C. Martin and H.P. Moravec. "Robot Evidence Grids," *CMU-RI-TR-96-06*, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Mar 1996, pp 1-48.
11. J. Pearl. *Probabilistic Reasoning in Intelligent Systems,* 2nd ed., Morgan Kaufmann Publishers, San Francisco, CA, 1988.
12. R. Smith and P. Cheeseman. "On the Estimation and Representation of Spatial Uncertainty," *International Journal of Robotics Research*, vol. 5, no. 4, Winter 1987, pp 56-68.