Trade-offs in the Use of Bayesian Filtering for Sensor Fusion

Anatole Gershman
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
anatole.gershman@cs.cmu.edu

Rayid Ghani
Gang Wei
Accenture Technology Labs
Chicago, IL 60601
rayid.ghani@accenture.com
gang.wei@accenture.com

Damian Roqueiro
University of Illinois Chicago
Chicago, IL 60606
roqueiro@cs.uic.edu

ABSTRACT

Robust identification and localization of moving objects via sensor networks relies on the quality of sensors, sensor coverage and the fusion of the information obtained from the sensors. Sensor fusion algorithms use domain and contextual knowledge to create the most plausible interpretation of all available data. Sensor fusion has been ac active researhc topic but little research has been done on the trade-offs between the factors that affect the performance of fusion systems. This paper presents an empirical study of these trade-offs for a classic sensor fusion algorithm based on Bayesian forward and backward propagation. To test the robustness of this algorithm under various conditions, we collected data by tracking 31 people in 112 locations using a set of 34 cameras and 70 badge readers, and created "virtual sensors". The performance characteristics of these virtual sensors were based on the data gathered. Our results show that even a relatively simple domain model enables robust performance of the algorithm even in the presence of poor sensors, thus providing a promising alternative to the expensive practice of installing better sensors and calibration procedures in order to improve surveillance systems.

Categories and Subject Descriptors

H.4.0 [**Information Systems**]: Information Systems Applications – *general*

Algorithms, Experimentation, Security

Keywords

Sensor Fusion, Bayesian Belief Propagation.

1. INTRODUCTION

The goal of most automated surveillance applications is the detection of specific or unusual activities. This requires the ability to identify and track moving objects and people based on the information provided by various sensors such as video cameras, motion detectors, smart badges and tags. Despite a great deal of research, especially in the area of video surveillance, this problem remains hard because of the inherent limitations of physical sensors, incomplete coverage of the network, and environmental constraints. The most common approach to solving this problem is the use of sensor fusion techniques that integrate information obtained from the sensors with domain knowledge to achieve the most plausible interpretation of observed reality. See [1] and [2] for an overview of video surveillance techniques and [3] and [4] for an overview of tag-based localization systems. Typically, research in this areas involves building systems that use a specific set of sensors (identity-based or anonymous) in a particular configuration (static or mobile), integrate the available domain knowledge into an ever more sophisticated fusion algorithm and report the achieved results. These systems serve as good case studies but may not generalize to systems using a different set of sensors with different levels of individual performance, placed in different environments. There has been very little attention given to analyzing the robustness of the fusion algorithms and the trade-offs between the quality and the number of sensors, the fidelity and the complexity of different elements of the domain model and the integration algorithms. Yet, these trade-offs are of great practical importance.

In order to improve the performance of localization systems, three different strategies can be used:

a) Improving the performance of individual sensors

b) Increasing the number of deployed sensors which increases the coverage and creates redundancy that can be exploited

c) Using additional domain knowledge to improve the effectiveness of sensor fusion.

Each of these strategies has its advantages and disadvantages. Improving the performance of physical sensors can be very costly or even impossible in many cases; increasing the number of poor-quality sensors requires more complex algorithms to interpret the results accurately; incorporating more domain knowledge can be expensive to create and maintain as it often requires considerable human effort.

This paper presents an empirical study of the trade-offs between these strategies in the context of people tracking using a sensor network in an office environment. The data used in this paper is obtained from sensors deployed in an office environment but our results generalize to other environments where people and object tracking takes place. The goal of our work was not to come up with another algorithm for sensor fusion so we used a fairly

KDD 2007 Workshop on Knowledge Discovery from Sensor Data

classic forward-backward Bayesian belief propagation algorithm BBP (Bi-directional Bayesian Propagation) similar to the one described in [5] to determine "topological" locations (specific office or a segment of a hallway) of a moving person. We tested the robustness of BBP along two dimensions: (1) different levels of domain knowledge and (2) various levels of individual sensor performance (recall and false positive rates), coverage, and breakdown rates. To create different testing conditions, we used "virtual sensors" whose performance characteristics, while based on the physical sensors, can be "stretched" to give us additional data points. The "virtual sensor" models and the "ground truth" data for testing were collected by tracking 31 people in 112 locations through a set of 34 cameras and 70 badge readers. Our results show that even a relatively simple domain model enables robust performance of the algorithm even in the presence of poor sensors, thus providing a promising alternative to the expensive practice of installing better sensors and calibration procedures in order to improve surveillance systems. We believe that the results that we show while studying these tradeoffs for BBP are certainly significant but our contribution in this paper also includes providing 1. A framework for studying these tradeoffs for any sensor fusion algorithm by using "virtual sensors" and 2. A labeled, real-world data set that was collected in our office and is now being made available for research use by the community at http://labs.accenture.com/data/.

We start by giving a more detailed description of the problem putting it in the context of previous works in this area. Second, we discuss our sensor model that allows us to generate simulated sensor readings based on real data collected in our environment. Third, we present and discuss the results of our experiments.

2. RELATED WORK

Following the taxonomy of Bayesian techniques for location estimation proposed in [3], our work falls into the "topological" group. We divide the space into discreet non-overlapping locations corresponding to offices, hallway segments, laboratories, etc. The goal of the topological tracking system is to determine the location of a person at each time "tick" (typically 1 second) of the observation period. Because we do not try to determine the X,Y coordinates of a person, we do not use either Kalman or Particle filter-based techniques. In terms of sensor fusion techniques, our work is most closely related to the SmartMoveX system developed at Microsoft Research and described in [6]. That system used RF active badges while we were using a combination of IR badges and surveillance video cameras. We will show how both active badge and camera-based systems can be modeled as "standard virtual sensors" to be treated uniformly by the sensor fusion algorithm.

[7] describes a more advanced system compared to SmartMoveX which uses two different types of sensors (IR badges and Ultra-sound "crickets") to determine most likely positions of a person on the Voronoi graph model of the Intel Research Lab in Seattle. Voronoi graphs is a compromise between the topological and the X,Y grid-based approaches. A person's position is estimated along the edges of the graph connecting discreet locations in an office environment. The authors report better efficiency and precision as compared to unconstrained particle filtering. Unfortunately, the paper does not report any comparison with "simple" topological localization techniques either in terms of accuracy or efficiency. We felt that topological localization is often sufficient for many applications and decided not to use Voronoi graphs or X,Y grid approaches in our study.

While active badges and other signal-emitting devices are particularly suitable for people localization, they have one major non-technological drawback – often people don't wear them or borrow each other badges since they might be required for access to certain areas. This observation is true in general but also specifically in our environment where we have had an active badge system deployed for a number of years. This was one of the main reasons why we wanted to integrate active badges with a passive video-surveillance system. There is a large body of literature on the use of video cameras for people tracking, see [2] for an overview. Our goal was to create a framework that enabled us to treat video cameras as "standard virtual sensors" for identity and location similar to active tags with the ultimate goal to integrate them into a hierarchical activity-detection system along the lines presented in [4]. This framework is described in the next section. Video-specific techniques we used and the related work are described in [8].

3. SENSOR MODEL

We have an environment with N locations $l_1, \ldots l_N$ where several people are moving around engaged in various tasks. A variety of sensors, from webcams, to badge and fingerprint readers, are used to observe and identify all the people in the area. The period of observation is divided into T short time intervals that we call time ticks (one second, for example). At each time tick t we obtain evidence E_t from all sensors that are active during that time tick. In our experiments, we tracked each person individually and independently of all other people. We will discuss the implications of this decision in the appropriate context throughout the paper. In the following discussion we assume that we are tracking a person named Joe. Joe may or may not be wearing an active tag and may be observed by one or more cameras.

Real, physical sensors may cover several locations and their outputs can take many different forms. To facilitate the integration of evidence from different types of sensors, we introduce a "standard" sensor model. In this model, at each time tick each sensor produces an output symbol from a finite set $Z=\{z_i\}$. The simplest sensor is the binary sensor with outputs 0 and 1, where 1 means that the sensor sees Joe and 0 means that it does not. A more nuanced sensor might produce symbols for "high confidence of seeing Joe", "low confidence of seeing Joe", "high confidence of not seeing Joe", and "low confidence of not seeing Joe". Some of the sensors may provide more information than the presence or absence of a person. A camera looking down a hallway may cover more than one section which we designate as separate locations. It can use the size of the extracted blob relative to the stationary objects in the scene and the direction to the blob to place Joe in a specific location. Similarly, the SmartMoveX system uses only 4 physical sensors (RF receiving stations) which cover all locations and from which they obtain the equivalent of a virtual sensor for each location under observation. To account for these capabilities, we "break" the sensors that cover more than one location into several virtual sensors each of which covers only one location.

The most important part of the sensor model is the relationship between its outputs and the physical conditions the sensor is supposed to report. In the HMM literature it is often referred to as an *observation* or *perceptual* model [1, 5]. The model can be represented as a table of conditional probabilities $b_j(k)=P(z_k|l_j)$ – of observing output z_k assuming that Joe is in

location l_j. These probabilities can be obtained either through empirical observation or through careful modeling of sensor properties.

The creators of the SmartMoveX system point out in [6] that it is very difficult either to construct an adequate physics-based model or to gather enough samples to cover every possible position and orientation of a person in each location. Instead, they used several dozen reference signal strength profiles for each location. Given the actual measurement, they picked the most similar profile and then computed the probability of the measurement assuming that the selected profile was the "true" position of the person. Similarly, we use several video frames of Joe (plus a few other characteristics) in each location as his profiles, compare the extracted blobs to them and select the one with the best match. We then calculate the likelihood of that match and use it as the output of the virtual sensor based on that camera. We use this output as an approximation of its conditional probability given Joe is in the location of the virtual sensor (see [8] for further details).

To compute the conditional probability of all evidence at the time tick t, we need to combine the evidence obtained from each sensor. To simplify the calculations, we are making a critical assumption of conditional independence of all sensor readings, provided that we know Joe's location. While this assumption seems to be reasonable, we did not conduct thorough experiments to verify it. With this assumption, the conditional probability $O_{j,t}=P(E_t|l_j)$ is simply the product of the conditional probabilities of the readings of all sensors.

The perceptual model of the sensor represented by the conditional probabilities table fully describes the behavior of the sensor with respect to a tracked person. To study the sensitivity of the sensor fusion algorithms with respect to the quality of sensors, we need to be able to vary this quality. To be able to do that, first we need to define what we mean by sensor quality. We can treat a sensor in location l_j as a binary classifier of Joe's location discriminating between l_j and *not* l_j, assuming that no other information is available and using the maximum likelihood decision rule. Two of the common performance measures for binary classifiers are *sensitivity* (or *recall*) and *specificity*. We did not use *precision* because under the above assumptions, *precision* is equal to *recall*. To make the measure more intuitive, we used the *false positive rate = 1 – specificity*. The *false positive* rate is the ratio of the number of false positive instances to the total number of false positive and true negative instances. Clearly, higher *recall* and lower *false positive* rates make better sensors.

In reality, changing the recall or the false positive rates of a physical sensor is almost impossible. This is true even for video cameras where the performance of the sensors is based more on the image-processing and vision algorithms than on the sensor hardware. We can spend years improving a blob extraction algorithm to gain a few percentage points in the sensor performance measures which may or may not significantly improve the overall accuracy of a localization system. We need a more practical approach to testing a sensor fusion system under a broad spectrum of sensor performance characteristics. In the experiments reported in this paper, we used virtual sensors which were created using actual deployed sensors in our environment but allow us to simulate and vary the behavior of the physical sensors at the higher and lower performance levels.

In every use of simulation to approximate realistic sensor readings, we must examine the limitations of our assumptions. We make three conditional independence assumptions which can be violated under certain circumstances. The condition is knowledge of the tracked person's location. These assumptions are:

1) Temporal independence – conditional independence of each sensor readings at two consecutive time ticks
2) Sensor independence – conditional independence of any two sensors looking at the same location
3) Subject independence – conditional independence of any two sensors looking at two different people

There are many situations when temporal independence of sensor readings can be violated. If a sensor malfunctions at time t, it is very likely to continue to malfunction at time $t+1$; or if a person puts his IR badge in his pocket or borrows someone else's badge the effects will persist over a period of time. Similarly, camera-based sensors can be persistently fooled by people borrowing each other jackets, by people remaining motionless, or by pieces of furniture that look like a person. Sensor independence can be violated by changing lighting conditions or by the use of disguises which affect both camera-based sensors. Subject independence is violated when the presence of several people can causes correlated errors in both cameras.

Some of these conditions, such as lighting changes, sensor malfunctions, motionless people or the presence of several people can be detected and the sensor readings appropriately adjusted. Other conditions such as use of disguises are much harder to detect and include in the sensor model. We did not encounter these problems in our data but the degree to which each of these assumptions will be violated depends on the specifics of the environments in which a sensor fusion system is deployed. In this paper, we did not address this issue but some of the future work involves coming up with diagnostic metrics that can measure the violations of our assumptions by analyzing the data collected in the environment and using active learning techniques to recover from the errors caused by these violations.

4. DOMAIN MODELS

The basis for sensor fusion is a domain model. Our domain model is a first order HMM composed of NxT states, where N is the number of locations and T is the number of time ticks in the observation period. Joe being in location l_j at time t defines the state $q_{j,t}$. The knowledge that makes the model useful has two parts: the prior probability distribution for Joe $\pi_j=P(l_j)$ and the transition probabilities for Joe moving from one location to another or for staying in the same location: $\psi_{i\rightarrow j}=P(q_{j,t+1}|q_{i,t})$. Different levels of domain knowledge lead to different levels of performance accuracy for the localization system. We used four levels of domain knowledge:

1. **Null Hypothesis** – no knowledge at all; all prior probabilities are equal and all transitions are equally probable
2. **Topological semantics** – takes walls and obstacles into account and only allows transitions to adjacent locations
3. **Location semantics** – distinguishes between offices, hallways, conference rooms, etc.
4. **Personal semantics** – takes into account personal patterns of behavior such as staying in one's office, visiting colleagues, etc.

Clearly, the more knowledge is embedded in the system, the better its performance should be. The question is by how much, and how this difference depends on the quality of sensors.

The domain models we are using are far from perfect. For example, we are assuming independence between the people we

are tracking which is problematic on two counts. First, the sensor accuracy may drop when the location is cluttered with several people. Second, the transition probabilities may change if, for example, Joe's boss is in the hallway in front of Joe's office making it more likely for Joe to come out. Our models should be considered as points of comparison for more sophisticated approaches.

5. THE SENSOR FUSION ALGORITHM

Our BBP sensor fusion algorithm is based on the forward-backward algorithm described in [5]. Rather than computing the Viterbi path – the most likely path given the evidence, we compute the probability distribution for each person (or object) among all locations at all points in time. With a minor modification the same algorithm can compute both the state probabilities and the Viterbi path. For completeness, we reproduce here the key steps of the algorithm.

The basis of the BBP algorithm is a simple observation that people do not disappear from one location to simply appear at the opposite end of the floor in the next second. A person's locations in two consecutive moments of time are related: in one second a person will most likely either remain in the same location or move to an adjacent location. Similarly, the current position of the person makes it more likely that this person was in one of the adjacent locations one second earlier.

We define $\alpha_{i,t}$ as the probability of Joe being in location l_i at time t, based on the evidence prior to and including t. For the first time tick, $\alpha_{i,1}=\nu\pi_i O_{i,1}$, where π_i is the prior probability for Joe being in location l_i, $O_{i,1}$ is the probability of the evidence obtained at the first tick, and ν is the normalization factor. The forward pass of the algorithm recursively computes alphas for all subsequent time ticks:

$$\alpha_{i,t+1}=\nu\, O_{i,t+1}\Sigma_{k=1:N}\, \alpha_{k,t}\, \psi_{k\rightarrow i}$$

The backward propagation computes $\beta_{i,t}$ defined as the conditional probability of all evidence subsequent to t, assuming that Joe was in location l_i at time t. All betas at the last moment of the observation period are assumed to be 1. The betas for all preceding moments are computed recursively as follows:

$$\beta_{i,t-1}=\Sigma_{k=1:N}\, \beta_{k,t}\, \psi_{i\rightarrow k}\, O_{k,t}$$

The probability $\gamma_{i,t}$ of Joe being in l_i at time t given all the evidence is the normalized product of the alphas and betas:

$$\gamma_{i,t}= \nu\, \alpha_{i,t}\, \beta_{i,t}.$$

6. EXPERIMENTAL SETUP

We describe experiments that are designed to test the robustness and performance of the BBP algorithm under varying sensor conditions and domain models. In order to base our experiments on real scenarios and real data, we conduct them using the environment that was created for the Multiple Sensor Indoor Surveillance (MSIS) project [9]. The data that we collected can be downloaded from http://labs.accenture.com/data/. The environment has 36 surveillance cameras and 91 infrared badge readers at the premises of the Accenture Technology Labs in Chicago that occupy the entire floor in a building. The floor was divided into 112 locations depicted in Figure 1. The creation of these locations was based on the following criteria:

a) Locations have to be large enough such that a person could not traverse more than one location in one time tick.
b) Locations have to be small enough so that they would require only one surveillance camera to cover it.
c) Locations are dedicated to specific activities (i.e., meeting rooms, hallways, offices).

For the purpose of determining the likelihood of different motion patterns, we have divided our locations into three main categories: 1- hallways; 2 - offices, rooms and cubicle areas; and 3 - open areas (e.g.: the lobby in front of the elevators). These areas are specific to our environment but generalize to most office locations, and even to non-office areas where dedicated areas serve specific purposes. Our sensors cover about 55 out of 112 locations.

Initially, we developed computer vision algorithms for localizing and tracking people using these sensors (cameras in our case) and the results of those algorithms gave us an average recall rate of about 50% and the false positive rate of about 2%. We tried varying the performance of these sensors artificially by degrading the vision algorithms but this degradation was not natural and did not allow us to experiment with the full range of the parameters described in our sensor model. Since the goal of our work was not to focus on one specific environment and one configuration of specific sensors, we systematically tested the fusion algorithm using virtual sensors with simulated outputs at varying levels of the recall, false positive and blackout rates. We kept the same space coverage as our physical sensors in the office environment.

To create the "ground truth" we observed the movements of 31 people during 1 hour (3600 ticks). We used the ground truth and the sensor parameters to generate Monte Carlo simulations of sensor readings to which we then applied our sensor fusion algorithm.

7. EXPERIMENTAL RESULTS

Our experiments are designed to test the performance of the Bayesian Belief Propagation under various levels of sensor performance and domain knowledge.

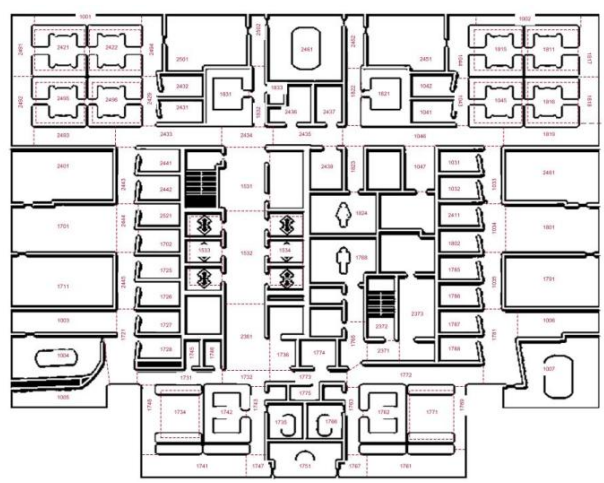

Figure 1. Experimental environment

7.1 What is the effect of adding more Domain Knowledge?

In the first set of our experiments we fixed the false positive rate of our sensors to 2% (matching the average of the physical sensors deployed in our office environment). This value was empirically determined by using computer vision algorithms on the data collected by the cameras to locate and identify the people in our office environment. More details about the algorithms and results of those experiments can be found in [9]. Using the ground truth data (the actual locations of a person during the observation period), we generated virtual sensor readings at 12 different levels of recall. We ran our sensor fusion algorithm using these readings and the four levels of domain knowledge (as described in Section 4):

1. **Null Hypothesis** – no knowledge at all; all prior probabilities are equal and all transitions are equally probable
2. **Topological semantics** – takes walls and obstacles into account and only allows transitions to adjacent locations
3. **Location semantics** – distinguishes between offices, hallways, conference rooms, etc.
4. **Personal semantics** – takes into account personal patterns of behavior such as staying in one's office, visiting colleagues, etc.

For each time tick, we picked the location with the maximum computed posterior probability (not necessarily on the Viterbi path). We compared these locations to the ground truth data at every time tick and computed the accuracy of the run. This procedure was repeated 10 times for each of the 31 tracked people yielding the average accuracy for each condition. The results of the first set of experiments are shown on Figure 2.

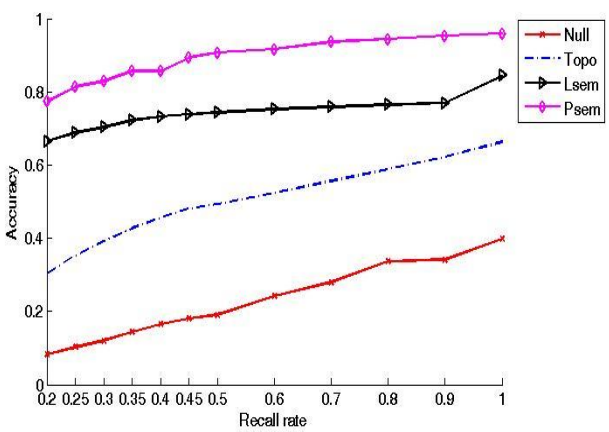

Figure 2. Knowledge level experiment

As we expected, the Null knowledge condition is very sensitive to the quality and positioning of sensors and performs the worst when the individual sensors have a very low recall rate. Adding topographical knowledge dramatically improves the algorithm's performance. Adding the additional knowledge about the patterns of use of individual locations (Lsem – Location Semantics) and the patterns of personal behavior (Psem – Personal Semantics) further improves not only the accuracy of the algorithm but also its robustness in the face of the decreasing recall rate of the sensors.

7.2 How does individual sensor performance affect overall accuracy?

In the first set of experiments, we fixed the false positive rate at 2%. The second set of experiments tested the algorithm at different false positive rates using personal semantics for domain knowledge. The results are shown on Figure 3.

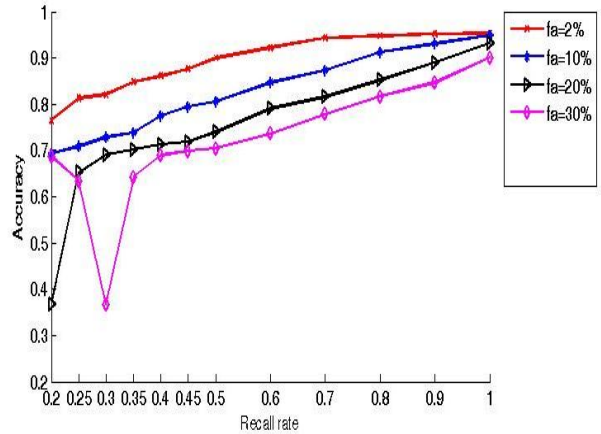

Figure 3. False positive sensitivity experiment

We can see that the decrease in accuracy is fairly gradual for a broad range of sensor settings. Even with 30% false positives and only 50% recall, our simple sensor fusion algorithm provides 70% localization accuracy.

7.3. What happens to the accuracy when some of the sensors fail?

The data we collected from our physical sensors shows that real sensors frequently experience "blackouts" – random losses of data. We factored this in our sensor models and tested our algorithm at various sensor blackout rates. Figure 4 shows the results of these tests at the 2% false positive rate using personal semantics.

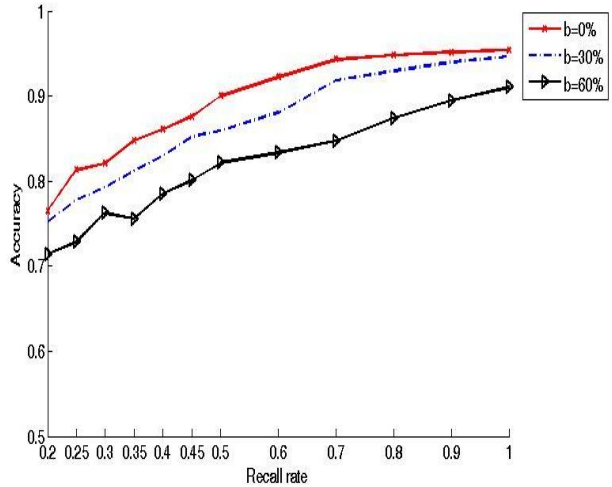

Figure 4. Blackout rate sensitivity experiment

The blackout rates of our physical sensors were below 10%, but as Figure 4 shows, that should not greatly affect the overall accuracy of our sensor fusion algorithm. Even at 30% blackouts, the accuracy does not drop by more than a couple of percentage points.

7.4 Are more sensors better?

In our fourth and final set of experiments we address the question of sensor coverage: the relationship between the number of sensors and the accuracy of the system.

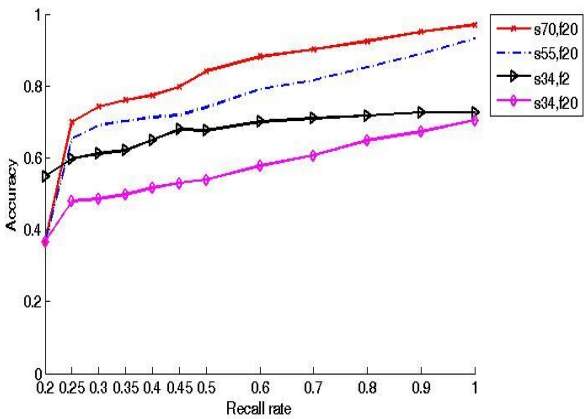

Figure 5. Coverage sensitivity experiment

The impact of sensor coverage has significant practical consequences. Suppose we had a set of 34 relatively poor but inexpensive sensors with an average recall rate of 60% and a false positive rate of 20% (the bottom curve in Figure 5). The BBP algorithm with personal semantics gives us the accuracy close to 57%. Suppose our application requires a localization accuracy of at least 80%. We could try to achieve this by boosting the recall rates and by reducing the false positive rates of our sensors – a very difficult and costly thing to achieve in the physical world. With 34 sensors, even if we could reach the recall level of 90% and the false positive rate of 2%, the average accuracy will still be about 72% (the second from the bottom curve on Figure 5). An alternative approach suggested by our experiments is to simply increase the number of sensors in the environment. Let's assume again a recall rate of 60% and a 20% false positive rate. Going from 34 sensors to 55 without any other improvements in the individual sensors gives us 79% accuracy (the second line from the top on Figure 5). If we deploy 70 sensors, the accuracy rate will go up to 88% (top line on Figure 5). With 70 sensors we can use even lower quality sensors: 70 sensors at 40% false positive rate are roughly equivalent to 55 sensors at 20% (not shown) and still much better than 34 sensors at an even harder to achieve 2% false positive rate.

By analyzing the performance of the fusion algorithm under varying levels of domain knowledge and individual sensor performance, we are able to study these tradeoffs and provide guidelines for setting up sensor environments. If a particular application required, say, 85% accuracy, our analysis would provide competing ways of achieving it – increasing the number of sensors, increasing the reliability of each sensor, increasing the performance of the sensors, creating more sophisticated domain models would all be candidate approaches. By studying the effect of these tradeoffs, we can compare the efficacy of each of the possibilities and find the optimal combination to support the application at hand.

8. CONCLUSIONS AND FUTURE WORK

As the number of applications that rely on information gathered by physical sensor networks continues to grow, the tradeoffs between the quality of sensors, their number, configuration and the sophistication of sensor fusion algorithms become increasingly important.

Most of the research in this area has typically focused on developing new algorithms optimized for specific environments and sensor configuration. There is certainly value in that research but what is also required is an understanding of the robustness of the proposed algorithms and the tradeoffs between sensor performance, sensor coverage and the sophistication of the fusion model. In this paper, we propose a general sensor model that allows us to fuse information from different kinds of sensors in a seamless manner. We also present a general framework as well as a real-world data set that can be used to study these tradeoffs for any given sensor fusion algorithm. We presented an empirical study of these trade-offs for BBP – a sensor fusion algorithm based on Bayesian forward and backward propagation. We tested its robustness under various levels of individual sensor performance (recall and false alarms rates), coverage, and breakdown rate in a people tracking task in an indoor office environment. The results show that our Sensor Fusion algorithm BBP armed with a relatively simple model of the environment enables robust performance even in the presence of poor sensors, thus providing a promising alternative to the expensive practice of installing better sensors and calibration procedures in order to improve surveillance systems.

Our contributions in this paper go beyond providing the results of these tradeoffs for one particular algorithm. We believe that 1. A framework for studying these tradeoffs for any sensor fusion algorithm by using "virtual sensors" and 2. A labeled, real-world data set that was collected in our office and is now being made available for research use by the community at http://labs.accenture.com/data/ are important contributions that allow this kind of study to be done for any sensor fusion algorithm.

We believe that the results of this study have strong practical implications for deploying sensor networks in real-world environments. Since the performance and accuracy requirements vary quite a bit for different applications, it is often critical for organizations deploying these networks to have a clear understanding of the tradeoffs involved. The results of our experiments show that there are often multiple ways to increase the localization and tracking performance. Using a more robust fusion algorithm, according to our experiments, allows organizations to increase this performance by deploying an increased number of the cheap, individually inaccurate, sensors and achieve comparable, or even better, performance than spending resources on improving the fidelity of the sensors or buying more expensive sensors. In certain cases, more accurate sensors may be out of the question – computer vision algorithms for localization and tracking have a certain level of accuracy today and increasing that accuracy to a desired level may require years of research.

There are several areas of future research that we plan on pursuing. The current work assumes a certain configuration of sensors in the environment and the sensor fusion algorithm is applied to data being captured by those sensors. We plan on investigating techniques that will allow us to optimally place these sensors in order to maximize the performance of the system. There has been work done in a related domain on optimal sensor placement (to minimize communication cost) in sensor networks designed to monitor spatial phenomena [10].

Another area we plan to investigate is the use of active learning to improve the sensor fusion system. The goal of active learning is to improve the accuracy of the system by interacting with human experts most effectively and efficiently. There are three questions we are considering when applying active learning techniques in our task: 1. When to query a human expert? 2. What should the query by about? 3. How should the feedback from the human expert be used to improve the sensor fusion system? By empirically studying the effect of different active learning strategies, we aim to build a broader system that is interactive, robust to noisy sensors,, and is able to utilize the redundancy in large-scale sensor networks for localization and tracking.

9. REFERENCES

[1] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankati, A. Senior, C. Shu, and Y.L. Tian, "Smart Video Surveillance", *IEEE Signal Processing Magazine*, March 2005

[2] G.L. Foresti, C. Micheloni, L. Snidaro, P. Remagnino, and T. Ellis, "Active Video-based Surveillance System", *IEEE Signal Processing Magazine*, March 2005

[3] D. Fox, J. Hightower, H. Kauz, L. Liao, and D.J. Patterson, "Bayesian Techniques for Location Estimation", *IEEE Pervasive Computing*, vol. 2, no 3, March 2003, pp. 24-33.

[4] J. Hightower, and G. Borriello, "Location Systems for Ubiquitous Computing", *Computer*, vol. 34, no 8, Aug. 2001, pp. 57-66.

[5] L.R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", In *Proceedings of the IEEE*, vol. 77, no 2, Feb. 1989, pp. 257-285.

[6] J. Krumm, L. Williams, and G. Smith, "SmartMoveX on a Graph: An Inexpensive Active Badge Tracker", In *Proceedings of the International Conference on Ubiquitous Computing* (Ubicom 02), LNCS Springer-Verlag, 2002, pp. 299-307.

[7] L. Liao, D. Fox, J. Hightower, H. Kautz, and D. Schulz, "Voronoi tracking: Location estimation using sparse and noisy sensor data", In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems.* (2003).

[8] V.A. Petrushin, G. Wei, and A.V. Gershman, "Multiple-Camera People Localization in an Indoor Environment", *Knowledge and Information Systems*, vol. 10, no 2, 2006, pp. 229-241.

[9] G. Wei, V.A.Petrushin, and A.V. Gershman, "Multiple-Camera People Localization in an Office Environment", in *Multimedia Data Mining and Knowledge Discovery*, Petrushin, V.A. and Khan, L. (eds.), Springer Verlag, (2007) 427–447

[10] C. Guestrin, A. Krause, A. Singh. Near-Optimal Sensor Placements in Gaussian Processes. *Proceedings of the International Conference on Machine Learning*. ICML (2005).