M.I.T Media Laboratory Perceptual Computing Section Technical Report No. 415
Edited version published in the Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition, April 14-16, 1998 in Nara, Japan.

Dynamic Models of Human Motion

Christopher R. Wren and Alex P. Pentland

MIT Media Laboratory; 20 Ames Street; Cambridge MA 02139 USA
{cwren}@media.mit.edu
http://www.media.mit.edu/vismod/

Abstract

This paper describes a framework for human motion understanding, defined here as estimation of the physical state of the body (the Plant) combined with interpretation of that part of the motion that cannot be predicted by the plant alone (the Behavior). The described behavior system operates in conjunction with a real-time, fully-dynamic, 3-D person tracking system that provides a mathematically concise formulation for incorporating a wide variety of physical constraints and probabilistic influences. The framework takes the form of a non-linear recursive filter that enables lower-level, probabilistic processes to take advantage of the contextual knowledge encoded in the higher-level models.

1 Introduction

This paper describes a real-time, fully-dynamic, 3-D person tracking system that is able to tolerate full (temporary) occlusions and whose performance is substantially unaffected by the presence of multiple people. The system is driven by 2-D *blob features* observed in two or more cameras [1, 22] and by behavior models that estimate control signals. These features and controls are then probabilistically integrated into a fully-dynamic 3-D skeletal model, which in turn drives the 2-D feature tracking process by setting appropriate prior probabilities. The intrinsic state of the skeletal model is also used by the behavior module to choose the appropriate control strategy.

The feedback between 3-D model and 2-D image features is an extended Kalman filter. One unusual aspect of our approach is that the filter directly couples raw pixel measurements with an articulated dynamic model of the human skeleton. Previous attempts at person tracking have utilized a generic set of image features (e.g., edges, optical flow) that were computed as a preprocessing step, without consideration of the task to be accomplished. In this aspect our system is similar to that of Dickmanns in automobile control [4], and we believe that we can obtain similar advantages in efficiency and stability though this direct coupling. A second unusual aspect is that the Kalman filter goes beyond passive physics of the body by incorporating various patterns of control (which we call 'behaviors') that are learned from observing humans while they perform various tasks.

This paper will illustrate the structure of the behavior system with some simple examples in Section 3. We will then briefly discuss the formulation of our 3-D skeletal model in Section 4, followed by an explanation of how to drive that model from 2-D probabilistic measurements, how to 2-D observations and feedback relate to that model in Section 5. Finally, we will report on experiments showing an increase in 3-D tracking accuracy, insensitivity to temporary occlusion, and the ability to handle multiple people in Section 7.

1.1 Related Work

In recent years there has been much interest in tracking the human body using 3-D models with kinematic and dynamic constraints. Perhaps the first efforts at body tracking were by Badler and O'Rourke 1980, followed by Hogg 1988 [11, 10]. These early efforts used edge information to drive a kinematic model of the human body. More recently, several authors have applied variations on this basic method to the body tracking problem [18, 2]. Because of the highly non-linear nature of the problem, all of these systems require fairly precise initialization, and can not handle the full range of common body motion.

Following this early work using kinematic models, some researchers began using dynamic constraints to track the human body. Pentland and Horowitz 1991 employed non-rigid finite element models driven by optical flow [12], and Metaxas and Terzopolous's 1993 system employing deformable superquadrics [7, 9] driven by 3-D point and 2-D edge measurements. Again, these systems required precise initialization and could handle a limited range of body motion.

More recently, Gavrila and Davis [5] and Rehg and Kanade [17], have demonstrated that an analysis-synthesis approach, using kinematic models driven by edge data, has the potential to deal with limited occlusions, and thus to handle a greater range of body motions.

The work described in this paper attempts to combine the the dynamic modeling work with the advantages of an analysis-synthesis approach, by use of an extended Kalman filter formulation that couples a fully dynamic skeletal model with observations of raw pixel values, as

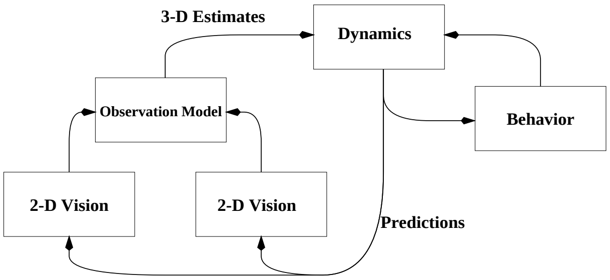

Figure 1: The flow of information though the system. 2-D observations are combined to drive a 3-D dynamic model of the skeleton. This dynamic model uses a control law that chooses 'typical' behaviors when it is necessary to choose among multiple possible legal trajectories. Predictive feedback from the dynamic model is provided by setting priors for the 2-D observations process.

modeled by probabilistic 'blob' models.

This system also attempts to explicitly incorporate learned patterns of control into the body model. The approach we take is based on the behavior modeling framework introduced in Pentland and Liu 1995 [14]; it is also related to the behavior modeling work of Blake 1996 [6] and Bregler 1997 [3]. However, this controller operates on a 3-D non-linear model of human motion that is closer to true body dynamics than 2-D linear models.

2 Mathematical Framework

The human body is a complex dynamic system, whose visual features are time-varying, noisy signals. Accurately tracking the state of such a system requires use of a recursive estimation framework, as illustrated in figure 1. The elements of the framework are the observation model relating noisy low-level features to the higher-level skeletal model and *vice versa*, the dynamic skeletal model, and a model of typical behaviors. Section 4 explains how, with a bit of work, it is possible to approximately model the physical aspects of the human body. Section 5 explains the observation model. The next section explains an approach for modeling human behaviors.

3 The Idea

Observations of the human body reveal an interplay between the passive evolution of a physical system (the human body) and the influences of an active, complex controller (the nervous system). Given the available tools for modeling physical systems, it is convenient to model these two elements separately, as illustrated in Figure 2.

In the classical Observer model, the Observer has direct access to the control being applied to the Plant, as shown in Figure 3.

In the case of human motion, direct access to control signals is very difficult, or impossible. So the situation we're presented with is better modeled by Figure 4. In

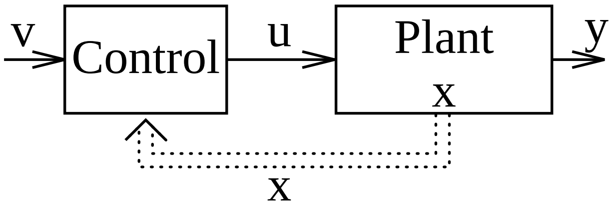

Figure 2: A block diagram of human motion. The **Plant** models the physical evolution of the system, but there is also significant structure in the peripheral nervous system, here lumped into **Control**

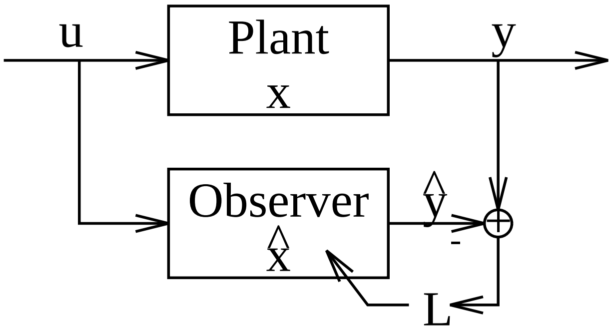

Figure 3: This is the classic Observer configuration, where the control input is available to to the Observer.

this formulation, in the place of the control signal, is an estimated control signal. The estimated control signal is a function of the observation error, and possibly the Observer state.

In Figure 4, $\tilde{y}$ is the observation error. When the above Observer is implemented as a Kalman filter, the observation errors are the Innovations. For an optimally tuned filter, the Innovations are a zero mean Gaussian noise process if the state disturbance is a zero mean Gaussian noise process. When the control signal isn't zero mean Gaussian, this is expressed as a bias in the Innovations. Thus the Innovations process is a natural place to look for structure in the controller.

The structure of the controller provides important constraints on the system evolution. Understanding those constraints is an important element of understanding motion that we're observing. However, it is *very* difficult to explicitly model the human nervous system, so the approach of using observed data to estimate probability distributions over this control space is very appealing.

3.1 A Model for Control

By collecting data from real human motion our system models behavior patterns as statistical densities over configuration space. Different configurations have different observation probabilities.

One very simple behavior model is the mixture model, in which distribution is modeled as a collection of Gaus-

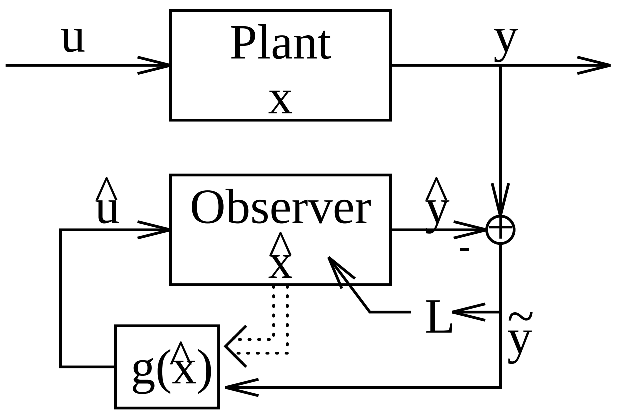

Figure 4: The proposed Observer configuration for the case when the control signal is unavailable.

sians. In this case the composite density is described by:

$$\sum_{k=1}^{N} P_k \cdot \Pr(\mathbf{O}|\lambda = k) \quad (1)$$

where P_k is the observed prior probability of sub-model k.

The mixture model represents a clustering of data into regions within the observation space. Since human motion evolves over time, in a complex way, it is advantageous to explicitly model temporal dependence and internal states. A Hidden Markov Model (HMM) is one way to do this, and has been shown to perform quite well recognizing human motion[19].

The probability that the model is in a certain state, S_j given a sequence of observations, $\mathbf{O}_1, \mathbf{O}_2, \ldots, \mathbf{O}_N$, is defined recursively. For two observations, the density is:

$$\Pr(\mathbf{O}_1, \mathbf{O}_2, \mathbf{q}_2 = S_j) = \left[\sum_{i=1}^{N} \pi_i b_i(\mathbf{O}_1)\mathbf{a}_{ij}\right] b_j(\mathbf{O}_2) \quad (2)$$

Where π_i is the prior probability of being in a state i, and $b_i(\mathbf{O})$ is the probability of making the observation $\mathbf{O}$ while in state i. This is the Forward algorithm for HMM models [16].

Estimation proceeds by identifying the most likely state given the current observation and the last state, and then using the observation density of that state as described above. We restrict the observation densities to be either a Gaussian or a mixture of Gaussians. There are well understood techniques for estimating the parameters of the HMM from data [16]. Figure 5 shows some representative HMM topologies.

3.2 A Simple Example

A simple example is helpful for illustrating the idea expressed in Section 3.1. This section explores the application of hybrid models to the domain of simple mouse gestures. Hundreds of examples of circles, triangles, and scribbles were collected. This data was than used to train two classes of HMMs. The HMMs were all initialized to have five states with the possibility of skipping up to two states per transition.

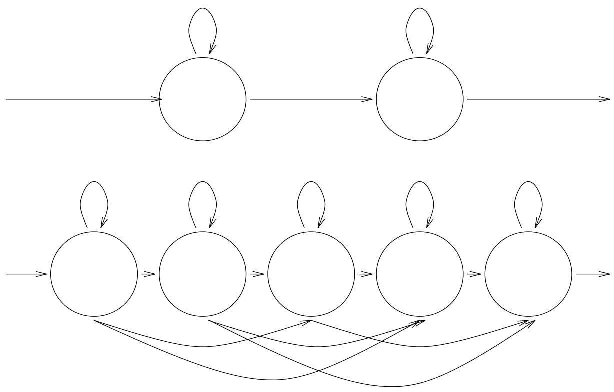

Figure 5: **Top:** A typical illustration of a two state HMM. Circles represent states with associated observation probabilities, and arrows represent non-zero transition arcs, with associated probability. **Bottom:** This is an illustration of a five state HMM. The arcs under the state circles model the possibility that some states may be skipped.

	circle	triangle	scribble
Deltas	100%	100%	100%
Innovations	100%	100%	100%

Table 1: Recognition rates for the two behavior models. Obviously both models are sufficiently powerful classifiers for this simple set of gestures. For comparison, models trained on absolute position perform at chance.

One class of HMMs, the Delta models, were trained on the differences between the last mouse state and the current state. This is a well known technique when using HMMs to recognize human gesture [19].

The other class of HMMs, the Innovation models, were trained on the innovations sequence from a Kalman filter. the innovation is the error between on observation and the prediction of that observation by the linear model inside the filter. That is, these HMMs were trained on the part of the motion that was not solely due to the evolution of a dynamic model.

Table 1 shows that from a classification point of view, with 100% recognition rates, both classes of model are describing the data very well. However, it is necessary that the models not only classify the actions of the user, but also allow prediction.

Figure 6 demonstrates the difference in the predictive power of two types of model. Locally, where we would expect predictive power to be the strongest, the Delta

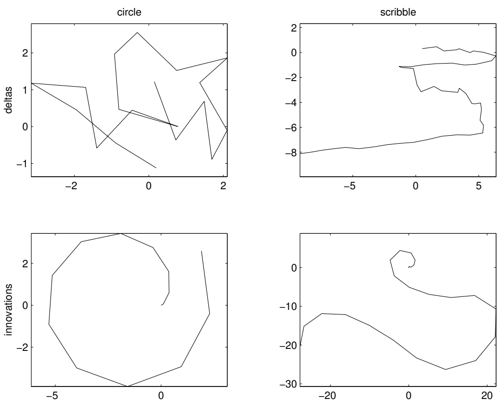

Figure 6: These plots show some examples of synthesized gestures. The top row contains examples of gestures produced by integrating over the outputs of the Delta models. The bottom row shows examples produced by Kalman filter driven by the outputs of the Innovation models. Notice the noise in the top row that makes this method less suitable for prediction.

models provide very noisy output. By contrast the Innovation models produce more reasonable output. The key difference is the lack, in the Delta models, of an explicit representation of the inherent dynamics of the data.

Since the Innovation models have an explicit model of the system dynamics, the HMM parameters can be used to model the innovation, that aspect of the signal that cannot be predicted by the dynamic model. We call patterns in the innovations "effects of control" or "behaviors". The next section examines more powerful dynamic models.

4 Dynamics

There are a wide variety of ways to model physical systems. The model needs to include parameters that describe the *links* that compose the system, as well as information about the hard *constraints* that connect these links to one another. A model that only includes this information is called a *kinematic* model, and can only describe the static states of a system. The state vector of a kinematic model consists of the model state, $\mathbf{q}$, and the model parameters, $\mathbf{p}$.

A system in motion is more completely modeled when the *dynamics* of the system are modeled as well. A dynamic model describes the state evolution of the system over time. In a dynamic model the state vector includes velocity as well as position: $\mathbf{q}, \dot{\mathbf{q}}; \mathbf{p})$. And state evolves according to Newton's First Law:

$$\ddot{\mathbf{q}} = \mathbf{W} \cdot \mathbf{Q} \tag{3}$$

Where $\mathbf{Q}$ is the vector of external forces applied to the system, and $\mathbf{W}$ is the inverse of the system mass matrix. The mass matrix describes the distribution of mass in the system.

4.1 Hard Constraints

Hard constraints represent absolute limitations imposed on the system. One example of a kinematic constraint is a skeletal joint. Our model instead follows the *virtual work* formulation [21]. In a virtual work formulation, all the links in a model have full range of unconstrained motion. Hard kinematic constraints on the system enforced by a special set of forces $\mathbf{c}$:

$$\ddot{\mathbf{q}} = \mathbf{W} \cdot (\mathbf{Q} + \mathbf{c}(\mathbf{q}, t)) \tag{4}$$

The formulas governing these constraints can be modified at run-time. It is essential that the constraint forces do not add energy to the system.

It can be shown that this requirement is satisfied if they are constructed so they lie in the null space complement of the constraint Jacobian:

$$\mathbf{c}(\mathbf{q}, t) = \lambda \frac{\partial \dot{\mathbf{c}}}{\partial \mathbf{q}} \tag{5}$$

Combining that equation with the definition of the constraints results in a linear system of equations with only the one unknown, λ:

$$-\left[\frac{\partial \mathbf{c}}{\partial \mathbf{q}}^T \mathbf{W} \frac{\partial \mathbf{c}}{\partial \mathbf{q}}\right] \lambda = \frac{\partial \mathbf{c}}{\partial \mathbf{q}}^T \mathbf{W}\mathbf{Q} + \frac{\partial \dot{\mathbf{c}}}{\partial \mathbf{q}} \dot{\mathbf{q}} + \frac{\partial^2 \mathbf{c}}{\partial t^2} \tag{6}$$

This equation can be rewritten to emphasize its linear nature. $\mathbf{J}$ is the constraint Jacobian, $\mathbf{K}$ is a known constant vector, and λ is the vector of unknown Lagrange multipliers:

$$-\mathbf{J}^T \mathbf{W} \mathbf{J} \lambda = \mathbf{K} \tag{7}$$

Many fast, stable methods exist for solving equations of this form[15].

4.2 Soft Constraints

Some constraints are probabilistic in nature. Noisy image measurements are a constraint of this sort, they influence the dynamic model but do not impose hard constraints on its behavior.

Soft constraints such as these can be expressed as a potential field acting on the dynamic system. The incorporation of a potential field function that models a probability density pushes the dynamic evolution of the model toward the most likely value, starting from the current model state.

Note that functions that take the model state as input, such as a the controller from Section ??, can be represented as a time-varying potential field. One relevant example is incorporation of a probability distribution over link position and velocity:

$$\mathbf{Q}_f = f(\mathbf{X}, \mathbf{q}, \dot{\mathbf{q}}) \tag{8}$$

5 The Observation Model

The low-level features extracted from video comprise the final element of our system. Our system tracks regions that are visually similar, and spatially coherent: blobs [13, 8]. We can represent these 2-D regions by their low-order statistics. Clusters of 2-D points have 2-D spatial means and covariance matrices, which we shall denote $\boldsymbol{\mu}$ and $\mathbf{K}$. The blob spatial statistics are described in terms of their second-order properties; for computational convenience we will interpret this as a Gaussian model:

$$\Pr(\mathbf{O}|\boldsymbol{\mu}_k, \mathbf{K}_k) = \frac{\exp(-\frac{1}{2}(\mathbf{O}-\boldsymbol{\mu}_k)^T \mathbf{K}_k^{-1}(\mathbf{O}-\boldsymbol{\mu}_k))}{(2\pi)^{\frac{m}{2}}|\mathbf{K}_k|^{\frac{1}{2}}} \quad (9)$$

The Gaussian interpretation is not terribly significant, because we also keep a pixel-by-pixel *support map* showing the actual occupancy [22].

These 2-D features are the input to the 3-D blob estimation equation used by Azarbayejani and Pentland [1]. This observation equation relates the 2-D distribution of pixel values to a tracked object's 3-D position and orientation.

These observations supply constraints on the underlying 3-D human model. Due to their statistical nature, observations are easily modeled as soft constraints. Observations are integrated into the dynamic evolution of the system by modeling them as descriptions of potential fields, as discussed in Section 4.2.

5.1 The Inverse Observation Model

In the open-loop system, the vision system uses a Maximum Likelihood (ML) framework to label individual pixels in the scene:

$$\Gamma_{ij} = \arg\max_k \left[\Pr(\mathbf{O}_{ij}|\boldsymbol{\mu}_k, \mathbf{K}_k)\right] \quad (10)$$

where Γ_{ij} is the labeling of pixel (i,j), and $(\boldsymbol{\mu}_k, \mathbf{K}_k)$ are the second-order statistics of model k.

To close the loop, we need to incorporate information from the 3-D model. Given the current state of the model $\mathbf{q}$, it is possible to compute the state of an individual link that matches a specific tracked feature (say the hand), and call it $\mathbf{v}$. Then, given a model of the camera, it is possible to calculate the perspective projection of that state into 2-D and call it $\mathbf{v}^*$.

Since the vision system uses a stochastic framework, it is necessary to represent this link projection as a statistical model: $\Pr(\mathbf{O}_{ij}|\mathbf{v}_k^*)$. Integrating this information into the 2-D statistical decision framework results in a Maximum *A Posteriori* decision rule:

$$\Gamma_{ij} = \arg\max_k \left[\Pr(\mathbf{O}_{ij}|\boldsymbol{\mu}_k, \mathbf{K}_k) \cdot \Pr(\mathbf{O}_{ij}|\mathbf{v}_k^*)\right] \quad (11)$$

6 Multiple Behavior Models

Human behavior, in all but the simplest tasks, is not as simple as a single dynamic model. The next most complex model of human behavior is to have *several* alternative models of the person's dynamics, one for each class of response. Then at each instant we can make observations of the person's state, decide which model applies, and then use that model for estimation. This is known as the *multiple model* or *generalized likelihood* approach, and produces a generalized maximum likelihood estimate of the current and future values of the state variables [20]. Moreover, the cost of the Kalman filter calculations is sufficiently small to make the approach quite practical.

Intuitively, this solution breaks the person's overall behavior down into several "prototypical" behaviors. For instance, we might have dynamic models corresponding to a relaxed state, a very "tight" state, and so forth. We then classify the behavior by determining which model best fits the observations.

Mathematically, this is accomplished by evaluating a dynamic model in the form of a Kalman filter:

$$\hat{\mathbf{X}}_k = \mathbf{X}_k + \mathbf{K}_k(\mathbf{Y}_k - \mathbf{h}(\mathbf{X}_k^*, t)) \quad (12)$$

The *measurement innovations process* for the model (and associated Kalman filter) is then

$$\Gamma_k = \mathbf{Y}_k - \mathbf{h}(\mathbf{X}_k^*, t) \quad (13)$$

The measurement innovations process is zero-mean with covariance $\mathcal{R}$.

The measurement innovations process is, intuitively, the part of the observation data that is unexplained by the dynamic model. The behavior model that explains the largest portion of the observations is, of course, the model most likely to be correct. Thus, at each time step, we calculate the probability $Pr^{(i)}$ of the m-dimensional observations $\mathbf{Y}_k$ given the i^{th} model using Equation 2 and choose the model with the largest probability. This model is then used to estimate the current value of the state variables, to predict their future values, and to choose among alternative responses.

Note that when optimizing predictions of measurements Δt in the future, Equation 13 must be modified slightly to test the predictive accuracy of state estimates from Δt in the past.

7 Results

The dynamic skeleton model currently includes the upper body and arms. Figure 7 shows the real-time response to various target postures. The model interpolates those portions of the body state that are not measured directly, such as the upper body and elbow orientation, by use of the model's intrinsic dynamics and the behavior (control) model. The model also rejects noise that is inconsistent with the dynamic model. Table 2 compares RMS noise in the dynamic model output with noise in the underlying feature tracker. The "line following" test measures error from the best-fit line to data produced by constraining

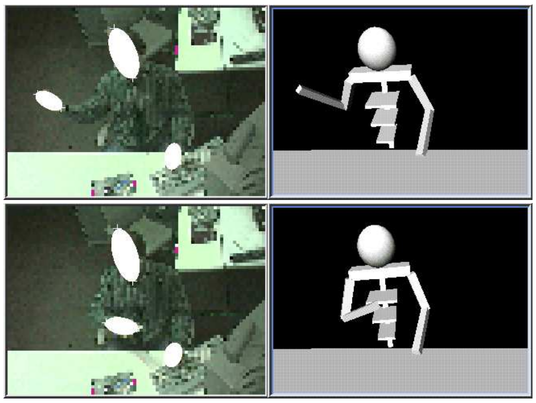

Figure 7: Frames on the left show video and 2-D blobs from one camera in the stereo pair. Frames on the right show corresponding configurations of the dynamic model at that instant in time.

	tracker	dynamic model
line following	1.4 cm	0.9 cm
rotational jitter	2.2 deg	0.6 deg

Table 2: Comparison of RMS tracking error for tracking with and without feedback.

the users hand to move along a linear trajectory. The "rotational jitter" measures error to a smoothed version of data obtained by smooth motions of the user's hand through a rotation.

It can be seen that Figure 8 illustrates another advantage of feedback from higher-level models to the low-level vision system. Without feedback, the 2-D tracker fails if there is even partial self-occlusion from a single camera's perspective. With feedback, information from the dynamic model can be used to resolve ambiguity during 2-D tracking.

Figure 9: Users sharing the workspace. Physical constraints stabilize the 2-D tracker with respect to competing targets.

The model predictions also stabilize tracking by providing constraints that help the tracking algorithm reject distractions in the environment. The addition of another person to the scene, as in Figure 9, produces many patches in the image that are similar to the target blobs. Without high-level model knowledge, the 2-D tracker can only reject these distractions based on some assumptions about the temporal stability of blobs. With the addition of high-level feedback, however, the 2-D tracker now has information about the physical constraints of the underlying system. Consequently, it is generally not distracted by competing targets (such as other people).

Figure 10 illustrates a simple occlusion example, where a hand can occlude the face from the viewpoint of one or both cameras. Without feedback to guide the low-level vision systems, tracking fails. The top-right graph in Figure 10 shows this case. The ambiguity caused by the occlusion caused one camera to mislabel the head and tight hand. This correspondence error causes stereo estimation to fail, resulting in an erroneous head position (second cluster of blue data points) and an erroneous path for the right hand (red data points).

The bottom two plots in Figure 10 show correct tracking when feedback is enabled, either with the observer-based system, or with the Kalman filters.

Figure 11 illustrates a more complex occlusion example, where the hands are moving in circular paths, repeatedly occluding each other simultaneously in both cameras. Without feedback, again, tracking fails. Even though the head is never occluded, the system fails badly enough to require several re-initializations. Ambiguities encountered in this process lead to the head being mislabeled in several instances. The scattering of blue (head position) data points in the top-right graph in Figure 11 is a result of these errors.

The bottom-left graph in Figure 11 shows the observer-based feedback system not faring much better. Compared to the previous case, there are very few head tracking errors. the feedback system has, at least, eliminated the need for constant, bottom-up re-initializations. The observer just can't keep up with the frequent occlusions. It seems to need more time to stabilize its estimates between occlusions.

The bottom-right plot in Figure 11 show correct tracking with the Kalman filter based feedback system. The track paths aren't entirely smooth. They should be smooth given that the actual physical process behind the data was smooth. These aberrations are probably caused by the significant increase in measurement noise associated with occlusions.

8 Conclusion

We have presented a framework for human motion understanding, defined as estimation of the physical state of the body combined with interpretation of that part of

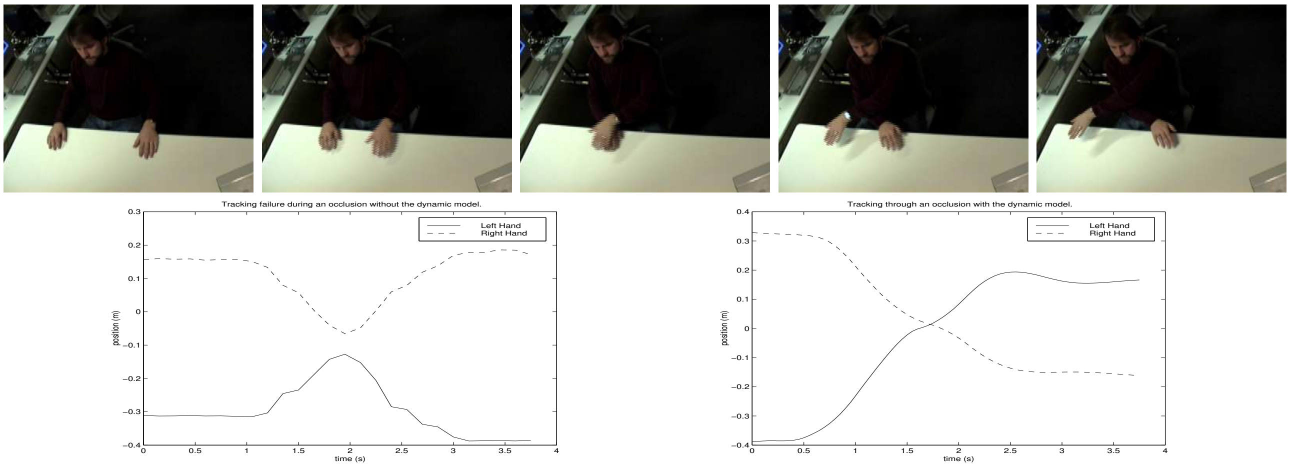

Figure 8: **Top:** frames showing self-occlusion during crossing. **Left:** tracking results without feedback. **Right:** correct tracking when feedback is enabled.

the motion that cannot be predicted by passive physics alone. The behavior system operates in conjunction with a real-time, fully-dynamic, 3-D person tracking system that provides a mathematically concise formulation for incorporating a wide variety of physical constraints and probabilistic influences. The framework takes the form of a non-linear recursive filter that enables lower-level, probabilistic processes to take advantage of the contextual knowledge encoded in the higher-level models.

9 References

[1] Ali Azarbayejani and Alex Pentland. Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features. In *Proceedings of 13th ICPR*, Vienna, Austria, August 1996. IEEE Computer Society Press.

[2] A. Baumberg and D. Hogg. An efficient method for contour tracking using active shape models. In *Proceeding of the Workshop on Motion of Nonrigid and Articulated Objects*. IEEE Computer Society, 1994.

[3] Christoph Bregler. Learning and recognizing human dynamics in video sequences. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, June 1997.

[4] Ernst D. Dickmanns and Birger D. Mysliwetz. Recursive 3-d road and relative ego-state recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(2):199–213, February 1992.

[5] D. M. Gavrila and L. S. Davis. Towards 3-d model-based tracking and recognition of human movement: a multi-view approach. In *International Workshop on Automatic Face- and Gesture-Recognition*. IEEE Computer Society, 1995. Zurich.

[6] Michael Isard and Andrew Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. European Conference on Computer Vision*, pages 343–356, Cambridge, UK, 1996.

[7] I. Kakadiaris, D. Metaxas, and R. Bajcsy. Active part-decomposition, shape and motion estimation of articulated objects: A physics-based approach. In *CVPR94*, pages 980–984, 1994.

[8] R. J. Kauth, A. P. Pentland, and G. S. Thomas. Blob: An unsupervised clustering approach to spatial preprocessing of mss imagery. In *11th Int'l Symposium on Remote Sensing of the Environment*, Ann Arbor, MI, April 1977.

[9] Dimitris Metaxas and Dimitris Terzopoulos. Shape and non-rigid motion estimation through physics-based synthesis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(6):580–591, 1993.

[10] K. Oatley, G. D. Sullivan, and D. Hogg. Drawing visual conclusions from analogy: preprocessing, cues and schemata in the perception of three dimensional objects. *Journal of Intelligent Systems*, 1(2):97–133, 1988.

[11] J. O'Rourke and N.I. Badler. Model-based image analysis of human motion using constraint propagation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2(6):522–536, November 1980.

[12] A. Pentland and B. Horowitz. Recovery of nonrigid motion and structure. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(7):730–742, July 1991.

[13] Alex Pentland. Classification by clustering. In *Proceedings of the Symposium on Machine Processing of Remotely Sensed Data*. IEEE, IEEE Computer Society Press, June 1976.

[14] Alex Pentland and Andrew Liu. Modeling and predicition of human behavior. In *IEEE Intelligent Vehicles 95*, September 1995.

[15] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C: the art of scientific computing.* Cambridge University Press, Cambridge, U.K., second edition, 1992.

[16] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of IEEE*, 77(2):257–285, 1989.

[17] J.M. Rehg and T. Kanade. Visual tracking of high dof articulated structures: An application to human hand tracking. In *European Conference on Computer Vision*, pages B:35–46, 1994.

[18] K. Rohr. Cvgipiu. *"Towards Model-Based Recognition of Human Movements in Image Sequences*, 1(59):94–115, 1994.

[19] Thad Starner and Alex Pentland. Real-time american sign language recognition from video using hidden markov models. In *Proceedings of International Symposium on Computer Vision*, Coral Gables, FL, USA, 1995. IEEE Computer Society Press.

[20] A. S. Willsky. Detection of abrupt changes in dynamic systems. In M. Basseville and A. Benveniste, editors, *Detection of Abrupt Changes in Signals and Dynamical Systems*, number 77 in Lecture Notes in Control and Information Sciences, pages 27–49. Springer-Verlag, 1986.

[21] Andrew Witkin, Michael Gleicher, and William Welch. Interactive dynamics. In *ACM SIGGraph, Computer Graphics*, volume 24:2, pages 11–21. ACM SIGgraph, March 1990.

[22] Christopher Wren, Ali Azarbayejani, Trevor Darrell, and Alex Pentland. Pfinder: Real-time tracking of the human body. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(7):780–785, July 1997.

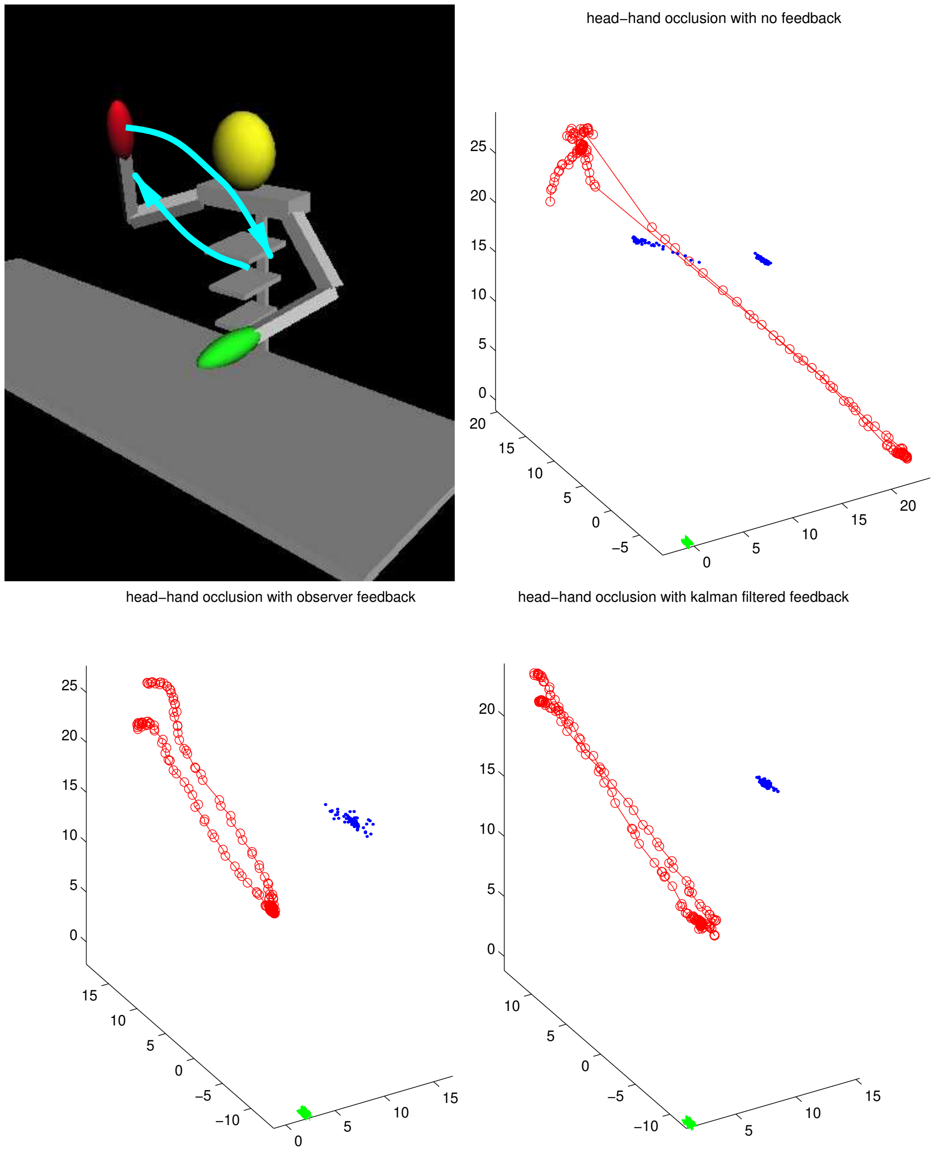

Figure 10: The left hand is stationary on the table. The right hand occludes the face. (top-left) Illustration of real motion (top-right) the system without feedback fails. (bottom-left) using an observer to provide feedback prevents the failure (bottom-right) Kalman filtered feedback also prevents the failure.

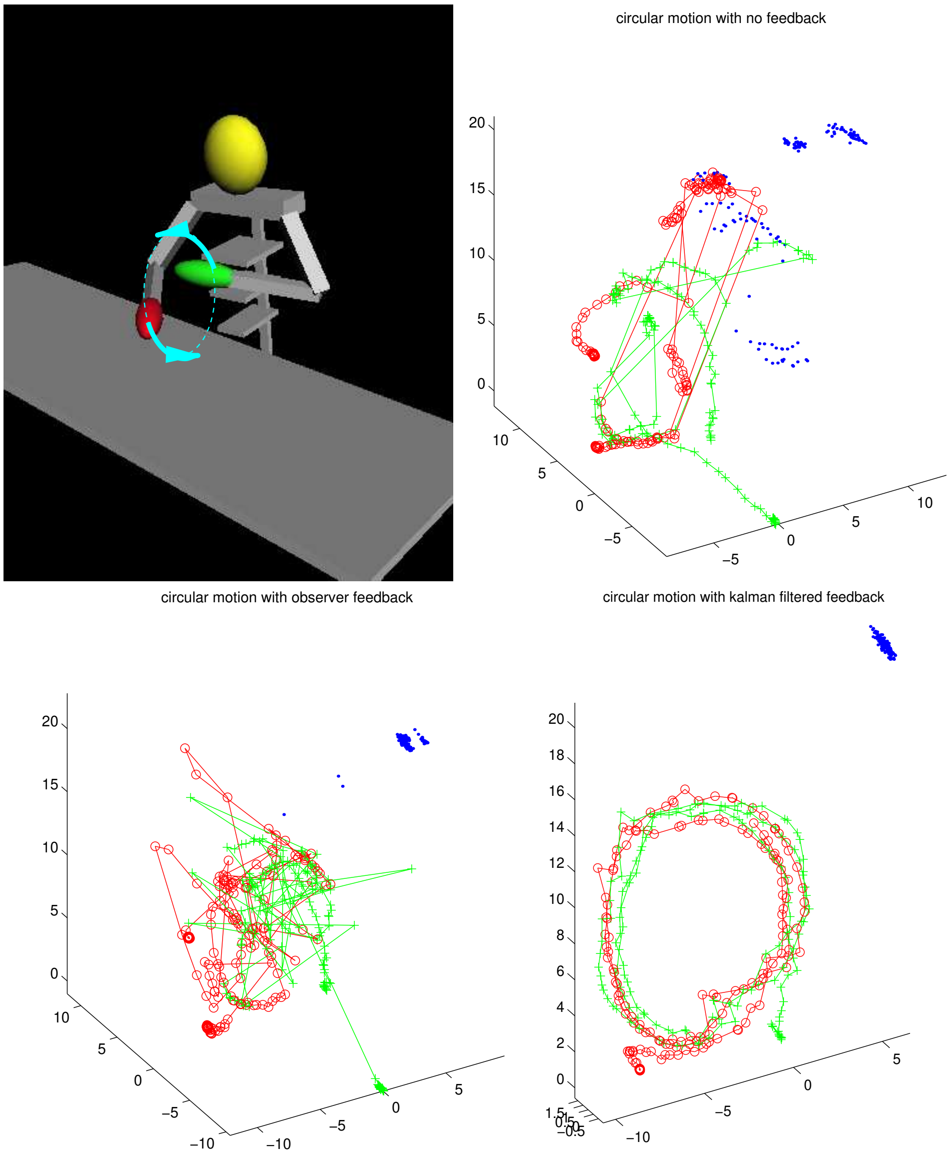

Figure 11: The left and right hands moving in a circular motion, repeatedly occluding each other. (top-left) Illustration of real motion (top-right) the system without feedback fails. (bottom-left) using an observer to provide feedback helps, but hand tracking still often fails. (bottom-right) Kalman filtered feedback prevents the failures.