Shape-and-Behavior-Encoded Tracking of Bee Dances

Ashok Veeraraghavan, *Student Member*, *IEEE*, Rama Chellappa, *Fellow*, *IEEE*, and Mandyam Srinivasan, *Member*, *IEEE*

Abstract—Behavior analysis of social insects has garnered impetus in recent years and has led to some advances in fields like control systems and flight navigation. Manual labeling of insect motions required for analyzing the behaviors of insects requires significant investment of time and effort. In this paper, we propose certain general principles that help in simultaneous automatic tracking and behavior analysis, with applications in tracking bees and recognizing specific behaviors that they exhibit. The state space for tracking is defined using the position, orientation, and current behavior of the insect being tracked. The position and the orientation are parameterized using a shape model, whereas the behavior is explicitly modeled using a three-tier hierarchical motion model. The first tier (dynamics) models the local motions exhibited, and the models built in this tier act as a vocabulary for behavior modeling. The second tier is a Markov motion model built on top of the local motion vocabulary, which serves as the behavior model. The third tier of the hierarchy models the switching between behaviors, and this is also modeled as a Markov model. We address issues in learning the three-tier behavioral model, in discriminating between models, and in detecting and modeling abnormal behaviors. Another important aspect of this work is that it leads to joint tracking and behavior analysis instead of the traditional "track-and-then-recognize" approach. We apply these principles for tracking bees in a hive while they are executing the waggle dance and the round dance.

Index Terms—Tracking, behavior analysis, activity analysis, waggle dance, bee dance.

1 Introduction

Behavioral research in the study of the organizational structure and communication forms in social insects like the ants and bees has received much attention in recent years [1], [2]. Such a study has provided some practical models for tasks like work organization, reliable distributed communication, and navigation [3], [4]. Usually, when such an experiment to study these insects is set up, the insects in an observation hive are videotaped. The hours of video data are then manually studied and hand labeled. This task of manually labeling the video data takes up the bulk of the time and effort in such experiments. In this paper, we discuss general methodologies for automatic labeling of such videos and provide an example by following the approach for analyzing the movement of bees in a beehive. Contrary to traditional approaches that first track the objects in video and then recognize the behaviors by using the extracted trajectories, we propose to simultaneously track and recognize the behaviors. In such a joint approach, accurate modeling of behaviors act as priors for motion tracking and significantly enhances motion tracking, whereas accurate and reliable motion tracking enables behavior analysis and recognition.

We present a system that can be used to analyze the behavior of insects and, more broadly, provide a general framework for the representation and analysis of complex behaviors. Such an automated system significantly speeds up the analysis of video data obtained from experiments and also reduces manual errors in the labeling of data. Moreover, parameters like the orientation of various body parts of the insects (which are of great interest to behavioral researchers) can be automatically extracted using such a framework. The system requires the technical input of a behavioral researcher (who would be the user) regarding the type of behaviors that would be exhibited by the insect being studied.

The salient characteristics of this paper are the following:

- We suggest a joint tracking and behavior analysis instead of the traditional "track-and-then-recognize" approach for activity analysis. The principles for simultaneous tracking and behavior analysis presented in this paper should be applicable to a wide range of scenarios like analyzing sports videos, activity monitoring, and surveillance.
- We show how the method can be extended to tackle multiple behaviors by using hierarchical Markov models to model various behaviors. We define instantaneous low-level motion states like hover, turn, and waggle and model each of the dances as a Markov model over these low-level motion states. Switching between behaviors (dances) is modeled as another Markov model over the discrete labels corresponding to the various dances.
- We also present methods for detecting and characterizing abnormal behaviors.
- In particular, we study the simultaneous tracking and analysis of bee dances in their hive. This is an appropriate setting, in which we can study the

- *A. Veeraraghavan and R. Chellappa are with the Department of Electrical and Computer Engineering, Center for Automation Research, University of Maryland, 4421 A.V. Williams Building, College Park, MD 20742. E-mail: {vashok, rama}@umiacs.umd.edu.*
- *M. Srinivasan is with the Visual Neuroscience Department, Queensland Brain Institute, The University of Queensland, Brisbane QLD 4072 Australia. E-mail: m.srinivasan@uq.edu.au.*

Manuscript received 24 May 2006; revised 20 Dec. 2006; accepted 8 May 2007; published online 30 May 2007.
Recommended for acceptance by C. Kambhamettu.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0399-0506.
Digital Object Identifier no. 10.1109/TPAMI.2007.70707.

"track-and-recognize-simultaneously" approach suggested by this paper, since the extreme clutter and presence of several similar bees make traditional tracking in such videos extremely difficult. Consequently, most tracking algorithms suffer frequent missed tracks, and the rich variety of structured behaviors that the bees exhibit enables a rigorous test of behavior modeling. We have modeled a few of the dances of the foraging bees and estimated the parameters of the waggle dance.

1.1 Prior Work in Tracking

There has been significant work on tracking objects in video. Most tracking methodologies can be classified as either deterministic or stochastic. Deterministic approaches solve an optimization problem under a prescribed cost function [5], [6]. Stochastic approaches estimate posterior distribution of the position of the object in the current frame by using a Kalman filter or particle filters [7], [8], [9], [10], [11], [12]. Most of these do not directly adapt well to tracking insects because they exhibit very specific forms of motion (for example, bees can turn by a right angle within two or three frames). In order to extend such tracking methods, it is important to consider the anatomy (body parts) of these insects and incorporate both their structure and the nature of their motions in the tracking algorithm.

The use of prior shape and motion models to facilitate tracking has been recently explored in several works for the problem of human body tracking. The shape of the human body has been modeled as anything ranging from a simple stick figure model [13] to a complex superquadric model [14]. Several tracking algorithms use motion models (like constant velocity model and random walk model) for tracking [9], [12], [15], [11]. There have also been some recent attempts to model specific motion characteristics of the human body to aid as priors in tracking [16], [17], [18], [19], [20].

Previous work on tracking insects has concentrated on the speed and reliability of estimating just the position of the center of insects in videos [12], [21]. Inspired by the studies in human body tracking mentioned above, we explore the effectiveness of higher level shape and motion models for the problem of tracking insects in their hives. We believe that such methods lead to algorithms where tracking and behavior analysis can both be performed simultaneously; that is, whereas these motion priors aid reliable tracking, the parameters of the motion models also encode information about the nature of behavior being exhibited. We model the behaviors exhibited by the insect by using Markov motion models and use these models as priors in a tracking framework to reliably estimate the location and the orientation of the various body parts of the insect. We also show that it is possible to make inferences about the behavior of the insect by using the parameters estimated via the motion model.

1.2 Bee Dances as a Means of Communication

When a worker honeybee returns to her nest after a visit to a nourishing food source, she performs a so-called "dance" on the vertical face of the honeycomb to inform her nestmates about the location of the food source [1]. This behavior serves the recruitment additional workers to the location, thus enabling the colony to exploit the food source effectively. Bees perform essentially two types of dances in the context of

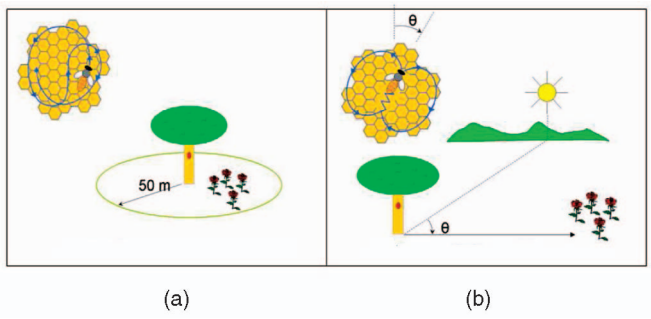


Fig. 1. Illustration of the (a) "round dance," (b) the "waggle dance," and their meaning.

communicating the location of food sites. When the site is very close to the nest (typically within a radius of 50 m), the bee performs a so-called "round dance." This dance consists of a series of alternating left-hand and right-hand loops, as shown in Fig. 1a. It informs the nestmates that there is an attractive source of food located within a radius of about 50 m from the nest. When the site is at a considerable distance away from the nest (typically farther than 100 m), the bee performs a different kind of dance, the so-called "waggle dance," as shown in Fig. 1b. In this dance, the transition between one loop and the next is punctuated by a "waggle phase," in which the bee waggles her abdomen from side to side while moving in a more or less straight line. Thus, the bee executes a left-hand loop, performs a waggle, executes a right-hand loop, performs a waggle, executes a left-hand loop, and so on. During the waggle phase, the abdomen is waved from side to side at an approximately constant frequency of about 12 Hz. The waggle phase contains valuable information about the location of the food source. The duration of the waggle phase (or equivalently, the number of waggles in the phase) is roughly proportional to the bee's perceived distance of the food source: the longer the duration, the greater the distance. The orientation of the waggle axis (the average direction of the bee's long axis during the waggle phase) with respect to the vertically upward direction conveys information about the direction of the food source. The angle between the waggle axis and the vertically upward direction is equal to the azimuthal angle between the sun and the direction of the food source. Thus, the waggle dance is used to convey the position of the food source in a polar coordinate system (in terms of distance and direction), with the nest being regarded as the origin and the sun being used as a directional compass [1]. The "attractiveness" of the food source is also conveyed in the waggle dance: the greater the attractiveness, the greater the number of loops that the bee performs in a given dance, and the shorter the duration of the return phase (the nonwaggle period) of each loop. The waggle frequency of 12 Hz is remarkably constant from bee to bee and from hive to hive [1]. The attractiveness of a food source, however, may depend upon the specific foraging circumstances such as the availability of other sources and their relative profitability, as well as an individual's knowledge and experience with the various sites. Thus, the number of dance loops and the duration of the return phase may vary from bee to bee and from one day to the next in a given bee [1].

There are additional dances that bees use to communicate other kinds of information [1]. For example, there is the so-called "jostling dance," where a returning bee runs rapidly through the nest, pushing nest mates aside,

apparently signaling that she has just discovered an excellent food source. The "tremble" dance [22], where a returning forager shakes her body from side to side and at the same time rotating her body axis by about 50 degrees every second or so, is used by a returning bee to inform her nestmates that there is too much nectar coming in, and she is consequently unable to unload her food to a food-storing bee [22]. There is also the "grooming dance," in which a standing bee raises her middle legs and shakes her body rapidly to and fro, beckoning other bees to assist her with her grooming activities. The "jerking dance," performed by a queen, consisting of up and down movements of the abdomen, usually precedes swarming or a nuptial flight. However, the pinnacle of communication in insects resides undoubtedly in the waggle dance. The surprisingly symbolic and abstract way in which this dance is used to convey information about the location of a food source has earned it the status of a "language" [1].

1.3 Prior Work in Analyzing Bee Dances

There is a great deal of interest and a significant need for developing automated methods for 1) detecting dancing bees in video sequences, 2) accurately tracking dance trajectories, and 3) extracting the dance parameters described above. However, in most of these cases, the experimenters manually study the videos of bee dances and annotate the various bee dances. This is usually time consuming, tiring, and error prone. Some recent efforts into automating such tasks have started emerging with the advances made in vision-based tracking systems. Feldman and Balch [23] suggest the use of Markov models for identifying certain segments of the dances, but this method relies on the availability of manually labeled data. Khan et al. [12] suggest the use of a Rao-Blackwellized particle filter to track the center of the bee during dances. The work does not address the issue of behavioral analysis once tracking is done. Moreover, some of the parameters of the dances that are essential for decoding the dance, like the orientation of the thorax during the waggle and so forth, are not estimated directly. Oh et al. [21] suggest the use of parametric switched linear dynamical system (p-SLDS) for learning motions that exhibit systematic temporal and spatial variations. They use the position tracking algorithm proposed in [12] and obtain trajectories of the bees in the videos. An Expectation-Maximization (EM)-based algorithm is used for learning the p-SLDS parameters from these trajectories. Much in the same spirit, we also model the various behaviors explicitly by using hierarchical Markov models (which can be viewed as an SLDS). Nevertheless, whereas position tracking and behavior interpretation are completely independent in their system, here, we close the loop between position tracking and behavior inference, thereby enabling a persistent and simultaneous tracking and behavior analysis. In such a "simultaneous tracking and behavioral analysis approach," the behavior modeling enhances tracking accuracy, whereas the tracking results enable accurate interpretation of behaviors.

1.4 Organization of the Paper

In Section 2, we discuss the shape model to track insects in videos and show how using the model helps in inferring parameters of interest about the motions exhibited by the insects. Section 3 discusses the issue of modeling behaviors, detecting, and characterizing abnormal behaviors. Section 4 discusses the tracking algorithm. Detailed experimental results for the problem of tracking and analyzing bee dances are provided in Section 5.

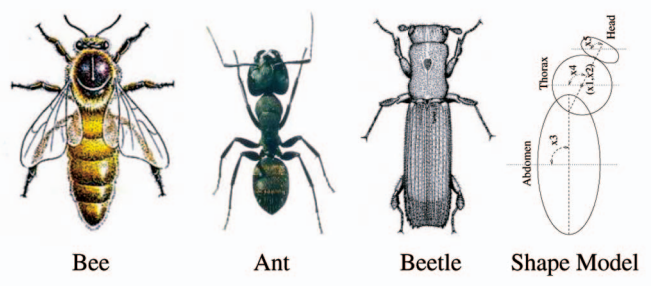


Fig. 2. A bee, an ant, a beetle, and a shape model.

2 Anatomical/Shape Model

Modeling the anatomy of insects is very important for reliable tracking because the structure of their body parts and their relative positions present some physical limits on their possible relative orientations. In spite of their great diversity, the anatomy of most insects is rather similar. All insects possess six legs. An insect body has a hard exoskeleton protecting a soft interior. The body is divided into three main parts: the head, the thorax, and the abdomen. The abdomen is divided into several smaller segments. Fig. 2 shows the image of a bee, an ant, and a beetle. Though there are individual differences in their body structure, the three main parts of the body are evidently visible. Each of these three parts can be regarded as rigid body parts for the purposes of video-based tracking. The interconnection between parts provide some physical limits for the relative movement of these parts. Most insects also move toward the direction of their head. Therefore, during specific movements such as turning, the orientation of the abdomen usually follows the orientation of the head and the thorax with some lag. Such interactions between body parts can be easily captured using a structural model for insects.

We model the bees with three ellipses, one for each body part. We neglect the effect of the wings and legs on the bees. Fig. 2 shows the shape model of a bee. Note that the same shape model can be used to adequately model most other insects also. The dimensions of the various ellipses are fixed during initialization. Currently, the initialization for the first frame is manual. It consists of clicking two points to indicate the enclosing rectangle for each ellipse. Automatic initialization is a challenging problem in itself and is outside the scope of our current work.

The location of the bee and its parts in any frame can be given by five parameters, namely, the location of the center of the thorax (two parameters), the orientation of the head, the orientation of the thorax, and the orientation of the abdomen (refer to Fig. 2). Tracking the bee over a video essentially amounts to estimating these five model parameters ($\mathbf{X} = [x_1\ x_2\ x_3\ x_4\ x_5]'$) for each frame. This five-parameter model has a direct physical significance in terms of defining the location and the orientation of the various body parts in each frame. These physical parameters are of importance to behavioral researchers.

2.1 Limitations of the Anatomical Model

We have assumed that the actual sizes of these ellipses do not change with time. This would, of course, be the case, as long as

the bee remains at the same distance from the camera. Since the behaviors that we study in our work (like the waggle dance) are performed on a vertical plane inside the beehive, and the optical axis of the video camera was perpendicular to this plane, the bees projected the same part sizes during the entire length of video captures. Nevertheless, it is very easy to incorporate the effect of distance from the camera in our shape model by introducing a scale factor as one more parameter in our state space. Moreover, the bees are quite small and were far enough from the camera that perspective effects could be ignored. The spatial resolution with which the bees appear in the video also limit the accuracy with which the physical model parameters can be recovered. For example, when the spatial resolution of the video is low, we may not be able to recover the orientation of the body parts individually.

3 Behavior Model

Insects, especially social insects like bees and ants, exhibit rich behaviors, as described in Section 1.2. Modeling such behaviors explicitly is helpful in accurate and robust tracking. Moreover, explicitly modeling such behaviors also leads to algorithms where position tracking and behavior analysis are tackled in a unified framework. Several algorithms use motion models (like the constant velocity model and random walk model) for tracking [9], [12], [15], [11]. We propose the use of behavioral models for the problem of tracking insects. Such behavioral models have been used for certain other specific applications like human locomotion [18], [19], [20]. The difference between motion models and behavioral models is the range of time scales at which modeling is done. Motion models typically model the probability distribution (pdf) of the position in the next frame as a function of the position in the current frame. Instead, behavioral models capture the pdf of position over time as a function of the behavior that the tracked object is exhibiting. We believe that the use of behavioral models presents a significant layer of abstraction that enhances the variety and complexity of the motions that can be tracked automatically.

3.1 Deliberation of the Behavior Model

The state space for tracking the position and body angles of the insect in each frame of the video sequence is determined by the choice of the shape model. In our specific case, this state space comprises of the x, y position of the center of the thorax and the orientation of the three body parts in each frame ($\mathbf{X} = [x_1\ x_2\ x_3\ x_4\ x_5]'$). A given behavior can be modeled as a dynamical model on this space. At one extreme, one can attempt to learn a dynamical model like an autoregressive model or an autoregressive and moving average (ARMA) model directly on this state space. A suitably and carefully selected model of this form might be able to capture large time scale interactions that are a characteristic of complex behaviors. However, these models constructed directly on the position state space suffer from two significant handicaps. First, to incorporate long-range interactions, these models would necessarily have a large number of parameters, and learning all these parameters from limited data would be brittle. It would be nice to somehow learn a compact set of parameters that can capture such large time range interactions. Second, these models are opaque to the behavioral researcher who is continuously interacting with the system during the learning phase. Since the system does not replace the behavioral researcher but rather assists him by tracking and analyzing behaviors of bees that the researcher selects, it is very important for the model to be easily amenable to the intended user of the system.

One can achieve both these objectives by abstracting out local motions like turning, hovering, and moving straight ahead and by modeling the behavior as a dynamical model on such local motions. Such a model would be simple and intuitive to the behavior researcher, and the number of parameters required to model behaviors would be dependent only on the number of local motions modeled. When the need to specify and learn new behaviors arises, the user would have to focus only on the dynamical model of the local motions, since the model for the local motions themselves would already be a part of system. In short, the local motions act as some sort of a vocabulary that enables the user to effectively interact with the system.

3.2 Choice of Markov Model

As described in the previous section, we first define pdf's for some basic motions such as moving straight ahead, turning, waggling, and hovering at the same location. Once these descriptions have been learned, we define each behavior by using an appropriate model on this space of possible local motions. Prior work [23] on analyzing the behaviors of bees has used Markov models to model the behaviors. That study reports promising results on recognizing behaviors using such Markov models. More recently, Oh et al. [21] used SLDS to model and analyze bee dances. They then noted that the models can be made more specific and accurate by incorporating a duration model within the framework of a linear dynamical system. They use this parameterized duration modeling with an SLDS and show improved performance [24]. We could, in principle, choose any of these models for analyzing bee dances. Note that the tracking algorithm would be identical, irrespective of the specific choice of model, since it is based on particle filtering and, therefore, just requires that we be able to efficiently sample from these motion models. The various dances that the bees perform are very structured behaviors and, consequently, we need these models to have enough expressive power to capture these structures. Nevertheless, we also note that at this stage, these models are acting as priors to the tracking algorithm, and therefore, if these models were very peaky/specific, then even a small change in the actual motion of the bees might cause a loss of track. Therefore, the model must also be fairly generic in the sense that it must be able to continue tracking, even if the insect deviates from the model. Taking these factors into account, we used Markov models very similar to those used in [23] to model bee behaviors. We noticed that even such a simple Markov model significantly aided tracking performance and enabled the tracker to continue maintaining the track in several scenarios where the traditional tracking algorithms failed (see Section 5.4). Another significant advantage of choosing a simple Markov model to act as behavior priors rather than more sophisticated and specific models is the fact that the very generality of the model makes the tracking algorithm fairly insensitive with respect to the initialization of the model parameters. In practice, we found that the tracking algorithm was fairly insensitive to the initialization of the model parameters and was quickly able to refine the corresponding model parameters within about 100-200 frames.

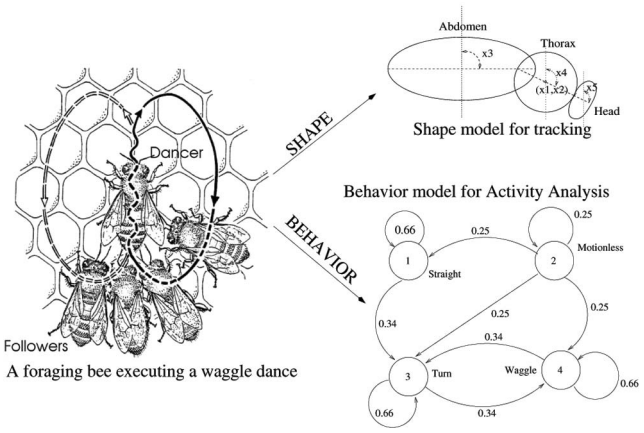


Fig. 3. A bee performing a waggle dance and the behavioral model for the waggle dance.

3.3 Mixture Markov Models for Behavior

Mixture models have been proposed and used successfully for tracking [25], [26]. Here, we advocate the use of Markovian mixture models in order to enable persistent tracking and behavior analysis. First, basic motions are modeled, creating a vocabulary of local motions. These basic motions are then regarded as states, and behaviors are modeled as being Markovian on this motion state space. Once each specific behavior has been modeled as a Markov process, our tracking system can simultaneously track the position and the behavior of insects in videos.

We model the pdf's of location parameters $\mathbf{X}$ for certain basic motions $(m_1{-}m_4)$. We model four different motions:

1. moving straight ahead,
2. turning,
3. waggling, and
4. motionless.

The basic motions moving straight, waggling, and motionless are modeled using Gaussian pdf's (p_{m1}, p_{m3}, p_{m4}) whereas a mixture of two Gaussians (p_{m2}) is used for modeling the turning motion (to accommodate the two possible turning directions):

$$p_{mi}(X_t/X_{t-1}) = N(X_{t-1} + \vec{\mu_{mi}}, \Sigma_{mi}), for\ i = 1, 3, 4, \quad (1)$$

$$\begin{aligned} p_{m2}(X_t/X_{t-1}) = {} & 0.5N(X_{t-1} + \vec{\mu}_{m2}, \Sigma_{m2}) \\ & + 0.5N(X_{t-1} - \vec{\mu}_{m2}, \Sigma_{m2}). \end{aligned} \quad (2)$$

Each behavior B_i is now modeled as a Markov process of order K_i on these motions, that is,

$$\mathbf{s}_t = \sum_{k=1}^{K_i} A_{B_i}^k \mathbf{s}_{t-k}, \quad (3)$$

where s_t is a vector whose jth element is $P(\text{motion state} = m_j)$, and K_i is the model order for the ith behavior B_i. The parameters of each behavior model are made of autoregressive parameters $A_{B_i}^k$ for $k = 1 \ldots K_i$. We discuss methods for learning the parameters of the behavior model later.

We have modeled three different behaviors: the waggle dance, the round dance, and a stationary bee using a first-order Markov model. For illustration, we discuss the manner in which the waggle dance is modeled. Fig. 3 shows the trajectory followed by a bee during a single run of the waggle dance. It also shows some followers who follow the dancer but do not waggle. A typical Markov model for the waggle dance is also shown in Fig. 3.

The trajectory of the bee can now be viewed as a realization from a random process following a mixture of behaviors. In addition, we assume that the behavior exhibited by the bee changes in a Markovian manner, that is,

$$B_t = T_B B_{t-1}, \quad (4)$$

where T_B is the transition probability matrix between behaviors. Note that T_B has a dominant diagonal. Estimating the trajectory and the specific behavior exhibited by the bee at any instant is then a state inference problem. This can be solved using one of the several techniques for estimating the state, given the observations.

Thus, the model consists of a three-tier hierarchy. At the first level, the dynamics of local motions are characterized. These act as a vocabulary enabling the behavior researcher to easily interact with the system in order to add new behaviors and analyze the output of the tracking algorithm without being bogged down by the particulars of the data capture. Behaviors that the bees exhibit are modeled as Markovian on the space of local motions forming the second tier of the hierarchy. Finally, switching between behaviors is modeled as a diagonal dominant Markov model, completing the model. The first two tiers of the hierarchy, dynamics and behavior, may be collapsed into a single tier. However, this would be disadvantageous, since it would 1) couple the specifics of data capture with the behavior models and 2) also make it significantly more difficult for the behavior researcher (user) to efficiently interact with the system.

3.4 Limitations and Implications of the Choice of Behavior Model

As described above, the choice of Markov model on a vocabulary of a set of low-level motions was motivated primarily from two design considerations: 1) ease of use for the user and 2) generality of the model, allowing the tracking algorithm to be robust to initialization parameters. However, this choice also leads to certain limitations. For one, it might indeed be possible to collapse the entire three-tier hierarchy of motion modeling into one large set of motion models, all at the dynamics stage. However, such a model would suffer from significant disadvantages, since the number of required parameters would significantly increase. Moreover, each new behavior must be modeled from scratch, whereas if we maintained the hierarchy, then the vocabulary of local motions learned at the lower tiers of the hierarchy can be used to simplify the learning problem for new behaviors. Fine et al. [27] provide a detailed characterization of the limitations and expressive power of such hierarchical Markov models, whereas Koutsoukos and Antsaklis [28] describe a methodology to analyze such linear hybrid dynamical systems. The hierarchical model also assumes that the various tiers of the hierarchy are semi-independent and that the particular current motion state does not have a direct influence on the behavior in subsequent frames. This would not necessarily be true, since particular behaviors might have specific end patterns of motion. In the future, we would like to study how one might introduce such state-based transition characteristics into the behavior model while retaining both the hierarchical nature of the model itself and keeping complexity of the model manageable.

3.5 Learning the Parameters of the Model

Learning the behavior model is now equivalent to learning the autoregressive parameters $A_{B_i}^k$ for $k = 1 \ldots K_i$ for each behavior B_i and also learning the transition probability matrix between behaviors given by T_B. This step can either be supervised or unsupervised.

3.5.1 Unsupervised Learning/Clustering

In an unsupervised learning, we are provided with only the sequence of motion states that are exhibited by the bee for frames 1 to N; that is, we are provided with a time series $s_1, s_2, s_3, \ldots, s_N$, where each s_i is one of the motion states $m_1 \ldots m_4$. We are not provided with any annotation of the behaviors exhibited by the bee; that is, we do not know the behavior exhibited by the bee in each of these frames. This is essentially a clustering problem. A maximum-likelihood (ML) approach to this clustering problem involves maximizing the probability of the state sequence, given the model parameters

$$\hat{Q} = \arg\max_{Q} P(s_{1:N}/Q), \tag{5}$$

where $Q = [A_{B_i}^k]_{k=1\ldots K_i}^{i=1:B}$ represents the model parameters. Such an approach to learning the parameters of a mixture model for a "juggling sequence" was shown in [29]. They show how EM can be combined with CONDENSATION to learn the parameters of a mixture model. However, as they point out, there is no guarantee that the clusters found will correspond to semantically meaningful behaviors. For our specific problem of interest, that is, tracking and annotating activities of insects, we would like to learn models for specific behaviors like the waggle dance. Therefore, we use a supervised method to learn the parameters of each behavior. Nevertheless, an unsupervised learning is useful while attempting to learn anomalous behaviors, and we will revisit this issue later.

3.5.2 Supervised Learning

Since it is important to maintain the semantic relationship between learned models and actual behaviors exhibited by the bee, we resort to a supervised learning of the model parameters. For a small training database of videos of bee dances, we obtain manual tracking and labeling of both the motion states and the behaviors exhibited; that is, for a training database, we first obtain the labeling over the three tiers of the hierarchy. For each frame j of the training video, we have the position X^j, the motion state m^j, and the behavior B^j.

Learning the dynamics. The first tier of the three-tier model involves the local motion states like moving straight, turning, waggling, and motionless. As described in (1) and (2), each of these local motion states is modeled either using a Gaussian or using a mixture of Gaussians. The mean and the variance of the corresponding Normal distributions are directly learned from the training as

$$\hat{\mu}_{mi} = E\left[(X^j - X^{j-1})|m^j = i\right] \tag{6}$$

$$= \frac{1}{N_i} \sum_{j=1,2,\ldots N}^{m^j = mi} (X^j - X^{j-1}), \tag{7}$$

$$\Sigma_{mi} = E\left[(X^j - X^{j-1} - \mu_{mi})(X^j - X^{j-1} - \mu_{mi})^T\right] \tag{8}$$

$$= \frac{\sum_{j=1,2,\ldots N}^{m^j = mi} (X^j - X^{j-1} - \hat{\mu}_{mi})(X^j - X^{j-1} - \hat{\mu}_{mi})^T}{N_i - 1}, \tag{9}$$

where the summations are carried out only for the frames in which the annotated motion state for that frame is mi, and the total number of such frames is denoted by N_i. In the case of a mixture of Gaussians model (for turning), we use the EM algorithm to learn the model parameters. In practice, learning the dynamics is the simplest of the three tiers of learning.

Learning the behavior. The second tier of the hierarchy involves the Markov model for each behavior. For the ith behavior B_i, we learn the model parameters by using an ML estimation. As an example, let us assume that the insect exhibited behavior B_i for frame 1 to N. In the training database, we have obtained a corresponding sequence of motion states $s_1, s_2, s_3, \ldots, s_N$, where s_j is one of the four possible motion states (straight, turn, waggle, and motionless) exhibited in frame j. We can learn the model parameters of the Markov model for behavior B_i by

$$\hat{Q}_i = \arg\max_{Q_i} P(s_{1:N}/Q_i). \tag{10}$$

Here, $Q_i = [A_{B_i}^k]^{k=1\ldots K_i}$ represents the model parameters for behavior B_i. In our current implementation, we have modeled behaviors for the waggle dance, the round dance and a stationary bee. We have used Markov models of order 1 so that we need to only estimate the transition probabilities between each motion state. These are estimated as follows:

$$\hat{A}_{B_i}(l,k) = E(P(s_t = k/s_{t-1} = l)) \tag{11}$$

$$= \frac{N_{kl}}{N_l}, \tag{12}$$

where E is the expectation operator, N_l is the number of frames in which the annotated motion state was ml, and N_{kl} is the number of times in which the annotated motion state mk appeared immediately after motion state ml. Note that since this step of the learning procedure concerns only a particular behavior B_i, only the frames whose annotated behavior is B_i are taken into account. Learning the model parameters of a particular behavior depends upon two factors: the inherent variability in the behavior and the amount of training data available for that particular behavior. Some behaviors have significant variability in their executions, and learning the model parameters for these behaviors could be unreliable. Moreover, some behaviors are uncommon, and therefore, the amount of training data available for these behaviors might be too little to accurately learn the model parameters. Experiments to indicate the minimum number of frames that one needs to observe a behavior before one can learn the model parameters are shown in Section 3.7.

Switching between behaviors. The third tier of the model involves the switching between behaviors. The switching between behaviors is also modeled as being Markovian, with the transition matrix denoted as T_B. The transition matrix T_B can be learned as

$$\hat{T}_B(l,k) = E[B^j = k|B^{j-1} = l]. \tag{13}$$

Learning the switching model is the most challenging part of the learning phase. First, within a given length of training data, there might be very few transitions observed and, therefore, sufficient data might not be available to learn the switching matrix T_B accurately. Second, there is really no particular ethological justification to model the transitions between behaviors by using a Markov model, though in practice, the model seems adequate. Therefore, once we learn the transition matrix T_B from the training data, we also ensure

that every transition is possible, that is, $T_B(l,k) \neq 0 \forall (l,k)$, by adding a small value ϵ to every element in the matrix and then normalizing the matrix so that it still represents a transition probability matrix (sum of each row $= 1$).

3.6 Discriminability among Behaviors

The disadvantage of using an supervised learning is that since learning for each behavior is independent of others, there is no guarantee that the learned models are sufficiently distinct for us to be able to distinguish among different behaviors. There is reason, however, to believe that this would be the case, since in actual practice, these behaviors are distinct enough. Nevertheless, we need some quantitative measure to characterize the discriminability between models. This would be of great help, especially when we have several behaviors.

3.6.1 Rabiner-Juang Distance

There are several measures for computing distances between Hidden Markov Models. In particular, one distance measure that is popular is the Rabiner-Juang distance [30]. However, such a distance measure is based on the Kullback–Leibler (KL) distance and, therefore, captures the distance between the asymptotic observation densities. However, in actual practice, we are always called upon to recognize the source model by using observation or state sequences of finite length. In fact, in our specific scenario, we need to reestimate the behavior exhibited by the bee every few frames. Therefore, in such situations, we need to know how long a state/observation sequence is required before we can disambiguate between two models.

3.6.2 Probability of N-Misclassification

Suppose we have D different Markov models $M_1 \ldots M_D$, with M_i being of order K_i. We define the Probability of N-Misclassification for Model M_i as the probability that a state sequence of length N that is generated by model M_i is misclassified to some model M_j, $j \neq i$ using an ML rule:

$$P_{M_i}(NMiscl) = 1 - \sum_{s_{1:N}} P(s_{1:N}/M_i) I(s_{1:N}, i), \tag{14}$$

where the summation is over all state sequences of length N, and $I(s_{1:N}, i)$ is an indicator function, which is 1 only when $P(s_{1:N}/M_i)$ is greater than $P(s_{1:N}/M_j)$ for all $j \neq i$. The number of terms in the summation is S^N, where S is the number of states in the state space. Even for moderate sizes of S and N, this is difficult to compute. However, the summation will be dominated by few of the most probable state sequences. Thus, a tight lower bound can be obtained by Monte Carlo methods of sampling. An approximation to the Probability of N-Misclassification can also be obtained using Monte Carlo sampling methods. This is done by generating K independent state sequences $Seq_1, Seq_2 \ldots Seq_K$, each of length N, randomly using model M_i. For reasonably large K

$$P_{M_i}(NMiscl) \approx 1 - 1/K \sum_{k=1,\ldots K} I(Seq_k, i). \tag{15}$$

Fig. 4 shows the Probability of N-Misclassification for the three modeled behaviors waggle, round, and the stationary bee for different values of N. We choose a window length $N = 25$, which provides us with sufficiently low misclassification errors while being small enough compared to an average length of behaviors so as to not smooth across behaviors.

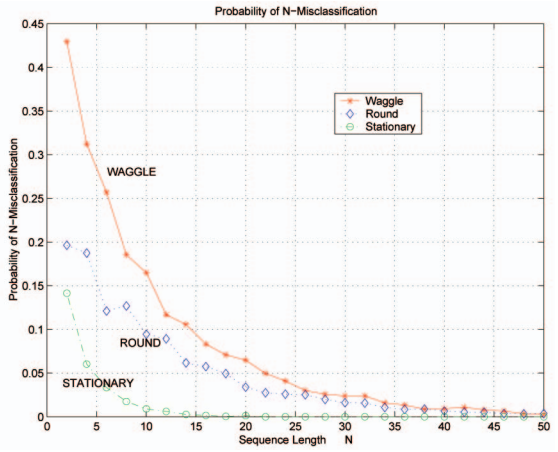


Fig. 4. Probability of N-Misclassification.

3.7 Detecting/Modeling Anomalous Behavior

A change in behavior of the insect would result in the behavior model not being able to explain the observed motion of the insect. When this happens, we need to be able to detect and characterize these abnormal behaviors so that the tracking algorithm is able to continue maintaining the track. A change in behavior can either be slow or drastic. We use the observation likelihood and the expected negative log-likelihood ((ELL) of the observation, given the model parameters) as proposed in [31] and [32] in order to detect drastic and slow changes in behavior.

Drastic change. When there is a drastic change in the behavior of the insect, this would cause the tracking algorithm to lose track. Once it loses track, the image within the shape model of the bee does not resemble the bee anymore. Therefore, the observation likelihood decreases rapidly. This can be used as a statistic to detect drastic changes in behavior. Once the anomalous behavior is detected, it would of course be left to the expert to manually identify and characterize the newly observed behavior.

Slow change. When the change in system parameters is slow, that is, the anomalous behavior is not drastic enough to cause the tracker to lose track, we use a statistic very closely related to the ELL proposed in [31], [32]. Let us assume that we have modeled behavior $M0$. Suppose the actual behavior exhibited by the insect is $M1$. We are required to decide whether the behavior exhibited is $M0$ or not with knowledge of the state sequence $x_{1:N}$ alone. Let hypothesis $H0$ be that the behavior being exhibited is $M0$. Let hypothesis $H1$ be that the behavior exhibited is not $M0$, that is, $\overline{M0}$. The likelihood ratio test for such a hypothesis is given as follows. The state sequence $x_{1:N}$ was generated by model $\overline{M0}$ if and only if

$$\frac{P(\overline{M0}/x_{1:N})}{P(M0/x_{1:N})} \geq \eta \quad \eta > 0 \tag{16}$$

$$\Rightarrow \frac{1 - P(M0/x_{1:N})}{P(M0/x_{1:N})} \geq \eta \quad \eta > 0 \tag{17}$$

$$\Rightarrow P(M0/x_{1:N}) \leq 1/(\eta + 1) \tag{18}$$

$$\Rightarrow P(x_{1:N}/M0)P(M0)/P(x_{1:N}) \leq 1/(\eta + 1) \tag{19}$$

$$\Rightarrow P(x_{1:N}/M0) \leq \beta \quad \beta > 0 \tag{20}$$

$$\Rightarrow D = -log(P(x_{1:N}/M0)) \geq T \quad T = -log(\beta), \tag{21}$$

where D is the decision statistic, and T is the decision threshold.

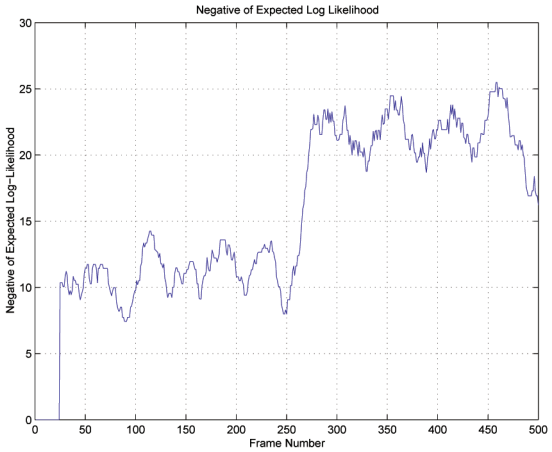


Fig. 5. Abnormality detection statistic.

When the bee exhibits an anomalous behavior, the likelihood that the state sequence observed was generated by the original model decreases, as shown above. Therefore, we can use D as a statistic to detect slow changes. When D increases beyond a certain threshold T, we detect am anomalous behavior. Once slow changes are detected, they can then be automatically modeled. This can be done by learning a mixture model for the observed state sequence by using the principles outlined in [29].

Since we did not have any real video sequence of an abnormal behavior, we performed an experiment on synthetic data. We generated an artificial sequence of motion states for 500 frames. The first 250 frames correspond to the model learned for the waggle dance. The succeeding 250 frames were from a Markov model of order 1, with transition probability matrix A. We computed the negative log-likelihood of the windowed state sequence, with a window length of 25. This statistic D is shown in Fig. 5. Changes in the model parameters are clearly visible at around frame 250, resulting in an increase in the negative log-likelihood (equivalent to an exponential decrease in the probability of the windowed sequence being generated from the waggle model). The anomalous behavior was automatically detected at frame 265. Moreover, we also used the next 150 frames to learn the parameters of the anomalous model (A). The estimated transition probability $\mathrm{matrix}(\hat{A})$ was very close to the actual model parameters

$$A = \begin{pmatrix} .30 & .30 & .20 & .20 \\ .20 & .25 & .25 & .30 \\ .80 & .10 & .05 & .05 \\ .50 & .10 & .20 & .20 \end{pmatrix} \quad \hat{A} = \begin{pmatrix} .30 & .22 & .23 & .25 \\ .30 & .18 & .22 & .30 \\ .78 & .13 & .04 & .05 \\ .47 & .06 & .28 & .19 \end{pmatrix}.$$

4 Shape-and-Behavior-Encoded Particle Filter

We address the tracking problem as a problem of estimating the state X_1^t, given the image observations Y_1^t. Since both the state-transition model and the observation model are nonlinear, methods like the Kalman filter are inadequate.

The particle filter [33], [8], [9] provides a method for recursively estimating the posterior pdf $P(X_t/Y_1^t)$ as a set of N weighted particles $\{X_t^{(i)}, \pi_t^{(i)}\}_{i=1}^N$. from a collection of noisy observations $\mathbf{Y}_1^t$. The state parameters to be estimated are the position and orientation of the bee in the current frame ($\mathbf{X}$). The observation is the color image of each frame ($\mathbf{Y}_t$), from which the appearance of the bee ($\mathbf{Z}_t^{(i)}$) can be computed for each hypothesized position ($\mathbf{X}_t^{(i)}$). The state-transition and the observation models are given by

$$State\ transition\ model: X_t = F_B(X_{t-1}, N_t), \tag{22}$$

$$Observation\ model: Y_t = G(X_t, W_t), \tag{23}$$

where N_t is the system noise, and W_t is the observation noise. The state-transition function F_B characterizes the state evolution for a certain behavior B. In usual tracking problems, the motion model is used to characterize the state-transition function. In our current algorithm, the behavioral model described in Section 3.1 is used as the state-transition function. Therefore, the state at time t, that is, (X_t), depends upon the state at the previous frame (X_{t-1}), the behavioral model, and the system noise. The observation function G models the appearance of the bee (in the current frame) as a function of its current position (state X_t) and observation noise. Once such a description for state evolution has been made, the particle filter provides a method for representing and estimating the posterior pdf $P(X_t/Y_1^t)$ as a set of N weighted particles $\{X_t^{(i)}, \pi_t^{(i)}\}_{i=1}^N$. Then, the state X_t can be estimated as the maximum a posteriori (MAP) estimate

$$\hat{X}_t^{MAP} = \arg\max_{X_t} \pi_t^{(i)}. \tag{24}$$

The complete algorithm is given as follows:

1. **Initialize** the tracker with a sample set according to a prior distribution $p(X_0)$.
2. **For** Frame $= 1, 2, \ldots$
   a) **For** sample $i = 1, 2, 3, \ldots N$
      - **Resample** $X_{t-1} = \{\pi_{t-1}^{(i)}\}$
      - **Predict** the sample $X_t^{(i)}$ by sampling from $F_B(X_{t-1}^{(i)}, N_t)$, where F_B is a Markov model for the behavior B estimated in the previous frame.
      - **Compute Weights** for the particle using the likelihood model, that is, $\pi_t^{(i)} = p(Y_t/X_t^{(i)})$. This is done by first computing the predicted appearance of the bee using the function G and then evaluating its probability from the observation noise model.
   b) **Normalize** the weights using $\pi_t^{(i)} = \pi_t^{(i)} / \sum_{i=1}^N \pi_t^{(i)}$ so that the particles represent a probability mass function.
   c) **Estimate** the MAP or minimum mean square error (MMSE) estimate of the state X_t by using the particles and their weights.
   d) **Compute** the ML estimate $(\hat{s}^t)$ for the current motion state, given the position and orientation in the current and previous frames.
   e) **Estimate** the behavior of the bee by using an ML estimate from the various behavior models as $\hat{B} = \arg\max_j P(\hat{s}_{t-24}^t / B_j)$., where B_j *for* $j = 1, 2, \ldots$ indicate the behaviors modeled.

4.1 Prediction and Likelihood Model

In typical tracking applications, it is customary to use motion models for prediction [9], [12], [15], [11]. We use behavioral models, in addition to motion models. The use of such models for prediction improves tracking performance significantly.

Given the location of the bee in the current frame (X_t) and the image observation given by (Y_t), we first compute the appearance (Z_t) of the bee in the current frame (that is, the color image of the three-ellipse anatomical model of the bee). Therefore, given this appearance $(Z_t^{(i)})$ for each hypothesized position $X_t^{(i)}$, the weight for the ith particle $(\pi_t^{(i)})$ is updated as

$$\pi_t^{(i)} = p\left(Y_t / X_t^{(i)}\right) = p\left(Z_t^{(i)} / X_t^{(i)}\right), \tag{25}$$

where Y_t is the observation. Since the appearance of the bee changes drastically over the video sequence, we use an appearance model consisting of multiple color exemplars $(A_1, A_2, \ldots, A_5)$. The red, green, and blue (RGB) components of color are treated independently and identically. The appearance of the bee in any given frame is assumed to be Gaussian centered around one of these five exemplars, that is

$$P(Z_t) = \frac{1}{5}\sum_{i=1}^{i=5} 0.2 \;\; N(Z; A_i, \Sigma_i), \tag{26}$$

where $N(Z; A_i, \Sigma_i)$ stands for the Normal distribution, with mean A_i and covariance Σ_i. In practice, we modeled the covariance matrix as a diagonal matrix with equal elements on the diagonal, that is, $\Sigma_i = \sigma I$, where I is the identity matrix. The mean observation intensities $A1 - A5$ are learned by specifying the location of the bee in five arbitrary frames of the video sequence. In practice, we also used four of these five exemplars from the training database, whereas the fifth exemplar was estimated from the initialization provided in the first frame of the current video sequence. In either case, the performance was similar. For extremely challenging sequences, with large variations in lighting, the former method performed better than the latter.

4.2 Inference of Dynamics, Motion, and Behavior

Inference on the three-tier hierarchical model is performed using a greedy approach. The inference for the lower tiers is first performed independently, and these estimates are then used in the inference for the next tier. Estimating the current position and orientation of the insect $(\hat{X}^t)$ is performed using a particle filter with observation and state-transition models, as described in the previous section. Once the position and the orientation are estimated using the particle filter, we then use these estimates to infer about the current motion state. The ML estimate for the current motion state, given the position and orientation in the current and previous state, is estimated as

$$\hat{s}^t_{ML} = \arg \max_{mi} \max_{i=1,2,3\ldots} P(\hat{X}^t - \hat{X}^{t-1} | s^t = mi). \tag{27}$$

Finally, we also need to estimate the behavior of the insect in the current frame. Once again, we assume that the inference for the lower tiers has been completed, and based on the estimated motion states $\hat{s}^{1:t}$, we infer the ML estimate for the current behavior. In order to perform this, we also need to decide an appropriate window length W. Based on Section 3.7, we see that a window length W of 25 is a good trade-off between recognition performance and smoothing across behavior transitions. Therefore, we do an ML estimation for the behavior by using a window length of 25 frames as

$$\hat{B} = \arg \max_j P\left(\hat{s}^t_{t-W+1} | B_j\right). \tag{28}$$

Since the behavior model B_j is a simple Markov model of order 1 given by the transition matrix T_{B_j}, this ML estimate is easily obtained as

$$\hat{B} = \arg \max_j P(\hat{s}^t_{t-W+1} | T_{B_j}) \tag{29}$$

$$= \arg \max_j \prod_{i=1,2,\ldots W} T_{B_j}(\hat{s}_{t-i}, \hat{s}_{t+1-i}). \tag{30}$$

5 Experimental Results

5.1 Experimental Methodology

For a training database of videos, manual tracking was performed; that is, at each frame the position, motion, and behavior of the bee were manually labeled. Following the steps outlined in Section 3.5.2, the model for dynamics, behavior, and the behavior transitions was learned. During the test phase, for every test video sequence, the user first identifies the bee to be tracked and initializes the position of the bee by identifying four extreme points on the abdomen, thorax, and head, respectively. Then, the tracking algorithm uses this initialization with a suitably chosen variance as the prior distribution $p(X_0)$ and automatically tracks both the position and the behavior of the bee, as described in Section 4. This is a significant difference in experimental methodology from most other previous work. In [23], they first obtain manually tracked data for the entire video sequence to be analyzed. Then, the Markov model is used in order to classify the various behaviors. In other related work, like [21] and [24], for each test video sequence, the tracking is independently accomplished using a tracking algorithm [12], which has no knowledge of the behavior models. Once the entire video sequence is tracked, an analysis of the tracked data is performed using specific behavior models. The training phase for our algorithm is similar to those in [23], [21], and [24] in the sense that all these algorithms use some kind of labeled data to learn the model parameters for each behavior. However, our algorithm differs from all the others mentioned above in that the behavior model thus learned is used as a prior for tracking, thus enhancing the tracking accuracy. Moreover, this also means that manual labeling is required only for the training sequences and not for any of the test videos.

5.2 Relation to Previous Work

Previous work in tracking and analyzing the behaviors of bees have dealt either with the visual tracking problem [12] or with that of accurately modeling and analyzing the tracked trajectories of the insects [21], [24], [23]. This is the first study that tackles both tracking and behavior modeling in a closed-loop manner. By closing the loop and enabling the tracking algorithm to be aware of the behavior models, we have improved the tracking performance significantly. Experiments in the next section will demonstrate the improvement of the tracking performance for two video sequences that have drastic motions. Once the results of the tracking algorithm are

available, one can, in principle, analyze the tracked trajectories by using any appropriate behavior model: the hierarchical Markov model or the p-SLDS. In all the experiments reported in this paper, we have used the hierarchical Markov motion model to analyze the behavior of the bees.

5.3 Tracking Dancing Bees in a Hive

We conducted tracking experiments on video sequences of bees in a hive. In all the experiments reported, the training data and the test data were mutually exclusive. In the videos, the bees exhibited three behaviors: the waggle dance, the round dance, and a stationary bee. In all our simulations, we used 300 to 600 particles. The video sequences ranged from 50 frames to about 700 frames long. It is noteworthy that when a similar tracking algorithm without a behavioral model was used for tracking, it lost track within 30-40 frames (see Table 1 for details). With our behavior-based tracking algorithm, we were able to track the bees during the entire length of these videos. We were also able to extract parameters like the orientation of the various body parts during each frame over the entire video sequences. We used these parameters to automatically identify the behaviors. We also verified this estimate manually and found it to be robust and accurate.

Fig. 6 shows the structural model of the tracked bee superimposed on the original image frame. In this particular video, the bee was exhibiting a waggle dance. The results are best viewed in color, since the tracking algorithm had color images as observations. The figure shows the top-five tracked particles (with blue being the best particle and red being the fifth best particle). As apparent from the sample frames, the appearance of the dancer varies significantly within the video. These images display the ability of the tracker to maintain track, even under extreme clutter and in the presence of several similar looking bees. Frames 30-34 show the bee executing a waggle dance. Notice that the abdomen of the bee waggles from one side to another.

TABLE 1
Comparison of Our Behavior-Based Tracking Algorithm (BT) with Visual Tracking (VT) [11] and the Same Visual Tracking Algorithm Enhanced with Our Shape Model (VT-S)

Video Name	Video 1			Video 2		
Total Frames	550			200		
Algorithm	VT	VT-S	BT	VT	VT-S	BT
Number of Particles	500	500	500	500	500	500
Successful Tracking	No	No	Yes	No	No	Yes
Number of Missed Tracks	14	10	0	5	5	0
Average No. of Frames Tracked	37	50	550	33	33	200

5.3.1 Occlusions

Fig. 7 shows the ability of the behavior-based tracker to maintain track during occlusions in two different video sequences. There is significant occlusion in frames 170, 172, and 187 of video sequence 1. In fact, in fame 172, occlusion forces the posterior pdf to become bimodal (another bee in close proximity). However, we see that the track is regained

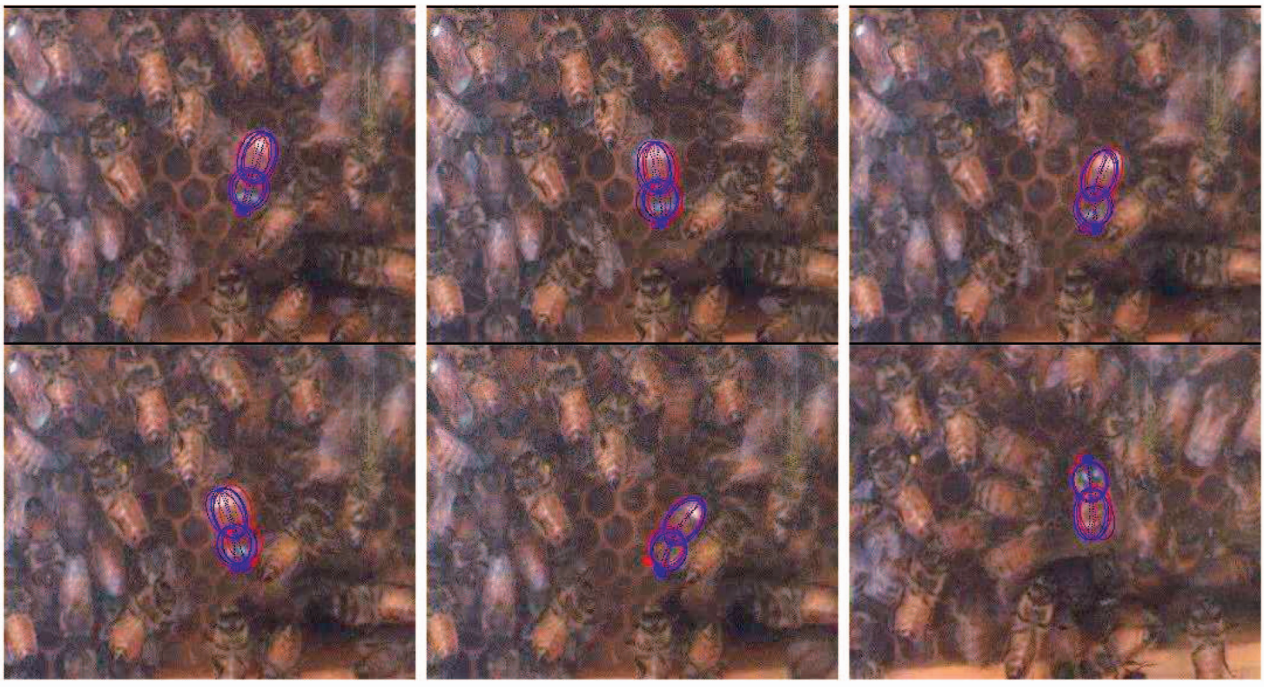


Fig. 6. Sample frames from a tracked sequence of a bee in a beehive. Images show the top-five particles superimposed on each frame. Blue denotes the best particle, whereas red denotes the fifth best particle. Frame numbers row-wise from top left: 30, 31, 32, 33, 34, and 90.

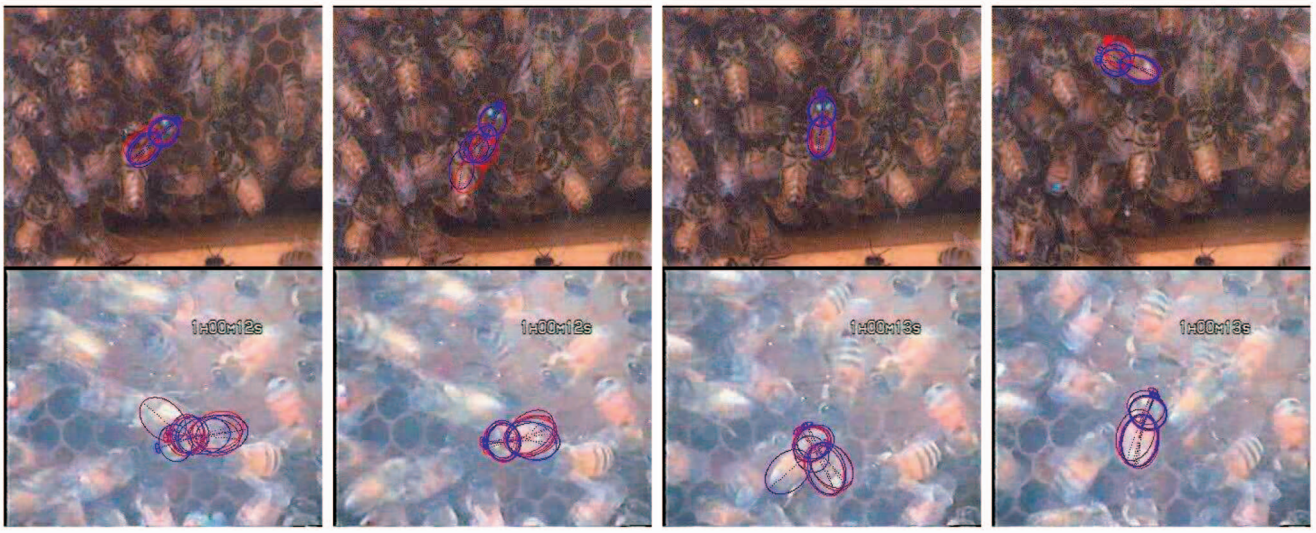


Fig. 7. Ability of the behavior-based tracker to maintain tracking during occlusions in two different video sequences. Images show the top-five particles superimposed on each frame. Blue denotes the best particle, and red denotes the fifth best particle. Row 1: video 1 frames 170, 172, 175, and 187. Row 2: video 2 frames 122, 123, 129, and 134.

when the bee emerges out of occlusion in frame 175. In frame 187, we see that the thorax and the head of the bee are occluded, whereas the abdomen of the bee is visible. Therefore, the estimate of the abdomen is very precise (all five particles shown indicate the same orientation of abdomen). Since the thorax is not visible, we see that there is a high variance in the estimate of the orientation of the thorax and the head. Structural modeling has ensured that in spite of the occlusion, only physically realizable orientations of the thorax and the head are maintained. In frame 122 of video sequence 2, we see that another bee completely occludes the bee being tracked. This creates confusion in the posterior distribution of the position and orientation. However, behavior modeling ensures that most particles still track the correct bee. Moreover, at the end of occlusion in frame 123, the track is regained. Frame 129 in video sequence 2, shows another case of severe occlusion. However, once again, we see that the tracker maintains track during occlusion and immediately after occlusion (frame 134). Thus, behavior modeling helps maintain tracking under extreme clutter and severe occlusions.

5.4 Importance of Shape and Behavioral Model for Tracking

To quantify the importance of the shape and the behavioral model in the above-mentioned tracking experiments, we also implemented another recent and successful tracking algorithm also based on a particle-filter-based inference. We implemented the visual tracking algorithm based on an adaptive appearance model described in [11]. We also implemented a minor variation of this algorithm by incorporating our shape model within their framework. In either case, we spent a significant amount of time and effort in varying the parameters of the algorithm so as to obtain the best possible tracking results with these algorithms. We compare the performance of our tracking algorithm to the two approaches mentioned above on two different video sequences in Table 1. Both these videos consisted of a handheld camera held over the vertical face of the beehive. There were several bees within the field of view of each of these videos, but we were interested in tracking the dancing bees in both videos. Thus, we initialized the tracking algorithm on the dancers in all these experiments. Moreover, these video sequences were also specifically chosen, since the bees exhibited drastic motion changes during the videos, and the illumination and lighting remained fairly consistent during the course of these videos. This gives us a nice testbed to evaluate the performance of the shape-and-behavior model fairly independent of other challenges in tracking like illumination. The incorporation of the shape constraints improves the performance of the tracking algorithm, showing that an anatomically correct model improves tracking performance. We declared that a tracking algorithm "lost track" when the distance between the estimated position of the bee and the actual position of the bee on the image was greater than half the length of the bee. We see that although the proposed tracking algorithm was able to successfully track the bee over the length of the entire video sequences, the other approaches implemented were not able to do so. The table also clearly shows that the behavior-aided tracking algorithm that we propose significantly outperforms the adaptive appearance-based tracking [11].

5.5 Comparison with Ground Truth

We validated a portion of the tracking result by comparing it with a "ground-truth" track obtained using manual ("point and click") tracking by an experienced human observer. We find that the tracking result obtained using the proposed method is very close to manual tracking. The mean differences between manual and automated tracking using our method are given in Table 2. The positional differences are small compared to the average length of the bee, which is about 80 pixels (from the front of its head to the tip of its abdomen).

5.6 Modes of Failure

Even in the presence of the improved behavior-model-based tracking algorithm, there are some extremely challenging

TABLE 2
Comparison of Our Tracking Algorithm with Ground Truth

	Average positional difference between Ground Truth and our algorithm
Center of Abdomen	4.5 pixels
Abdomen Orientation	0.20 radians (11.5 deg)
Center of Thorax	3.5 pixels
Thorax orientation	0.15 radians (8.6 deg)

video sequences, where the improved tracking algorithm resulted in some missed tracks. The primary modes of failure are given as follows:

- **Illumination.** We are interested in studying and analyzing bee dances. Bee dances are typically performed in the dark environment of the beehive. Since the bees typically prefer to dance only in minimal lighting, some of the videos end up being quite dark. Moreover, there are also significant illumination changes, depending upon the exact position of the dancer on the beehive. These illumination changes posed the most significant challenge for the tracking algorithm, and most of the tracking failures can be attributed to illumination-based challenges in tracking. Even in such videos, the tracking algorithm with the behavior-and-anatomical model outperforms the adaptive appearance-based tracking algorithm [11]. Recently, a lot of research effort has been invested in studying and developing appearance models that are either robust or invariant to illumination changes [34], [35]. Augmenting the appearance model with illumination-invariant appearance models might reduce some of the errors caused due to illumination changes. Since the focus of this work was on behavior modeling, we did not systematically analyze the effect of incorporating such illumination invariant appearance models in our algorithm.
- **Occlusions.** Another reason for some of the observed tracking failures is occlusions. The beehive is full of several bees, which are very similar in appearance. Sometimes, the dancing bee disappears below other bees and then reappears after a few frames. As described in Section 5.3.1, when the dancing bee is occluded for a relatively small number of frames, the algorithm is able to regain track when the bee emerges out of occlusions (refer to Fig. 7). However, in some videos, the dancing bee remains occluded for over 30 frames or more. During such cases of extreme occlusions, the tracking algorithm is unable to regain track, and the only reasonable way to regain track would be to design an initialization algorithm that can potentially discover dancing bees in a hive. This would be an extremely challenging task, considering the complex nature of motions in a beehive and the fact that there are several moving bees in every frame of the video. In practice, it might be a good idea to perform manual reinitialization in such videos.

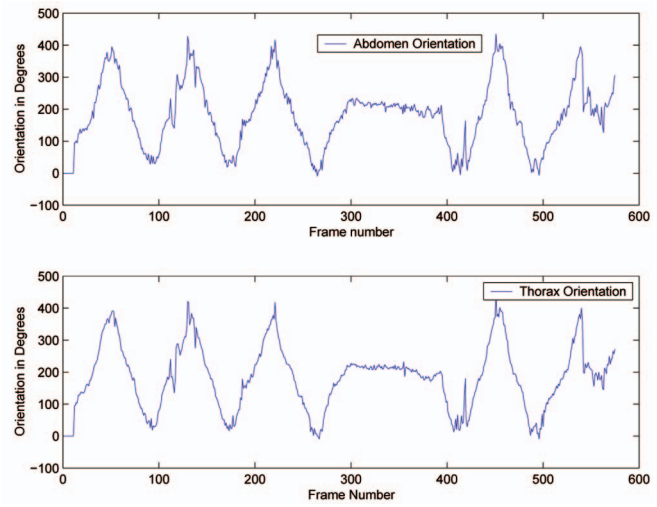


Fig. 8. The orientation of the abdomen and the thorax of a bee in a video sequence of about 600 frames.

5.7 Estimating Parameters of the Waggle Dance

Foraging honeybees communicate the distance, the direction, and the attractiveness of the food source through the waggle dance. The details of the waggle dance were discussed in detail in Section 1.2. The duration of the waggle portion of the dance and the orientation of the waggle axis are some of the parameters of interest while analyzing the bee dances. The duration of the waggle portion of the dance may be estimated by carefully filtering the orientation of the thorax and the abdomen of a honeybee as it moves around in its hive. Moreover, the orientation of the waggle axis can also be estimated from the orientation of the thorax during the periods of waggle.

Fig. 8 shows the estimated orientation of the abdomen and the thorax in a video sequence of around 600 frames. The orientation is measured with respect to the vertically upward direction in each image frame, and a clockwise rotation would increase the angle of orientation, whereas a counterclockwise rotation would decrease the angle of orientation.

The waggle dance is characterized by the central waggling portion, which is immediately followed by a turn, a straight run, another turn, and a return to the waggling section, as shown in Fig. 3. After every alternate waggling section, the direction of the turning is reversed. This is clearly seen in the orientation of both the abdomen and the thorax. The sudden change in slope (from positive to negative, or vice versa) of the angle of orientation denotes the reversal of turning direction. During the waggle portion of the dance, the bee moves its abdomen from one side to another while continuing to move forward slowly. The large local variation in the orientation of the abdomen just before every reversal of direction shows the waggling nature of the abdomen. Moreover, the average angle of the thorax during the waggle segments denotes the direction of the waggle axis.

In order to estimate the parameters of the waggle dance, we use some heuristics described below. During the waggling

TABLE 3
Comparison of Waggle Detection with Hand Labeling by Expert

	Automated Labeling (Frame Numbers)	Expert Labeling (Frame Numbers)
Waggle 1	46 - 55	46 - 56
Waggle 2	88 - 95	89 - 97
Waggle 3	127 - 141	127 - 140
Waggle 4	171 - 180	171 - 181
Waggle 5	210 - 222	211 - 222
Waggle 6	255 - 274	257 - 274
Waggle 7	406 - 424	407 - 423
Waggle 8	444 - 461	444 - 461
Waggle 9	486 - 502	486 - 502
Waggle 10	532 - 543	534 - 544

portion of the dance, the bee moves its abdomen from one side to another in the direction transverse to the direction of motion. The average absolute motion of the center of the abdomen about an axis transverse to the axis of motion is used as a waggle detection statistic. When this statistic is large, the probability of waggle during that particular frame is large. Moreover, we also recognize that the waggle portion of the dance is followed by a change in the direction of turning. Therefore, only those frames that are followed by a change in direction of turning and have a high "waggle detection statistic" are labeled as waggle frames. Once the frames in which the bee waggles are estimated, it is then relatively straightforward to estimate the waggle axis. The waggle axis is estimated as the average orientation of the thorax during a single waggle run. Table 3 shows the frames that were detected as waggle frames automatically. We also had the same video sequence hand labeled by an expert. The table also shows the frames that were labeled as "waggle" by the expert. There were a total of 138 frames that were labeled as "waggle." Of these 138 frames, 133 frames were correctly labeled automatically using the procedure described above.

6 Conclusions and Future Work

We proposed a method using behavioral models to reliably track the position/orientation and the behavior of an insect and applied it to the problem of tracking bees in a hive. We also discussed issues in learning models, discriminating between behaviors, and detecting and modeling abnormal behaviors. Specifically, for the waggle dance, we also proposed and used some simple statistical measures to estimate the parameters of interest in a waggle dance. The modeling methodology is quite generic and can be used to model activities of humans by using appropriate features. We are working to extend the behavior model by modeling interactions among insects. We are also looking to extend the method to problems like analyzing human activities.

Acknowledgments

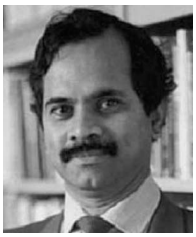
This work was done when Dr. Srinivasan was at the Australian National University, Canberra. This work was partially supported by the NSF-ITR Grant 0325119, US Army Research Office MURI ARMY-W911NF0410176, Technical Monitor Dr. Tom Doligalski, US AFOSR Contract F62562, US AOARD contract FA4869-07-1-0010, and Australian Research Council Grants FF0241328, CE0561903, and DP020863.

References

[1] V. Frisch, *The Dance Language and Orientation of Bees.* Harvard Univ. Press, 1993.
[2] M. Srinivasan, S. Zhang, M. Lehrer, and T. Collett, "Honeybee Navigation en Route to the Goal: Visual Flight Control and Odometry," *J. Experimental Biology,* vol. 199, pp. 237-244, 1996.
[3] T. Neumann and H. Bulthoff, "Insect-Inspired Visual Control of Translatory Flight," *Proc. Sixth European Conf. Artificial Life,* pp. 627-636, 2001.
[4] F. Mura and N. Franceschini, "Visual Control of Altitude and Speed in a Flight Agent," *Proc. Third Int'l Conf. Simulation of Adaptive Behavior: From Animal to Animats,* pp. 91-99, 1994.
[5] G. Hager and P. Belhumeur, "Efficient Region Tracking with Parametric Models of Geometry and Illumination," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, pp. 1025-1039, 1998.
[6] D. Comaniciu, V. Ramesh, and P. Meer, "Real-Time Tracking of Non-Rigid Objects Using Mean-Shift," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 142-149, 2000.
[7] T. Broida, S. Chandra, and R. Chellappa, "Recursive Techniques for the Estimation of 3D Translation and Rotation Parameters from Noisy Image Sequences," *IEEE Trans. Aerospace and Electronic Systems,* vol. 26, pp. 639-656, 1990.
[8] A. Doucet, N. Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice.* Springer-Verlag, 2001.
[9] M. Isard and A. Blake, "Contour Tracking by Stochastic Propagation of Conditional Density," *Proc. Fourth European Conf. Computer Vision,* pp. 343-356, 1996.
[10] J. Liu and R. Chen, "Sequential Monte Carlo for Dynamical Systems," *J. Am. Statistical Assoc.,* vol. 93, pp. 1031-1041, 1998.
[11] S. Zhou, R. Chellappa, and B. Moghaddam, "Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filters," *IEEE Trans. Image Processing,* vol. 11, pp. 1434-1456, 2004.
[12] Z. Khan, T. Balch, and F. Dellaert, "A Rao-Blackwellized Particle Filter for Eigen Tracking," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2004.
[13] H. Lee and Z. Chen, "Determination of 3D Human Body Posture from a Single View," *Computer Vision, Graphics, Image Processing,* vol. 30, pp. 148-168, 1985.
[14] C. Sminchisescu and B. Triggs, "Covariance Scaled Tracking for Monocular 3D Body Tracking," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2001.
[15] M. Black and A. Jepson, "A Probabilistic Framework for Matching Temporal Trajectories," *Proc. Seventh IEEE Int'l Conf. Computer Vision,* vol. 22, pp. 176-181, 1999.
[16] T. Zhao, T. Wang, and H. Shum, "Learning a Highly Structured Motion Model for 3D Human Tracking," *Proc. Fifth Asian Conf. Computer Vision,* 2002.
[17] J. Cheng and J. Moura, "Capture and Representation of Human Walking in Live Video Sequence," *IEEE Trans. Multimedia,* vol. 1, no. 2, pp. 144-156, 1999.
[18] C. Bregler, "Learning and Recognizing Human Dynamics in Video Sequences," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 1997.
[19] T. Zhao and R. Nevatia, "3D Tracking of Human Locomotion: A Tracking as Recognition Approach," *Proc. 16th Int'l Conf. Pattern Recognition,* 2002.
[20] V. Pavlovic, J. Rehg, T. Cham, and K. Murphy, "A Dynamic Bayesian Network Approach to Figure Tracking Using Learned Dynamic Models," *Proc. Seventh IEEE Int'l Conf. Computer Vision,* 1999.
[21] S.M. Oh, J.M. Rehg, T. Balch, and F. Dellaert, "Learning and Inference in Parametric Switching Linear Dynamic Systems," *Proc. 10th IEEE Int'l Conf. Computer Vision,* 2005.

[22] T.D. Seeley, "The Tremble Dance of the Honeybee: Message and Meanings," *Behavioral Ecology and Sociobiology,* vol. 31, pp. 375-383, 1992.
[23] A. Feldman and T. Balch, "Automatic Identification of Bee Movement Using Human Trainable Models of Behavior," *Math. and Algorithms of Social Insects,* Dec. 2003.
[24] S.M. Oh, J.M. Rehg, T. Balch, and F. Dellaert, "Parameterized Duration Modeling for Switching Linear Dynamic Systems," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2006.
[25] M. Isard and A. Blake, "A Mixed-State Condensation Tracker with Automatic Model-Switching," *Proc. Sixth IEEE Int'l Conf. Computer Vision,* 1998.
[26] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems.* Artech House, 1999.
[27] S. Fine, Y. Singer, and N. Tishby, "The Hierarchical Hidden Markov Model: Analysis and Applications," *Machine Learning,* vol. 32, no. 1, pp. 41-62, 1998.
[28] X. Koutsoukos and P. Antsaklis, "Hierarchical Control of Piecewise Linear Hybrid Dynamical Systems Based on Discrete Abstractions," ISIS technical report, Feb. 2001.
[29] A. Blake, B. North, and M. Isard, "Learning Multi-Class Dynamics," *Advances in Neural Information Processing Systems,* pp. 389-395, 1999.
[30] B. Juang and L. Rabiner, "A Probabilistic Distance Measure for Hidden Markov Models," *AT&T Technical J.,* vol. 64, pp. 391-408, 1985.
[31] N. Vaswani, "Additive Change Detection in Nonlinear Systems with Unknown Change Parameters," *IEEE Trans. Signal Processing,* 2006.
[32] N. Vaswani, "Change Detection in Partially Observed Nonlinear Dynamic Systems with Unknown Change Parameters," *Am. Control Conf.,* 2004.
[33] N. Gordon, D. Salmond, and A. Smith, "Novel Approach to Non-Linear/Non-Gaussian Bayesian State Estimation," *Proc. IEE Radar and Signal Processing,* vol. 140, pp. 107-113, 1993.
[34] D. Freedman and M. Turek, "Illumination-Invariant Tracking via Graph Cuts," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2005.
[35] Y. Xu and A. Roy-Chowdhury, "Integrating Motion, Illumination and Structure in Video Sequences, with Applications in Illumination-Invariant Tracking," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 5, May 2006.

Ashok Veeraraghavan received the BTech degree in electrical engineering from the Indian Institute of Technology, Madras, in 2002 and the MS degree from the Department of Electrical and Computer Engineering, University of Maryland, College Park, in 2004. He is currently working toward the PhD degree in the Department of Electrical and Computer Engineering, University of Maryland. His research interests are signal, image, and video processing, computer vision, pattern recognition, and graphics. He is a student member of the IEEE and the IEEE Computer Society.



Rama Chellappa received the MSEE and PhD degrees in electrical engineering from Purdue University, West Lafayette, Indiana, in 1978 and 1981, respectively. Since 1991, he has been a professor of electrical engineering and an affiliate professor of computer science at the University of Maryland, College Park. He is with the Center for Automation Research as a director and also with the Institute for Advanced Computer Studies as a permanent member. Recently, he was named a Minta Martin Professor of Engineering. Prior to joining the University of Maryland, he was with the University of Southern California (USC), Los Angeles as an assistant from 1981 to 1986, an associate professor from 1986 to 1991, and the director of the Signal and Image Processing Institute from 1988 to 1990. Over the last 26 years, he has published numerous book chapters and peer-reviewed journal and conference papers. He has also coedited and coauthored many research monographs. His current research interests are face and gait analysis, 3D modeling from video, automatic target recognition from stationary and moving platforms, surveillance and monitoring, hyperspectral processing, image understanding, and commercial applications of image processing and understanding. Dr. Chellappa has served as an associate editor of four IEEE Transactions. He was a co-Editor-in-Chief of *Graphical Models and Image Processing*. He also served as the Editor-in-Chief of the *IEEE Transactions on Pattern Analysis and Machine Intelligence*. He served as a member of the IEEE Signal Processing Society Board of Governors and as its Vice President of Awards and Membership. He has received several awards, including the National Science Foundation Presidential Young Investigator Award in 1985, three IBM Faculty Development Awards, the 1990 Excellence in Teaching Award from the School of Engineering at USC, the 1992 Best Industry Related Paper Award from the International Association of Pattern Recognition (with Q. Zheng), and the 2000 Technical Achievement Award from the IEEE Signal Processing Society. He was elected as a Distinguished Faculty Research Fellow (1996 to 1998), and as a Distinguished Scholar-Teacher (2003) at University of Maryland. He coauthored a paper that received the Best Student Paper in the Computer Vision Track at the International Association of Pattern Recognition in 2006. He is a corecipient of the 2007 Outstanding Innovator of the Year Award (with A. Sundaresan) from the Office of Technology Commercialization at University of Maryland and received the A.J. Clark School of Engineering Faculty Outstanding Research Award. He was recently elected to serve as a Distinguished Lecturer of the Signal Processing Society. He is a Fellow of the International Association for Pattern Recognition. He has served as a General the Technical Program Chair for several IEEE international and national conferences and workshops. He is a Golden Core Member of IEEE Computer Society and received its Meritorious Service Award in 2004. He is a fellow of the IEEE.

Mandyam Srinivasan received the bachelor's degree in electrical engineering from Bangalore University, a master's degree in electronics from the Indian Institute of Science, a PhD degree in engineering and applied science from Yale University, a DSc degree in neuroethology from the Australian National University, and an honorary doctorate (doctor *honoris causa*) from the University of Zurich. He is currently a professor of visual neuroscience at the Queensland Brain Institute, University of Queensland. His research focuses on the principles of visual processing in simple natural systems and on the application of these principles to machine vision and robotics. He is a fellow of the Australian Academy of Science, a fellow of the Royal Society of London, and an Inaugural Australian Research Council Federation Fellow. He is a member of the IEEE.