Hierarchical Representations for Learning and Inferring Office Activity from Multiple Sensory Channels

Nuria Oliver, Eric Horvitz {NURIA,HORVITZ}@MICROSOFT.COM
Adaptive Systems & Interaction, Microsoft Research, One Microsoft Way, Redmond, WA 99052 USA

Ashutosh Garg ASHUTOSH@IFP.UIUC.EDU
Beckman Institute, University of Illinois, Urbana-Champaign, 405 N. Mathews, Urbana, IL 61801,USA

NOTE: A version of this paper has been submitted to the AAAI 2002 conference

Abstract

We present the use of hierarchical probabilistic representations for sensing, learning, and doing inference from multiple sensory streams at multiple levels of temporal granularity and abstraction. The methods show robustness to subtle changes in the environment and enable adaptation with minimal retraining. The approach centers on the use of a Hierarchy of Hidden Markov Models (HHMMs), whose parameters are learned from data. We have found that HHMMs are a promising means for making inferences about context and activity from perceptual signals. After presenting key properties of the representation, we review experiments with HHMMs applied in an office-awareness real-time multimodal prototype.

1 Introduction

Researchers and application developers have long been interested in the promise of performing automatic and semi-automatic recognition of activity and context from multiple perceptual cues. Developing methods for learning and reasoning about streams of perceptual evidence from visual, acoustic, and kinesthetic channels could accelerate the development of a variety of compelling applications and services that hinge on the identification of rich, human-centric notions of context. We believe that creating more robust learning and modeling methods for reasoning over multiple channels will unleash a great deal of creativity in such realms as multimodal human–computer interaction (HCI), intelligent environments, and visual surveillance.

We address in this paper the challenge of performing inferences that take as inputs raw signals coming from multiple sensors and that yield high-level abstract descriptions of the human activities. The task of moving from low-level signals to more abstract hypotheses about activity brings into focus a consideration of a spectrum of approaches. Potentially valuable methods include template matching, context-free grammars, and various statistical methods.

In this paper, we introduce a hierarchical statistical technique for detecting and recognizing human activities based on multiple streams of sensory information. The method is based on a formulation of dynamic graphical models which we refer to as a Hierarchy of Hidden Markov Models (HHMMs). Our research on HHMMs is motivated by the challenge of building *robust* context-recognition systems that can handle multiple levels of time granularity. Straightforward modeling techniques can be very sensitive to subtle variations in a target environment and can require extensive retraining when moved to another similar environment. As we shall describe below, our hierarchical statistical models allow a factoring of the learning and inference problems into multiple sub-problems, that produce a decomposition along the lines of variation and stability.

The paper is organized as follows: We shall review relevant prior work in the Section 2. In Section 3 we describe the learning and inference algorithms for our Hierarchy of Hidden Markov Models. Section 4 reviews the application of our hierarchical representation in the context of Seer, a real-time multimodal activity-recognition prototype. Experimental results are presented in Section 5, and finally Section 6 summarizes our work and highlights several

conclusions and future directions of research.

2 Previous Work

Multiple research teams have explored the fusion of multiple sources of information to reason about higher-level abstractions of context. Recent work on probabilistic models for reasoning about a user's location, intentions, and focus of attention have highlighted opportunities for building new kinds of applications and services (Horvitz, 1999; Horvitz et al., 1999). A portion of the work on leveraging perceptual information to recognize human activities has centered on the identification of a specific type of activity in a particular scenario. Many of these techniques are targeted at recognizing single, simple events, *e.g.,* 'waving the hand' or 'sitting on a chair'. However, less effort has been applied to research on methods for identifying more complex patterns of human behavior, extending over longer periods of time.

Dynamic models of periodic patterns of people's movements are used by Davis et al. (Davis & Bobick, 1997) to capture the periodicity of activities such as walking. Other approaches to the recognition of human activity employ graphical models. A significant portion of work in this arena has made use of Hidden Markov Models (HMMs) (Rabiner, 1989). Starner and Pentland in (Starner & Pentland, 1995) use an HMM for recognizing hand movements used to relay symbols in American Sign Language. The different signs are recognized by computing the probabilities that models for different symbols would have produced the observed visual sequence. More complex models, such as Parameterized-HMM (PHMM) (Wilson & Bobick, 1998), Entropic-HMM (Brand & Kettnaker, 2000), Variable-length HMM (VHMM) (Galata et al., 2001) and Coupled-HMM (CHMM) (Brand et al., 1996; Oliver, 2000), have been used to recognize more complex activities such as the interaction between two people. Bobick and Ivanov (Ivanov & Bobick, 2000), propose the use of a stochastic context-free grammar to compute the probability of a temporally consistent sequence of primitive actions recognized by HMMs. Clarkson and Pentland model events and scenes from audio-visual information in (Clarkson & Pentland, 1999). They have developed a wearable computer system that uses a hierarchy of HMMs for recognizing the user's location, *e.g.*, in the office, at the bank, etc. Brand and Kettnaker in (Brand & Kettnaker, 2000) propose an entropic-HMM approach to organize the observed video activities (office activity and outdoor traffic) into meaningful states. They illustrate their models in video monitoring of office activity and outdoor traffic. In (S. Hongeng & Nevatia, 2000), a probabilistic finite-state automaton (a variation of structured HMMs) is used for recognizing different scenarios, such as monitoring pedestrians or cars on a freeway. Although HMMs appear to be robust to changes in the temporal segmentation of observations, they suffer from a lack of structure, an excess of parameters, and an associated over-fitting problem when applied to reason about long and complex temporal sequences with insufficient training data. Finally, in recent years, more complex Bayesian networks have also been adopted for the modeling and recognition of human activities (Binder et al., 1997; Buxton & Gong, 1995; Horvitz et al., 1999; Intille & Bobick, 1999; Madabhushi & Aggarwal, 1999; Huang et al., 1994; Fernyhough et al., 1998).

To date, however, there has been little research on methods for exploiting statistical methods to fuse multiple sensory streams that address problems with robustness and training effort. We deal with these issues by means of a hierarchical probabilistic representation that allows us to make critical decompositions of the model and associated learning parameter space.

3 Hierarchical Hidden Markov Models (HHMMs)

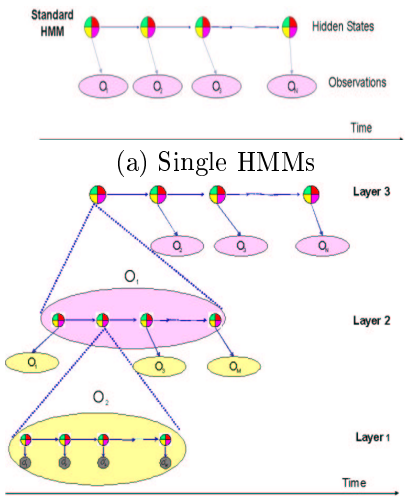
Our hierarchical approach to learning human activities from data employs directed acyclic graphs (DAGs), also referred to as Dynamic Bayesian Networks or Dynamic Graphical Models. Statistical DAGs (Dean & Kanazawa, 1988) can provide a computationally efficient and sufficiently expressive solution to the problem of human activity modeling and recognition. HMMs and their extensions, e.g. CHMMs, PHMMs, VHMMs, including the architecture proposed in this paper, HHMMs, can be viewed as particular cases of temporal DAGs.

DAGs consist of a set of random variables represented as nodes as well as directed edges or links between them. They define a mathematical form of the joint or conditional Probability Distribution Function (PDF) between the random variables. More importantly, from a human behavior perspective, DAGs are important because they constitute a graphical representation of causal dependencies among variables. The absence of directed links between nodes implies a conditional independence. Moreover a family of transformations can be performed on the graphical structure that has a direct translation in terms of mathematical operations applied to the underlying PDF. Finally they are modular, *i.e.,* one can express the joint global PDF as

the product of local conditional PDFs.

DAGs present several important advantages that are relevant to the problem of human behavior modeling from multiple sensors: 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 doing predictive inference; they offer a framework for combining prior knowledge and data; and finally they are modular and parallelizable.

Hidden Markov Models are one of the most popular examples of DAGs for modeling processes that have structure in time. They have a clear Bayesian semantics, efficient algorithms for state and parameter estimation, and they automatically perform dynamic time warping. An HMM is essentially a quantization of a system's configuration space into a small number of discrete states, together with probabilities for transitions between states. A single finite discrete variable indexes the current state of the system. Any information about the history of the process needed for future inferences must be reflected in the current value of this state variable. Graphically HMMs are often depicted 'rolled-out in time' as DAGs, such as in Figure 1.



(a) Single HMMs

(b) HHMMs

Figure 1. Graphical Representation of HMMs rolled-out in time (top) and Graphical Representation of Hierarchical HMMs. Note how each level handles a different temporal granularity (bottom)

We have explored a hierarchical representation for reasons of robustness, training efficacy, and naturalness. In developing systems that can make inferences about context from multiple streams of perceptual information, we faced the challenge of carrying out learning and inference that are robust to typical variations within office environments (e.g., lighting and acoustics differences). Beyond robustness in a single office, we desired a representation that would allow the models to perform well when transferred to new office spaces with minimal tuning through retraining.

We converged on the use of a multilevel representation of observations that allows for explanations at different granularities of time, by capturing different levels of temporal detail. For example, in the domain of office awareness, one level of description is the analysis and classification of the raw sensor signals. This level corresponds to a fine time granularity on the order of milliseconds. Another level of description is the detection of the user's presence, by use of audio, keyboard, mouse, and video. In this case, the time granularity is of several seconds. At another level, one could describe what the user has done in the last N minutes. Finally, one could provide an explanation of the user's activities during the day. The graphical structure of our HHMMs architecture is displayed in Figure 1.

In addition, employing a hierarchical structure provides several valuable properties. A hierarchical formulation makes it feasible to decouple different levels of analysis for training and inference. As it is further explained below, each level of our hierarchy is trained independently, with different feature vectors and time granularities. Once the system has been trained, inference can be carried out at any level of the hierarchy. One could retrain the lowest (most sensitive to variations in the environment) level, for example, without having to retrain any other level in the hierarchy.

3.1 Learning

For an HMM, the problem of learning the model parameters is solved by the forward-backward or Baum-Welch algorithm. This algorithm provides expressions for the α and β variables, whose product leads to the *likelihood* of a sequence at each instant of time. In particular, the α variable is given by $\alpha_{j,t+1} = [\sum_{i=1}^{N} \alpha_{i,t} P_{j|i}] p_j(o_t)$, and the β variable by $\beta_{i,t} = [\sum_{j=1}^{N} \beta_{j,t+1} P_{j|i} p_j(o_{t+1})]$, where N is the number of hidden states, $P_{i|j}$ is the probability of state i given state j and $p_i(o_t)$ is the probability for state i of the observation at time t. From the α and β variables one can obtain the model parameters, i.e. the observation and transition probabilities.

A formulation for hierarchical HMMs is first proposed in (Fine et al., 1998) in work that extends the

standard Baum-Welch algorithm and presents an efficient estimation procedure of the model parameters from unlabeled data. A trained model is applied to an automatic hierarchical parser of an observation sequence as a dendrogram. Because of the computational complexity of the original algorithm, the authors suggest an efficient approximation to the full estimation scheme. The approximation could further be used to construct models that adapt both their topology and parameters. The authors briefly illustrate the performance of their models on natural written English text interpretation and in English handwriting recognition.

Our HHMMs algorithm differs in fundamental ways from the hierarchical approach explored in (Fine et al., 1998). For HHMMs, each layer of the architecture is connected to the next layer via its inferential results (log likelihoods). In contrast, in (Fine et al., 1998), each state of the architecture is another HMM, and therefore represents a time sequence of the raw signals. Moreover, rather than training all the levels at the same time, the parameters for each level of our HHMMs can be trained independently by means of the Baum-Welch algorithm. The inputs (observations) of each level are the outputs of the previous level. At the lowest level, the observations (the leaves of the tree) are the feature vectors extracted directly from sensor signals.

3.2 Inference

The Viterbi algorithm (Rabiner, 1989) yields the most likely sequence of states $\hat{S}$ within a model given the observation sequence $O = \{o_1, \ldots, o_n\}$. This most likely sequence is obtained by $\hat{S} = argmax_S P(S|O)$.

In the case of HMMs the posterior state sequence probability $P(S|O)$ is given by

$$P(S|O) = \frac{P_{s_1} p_{s_1}(o_1) \prod_{t=2}^{T} p_{s_t}(o_t) P_{s_t|s_{t-1}}}{P(O)} \quad (1)$$

where $S = \{a_1, \ldots, a_N\}$ is the set of discrete states, $s_t \in S$ corresponds to the state at time t. $P_{i|j} \doteq P_{s_t=a_i|s_{t-1}=a_j}$ is the state-to-state transition probability (i.e. probability of being in state a_i at time t given that the system was in state a_j at time $t-1$). The prior probabilities for the initial state are $P_i \doteq P_{s_1=a_i} = P_{s_1}$. And finally $p_i(o_t) \doteq p_{s_t=a_i}(o_t) = p_{s_t}(o_t)$ are the output probabilities for each state, (i.e. the probability of observing o_t given state a_i at time t).

For HHMMs, inference can be performed at each level of the hierarchy by means of the same Viterbi algorithm as the one used for HMMs. Each level encodes a different temporal abstraction, going from the finest time granularity, at the leaves, to the lowest time granularity (highest level of temporal abstraction), at the root of the tree.

4 Sample Application: HHMMs and Office Awareness

We tested HHMMs within an office-awareness application, and built a system named Seer, that exploits the HHMM representation to learn about and detect activities in an office. Seer is composed of a three-layer HHMM, as it is displayed in Figure 2:

(1) At the lowest level, the sensor signals (described in section 4.1), *i.e.*, images, audio and keyboard and mouse activity, are captured and processed, resulting in a feature vector for each modality. The time granularity at this level is of windows of duration less than 100 milliseconds.

(2) In the middle layer, the sound is classified using discriminative HMMs [1]. Typical office sounds, such as human speech, music, silence, ambient noise, phone ringing and keyboard typing, are trained and learned in real time, with one HMM per type of sound. The source of the sound is also localized using a technique based on the Time Delay of Arrival (TDOA) (Brandstein & Silverman, 1997). The video signals are classified using discriminative HMMs to implement a person detector. At this level, the system detects whether one person is present in the room (semi-static), one active person is present, multiple people are present or there is nobody in the office. The inferential results from this layer (audio and video classifiers), the derivative of the sound localization component, and the history of keyboard and mouse activities constitute a feature vector that is passed to the next higher (third) layer in the hierarchy. The time granularity at this level is of less than 3 seconds.

(3) The third layer handles concepts that have longer temporal extent. Such concepts include the user's high-level activities in or near an office, corresponding to a time granularity of about $10 - 15$ seconds. The behavior models are HMMs whose observations are the inferential outputs of the previous level. Office activities recognized by Seer include PHONE CONVERSATION, FACE TO FACE CONVERSATION, PRESENTATION, DISTANT CONVERSATION, NOBODY IN THE OFFICE and USER PRESENT, ENGAGED IN SOME OTHER ACTIVITY. Some of these activities have been

[1] By discriminative HMMs we denote HMMs that have been trained to recognize a particular sound in this case. When classifying the sounds, inference is performed in all the models in parallel. At each instant, the model with the highest likelihood is selected.

proposed in the past as indicators of a person's availability (Johnson & Greenberg, 1999).

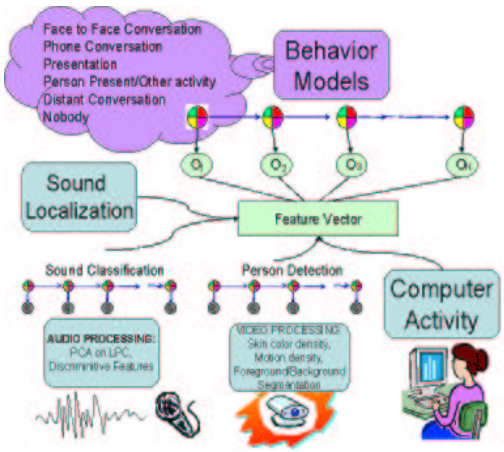


Figure 2. Architecture of the multimodal Seer system.

4.1 Perceptual Input

Seer is multimodal, accessing perceptual information from the following sources: (1) **Binaural microphones:** Two mini-microphones ($20-16000$ Hz, SNR 58 dB) are being used. The audio signal is sampled at 44100 KHz. The microphones are used for sound classification and localization; (2) **USB camera:** The video signal is obtained via a standard USB camera (Intel), sampled at 30 f.p.s. The video input is used to determine the number of persons present in the scene; (3) **Keyboard and mouse:** The system keeps a history of keyboard and mouse activities during the past 5, 60 and 600 seconds.

4.2 Feature Extraction and Selection

The raw sensor signals are preprocessed to obtain feature vectors (*i.e.* observations) for the lowest-level HMMs. On the audio side, Linear Predictive Coding (LPC) coefficients (Deller et al., 1993) are computed. Feature selection is applied on these coefficients by means of principal component analysis (PCA). The number of features is selected such that more than 95% of the variability in the data is maintained, which is typically achieved with no more than 7 features. Other higher-level features are also extracted from the audio signal, such as the energy, the mean and variance of the fundamental frequency over a time window and the zero crossing rate (ZCR) (Rabiner & Huang, 1993), given by $Zero(f) = \frac{1}{L}\sum_{i=(f-1)\times L+1}^{f\times L-1} \frac{|sign(s(i+1))-sign(s(i))|}{2}\cdot w(n-i)$, where f is the frame number, L is the frame length, w is a window function and $s(i)$ is the digitized speech signal at an index indicator i.

On the video side, three features are computed: the density of skin color in the image (obtained via a discriminative histogram between skin and non-skin in HSV space), the density of motion in the image (obtained by image differences), and the density of foreground pixels in the image (obtained by background subtraction, after having learned the background). Finally, a history of the last 5, 60 and 600 seconds of mouse and keyboard activities is logged.

The source of the sound is localized using the Time Delay of Arrival (TDOA) (Brandstein & Silverman, 1997) method. In TDOA, the measure in question is not the acoustic data received by the sensors, but rather the time delays between the signals coming from each sensor. Typically, TDOA-based approaches have two steps: the time delay estimation (TDE) and the sound source localization (SSL). Let $s(n)$ be the source signal and be $x_i(n)$ the i-th sensor received signal. If we assume no reverberation, we have $x_i(n) = a_i s(n - t_i) + b_i(n)$. To the model reverberation, we add the non-linear reverberation function: $x_i(n) = g_i * s(n-t_i)+b_i(n)$, where a_i is the attenuation factor, b_i is additive noise and g_i is the response between the source and the sensor. Seer includes multiple approaches for estimating the time delay of arrival between the left and right audio signals. We have obtained the best performance by estimating the peak of the time cross-correlation function between the left and right audio signals over a finite time window $[N_1, N_2]$, i.e.: $r_{lr}(d) = \sum_{n=N_1}^{n=N_2} l(n)r(n-d)$.

5 Experiments with Seer

We have been running Seer in multiple offices, with different users and respective environments for several weeks. In our preliminary tests, we have found that the high-level layers of Seer are quite robust to changes in the environment. In all the cases, when we moved Seer from one office to another, we obtained nearly perfect performance *without* the need for retraining the higher levels of the hierarchy. Only some of the lowest-level features (such as the ambient noise, the background image, and the skin color threshold) required re-training to tune the lowest level to the new conditions. In summary, the hierarchical structure greatly contributes to the overall robustness of the system given changes in the environment.

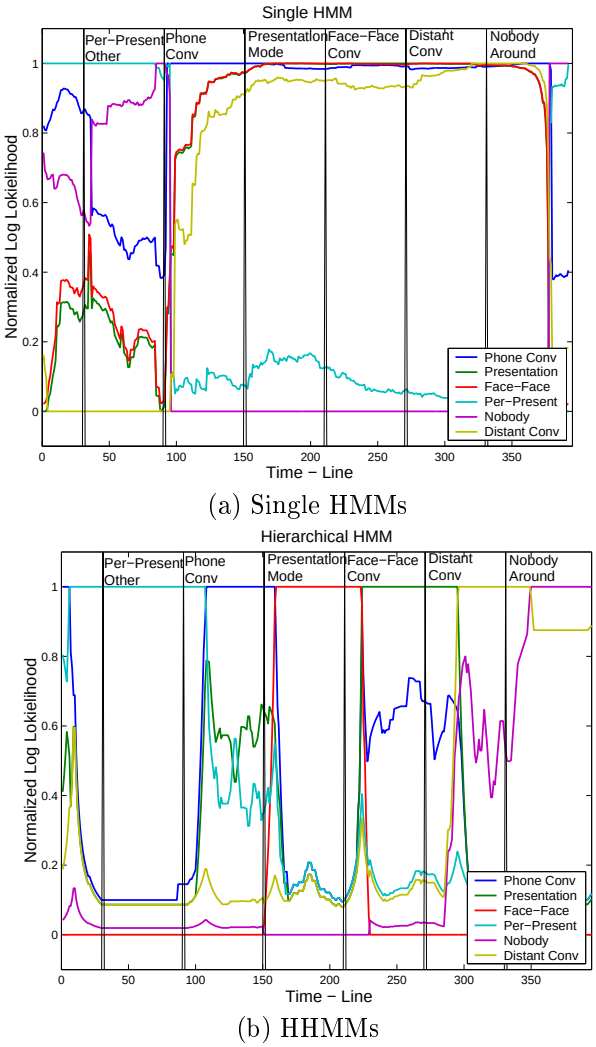
In a more quantitative study, we compared the performance of our model with that of single, standard HMMs. The feature vector in the latter case results from the concatenation of the audio, video and keyboard/mouse activities features in one long feature vector. We refer to these HMMs as the Carte-

sian Product HMMs. Note that a five-state HMM with single Gaussian observations of dimensionality 16 would have $5 * 16 * (16 + 1) = 1360$ parameters to estimate. An equivalent HHMM with 2 levels, two five-state HMMs at the lowest level (audio and video, with dimensionalities 10 and 3 respectively) and one five-state HMM at the highest level (of dimensionality 3), would have $5 * 10 * (10 + 1) + 5 * 3 * (3 + 1) + 5 * 3 * (3 + 1) = 670$ parameters. Note how encoding prior knowledge about the problem in the structure of the models significantly reduces the dimensionality of the problem. Therefore, for the same amount of training data, it is expected for HHMMs to have superior performance than HMMs. Our experimental results confirm such expectation.

Figure 3 illustrates the per-frame normalized likelihoods on testing in *real-time* both HMMs and HHMMs with the different office activities. By 'normalized' likelihoods, we denote the likelihoods whose values have been bounded between 0 and 1. They are given by: $NormLike_i = \frac{Like_i - min_j(Like_j)}{max_j(Like_j) - min_j(Like_j)}$, for $i = 1, ..., N$, $j = 1, ...N$, and N models. We only plot the likelihoods for the last half of the testing data to avoid instabilities in the transitions.

The accuracies of both HMMs and HHMMs when tested on 8 real-time sequences of each class were of 72.68% (STD 8.15) and 99.5% (STD 0.95) respectively.

Finally, we compared the performance on 30 minutes of office activity data (5 minutes per activity and 6 activities) of HHMMs and HMMs. The results are summarized in table 1. The HMMs were specifically tuned to the particular testing data. Their performance was otherwise so poor that we could not make any meaningful comparison with the equivalent HHMMs. On the other hand, the HHMMs had been trained many days before, under different office conditions than that of testing.



(a) Single HMMs

(b) HHMMs

Figure 3. Log Likelihoods for each of the activity models over time when tested in real time

5.1 Discussion

From our experiments we would highlight:

(1) *For the same amount of data, the accuracy of HHMMs is significantly higher than that of HMMs.* There are several reasons for the better performance of HHMMs when compared to HMMs: (a) The number of parameters of HMMs is about double that of HHMMs for the office activities being modeled in our experiments. As a consequence, for the same amount of training data, HMMs have many more parameters to estimate than HHMMs. Therefore HMMs are more prone to overfitting and worse generalization than HHMMs, especially when trained with limited amounts of data. (b) HMMs carry out high-level inferences about the user's activity, directly from the raw sensor signals. HHMMs, on the other hand, isolate the sensor signals in different sub-HMMs for each input modality. The inferential results of these models feed the HMMs in the next layer, that characterizes the office activities. Due to its hierarchical structure, HHMMs are more robust to noise in the sensor signals and have better generalization performance than HMMs.

(2) *HHMMs are more robust to changes in the environment than HMMs.* We could not obtain any reasonable performance on HMMs had they not been *highly tuned* to the particular testing environment and conditions. We had to retrain the HMMs every time we needed to test them on some particular data. On the contrary, HHMMs did *not* require

Confusion Matrix for highly-tuned HMMs						
	PC	FFC	P	O	NA	DC
PC	0.8145	0.0679	0.0676	0.0	0.0	0.05
FFC	0.0014	0.9986	0.0	0.0	0.0	0.0
P	0.0	0.0052	0.9948	0.0	0.0	0.0
O	0.0345	0.0041	0.003	0.9610	0.0	0.0
NA	0.0341	0.0038	0.0010	0.2524	0.7086	0.0
DC	0.0076	0.0059	0.0065	0.0	0.0	0.98
Confusion Matrix for generic HHMMs						
	PC	FFC	P	O	NA	DC
PC	1.0	0.0	0.0	0.0	0.0	0.0
FFC	0.0	1.0	0.0	0.0	0.0	0.0
P	0.0	0.0	1.0	0.0	0.0	0.0
O	0.0	0.0	0.0	1.0	0.0	0.0
NA	0.0	0.0	0.0	0.0	1.0	0.0
DC	0.0	0.0	0.0	0.0	0.0034	0.9966

Table 1. Confusion matrix for highly-tuned HMMs and generic HHMMs on 30 min of real data, where PC=Phone Conversation; FFC=Face to Face Conversation; P=Presentation; O=Other Activity; NA=Nobody Around; DC=Distant Conversation.

retraining, despite the changes in office conditions.

(3) *The discriminative power of HHMMs is notably higher than that of HMMs.* By *discriminative power*, we mean the distance between the log-likelihood of the two most likely models. The log likelihoods for the HMMs tend to be much closer to each other, making them prone to instability and errors in the classification. Note in Figure 3 how the normalized likelihoods between the two best models in HMMs are much closer than that in HHMMs. This phenomenon is particularly noticeable in the Presentation, Face to Face Conversation, Distant Conversation and Nobody Around activities.

6 Summary and Future Directions

In this paper we have introduced hierarchical probabilistic representations –in particular a Hierarchy of Hidden Markov Models (HHMMs)–, for sensing, learning, and inference. The representation was motivated by the challenge of building reasoning systems that infer context from multiple sensory streams. We believe that our representation provides a natural means for doing learning and inference at multiple levels of temporal granularity and abstraction. Moreover, we have sought a representation that maps naturally onto the problem space. Psychologists have found that human behaviors are hierarchically structured in many cases (Zacks & Tversky, 2001). We have pursued a representation that could capture such hierarchical properties in an elegant manner.

We have found that HHMMs provide promising means for making inferences about context and activity from perceptual signals. After presenting key properties of the representation, we have reviewed experiments with HHMMs applied in an office-awareness prototype. Some important characteristics of HHMMs when compared to HMMs are: (1) HHMMs encode the hierarchical temporal structure of the discrimination problem; thus, the dimensionality of the state space that needs to be learned from data is much smaller than that of their corresponding Cartesian Product HMMs; (2) HHMMs are easier to interpret, and, thus, easier to refine and improve, than the corresponding Cartesian Product HMMs; (3) HHMMs can encode different levels of abstraction and time granularities that can be linked to different levels of representation for human behaviors; (4) the modularity of HHMMs allows the selective retraining of the levels that are most sensitive to environmental or sensor variation, minimizing the burden of training during transfer among different environments.

We have demonstrated the performance of HHMMs in Seer, a real-time system for recognizing typical office activities. Seer can accurately recognize when a user is engaged in a phone conversation, giving a presentation, involved in a face-to-face conversation, or doing some other work in the office—or when a distant conversation is occurring in the corridor. We believe that HHMMs can be used to enhance a variety of applications that rely on the identification of contexts from perceptual cues.

We are currently exploring several theoretical and engineering challenges with the refinement of HHMMs, including efforts to understand the influence of the hierarchical decomposition on the size of the parameter space, and the resulting effects on learning requirements and accuracy of inference for different amounts of training. Alternate decompositions lead to layers of different configurations and structure; we are interested in understanding better how to optimize the decompositions. We are also exploring the use of unsupervised and semi-supervised methods for training one or more layers of the HHMMs without explicit training effort. Finally, we are exploring several applications of inference about context.

References

Binder, J., Koller, D., Russell, S. J., & Kanazawa, K. (1997). Adaptive probabilistic networks with hidden variables. *Machine Learning*, *29*, 213–244.

Brand, M., & Kettnaker, V. (2000). Discovery and

segmentation of activities in video. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8).*

Brand, M., Oliver, N., & Pentland, A. (1996). Coupled hidden markov models for complex action recognition. *Proc. of CVPR97* (pp. 994–999).

Brandstein, M., & Silverman, H. (1997). A practical methodology for speech source localization with microphone arrays. *11(2)*, 91–126.

Buxton, H., & Gong, S. (1995). Advanced visual surveillance using bayesian networks. *International Conference on Computer Vision.* Cambridge, Massachusetts.

Clarkson, B., & Pentland, A. (1999). Unsupervised clustering of ambulatory audio and video. *International Conference on Acoustics, Speech and Signal Processing, ICASSP'99* (pp. 3037–3040).

Davis, J., & Bobick, A. (1997). The representation and recognition of action using temporal templates. *Proc. of Computer Vision and Pattern Recognition (CVPR'97)* (pp. 928–934).

Dean, T., & Kanazawa, K. (1988). Probabilistic temporal reasoning. *Seventh National Conference on Artificial Intelligence (AAAI-88)* (pp. 524–529). St.Paul, Minnesota.

Deller, J., Proakis, J., & Hansen, J. (1993). *Discrete time processing of speech signals.* Macmillan Series for Prentice-Hall Publishers, New York.

Fernyhough, J., Cohn, A., & Hogg, D. (1998). Building qualitative event models automatically from visual input. *ICCV98* (pp. 350–355).

Fine, S., Singer, Y., & Tishby, N. (1998). The hierarchical hidden markov model: Analysis and applications. *Machine Learning, 32*, 41–62.

Galata, A., Johnson, N., & Hogg, D. (2001). Learning variable length markov models of behaviour. *International Journal on Computer Vision, IJCV-2001*, 398–413.

Horvitz, E. (1999). Principles of mixed-initiative user interfaces. *CHI* (pp. 159–166).

Horvitz, E., Jacobs, A., & Hovel, D. (1999). Attention-sensitive alerting. *Proc. of Conf. on Uncertainty in Artificial Intelligence* (pp. 305–313).

Huang, T., Koller, D., Malik, J., Ogasawara, G., Rao, B., Russel, S., & Weber, J. (1994). Automatic symbolic traffic scene analysis using belief networks (pp. 966–972.).

Intille, S. S., & Bobick, A. F. (1999). A framework for recognizing multi-agent action from visual evidence. *AAAI/IAAI* (pp. 518–525).

Ivanov, Y., & Bobick, A. (2000). Recognition of visual activities and interactions by stochastic parsing. *22(8)*, 852–872.

Johnson, B., & Greenberg, S. (1999). Judging people's availability for interaction from video snapshots. *Proc. of the IEEE Hawaii International Conference on System Sciences, HICS'99.*

Madabhushi, A., & Aggarwal, J. (1999). A bayesian approach to human activity recognition. *In Proc. of the 2nd International Workshop on Visual Surveillance* (pp. 25–30).

Oliver, N. (2000). *Towards perceptual intelligence: Statistical modeling of human individual and interactive behaviors.* Doctoral dissertation, Massachusetts Institute of Technology, MIT.

Rabiner, L., & Huang, B. (1993). *Fundamentals of speech recognition*, chapter 3.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *IEEE, 77*, 257–286.

S. Hongeng, F. B., & Nevatia, R. (2000). Representation and optimal recognition of human activities. *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition CVPR00.*

Starner, T., & Pentland, A. (1995). Real-time american sign language recognition from video using hidden markov models. *SCV95* (pp. 265–270).

Wilson, A., & Bobick, A. (1998). Recognition and interpretation of parametric gesture. *Proc. of International Conference on Computer Vision (ICCV'98)* (pp. 329–336).

Zacks, J., & Tversky, B. (2001). Event structure in perception and cognition. *Psychological Bulletin, 127(1)*, 3–21.