Modeling and Prediction of Human Behavior

Alex Pentland
Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.

Andrew Liu
Nissan Cambridge Basic Research, Cambridge, MA 02142, U.S.A.

We propose that many human behaviors can be accurately described as a set of dynamic models (e.g., Kalman filters) sequenced together by a Markov chain. We then use these dynamic Markov models to recognize human behaviors from sensory data and to predict human behaviors over a few seconds time. To test the power of this modeling approach, we report an experiment in which we were able to achieve 95% accuracy at predicting automobile drivers' subsequent actions from their initial preparatory movements.

1 Introduction

Our approach to modeling human behavior is to consider the human as a device with a large number of internal mental states, each with its own particular control behavior and interstate transition probabilities. Perhaps the canonical example of this type of model would be a bank of standard linear controllers (e.g., Kalman filters plus a simple control law), each using different dynamics and measurements, sequenced together with a Markov network of probabilistic transitions. The states of the model can be hierarchically organized to describe both short-term and longer-term behaviors; for instance, in the case of driving an automobile, the longer-term behaviors might be passing, following, and turning, while shorter-term behaviors would be maintaining lane position and releasing the brake.

Such a model of human behavior could be used to produce improved human-machine systems. If the machine could recognize the human's behavior or, even better, if it could anticipate the human's behavior, it could adjust itself to serve the human's needs better. To accomplish this, the machine would need to be able to determine which of the human's control states was currently active and to predict transitions between control states. It could then configure itself to achieve its best overall performance.

Because the internal states of the human are not directly observable, this scenario requires that the human's internal state be determined through an indirect estimation process. To accomplish this, we have adapted the expectation-maximization methods developed for use with hidden Markov

Neural Computation **11, 229–242** (1999)

models (HMM). By using these methods to identify a user's current pattern of control and predict the most likely pattern of subsequent control states, we have been able to recognize human driving behaviors accurately and anticipate the human's behavior for several seconds into the future.

Our research builds on the observation that although human behaviors such as speech (Rabinee & Juang, 1986), handwriting (Starner, Makhoul, Schwartz, & Chou, 1994), hand gestures (Yang, Xu, & Chen, 1997; Pentland, 1996), and even American Sign Language (Pentland, 1996; Starner & Pentland, 1995) can be accurately recognized by use of HMMs, they do not produce a model of the observations that is accurate enough for simulation or prediction. In these cases, the human behavior displays additional properties, such as smoothness and continuity, that are not captured within the HMM statistical framework.

We believe that these missing additional constraints are typically due to the physical properties of human movement and consequently best described by dynamic models such as the well-known Kalman filter (Kalman & Bucy, 1961). Our proposal is to describe the small-scale structure of human behavior by a set of dynamic models (thus incorporating constraints such as smoothness and continuity) and the large-scale structure by coupling together these control states into a Markov chain. It has been proposed that the basic element of cortical processing can be modeled as a Kalman filter (e.g., Pentland, 1992; Rao & Ballard, 1997); in this article, we are proposing that these basic elements are chained together to form larger behaviors.

The resulting framework, first proposed by Pentland and Liu (1995), is related to research in robot control (Meila & Jordan 1995), and machine vision (Isard & Blake, 1996; Bregler, 1997), in which elements from dynamic modeling or control theory are combined with stochastic transitions. These efforts have shown utility in tracking human motion and recognizing atomic actions such as grasping or running. Our approach goes beyond this to describe and classify more extended and elaborate behaviors, such as passing a vehicle while driving, which consist of several atomic actions chained together in a particular sequence. Our framework has consequently allowed us to predict sequences of human behaviors from initial, preparatory motions.

2 Simple Dynamic Models

Among the simplest nontrivial models that have been considered for modeling human behavior are single dynamic processes,

$$\dot{\mathbf{X}}_k = \mathbf{f}(\mathbf{X}_k, t) + \xi(t), \tag{2.1}$$

where the function $\mathbf{f}$ models the dynamic evolution of state vector $\mathbf{X}_k$ at time k. Let us define an observation process,

$$\mathbf{Y}_k = \mathbf{h}(\mathbf{X}_k, t) + \eta(t), \tag{2.2}$$

where the sensor observations $\mathbf{Y}$ are a function $\mathbf{h}$ of the state vector and time. Both ξ and η are white noise processes having known spectral density matrices.

Using Kalman's result, we can then obtain the optimal linear estimate $\hat{\mathbf{X}}_k$ of the state vector $\mathbf{X}_k$ by use of the following Kalman filter,

$$\hat{\mathbf{X}}_k = \mathbf{X}_k^* + \mathbf{K}_k(\mathbf{Y}_k - \mathbf{h}(\mathbf{X}_k^*, t)), \tag{2.3}$$

provided that the Kalman gain matrix $\mathbf{K}_k$ is chosen correctly (Kalman & Bucy, 1961). At each time step k, the filter algorithm uses a state prediction $\mathbf{X}_k^*$, an error covariance matrix prediction $\mathbf{P}_k^*$, and a sensor measurement $\mathbf{Y}_k$ to determine an optimal linear state estimate $\hat{\mathbf{X}}_k$, error covariance matrix estimate $\hat{\mathbf{P}}_k$, and predictions $\mathbf{X}_{k+1}^*$, $\mathbf{P}_{k+1}^*$ for the next time step.

The prediction of the state vector $\mathbf{X}_{k+1}^*$ at the next time step is obtained by combining the optimal state estimate $\hat{\mathbf{X}}_k$ and equation 2.1:

$$\mathbf{X}_{k+1}^* = \hat{\mathbf{X}}_k + \mathbf{f}(\hat{\mathbf{X}}_k, t)\Delta t. \tag{2.4}$$

In some applications this prediction equation is also used with larger time steps, to predict the human's future state. For instance, in a car, such a prediction capability can allow us to maintain synchrony with the driver by giving us the lead time needed to alter suspension components. In our experience, this type of prediction is useful only for short time periods, for instance, in the case of quick hand motions for up to one-tenth of a second (Friedmann, Starner, & Pentland, 1992a).

Classically $\mathbf{f}$, $\mathbf{h}$ are linear functions and ξ, η assumed gaussian. It is common practice to extend this formulation to "well-behaved" nonlinear problems by locally approximating the nonlinear system by linear functions using a local Taylor expansion; this is known as an extended Kalman filter. However, for strongly nonlinear problems such as are addressed in this article, one must either employ nonlinear functions and/or multimodal noises, or adopt the multiple-model and sequence-of-models approach described in the following sections.

3 Multiple Dynamic Models

Human behavior is normally 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 (Willsky, 1986). Then at each instant we can make observations of the person's state, decide which model applies, and make our response based on that model. This multiple model approach produces a generalized maximum likelihood estimate of the current and future values of the state variables. Moreover, the cost of the Kalman filter calculations is sufficiently small to make the approach quite practical, even for real-time applications.

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

Mathematically, this is accomplished by setting up a set of states S, each associated with a Kalman filter and a particular dynamic model,

$$\hat{\mathbf{X}}_k^{(i)} = \mathbf{X}_k^{*(i)} + \mathbf{K}_k^{(i)}(\mathbf{Y}_k - \mathbf{h}^{(i)}(\mathbf{X}_k^{*(i)}, t)), \tag{3.1}$$

where the superscript (i) denotes the ith Kalman filter. The measurement innovations process for the ith model (and associated Kalman filter) is then

$$\Gamma_k^{(i)} = \mathbf{Y}_k - \mathbf{h}^{(i)}(\mathbf{X}_k^{*(i)}, t). \tag{3.2}$$

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

The ith measurement innovations process is, intuitively, the part of the observation data that is unexplained by the ith model. The 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 ith model's dynamics,

$$Pr^{(i)}(\mathbf{Y}_k|\mathbf{X}_k^*) = \frac{\exp\left(-\frac{1}{2}\Gamma_k^{(i)T}\mathcal{R}^{-1}\Gamma_k^{(i)}\right)}{(2\pi)^{m/2}\mathrm{Det}(\mathcal{R})^{1/2}}, \tag{3.3}$$

and choose the model with the largest probability. This model is then used to estimate the current value of the state variables, predict their future values, and choose among alternative responses. After the first time step, where $\mathcal{R}$ and $\mathbf{P}_k$ are assumed known a priori, they may be estimated from the incoming data (Kalman & Bucy, 1961).

Note that when optimizing predictions of measurements Δt in the future, equation 3.2 must be modified slightly to test the predictive accuracy of state estimates from Δt in the past:

$$\Gamma_k^{(i)} = \mathbf{Y}_k - \mathbf{h}^{(i)}(\mathbf{X}_{k-\Delta t}^{*(i)} + \mathbf{f}^{(i)}(\hat{\mathbf{X}}_{k-\Delta t}^{(i)}, \Delta t)\Delta t, t)). \tag{3.4}$$

We have used this method accurately to remove lag in a high-speed telemanipulation task by continuously reestimating the user's arm dynamics (e.g., tense and stiff, versus relaxed and inertia dominated) (Friedmann, Starner, & Pentland, 1992b). We found that using this multiple-model approach, we were able to obtain significantly better predictions of the user's hand position than was possible using a single dynamic or static model.

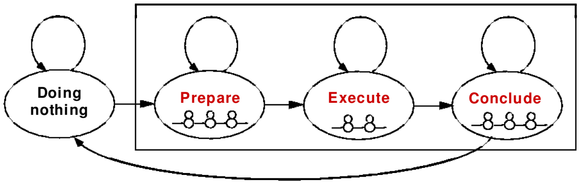

Figure 1: A Markov dynamic model of driver action. Only the substates in the Prepare state will be used for action recognition.

4 Markov Dynamic Models

In the multiple dynamic model, all the processes have a fixed likelihood at each time step. However, this is uncharacteristic of most situations, where there is often a fixed sequence of internal states, each with its own dynamics. Consider driving through a curve. The driver may be modeled as having transitioned through a series of states $\lambda = (s_1, s_2, \ldots, s_k)$, $s_i \epsilon S$, for instance, entering a curve, in the curve, and exiting a curve. Transitions between these states happened only in the order indicated.

Thus, in considering state transitions among a set of dynamic models, we should make use of our current estimate of the driver's internal state. We can accomplish this fairly generally by considering the Markov probability structure of the transitions between the different states. The input to decide the person's current internal state (e.g., which dynamic model currently applies) will be the measurement innovations process as above, but instead of using this directly in equation 3.3, we will also consider the Markov interstate transition probabilities.

We will call this type of multiple dynamic model a Markov dynamic model (MDM). Conceptually, MDMs are exactly like HMMs except that the observations are the innovations (roughly, prediction errors) of a Kalman filter or other dynamic, predictive process. In the case of the dynamic processes used here, these innovations correspond to accelerations that were not anticipated by the model. Thus, our MDMs describe how a set of dynamic processes must be controlled in order to generate the observed signal rather than attempting to describe the signal directly.

The initial topology for an MDM can be determined by estimating how many different states are involved in the observed phenomenon. Fine-tuning this topology can be performed empirically. Figure 1, for instance, shows a four-state MDM to describe long-time-scale driver behavior. Each state has substates, again described by an MDM, to describe the fine-grain structure of the various behaviors.

As with HMMs, there are three key problems in MDM use (Huang, Ariki, & Jack, 1990): the evaluation, estimation, and decoding problems. The evaluation problem is that given an observation sequence and a model, what is the probability that the observed sequence was generated by the model ($Pr(\mathbf{Y}|\lambda)$)? If this can be evaluated for all competing models for an observation sequence, then the model with the highest probability can be chosen for recognition.

As with HMMs, the Viterbi algorithm provides a quick means of evaluating a set of MDMs as well as providing a solution for the decoding problem (Huang et al., 1990; Rabiner & Juang, 1986). In decoding, the goal is to recover the state sequence given an observation sequence. The Viterbi algorithm can be viewed as a special form of the forward-backward algorithm where only the maximum path at each time step is taken instead of all paths. This optimization reduces computational load and allows the recovery of the most likely state sequence.

Since Viterbi guarantees only the maximum of $Pr(\mathbf{Y}, S|\lambda)$ over all state sequences S (as a result of the first-order Markov assumption) instead of the sum over all possible state sequences, the resultant scores are only an approximation. However, Rabiner and Juang (1986) show that this is typically sufficient.

Because the innovations processes that drive the MDM interstate transitions are continuous, we must employ the actual probability densities for the innovations processes. Fortunately, Baum-Welch parameter estimation, the Viterbi algorithm, and the forward-backward algorithms can be modified to handle a variety of characteristic densities (Huang et al., 1990; Juang, 1985). However, in this article, the densities will be assumed to be as in equation 3.3.

5 An Experiment Using Markov Dynamic Models

Driving is an important, natural-feeling, and familiar type of human behavior that exhibits complex patterns that last for several seconds. From an experimental point of view, it is important that the number of distinct driving behaviors is limited by the heavily engineered nature of the road system, and it is easy to instrument a car to record human hand and foot motions. These characteristics make driving a nearly ideal experimental testbed for modeling human behaviors. We have therefore applied MDMs to try to identify automobile drivers' current internal (intentional) state and to predict the most likely subsequent sequence of internal states. In the case of driving, the macroscopic actions are events like turning left, stopping, or changing lanes. The internal states are the individual steps that make up the action, and the observed variables will be changes in heading and acceleration of the car.

The intuition is that even apparently simple driving actions can be broken down into a long chain of simpler subactions. A lane change, for instance, may consist of the following steps: (1) a preparatory centering the car in the current lane, (2) looking around to make sure the adjacent lane is clear, (3) steering to initiate the lane change, (4) the change itself, (5) steering to terminate the lane change, and (6) a final recentering of the car in the new lane.

In this article we are statistically characterizing the sequence of steps within each action and then using the first few preparatory steps to identify which action is being initiated. To continue the example, the substates of "prepare" shown in Figure 1 might correspond to centering the car, checking the adjacent lane, and steering to initiate the change.

To recognize which action is occurring, one compares the observed pattern of driver behavior to Markov dynamic models of each action, in order to determine which action is most likely given the observed pattern of steering and acceleration and braking. This matching can be done in real time on current microprocessors, thus potentially allowing us to recognize a driver's intended action from his or her preparatory movements.

If the pattern of steering and acceleration is monitored internally by the automobile, then the ability to recognize which action the driver is beginning to initiate can allow intelligent cooperation by the vehicle. If heading and acceleration are monitored externally by video cameras (as in Boer, Fernandez, Pentland, & Liu, 1996), then we can more intelligently control the traffic flow.

5.1 Experimental Design. The goal is to test the ability of our framework to characterize the driver's steering and acceleration and braking patterns in order to classify the driver's intended action. The experiment was conducted within the Nissan Cambridge Basic Research driving simulator, shown in Figure 2a. The simulator consists of the front half of a Nissan 240SX convertible and a 60 degree (horizontal) by 40 degree (vertical) image projected onto the wall facing the driver. The 240SX is instrumented to record driver control input such as steering wheel angle, brake position, and accelerator position.

Subjects were instructed to use this simulator to drive through an extensive computer graphics world, illustrated in Figure 2b. This world contains a large number of buildings, many roads with standard markings, and other moving cars. Each subject drove through the simulated world for approximately 20 minutes; during that time, the driver's control of steering angle and steering velocity, car velocity, and car acceleration were recorded at 1/10 second intervals. Drivers were instructed to maintain a normal driving speed of 30 to 35 miles per hour (13–15 meters per second).

From time to time during this drive, text commands were presented onscreen for 1 second, whereupon the subjects had to assess the surrounding situation, formulate a plan to carry out the command, and then act to execute the command. The commands were: (1) stop at the next intersection, (2) turn left at the next intersection, (3) turn right at the next intersection, (4) change

(a) (b)

Figure 2: (a) Nissan Cambridge Basic Research simulator. (b) Part of the simulated world seen by the subjects.

lanes, (5) pass the car in front of you, and (6) do nothing (e.g., drive normally, with no turns or lane changes). A total of 72 stop, 262 turn, 47 lane change, 24 passing, and 208 drive-normal episodes were recorded from eight adult male subjects. The time needed to complete each command varied from approximately 5 to 10 or more seconds, depending on the complexity of both the action and the surrounding situation.

Command presentation was timed relative to the surroundings in order to allow the driver to execute the command in a normal manner. For turns, commands were presented 40 meters before an intersection ($\approx$ 3 seconds, a headway used in some commercially available navigation aids), for passing the headway was 30 meters ($\approx$ 2 seconds, which is the mean headway observed on real highways), and for stopping the headway was 70 meters ($\approx$ 5 seconds). The variables of command location, road type, surrounding buildings, and traffic conditions were varied randomly thoughout the experiment. The dynamic models used were specific to a Nissan 240SX.

Using the steering and acceleration data recorded while subjects carried out these commands, we built three-state models of each type of driver action (stopping, turn left, turn right, lane change, car passing, and do nothing) using expectation-maximization (EM) for the parameters of both the Markov chain and the state variables (heading, acceleration) of the dynamic models (Baum, 1972; Juang, 1985; Rabiner & Juang, 1986). Three states were used because preliminary investigation on informally collected data showed that the three state models performed slightly better than four or five state models. The form of the dynamic models employed is described in the appendix.

To assess the classification accuracy of these models, we combined them with the Viterbi recognition algorithm and examined the stream of drivers' steering and acceleration innovations in order to detect and classify each driver's actions. All of the data were labeled, with the "do nothing" label serving as a "garbage class" for any movement pattern other than the five actions of interest. We then examined the computer's classifications of the data immediately following each command and recorded whether the computer had correctly labeled the action. To obtain unbiased estimates of recognition performance, we employed the "leaving one out" method, and so can report both the mean and variance of the recognition rate.

Recognition results were tabulated at 2 seconds after the beginning of the presentation of a command to the subject, thus allowing the driver up to 2 seconds to read the command and begin responding. As will be seen in the next section, the 2 second time point is before there is any large, easily recognizable change in car position, heading, or velocity. Because the driving situation allows visual preview of possible locations for turning, passing, and so forth, we may presume that the driver was primed to react swiftly. As the minimum response time to a command is approximately 0.5 second, the 2-second point is at most 1.5 seconds after the beginning of the driver's action, which is on average 20% of the way through the action.

5.2 Results. Because the MDM framework is fairly complex, we must first try simpler methods to classify the data. For comparison, therefore, we used Bayesian classification, where the data were modeled using multivariate gaussian, nearest-neighbor, or three-state HMMs. The modeled data were the measured accelerator, brake, and steering wheel positions over the previous 1 second.

5.2.1 Results Using Classical Methods. At 2 seconds after the onset of the command text (approximately 1.5 seconds after the beginning of action, or roughly 20% of the way through the action) mean recognition accuracy (as evaluated using the leaving-one-out method) was not statistically different from chance performance for any of these three methods. Many different parameter settings and similar variations on these techniques were also tried, with no success.

Examination of the data makes the reasons for these failures fairly clear. First, each action can occur over a substantial range of time scales, so a successful recognition method must incorporate some form of time warping. Second, even for similar actions, the pattern of brake taps, steering corrections, and accelerator motions varies almost randomly, as the exact pattern depends on microevents in the environment, variations in the driver's attention, and so forth.

It is only when we integrate these inputs via the Kalman filter's physical model, to obtain the car's motion state variables (e.g., velocity, acceleration, heading), that we see similar patterns for similar actions. This is similar to many human behaviors, where the exact sequence of joint angles and muscle activations is unimportant; it is the trajectory of the end effectors or center of mass that matters.

These data characteristics cause methods such as multivariate gaussian and nearest-neighbor to fail because of time scale variations; HMM and time

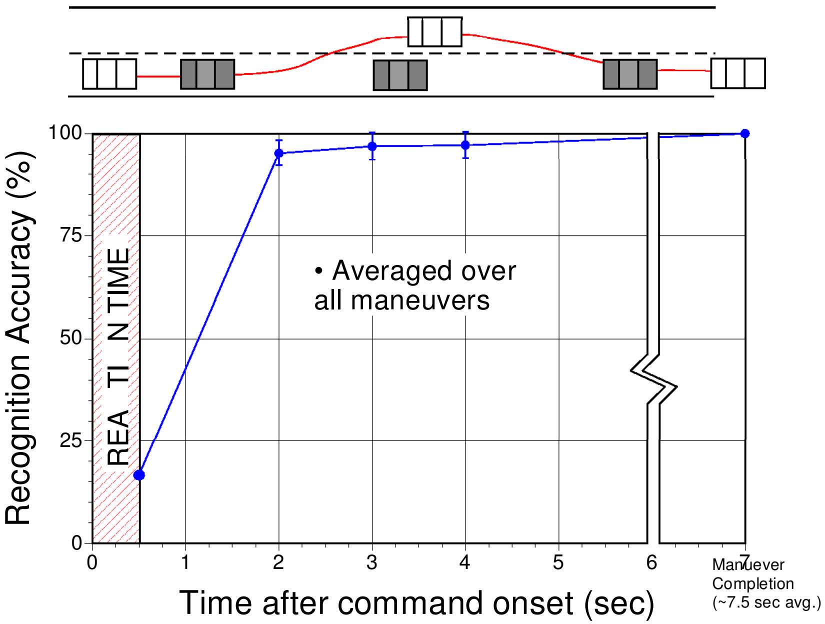

Figure 3: Recognition accuracy versus time. Very good recognition accuracy is obtained well before the main or functional part of the behavior.

warping methods fail because they operate on the control inputs rather than the intrinsic motion variables. By inserting an integrating dynamic model between the control inputs and the HMM, we bring out the underlying control intended by the human.

5.2.2 Results Using MDMs. At 2 seconds after the onset of the command text (approximately 1.5 seconds after the beginning of action, or roughly 20% of the way through the action), mean recognition accuracy was 95.24% ± 3.1%. These results, plus accuracies at longer lag times, are illustrated in Figure 3. As can be seen, the system is able to classify these behaviors accurately very early in the sequence of driver motions.

It could be argued that the drivers are already beginning the main or functional portion of the various behaviors at the 2 second point, and that it is these large motions that are being recognized. However, the failure of standard statistical techniques to classify the behaviors demonstrates that there are no changes in mean position, heading, or velocity at the 2 second point that are statistically different from normal driving. Thus, the behaviors are being recognized from observation of the driver's preparatory movements.

To illustrate this point, we observe that 2 seconds after the onset of a lane change command, the MDM system was able to recognize the driver's action with 93.3% accuracy ($N = 47$), even though the vehicle's lateral offset averaged only 0.8 ± 1.38 meters (lateral offset while going straight has a standard deviation of $\sigma = 0.51$ meters). Since this lateral displacement is subtantially before the vehicle exits the lane (lane width is 4 meters), it is clear that the MDM system is detecting the maneuver well before the main or functional part of the behavior. For comparison, to achieve a 93% accuracy by thresholding lateral position, one would need to tolerate a 99% false alarm rate while going straight.

To test whether our sample is sufficiently large to encompass the range of between-driver variation adequately, we compared these results to the case in which we train on all subjects and then test on the training data. In the test-on-training case the recognition accuracy was 98.8%, indicating that we have a sufficiently large sample of driving behavior in this experiment.

5.2.3 Discussion. We believe that these results support the view that human actions are best described as a sequence of control steps rather than as a sequence of raw positions and velocities. In the case of driving, this means that it is the pattern of acceleration and heading that defines the action. There are lots of ways to manipulate the car's controls to achieve a particular acceleration or heading, and consequently no simple pattern of hand-foot movement that defines a driving action.

Although these results are promising, caution must be taken in transferring them to other human actions or even to real-world driving. It is possible, for instance, that there are driving styles not seen in any of our subjects. Similarly, the driving conditions found in our simulator do not span the entire range of real driving situations. We believe, however, that our simulator is sufficiently realistic that comparable accuracies can be obtained in real driving. Moreover, there is no strong need for models that suit all drivers; most cars are driven by a relatively small number of drivers, and this fact can be used to increase classification accuracy. We are exploring these questions, and the initial results support our optimism.

6 Conclusion

We have demonstrated that we can accurately categorize human driving actions very soon after the beginning of the action using our behavior modeling methodology. Because of the generic nature of the driving task, there is reason to believe that this approach to modeling human behavior will generalize to other dynamic human-machine systems. This would allow us to recognize automatically people's intended action, and thus build control systems that dynamically adapt to suit the human's purpose better.

Appendix: Dynamic Car Models

The kinematic model is a three-wheeled model with two front wheels and one rear wheel. Given access to a system clock, we can figure out the amount of time elapsed since the last frame, Δt. To calculate the distance traveled since the last frame, we calculate

$$d_{lin} = v\Delta t,$$

where the function d_{lin} is the linear distance traveled since the last frame and v is the speed of the car. To calculate the new position of the car, we must consider whether the car is turning or moving straight ahead. If moving ahead, then

$$\vec{P}_{new} = \vec{P} + (d_{lin}\vec{H}).$$

If the car is making a turn, then

$$\vec{P}_{new} = \vec{P} + (r_{circle} \sin(\theta_d)\vec{H}),$$

where the angular travel of the car θ_d is given by

$$\theta_d = d_{lin}/r_{circle},$$

and the turning circle of the car, r_{circle}, is given by

$$r_{circle} = \frac{d_{WB}}{\tan \theta_{FW}},$$

where d_{WB} is the wheelbase of the car.

The new heading of the car is figured from the following equation,

$$\vec{H}_{new} = 0.5(\vec{V}_{LFW} + \vec{V}_{RFW}) - \vec{P}_{new},$$

where $\vec{V}_{LFW}$ is the vector position of the front left wheel and $\vec{V}_{RFW}$ is the vector position of the front right wheel. Finally, we normalize the new heading vector,

$$\vec{H}_{new} = \frac{\vec{H}_{new}}{\|\vec{H}_{new}\|}.$$

The dynamical equations are equally simple for the vehicle. The new velocity of the car is computed based on the energy produced by the engine balanced by the drag from air and rolling resistance and braking.

To calculate the new velocity, we first compute the kinetic energy based on the present velocity using

$$\boldsymbol{KE}_{old} = \frac{1}{2}Mv^2,$$

where M is the mass of the car. Then we calculate the work done by the engine to propel the car forward, which is simply given by

$$W_{engine} = aP_{maxeng}\Delta t,$$

where P_{maxeng} is the maximum power output of the engine and a is the position of the accelerator ($0 \rightarrow 1$). The limitation of this model is that the work is independent of the rpm. The work done by the brakes to slow the car is given by

$$W_{brake} = F_{maxbrake} d_{lin} (b + 0.6\, b_{park}),$$

where $F_{maxbrake}$ is the maximum braking force of the brakes, b is the postion of the brake pedal ($0 \rightarrow 1$), and b_{park} is the parking brake, which is either 0 or 1.

The contribution of potential energy, if the world is not flat, can be expressed as:

$$\Delta \boldsymbol{PE} = (P^y_{new} - P^y) M g,$$

where P^y is the current y-coordinate and P^y_{new} is the newly computed y-coordinate. The acceleration of gravity, g, is given in $\mathrm{m/s^2}$.

To calculate the effect of drag, we first find the total force resulting from the air drag and road friction,

$$F_{drag} = \mu_{air} v^2 + \mu_{road} M g,$$

where μ_{road} is the coefficient of friction of the road and μ_{air} is the drag coefficient of the car. The total deceleration energy is the sum of the drag energy and the brake energy,

$$E_{drag} = d_{lin} F_{drag} + W_{brake}.$$

Our final energy balance equation is:

$$\boldsymbol{KE}_{new} = \boldsymbol{KE}_{old} + W_{engine} - \Delta \boldsymbol{PE} - E_{drag}$$

$$v_{new} = \sqrt{\frac{2\boldsymbol{KE}_{new}}{M}}.$$

The skid state of the car depends on the speed and steering wheel angle and road surface coefficient of friction. In our experiments, we used constants that apply to a Nissan 240SX and a specified road-tire configuration. In this model, if the speed is below 11.2 mph, then the car always regains traction. Basically, if the following condition is satisfied,

$$\frac{\theta_{SW}}{e^{-0.0699 v}} < 24.51,$$

then the tires are within their adhesion limits and the car will not skid.

References

Baum, L. (1972). An inequality and associated maximization technique in statistical estimation of probabilistic functions of Markov processes. *Inequalities, 3*, 1–8.

Boer, E., Fernandez, M., Pentland, A., & Liu, A. (1996). Method for evaluating human and simulated drivers in real traffic situations. In *IEEE Vehicular Tech. Conf.* (pp. 1810–1814). Atlanta, GA.

Bregler, C. (1997). Learning and recognizing human dynamics in video sequences. In *IEEE Conf. on Computer Vision and Pattern Recognition* (pp. 568–574). San José, P.R.

Friedmann, M., Starner, S., & Pentland, A. (1992a). Synchronization in virtual realities. *Presence, 1*, 139–144.

Friedmann, M., Starner, T., & Pentland, A. (1992b). Device synchronization using an optimal linear filter. In *Proc. ACM 1992 Symposium on Interactive 3D Graphics* (pp. 128–134). Boston.

Huang, X., Ariki, Y., & Jack, M. (1990). *Hidden Markov models for speech recognition*. Edinburgh: Edinburgh University Press.

Isard, M., & Blake, A. (1996). Contour tracking by stochastic propagation of conditional density. In *1996 European Conf. on Computer Vision* (pp. 343–356). Cambridge, U.K.

Juang, B. (1985). Maximum likelihood estimation for mixture multivariate observations of Markov chains. *AT&T Technical Journal, 64*, 1235–1249.

Kalman, R., & Bucy, R. (1961). New results in linear filtering and prediction theory. *Transaction ASME, 83D*, 95–108.

Meila, M., & Jordan, M. (1995). *Learning fine motion by Markov mixtures of experts* (Tech. Memo No. 1567). Cambridge, MA: MIT, AI Laboratory.

Pentland, A. (1992). Dynamic vision. In G. A. Carpenter & S. Grossburg (Ed.), *Neural networks for vision and image processing* (pp. 133–159). Cambridge, MA: MIT Press.

Pentland, A. (1996). Smart rooms. *Scientific American, 274*, 68–76.

Pentland, A., & Liu, A. (1995). Toward augmented control systems. In *Proc. Intelligent Vehicles '95* (pp. 350–355). Detroit.

Rabiner, L., & Juang, B. (1986, January). An introduction to hidden Markov models. *IEEE ASSP Magazine*, pp. 4–16.

Starner, T., Makhoul, J., Schwartz, R., & Chou, G. (1994). On-line cursive handwriting recognition using speech recognition methods. In *ICASSP* (vol. 5, pp. 1234–1244). Adelaide, Australia.

Starner, T., & Pentland, A. (1995). Visual recognition of American Sign Language using hidden Markov models. In *Proc. Int'l Workshop on Automatic Face- and Gesture-Recognition* (pp. 38–44). Zurich, Switzerland.

Rao, P., & Ballard, D. (1997). Dynamic model of visual recognition predicts neural response properties in the visual cortex. *Neural Computation, 9*, 721–763.

Willsky, A. (1986). Detection of abrupt changes in dynamic systems. In M. Basseville & A. Benveniste (Eds.), *Detection of abrupt changes in signals and dynamical systems* (pp. 27–49). Berlin: Springer-Verlag.

Yang, J., Xu, Y., & Chen, C. S. (1997). Human action learning via hidden Markov model. *IEEE Trans. Systems, Man, and Cybernetics, 27*, 34–44.

Received December 5, 1997; accepted June 24, 1998.