A Comparison of HMMs and Dynamic Bayesian Networks for Recognizing Office Activities

Nuria Oliver and Eric Horvitz

Adaptive Systems & Interaction
Microsoft Research
Redmond, WA USA
{nuria,horvitz}@microsoft.com

Abstract. We present a comparative analysis of a layered architecture of Hidden Markov Models (HMMs) and dynamic Bayesian networks (DBNs) for identifying human activites from multimodal sensor information. We use the two representations to diagnose users' activities in S-SEER, a multimodal system for recognizing office activity from real-time streams of evidence from video, audio and computer (keyboard and mouse) interactions. As the computation required for sensing and processing perceptual information can impose significant burdens on personal computers, the system is designed to perform selective perception using expected-value-of-information (EVI) to limit sensing and analysis. We discuss the relative performance of HMMs and DBNs in the context of diagnosis and EVI computation.

1 Introduction

We explore in this paper a better understanding of the relative performance of Hidden Markov Models (HMMs) and dynamic Bayesian networks (DBNs) for recognizing office activities within a component of a multilevel signal processing and inference architecture, named S-SEER. S-SEER is a multimodal probabilistic reasoning system that provides real-time interpretations of human activity in and around an office [15, 13]. Our research to date on the system has addressed two main challenges. On one front, we have explored the use of a hierarchical reasoning architecture for processing low-level signals into higher-level interpretations. We have demonstrated several valuable properties of the multilevel architecture, including its value in significantly shrinking the dimensionality of the parameter space, thus reducing the training requirements of the system [15]. On another front, we have investigated the use of value of information to limit computation by selecting in a dynamic manner specific subsets of sensors to use. We have shown how the selective use of sensors and associated computation reduces the overall computational burden in return for small degradations in the accuracy of the system [13].

To date, we have employed HMMs at all levels of S-SEER. In this paper we extend S-SEER with a comparative analyis of HMMs and DBNs at the highest level of reasoning. The research was motivated by the challenge of reasoning with unobserved sets of variables –a situation underscored by our work with selective perception.

This paper is organized as follows: We first provide background on multimodal systems in Sect. 2. Section 3 describes our work on learning dynamic graphical models (HMMs and DBNs) to model office activities. In Sect. 4 we briefly describe the decision-theoretic selective perception strategy that we have incorporated in S-SEER. Section 5 provides background on the S-SEER system. Experimental results with the use of a layered architecture of HMMs and DBNs in S-SEER are presented in Sect. 6. We also perform a supportive study to probe the value of richer temporal relationships among states and unobserved variables with DBNs. Finally, we summarize our work in Sect. 7.

2 Prior Related Work on Human Activity Recognition

We shall review here some of the most relevant previous work on human activity recognition from perceptual data using dynamic graphical models. For a more complete overview of the prior related work, we direct the reader to [15, 13].

Most of the early work in this area centered on the identification of a specific activity in a particular scenario, and in particular, single events such as "waving the hand" or "sitting on a chair". More recently there has been increasing interest on modeling more complex patterns of behavior, and especially patterns that extend over long periods of time. Hidden Markov Models (HMMs) [16] and extensions have been one of the most popular modeling techniques. Some of the earliest work was done by Starner and Pentland in [18] where they used HMMs for recognizing hand movements in American Sign Language and by Oliver *et al* [12] to recognize facial expressions. More complex models, such as Parameterized-HMMs [19], Entropic-HMMs [1], Variable-length HMMs [7], Coupled-HMMs [2], structured HMMs [17] and context-free grammars [8] have been used to recognize more complex activities such as the interaction between two people.

Moving beyond the HMM representation and solution paradigm, researchers have investigated more general temporal dependency models, such as dynamic Bayesian networks (DBNs) (also known as dynamic graphical models). DBNs have been adopted by several researchers for the modeling and recognition of human activities [3, 5, 10].

HMMs can be viewed as a specific case of the more general dynamic graphical models, where particular dependencies are assumed. Thus, HMMs and their variants can be interpreted as examples of DBNs.

DBNs present several advantages to the problem of user modeling from multi-sensory information: they can handle incomplete data as well as uncertainty; they are trainable and provide means for avoiding overfitting; they encode causality in a natural way; algorithms exist for learning the structure of the networks and doing predictive inference; they offer a framework for combining prior knowledge and data; finally, they are modular and parallelizable. However, they pose, in the general case, difficult inference problems, especially with loopy graphs and continous data. Several efficient optimizations available for learning and solving HMMs are not available for general DBNs.

With different representations available, there is still the open question of how suitable a particular representation might be for a specific task. We explore in this paper the power and tradeoffs of HMMs versus more general DBNs when applied to the task of recognizing in real-time typical office activities from sensor data. Our main contribution is a comparison of a layered architecture of HMMs with a layered architecture of HMMs and DBNs for modeling office activities. We examine base-level inference as well as the use of value of information to select the best subset of sensors to use.

3 Layered Dynamic Graphical Models for User Modeling

We shall now review the layered dynamic graphical model approach that we have used for modeling the user's behavior in and around the office. We direct the reader to [15] for more detail on the motivation of our layered architecture and its performance compared to standard single-layer HMMs.

3.1 Layered HMMs (LHMMs)

In [15] we describe the use of a multilayer representation of HMMs, named LHMMs, that reasons in parallel at different levels of temporal detail. Such an architecture has the ability to decompose the parameter space in a manner that reduces the training and tuning requirements. Each layer of the architecture is connected to the next layer via its inferential results. The representation segments the problem into distinct layers that operate at different temporal granularities[1] —allowing for temporal abstractions from pointwise observations at particular times into explanations over varying temporal intervals. This architecture can be characterized as a *stacked classifier*.

The layered formulation makes it feasible to decouple different levels of analysis for training and inference. As we review in [15], each level of the hierarchy is trained independently, with different feature vectors and time granularities. Thus, the lowest signal-analysis layer that is most sensitive to variations in the environment can be retrained, while leaving the higher-level layers unchanged.

3.2 Layered HMMs and DBNs

We focus here on extending the layered HMM architecture to include DBNs at the highest level,while the lower level is still based on HMMs for simplicity[2].

We learn the DBNs from observed data using structural learning [6, 11]. In particular, we have extended a Bayesian network tool named WinMine [4] developed by Microsoft Research, to consider variables at different time steps and therefore learn a DBN. WinMine uses a Bayesian score to learn the structure and parameters of the model, given some basic constraints supplied *a priori*, such as prohibiting edges between nodes at time t and nodes at time $t-1$, *i.e.* forcing the connections to be either co-temporal or go forward in time. The learned distributions are decision trees and the Bayesian score is used to choose the splits in the trees. The tree-growing algorithm for Bayesian networks is to score every possible split in every leaf of every node, and then perform the best one that does not result in a cycle in the network (a split in a tree corresponds to a parent in the DBN).

4 Decision Theoretic Selective Perception

An important challenge in multimodal real-time perceptual systems is CPU consumption. Processing video and audio sensor information to make inferences usually consumes a large portion of the available CPU time. We integrated into

[1] The "time granularity" in this context corresponds to the window size or vector length of the observation sequences in the HMMs.

[2] This level interfaces with the sensor data which is a continous dynamic time series.

S-SEER several methods for selecting features dynamically [13], including an *EVI-based* method, based on calculations of the expected value of information. In the experiments described in [13] we studied the performance and overall computational cost of the system using these methods.

In this paper we focus on using an *EVI-based* method to perform real-time, one step look-ahead sensor selection both in HMMs an DBNs.

5 Implementation of S-SEER

S-SEER consists of a two-level architecture with three processing layers as illustrated in Fig. 1. For a more detailed description we direct the reader to [15, 13].

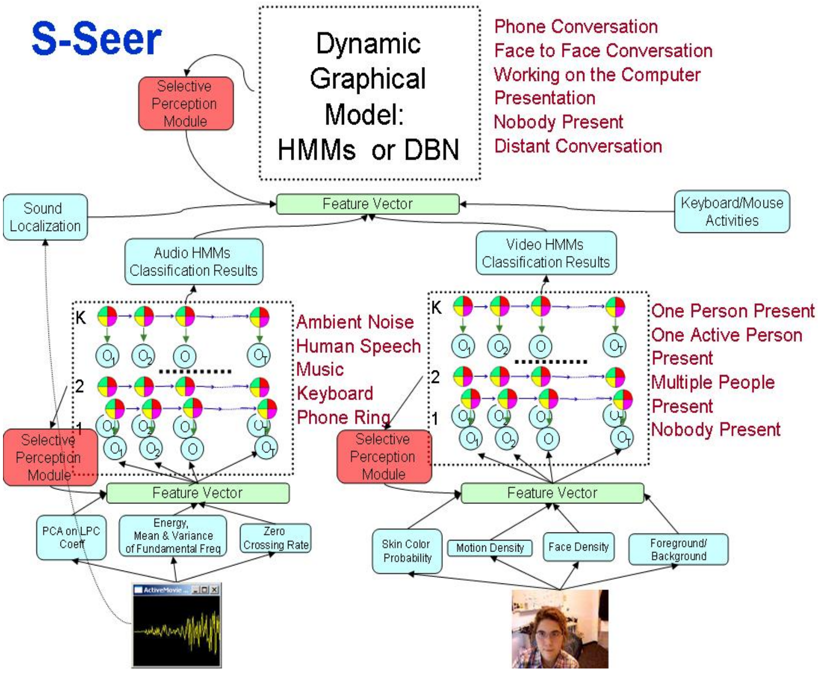


Fig. 1. Architecture of S-SEER.

5.1 Sensors and Feature Extraction

In S-SEER we explore the challenge of fusing information from three different sensors. The raw sensor signals are preprocessed to obtain feature vectors (*i.e.* observations) for the first layer of HMMs.

(1) **Audio**: Two mini-microphones ($20 - 16000$ Hz, SNR 58 dB) capture ambient audio information. They are used for sound classification and localization. The audio signal is sampled at 44100 KHz. We compute Linear Predictive

Coding coefficients [16] on the audio signal. Feature selection is applied to these coefficients via principal component analysis. We select the number of coefficients such that at least 95% of the variability in the data is kept, which is typically achieved with no more than 7 features. We also extract higher-level features from the audio signal such as its energy, the mean and variance of the fundamental frequency over a time window, and the zero crossing rate [16]. The source of the sound is localized using the Time Delay of Arrival (TDOA) method.

(2) **Video**: A standard Firewire camera, sampled at 30 f.p.s, is used to determine the number of persons present in the scene. We extract four features from the video signal: the density[3] of skin color pixels in the image (obtained by discriminating between skin and non-skin models, consisting of histograms in YUV color space), the density of motion pixels in the image (obtained by image differences), the density of foreground pixels in the image (obtained by background subtraction, using an adaptive background technique), and the density of face pixels in the image (obtained by means of a real-time face detector [9]).

(3) **Keyboard and Mouse**: A history of the last 1, 5 and 60 seconds of mouse and keyboard activities is logged.

5.2 Continuous HMMs at the First Level

The first level of HMMs includes two banks of distinct HMMs for classifying the audio and video feature vectors. The feature vectors at this level are a stream of continous floating point data. The structure for each of these HMMs is determined by means of cross-validation on a validation set of real-time data. On the audio side, we train one HMM for each of the following office sounds: *human speech, music, silence, ambient noise, phone ringing*, and the sounds of *keyboard typing*. In the architecture, all the HMMs are run in parallel. At each time slice, the model with the highest likelihood is selected and the data –*e.g.* sound in the case of the audio HMMs– is classified correspondingly. We will refer to this kind of HMMs as *discriminative* HMMs. The video signals are classified using another bank of discriminative HMMs that implement a person detector. At this level, the system detects whether *nobody, one person (semi-static), one active person, or multiple people* are present in the office. Each bank of HMMs can use selective perception strategies [13] to determine which features to use.

5.3 Second Level Dynamic Graphical Models

The next level in the architecture processes the inferential results[4] from the previous layer (*i.e.* the outputs of the audio and video classifiers), the derivative of the sound localization component, and the history of keyboard and mouse activities. This layer handles concepts with longer temporal extent and of discrete nature. Such concepts include the user's typical activities in or near an office. In particular, the activities modeled are: (1) *Phone conversation*; (2) *Presentation*; (3) *Face-to-face conversation*; (4) *User present, engaged in some other activity;* (5) *Distant conversation* (outside the field of view); (6) *Nobody present*. Some of

[3] By "density" we mean the number of pixels that satisfy a certain property, divided by the total number of pixels.

[4] See [14] for a detailed description of how we use these inferential results.

these activities can be used in a variety of ways in services, such as those that identify a person's availability.

This is the level of description where we have implemented and compared two different models: discrete HMMs and DBNs, both learned from data.

(1) HMMs: A bank of discriminative HMMs with selective perception policies to determine which inputs from the previous layer to use. Figure 2 (a) (left) illustrates the architecture with HMMs at the highest level.

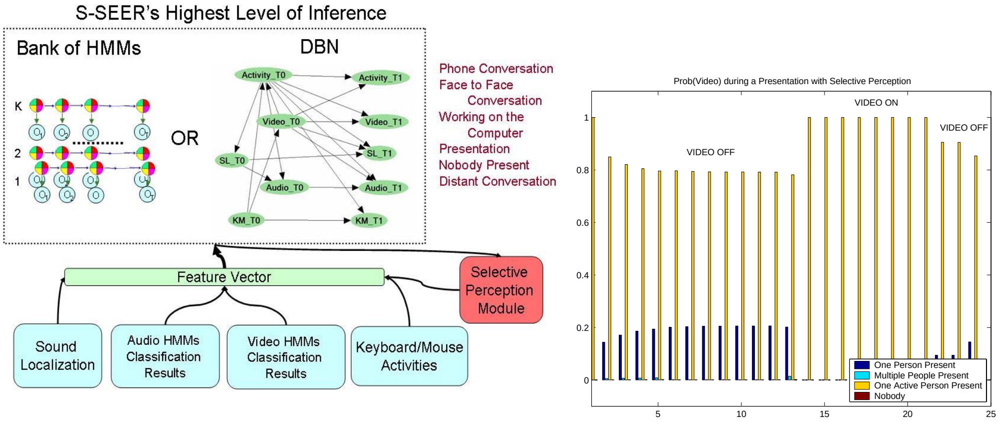
(2) DBNs: A single DBN with selective perception and a hidden "Activity" node is learned from data. Figure 2 (a) (right) depicts the network learned and used in our experiments.

The figure shows two time slices of the DBN, corresponding to time $T0$ and time $T1$. The complete network consists of extending the DBN up to time $T9$, *i.e.* for 10 time steps. There are five different discrete variables to be modeled, all of them with a subscript corresponding to the time slice: "Activity", which is a hidden variable that contains the value of current activity that is taking place in the office, i.e. (0) *Phone conversation*; (1) *Presentation*; (2) *Face-to-face conversation*; (3) *User present, engaged in some other activity;* (4) *Distant conversation* (outside the field of view); (5) *Nobody present*; "Video", an observed variable that contains the inferential results of the bank of HMMs classifying the video signal. It has one of the following values: (0) *One person present*; (1) *Multiple people present*; (2) *One active person present*; (3) *Nobody present*; "Audio", an observed variable corresponding to the inferential results of the bank of HMMs classifying the audio signal. Its possible values are: (0) *Ambient Noise*; (1) *Speech*; (2) *Music*; (3) *Phone Ringing;* (4) *Keyboard typing*; "SL", an observed variable with the sound localization results: (0) *Left of the monitor*; (1) *Center of the monitor*; (2) *Right of the monitor*; "KM", an observed variable with the history of keyboard and mouse activities. Its values are: (0) *No activity*; (1) *Current Mouse Activity*; (2) *Current Keyboard Activity*; (3) *Keyboard or mouse activity in the past second.*

The learned model highlights the enhanced expressiveness of more general dynamic graphical models. Note how the learned structure of the DBN differs from that of an HMM. The DBN has new dependencies that are missing on the HMM, such as the edge between the keyboard and mouse node and the video node, the edge between the video node at time $T0$ and the sound localization node at time $T1$, and the edge between the video node at time $T0$ and the audio node at time $T1$. The DBN has discovered in the data: (1) A co-temporal dependency between the sound localization and the audio nodes, and between the keyboard and mouse, and the video nodes; (2) A causal relationship between the presence information obtained from the video sensor and the audio and sound localization nodes. These new connections make intuitive sense. For example, if the keyboard and mouse are in use at time $T0$ it is very unlikely that the video sensor would determine that there is nobody there at that same time $T0$; or if the vision sensor detects that there is one person present at time $T0$, it is quite likely that there will be some speech at time $T1$ and that the sound will come from the center of the monitor.

6 Experiments

In our experiments we were particularly interested in comparing: (1) The accuracy of HMMs versus DBNs with and without selective perception, and (2)



(a) (b)

Fig. 2. (a) Highest level of S-SEER with HMMs (left) and a DBN (right); (b) Evolution over 25 consecutive time slices of the probability distribution of a "Video_T" node in the DBN with selective perception and during a *Presentation*.

evaluating the advantages and disadvantages of both models from a practical perspective.

We trained S-SEER both with HMMs and with the DBN at the highest processing level[5] with 1800 samples (300 samples per activity) of each of the office activities of interest, i.e., *Phone conversation*; *Presentation*; *Face-to-face conversation*; *User present, engaged in some other activity; Distant conversation* (outside the field of view); *Nobody present*. All the samples in the experiments below correspond to the same user. We used leave-one-out cross-validation to determine that 10 was the optimal number of time steps for the DBN.

To test the performance of both models we collected about 90 minutes of activity data (about 15 minutes per activity). We ran accuracy tests of the HMMs and the DBN with and without selective perception. The results are displayed in Table 1 (a) where we use the abbreviations: PC=Phone Conversation; FFC=Face to Face Conversation; P=Presentation; O=Other Activity; NP=Nobody Present; DC=Distant Conversation.

Observations that can be noted from our experiements are that the DBN has better recognition accuracies than HMMs for the problem we are solving, and that employing selective perception policies leads to a more significant degradation in the performance of HMMs than that of the DBN. An important factor for this difference in behavior is how unobserved variables are treated in each model. In HMMs, we marginalize over the unobserved variables whereas in DBNs we do not enter evidence in the unobserved nodes. Rather previous states and observations in the last time slice influence inference about the state of the unobserved variables. We will return to this below.

[5] Note that the signal processing module and the first level of HMMs is identical in both cases. We are comparing HMMs with DBNs at the highest level of inference in S-SEER.

Table 1. (a) Average accuracies for S-SEER with HMMs and DBNs, with and without selective perception; (b) Percentage of time that each sensor was in use with HMMs and DBNs.

Recognition Accuracy without/with Selective Perception (%)		
	HMMs	DBNs
PC	97/98	95/90
FFC	99/97	97/97
P	86/88	99/99
O	93/100	99/99
NP	100/100	100/99
DC	91/70	96/96
Average Accuracy	94.3/92.2	97.7/96.7

(a)

	Percentage of use of each sensor in HMMs/DBN			
	Video	Audio	Sound Loc.	Keyboard/ Mouse
PC	22.7/99.8	22.7/2.3	0.0/0.0	100.0/100.0
FFC	27.3/100.0	27.3/0.0	0.0/0.0	100.0/100.0
P	0.3/5.6	0.3/0.0	0.0/0.0	100.0/100.0
O	0.0/2.6	0.0/4.7	0.0/0.0	100.0/100.0
NP	24.1/3.4	24.1/98.1	0.0/0.0	100.0/100.0
DC	23.6/97.4	23.6/98.1	0.0/0.0	100.0/100.0

(b)

We also observed that in most of our experiments, S-SEER –both with HMMs and DBNs, never turned the *sound localization* feature on, due to its high computational cost versus the relatively low informational value this acoustical feature provides. On the other hand, the keyboard and mouse sensors were at use all the time. Thus, we have learned information that is valuable in learning designs for an activity sensing system in this domain.

To better understand the behavior of the EVI-based selective perception policy in HMMs and DBNs we tracked the percentage of time that each sensor was used in our experiments. Table 1 (b) reflects the results. Note how HMMs tend to use the video and audio sensors quite in synchrony, whereas the DBN exhibits are more asymmetric behavior. On top of the keyboard and mouse – that are constantly used, there are activities where the DBN heavily relies on one other sensor, such as the video sensor during a *Phone Conversation* (99.8% use) or the audio sensor when there is *Nobody Present* (98.1% use).

We note that S-SEER's high accuracy without selective perception, may indicate that the task is too easy for the model and that is the reason why the selective perception policies have reasonable accuracies as well. We emphasize that the results reflected on the table correspond to a particular test set. We are also exploring more challenging scenarios for S-SEER, both in terms of the number of activities to classify from and their complexity.

Persistence of the Observed Data

As mentioned above, we use HMMs in a discriminatory fashion, which implies learning one HMM per class, running all HMMs in parallel and choosing the HMM with the highest likelihood as the most likely model. On the other hand, we learn a single DBN that has a hidden "Activity" node that provides us with the likelihood of each office activity at each time slice[6].

We are interested in understanding the persistence versus volatility of observational states in the world. Rather than consider findings unobserved at a particular time slice if the corresponding sensory analyses have not been im-

[6] The duration of a time slice depends on the level of inference: typical durations for the time slices at the lowest level are of 50ms, and of .5s at the highest level.

mediately performed, we would like to smooth out the value of the unobserved variables over time. DBNs allow for such a consideration because we have a single model for all activities, they encode a probability distribution for each variable and inference is performed with the network moving forward in time for any number of time slices with or without entering new evidence.

In a second set of experiments, we tracked the evolution of the probability distribution over all possible values of a particular node when using selective perception. Our goal was to see how the values of such variables change over time when a particular sensor is not used. Figure 2 (b) illustrates a typical behavior of S-SEER with a DBN and selective perception. The figure shows the probabilities over 25 consecutive time slices of a "Video_T" node during a *Presentation*. At time 1 the video sensor was used and therefore the probability of *One Active Person Present* was clamped to 1.0. From time slice 2 until time 14 the video sensor was not in use. The probability of *One Active Person Present* smoothly declines over time while the probability of *One Person Present* increases over time. Then, at time 15, the system decides to use the video sensor again until time 22 when it turns off the video sensor. We believe that this probabilistic smoothing over time in the presence of missing data is a valuable property of DBNs.

7 Summary

We have explored and compared the use of HMMs and DBNs for recognizing office activities with and without selective perception. Our testbed is a multimodal, multi-layer, real-time office activity recognition system named S-SEER.

HMMs have been used successfully in the area of human behavior modeling and this representation formed the core of the early work in S-SEER. Motivated by the case of missing observations associated with the use of a selective perception policy, we pursued a comparative analysis of the use of dynamic Bayesian network models in a component of S-SEER. In experiments, we have identified some differences and tradeoffs in the use of DBNs when compared to HMMs. We found that (1) DBNs can learn dependencies between variables that were assumed independent in HMMs; (2) DBNs provide a unified probability model as opposed to having one model per activity as in discriminative HMMs; and (3) the accuracy of inference by DBNs seems to be less sensitive than HMMs to the loss of access to sets of observations, per a specific selective perception algorithm that we have implemented. We believe that one reason for their lower degradation of the performance is the fact that unobserved variables in DBNs change smoothly over time, whereas HMMs marginalize over the unobserved variables. On the other hand, HMMs are simpler to train and to do inference with, they can handle continous data, and they impose less computational burden than arbitrary DBNs.

Thus, the best representation depends on several factors, including the resources available for training and testing, the likelihood that variables will not be observed, the nature of the data and the complexity of the domain. We advocate considering the merits of each approach in building human activity recognition systems.

References

1. M. Brand and V. Kettnaker. Discovery and segmentation of activities in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), 2000.
2. M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In *Proc. of CVPR97*, 994–999, 1996.
3. Hilary Buxton and Shaogang Gong. Advanced Visual Surveillance using Bayesian Networks. In *International Conference on Computer Vision*, 111–123, Cambridge, Massachusetts, June 1995.
4. D.M. Chickering. The winmine toolkit. Technical Report MSR-TR-2002-103, Microsoft, Redmond, WA, 2002.
5. J. Forbes, T. Huang, K. Kanazawa, and S. Russell. The batmobile: Towards a bayesian automated taxi. In *Proc. Fourteenth International Joint Conference on Artificial Intelligence, IJCAI'95*, 1995.
6. N. Friedman, K. Murphy, and S. Russell. Learning the structure of dynamic probabilistic networks. In *Proceedings of the 1st Annual Conference on Uncertainty in Artificial Intelligence (UAI-98)*, 139–147, New York, NY, 1998. Elsevier Science Publishing Comapny, Inc.
7. A. Galata, N. Johnson, and D. Hogg. Learning variable length markov models of behaviour. *International Journal on Computer Vision, IJCV*, 398–413, 2001.
8. Y. Ivanov and A. Bobick. Recognition of visual activities and interactions by stochastic parsing. *IEEE Trans. on Pattern Analysis and Machine Intelligence, TPAMI*, 22(8):852–872, 2000.
9. S.Z. Li, X.L. Zou, Y.X. Hu, Z.Q. Zhang, S.C. Yan, X.H. Peng, L. Huang, and H.J. Zhang. Real-time multi-view face detection, tracking, pose estimation, alignment, and recognition, 2001.
10. L. Liao, D. Fox, and H. Kautz. Learning and inferring transportation routines. In *Proceedings of AAAI'04*, 348–353, 2004.
11. K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning.* PhD thesis, U.C. Berkeley, 2002.
12. N. Oliver, F. Berard, and A. Pentland. Lafter: Lips and face tracking. In *Proceed. of IEEE International Conference on Computer Vision and Pattern Recognition, CVPR'97*, S.Juan, Puerto Rico, June 1997.
13. N. Oliver and E. Horvitz. Selective perception policies for guiding sensing and computation in multimodal systems: a comparative analysis. In *Proc. of Int. Conf. on Multimodal Interfaces*, 36–43, 2003.
14. N. Oliver, E. Horvitz, and A. Garg. Layered representations for human activity recognition. In *Proc. of Int. Conf. on Multimodal Interfaces*, 3–8, 2002.
15. N. Oliver, E. Horvitz, and A. Garg. Layered representations for human activity recognition. *Computer Vision and Image Understanding Journal*, 96:2:163–180, 2004.
16. L. Rabiner and B.H. Huang. *Fundamentals of Speech Recognition.* 1993.
17. F. Bremond S. Hongeng and R. Nevatia. Representation and optimal recognition of human activities. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'00*, 2000.
18. T. Starner and A. Pentland. Real-time american sign language recognition from video using hidden markov models. In *Proceed. of SCV'95*, 265–270, 1995.
19. A. Wilson and A. Bobick. Recognition and interpretation of parametric gesture. In *Proc. of International Conference on Computer Vision, ICCV'98*, 329–336, 1998.