Lane-Change Detection Using a Computational Driver Model

Dario D. Salvucci and **Hiren M. Mandalia,** Drexel University, Philadelphia, Pennsylvania, and **Nobuyuki Kuge** and **Tomohiro Yamamura,** Nissan Motor Company, Atsugi, Japan

Objective: This paper introduces a robust, real-time system for detecting driver lane changes. **Background:** As intelligent transportation systems evolve to assist drivers in their intended behaviors, the systems have demonstrated a need for methods of inferring driver intentions and detecting intended maneuvers. **Method:** Using a "model tracing" methodology, our system simulates a set of possible driver intentions and their resulting behaviors using a simplification of a previously validated computational model of driver behavior. The system compares the model's simulated behavior with a driver's actual observed behavior and thus continually infers the driver's unobservable intentions from her or his observable actions. **Results:** For data collected in a driving simulator, the system detects 82% of lane changes within 0.5 s of maneuver onset (assuming a 5% false alarm rate), 93% within 1 s, and 95% before the vehicle moves one fourth of the lane width laterally. For data collected from an instrumented vehicle, the system detects 61% within 0.5 s, 77% within 1 s, and 84% before the vehicle moves one-fourth of the lane width laterally. **Conclusion:** The model-tracing system is the first system to demonstrate high sample-by-sample accuracy at low false alarm rates as well as high accuracy over the course of a lane change with respect to time and lateral movement. **Application:** By providing robust real-time detection of driver lane changes, the system shows good promise for incorporation into the next generation of intelligent transportation systems.

INTRODUCTION

Intelligent transportation systems (ITSs) represent a core component of research and development of the next generation of vehicles (see, e.g., Barfield & Dingus, 1997; Michon, 1993). Although the many varieties of ITS span a broad scope of devices, infrastructure, and technology, one major goal of ITSs in general is to help drivers do what they intend to do more safely and effectively. Embedded in this goal is a simple idea: For systems to help drivers do what they intend, they must somehow infer drivers' intentions from their observable behavior. Humans infer one another's intentions continually – for instance, when a passenger warns the driver of a car in the blind spot, the passenger has likely first inferred the driver's intention to change lanes; otherwise the warning would be needless and distracting. Similarly, intelligent machine systems need some way of inferring driver intent to provide safe assistance in both mundane and critical driving situations.

The general problem of inferring human intentions from actions has been studied in a wide variety of domains: recognition systems for speech (e.g., Rabiner, 1989; Waibel & Lee, 1990) and handwriting (e.g., Jaeger, Manke, Reichert, & Waibel, 2001; Tappert, Suen, & Wakahara, 1990) translate sound signals or pen strokes into intended words using sophisticated stochastic models; intelligent user interfaces infer the user's current goals and thus provide focused, contextual help (e.g., Horvitz, Breese, Heckerman, Hovel, & Rommelse, 1998); intelligent operator systems understand and support tasks of human operators (e.g., Rubin, Jones, & Mitchell, 1988); and intelligent tutoring systems infer a student's current skill level through observation of his or her solutions to presented problems (e.g., Anderson, Boyle, Corbett, & Lewis, 1990). In all these cases, the system analyzes a person's (often highly variable) observed

Address correspondence to Dario D. Salvucci, Department of Computer Science, Drexel University, 3141 Chestnut St., Philadelphia, PA 19104; salvucci@cs.drexel.edu. *HUMAN FACTORS,* Vol. 49, No. 3, June 2007, pp. 532–542. DOI 10.1518/001872007X200157.Copyright © 2007, Human Factors and Ergonomics Society. All rights reserved.

behavior and attempts to deduce the cognitive processes that resulted in that behavior.

Given the many examples of human intent inference, surprisingly little research has been focused specifically on the challenging problem of inferring driver intentions. Pentland and Liu (1999) and Oliver and Pentland (2000) employed stochastic modeling with hidden Markov models (common in speech and handwriting recognition) to classify driver behaviors as one of a set of possible behaviors (turn left/right, change lanes, pass, stop, etc.). Pentland and Liu (1999) achieved a 95% recognition accuracy approximately 1.5 s after initiation of the maneuver; however, their work did not analyze recognition accuracy with respect to elapsed time or vehicle position at a finer grain level. Oliver and Pentland (2000) emphasized the importance of including environmental data beyond the typical data available from the vehicle (steering angle, pedal depression, etc.); they also did not analyze recognition in temporal detail, except to note, as they observed, that the system detected maneuvers before any "significant" movement of the vehicle in the lane.

These previous approaches focused on classifying an entire sequence, or window, of data points at one time but did not address the more fine-grained problem of performing incremental "sample-by-sample" classification – classifying new data samples, and possibly reinterpreting previous samples, incrementally as each sample is observed. Kuge, Yamamura, Shimoyama, and Liu (2000) developed a recognition system for lane changes that operated at a sample-by-sample basis but used no information about the surrounding environment, instead focusing only on steering-based features. Their system achieved almost perfect accuracy in classifying entire maneuvers, but sample-by-sample recognition was much less accurate and prone to frequent "blips" of erroneous detections. Thus, these earlier efforts offer promising inroads toward a lane change inference system, but none offers a true sample-by-sample detection system with high accuracy and low false alarm rates, as would be necessary in a real-world intelligent vehicle system.

In this paper we describe a novel methodology for inferring driver intentions using "model tracing." The model-tracing methodology is a computational framework for representing intentions and mapping observable behaviors to these intentions, originally designed for recognizing student intentions for intelligent tutoring systems (e.g., Anderson et al., 1990). Model tracing centers on a set of behavioral models, each of which embodies a different intention, allowing the system to "think along" with the driver for a wide range of possible intentions and scenarios. At the same time, the system gathers observable data from the driver and compares each model's simulated behavior with the driver's actual observed behavior. In doing so, model tracing determines the best matching model and thus infers the driver's most likely current intention.

To test model tracing in the context of driving, this paper describes an application of the methodology to the problem of detecting driver lane changes. Various past efforts have examined driver behavior during lane changes, in studies ranging from broad data collection efforts in naturalistic settings (e.g., Lee, Olsen, & Wierwille, 2004) to more targeted studies focusing on issues such as eye glance behavior (e.g., Salvucci & Liu, 2002; Tijerina, Garrott, Stoltzfus, & Parmer, 2005). This work has also led to explorations and applications of this knowledge toward the development of lane-change collision avoidance systems and guidelines (see, e.g., Olsen, 2003; Tijerina, 1999). In spite of our understanding of lane-change behavior, the problem of lane-change detection, as indicated earlier, has proven surprisingly challenging even for common, powerful methods of pattern recognition. Our own first attempt at lane-change detection with a prototype system (Salvucci & Siedlecki, 2003) yielded reasonable overall detection rates but, in practice, suffered inconsistent predictions for sample-by-sample detection. This paper represents a fuller specification of model tracing (initially described by Salvucci, 2004, and greatly expanded here) that offers promise as a robust, real-time algorithm ready for incorporation into intelligent vehicle systems. The paper also includes a validation study using data from both a fixed-base driving simulator and a real instrumented vehicle, allowing for comparison of system performance with both perfect information (simulator data) and noisy, more realistic data (instrumented vehicle data).

MODEL TRACING AND LANE CHANGING

The model-tracing methodology is a computational framework for mapping a person's

observable actions to his or her unobservable intentions. At its core, model tracing uses a computational model that is capable of predicting probable behaviors, given a particular intention – for instance, predicting the driver behavior resulting from the intention to change lanes, turn, or simply stay in the current lane. For our purposes in the context of driving, the process can be described as an iterating cycle of four steps: data collection, model simulation, action tracking, and thought inference. The following sections describe these steps and how they are instantiated for the particular goal of detecting the intention to change lanes.

The Model-Tracing Process

1. Data collection. The first step of model tracing involves collecting a person's observable behavior and recording the behavior as a time-ordered vector of multimodal data. The data are typically sampled at a constant rate set in consideration of both the temporal density of the data's information and the density of predictions from the computational model. In addition to any observable data from the driver, model tracing also records current environmental data in order to enable the association of environment factors with resulting behavior.

For the lane-change application herein, the system records and utilizes the following data: steering wheel angle; accelerator depression; lateral position; longitudinal distance and time headway to a lead vehicle; longitudinal distance, front and back, to vehicles in adjacent lanes; and the presence or absence of a lane to the left and right of the current travel lane. For the driving simulator data, all data are, of course, easily collectable and available. For the instrumented vehicle data, certain features may not be available easily, if at all. As described in the next section, the instrumented vehicle can easily obtain steering and accelerator data; it obtains lateral position and lead car distance and time headway by means of its built-in lane-keep support system; and it cannot obtain any information about the presence of adjacent lanes or vehicles in these lanes. This final category of data is used by the system to restrict whether a lane change is feasible or likely; because the data are not available for use in the instrumented vehicle, the system simply removes the restriction and assumes that lane changes are always possible in either direction.

2. Model simulation. The second step of the process involves running several versions of the behavioral model in parallel (conceptually if not computationally), each representing a particular stream of possible intentions and actions. The behavioral model itself is a computational representation of a person's intentions and actions. The model used here is based on a two-point steering control model (Salvucci & Gray, 2004) as well as a cognitive model of driver behavior (Salvucci, 2006) implemented in the Adaptive Control of Thought-Rational (ACT-R) cognitive architecture (Anderson et al., 2004). The ACT-R driver model includes a cognitively plausible model of lateral and longitudinal control and has been validated to behave like human drivers in many aspects of common driving scenarios, particularly curve negotiation and lane changing on a multilane highway. Although the full driver model would suit our purposes here, a far simpler model based on this one suffices for tracking lane changes and incurs much less computational complexity and overhead.

The driver model used here is structured as follows. For lateral control, we assume that the model has access to two salient visual features – namely, the orthogonal lateral distance x_{near} (in meters) of the road 10 m ahead to the vehicle's current heading and the analogous quantity x_{far} calculated at 30 m ahead; these quantities reflect a driver's use of near and far information while steering (see Donges, 1978; Land & Horwood, 1995) as instantiated in a recent two-level control model of steering (Salvucci & Gray, 2004). Using this information, the model calculates a desired steering angle, ϕ, as

$$\varphi = k_{near}(x_{near} + x_{lc}) + k_{far}(x_{near} + x_{lc}),$$

in which x_{lc} is zero during lane keeping and nonzero when lane changing, representing the desired displacement of the vehicle during the maneuver (roughly representative of desired lateral speed), with a sign dependent on the desired lane-change direction (left or right). The model also sets the accelerator position, α, based on another environmental variable: the minimum time headway, *thw,* to either the lead vehicle or, if changing lanes, the lead vehicle in the destination lane:

$$\alpha = \alpha_0 + k_{acc}(thw - thw_{follow})$$

In this formulation, thw_{follow} is the desired following time headway and α_0 is a baseline accelerator depression for maintaining normal highway speed. The resulting value, α, can be positive (for throttle depression) or negative (for brake depression) and is cut off at a maximum depression α_{max} (or minimum $-\alpha_{max}$) to represent the fact that, during normal driving, drivers typically have a desired maximum depression that may be less than pressing the pedals to the floor; if no lead vehicle is present, acceleration is set to α_{max}. Admittedly this model of the driver is grossly simplified – for instance, the steering angle does not take into account the vehicle's current speed. Nevertheless, we have found that this simple model is quite sufficient in producing the desired result – effective tracking of driver intent – and is also computationally straightforward, making possible the real-time version of the full system presented here.

Given this formulation, the system runs simultaneous simulations of the model. Specifically, it maintains a set of models and spawns off new models for the next time step using the following three rules:

- Any model that is lane keeping (LK) stays LK and also spawns new lane-changing models for changing left and right (LC-L and LC-R). The lane-changing models are spawned only if the respective lane is actually present and there are no other vehicles in this lane within a longitudinal distance d_{clear} (forward and backward).
- Any model changing lanes left (LC-L) stays LC-L until it reaches the destination lane, then changes to LK.
- Similarly, any model changing lanes right (LC-L) stays LC-R until it reaches the destination lane, then changes to LK.

These rules are applied at each time step – that is, at the same rate as data collection from the vehicle and environment. A moving time window of size w seconds is maintained for each simulation along with the human data, and at each time step any redundant models are deleted. Because the lane-change models eventually return to lane-keeping models, these are eventually pruned to eliminate redundancy, and thus the set of models reaches a steady state with a roughly constant size (dependent on whether or not the vehicle can change lanes at a given time).

The model parameters, including constants k, were approximated from the original ACT-R driver model and adjusted informally to obtain the best tracking performance across both data sets in the validation study (specifically, to maximize accuracy as reflected by the ROC curves in the next section). The final parameter values were as follows: $k_{near} = 2$, $k_{far} = 20$, $k_{acc} = 1$, $x_{lc} = 1.75$, $\alpha_0 = .3$, $\alpha_{max} = .8$, $thw_{follow} = 1.0$, $d_{clear} = 5$, and $w = 2$.

3. Action tracking. The third step of model tracing involves matching the observed behavior of the human driver with the predicted behaviors of the model simulations. Because each model generates an action sequence analogous to the human driver, one can compare the sequences directly and determine which model sequence best matches the human sequence. This requires a similarity metric between a model M's simulation and the observed human data, computed as

$$S(M) = \prod_i G(\varphi_i^M, \hat{\varphi}_i, \sigma_\varphi) \cdot G(\alpha_i^M, \hat{\alpha}_i, \sigma_\alpha)$$

as the product over all sample indices i in the moving window. In the equation, φ_i^M and α_i^M are the steering angle and accelerator position (respectively) for the model M at sample i; $\hat{\varphi}_i$ and $\hat{\alpha}_i$ are the analogous quantities observed from the human driver; and G is the value of the Gaussian distribution at the given value, mean, and standard deviation (in which σ_ϕ and σ_α are estimated along with the model parameters; for our system, $\sigma_\phi = .9$ and $\sigma_\alpha = 4$). Finally, a lane-change score is computed using the most probable models LK for lane keeping and LC for lane changing as

$$Score = \frac{\log S(LK)}{\log S(LC) + \log S(LK)}.$$

Typically, a score >.5 would indicate a lane change; however, the threshold can vary in the range [0,1] to balance detection accuracy and false alarms, as will be exploited to analyze system performance in the validation study.

4. Intent inference. In the final step, model tracing determines the inferred driver intentions simply by examining the "thoughts" of the best-matching model – that is, the intentions that produced this model's action sequence. Thus, the end result of model tracing is the inferred sequence of intentions over the length of the moving window. The process then repeats, shifting the window by one sample and iterating the four-step process.

Relation to Previous Work

As mentioned, existing methods of inferring driver intentions (e.g., Kuge et al., 2000; Oliver & Pentland, 2000; Pentland & Liu, 1999) have exclusively used the stochastic techniques employed in other domains, such as speech and handwriting recognition – specifically, hidden Markov models and their variants (see Rabiner, 1989). Although this work clearly has the same goals as our own, the techniques are quite different from model tracing, and it is very difficult to compare them algorithmically. Arguably, the only significant area of conceptual overlap is model tracing's use of a probabilistic similarity metric to compare observed behavioral data with each model's predicted data.

In fact, the most similar existing work to the proposed methodology arises not in the driving domain but, rather, in the domain in which model tracing has had its most significant impact: intelligent tutoring systems. For over a decade, so-called cognitive tutors have utilized predictive cognitive models to infer student intentions (e.g., Anderson et al., 1990; Anderson, Corbett, Koedinger, & Pelletier, 1995; Fredericksen & White, 1990). For instance, the model-tracing method used by Anderson et al. (1990) incorporates a cognitive model of student problem solving in mathematical domains such as algebra and geometry. By predicting a student's possible next step or steps in problem solving, model tracing matches observed student actions with these models to infer the most likely student intentions. These intentions form the basis of a "knowledge tracing" algorithm (Corbett & Anderson, 1995) in which a separate model estimates the student's current level of skill in the various components of problem solving. The proposed model-tracing method for driver behavior is in essence quite similar, but the driving domain poses significant challenges beyond what is needed for intelligent tutoring: Most significantly, data are sampled much more frequently in driving (many times per second vs. once every several seconds for tutoring), and the data tend to be more variable because of behavioral differences among individual drivers as well as sensor noise in data collection from the environment.

VALIDATION STUDY

To test our application of the model-tracing methodology to lane-change detection, we performed a validation study that included analysis of both simulator and vehicle data. The simulator data, collected from a medium-fidelity fixed-base driving simulator, tested the model-tracing algorithm on "perfect" nonnoisy environmental information as obtainable from the simulation environment. The vehicle data, collected using an instrumented vehicle through existing sensors, tested the algorithm on imperfect noisy environmental information including, sensor noise. The details of this study now follow.

Data Sets

The simulator data set came from a recent study (Salvucci, Boer, & Liu, 2001) involving natural driving in a fixed-base driving simulator built using the front half of a 1992 Nissan 240sx convertible. This study employed a four-lane highway – two lanes in each direction – with automated vehicles that drove different speeds and changed lanes if necessary to pass other vehicles. In the study, drivers were asked to drive naturally as they would in a real-world vehicle, going at their desired speed and changing lanes as they felt appropriate; drivers were asked to stay in the right lane for normal driving and use the left lane for passing. The data set, collected from 11 drivers at a sampling rate of approximately 13 Hz, includes a total of 311 min of driving and 433 lane changes.

The real-world vehicle data set was collected at the Nissan Research Center in Yokosuka, Japan, using an instrumented vehicle. The base vehicle was a Japanese model 2001 Nissan Cima (equivalent to an Infiniti Q45 in the United States) with a factory-installed Lane Keep Support System. This support system was used only to record the vehicle's lateral position as determined using its forward-view CCD camera; the support functionality was turned off at all times. (Because the simulator and vehicle data sets were collected in countries with different driving customs – namely, driving on the right in the United States vs. on the left in Japan – lateral position in the Japanese data set was reversed in our analysis with no loss of generality.) As in the simulator study, drivers were asked to drive as naturally as possible, changing lanes whenever they felt appropriate. The data set, collected from 9 drivers at a rate of 10 Hz, includes a total of 356 min of driving and 255 lane changes.

Defining a Lane Change

Analysis of both data sets requires a rigorous target definition of a lane change against which to compare the tracker's predictions; in other words, we needed to define what we consider a lane change and thus define what we mean by "accuracy" with regard to the model-tracing results. To this end, we defined a lane change as a segment in which the vehicle starts moving toward another lane and continues, without reversal, through to that lane. We also used a minimum threshold on lateral velocity to eliminate slow, possibly unintended drifts from our analysis to focus on typical intended lane-change maneuvers. The value of this threshold was set at 0.35 m/s, which represents a conservative threshold considering that, assuming a 3.5-m lane width, a lane change below this threshold would require at least 10 s, whereas average lane-change durations fall in the range of 3 to 7 s (see Olsen, 2003, pp. 19–20, for a review of several relevant studies). According to this definition, then, the onset of the maneuver (time 0 in the upcoming analysis) corresponds to the point at which the vehicle achieves the minimum lateral velocity and proceeds, without a lateral reversal, through the lane boundary into the destination lane.

For purposes of understanding the critical goal of our system, it is important to note the relationship between this definition and the inferences drawn by the model-tracing system. This definition, in essence, classifies lane-change points based on both past and future information – most notably, the classification of a particular data point depends on whether the vehicle continues on to actually cross the lane boundary, as indicated in future data points. In contrast, the basic problem of detection being addressed by our model-tracing system is the classification of points based on past information alone, thus attempting to predict whether the vehicle is in fact changing lanes at a given point in time with no information about future data (as is necessary for real-time detection).

Results

The model-tracing system classifies each data point as an instance of either lane changing or lane keeping, which can then be compared with the "true" labeling of the data points given by our definition of a lane change, as described previously. In doing so, the system achieves an overall sample-by-sample accuracy of 95% for both data sets. However, this overall accuracy can be misleading; for example, if the system simply guesses that all data points are lane-keeping points, it would achieve a high accuracy given that the majority of data points represent lane keeping. As a better overall measure, Figure 1 graphs the system's true positive rate (accuracy)

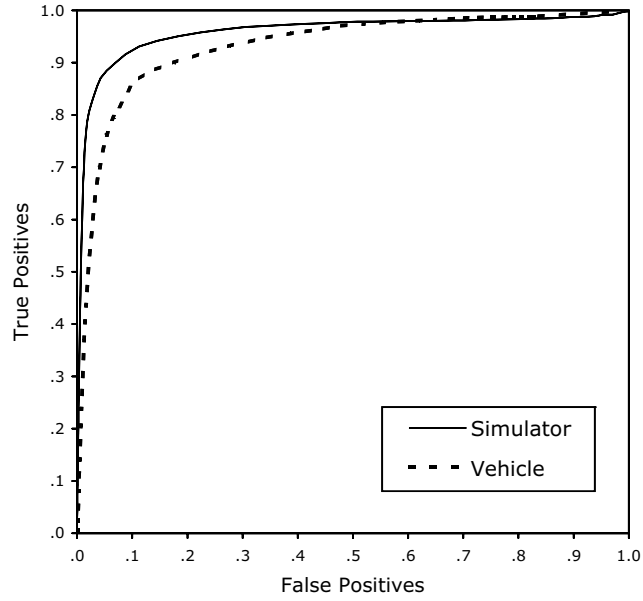

Figure 1. True versus false positives (ROC curves) for the driving simulator and instrumented vehicle data.

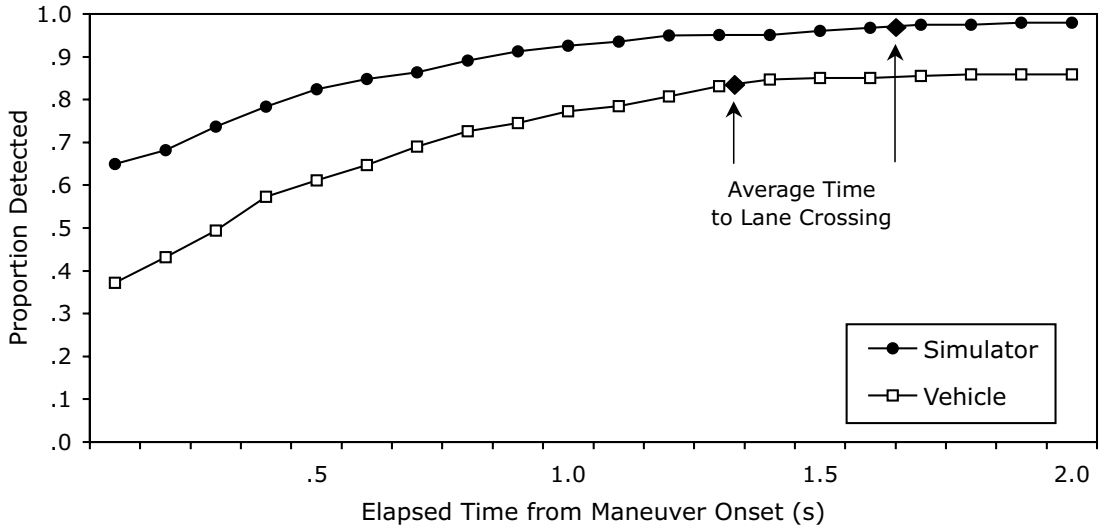

Figure 2. Lane changes detected by elapsed time at a 5% false positive rate for the simulator and vehicle data. Average time to lane crossing indicates the average time at which the vehicle center crossed over the lane boundary.

versus false positive rate (false alarms) for the simulator and vehicle data, where classifications were done on a sample-by-sample basis. (This so-called receiver-operator-characteristic [ROC] curve indicates the balance between true and false positives where perfect recognition would pull the curve through the point [0,1].) Overall, the system performs very well for both data sets. In particular, at the 0.5 score threshold, model tracing achieves 85% accuracy at a 4% false alarm rate for the simulator data and 86% accuracy at a 10% false alarm rate for the vehicle data. All points of the two curves represent different balances of accuracy and false alarms; however, the left side of the graph is especially important, given that warning systems using lane-change detection would likely require a low false alarm rate. In this region, model tracing does quite well for the simulator data and, perhaps surprisingly, even reasonably well for the vehicle data.

The ROC curve in Figure 1 represents the aggregate accuracy of the algorithms across the entire span (i.e., all data points) of the lane-change maneuvers. It is also useful to examine algorithm performance during the span of the maneuvers, specifically as a function of either time or lateral movement. Figure 2 shows the proportion of lane changes detected over time from the start of the maneuver as measured with an assumed 5% false alarm rate. At the very start of a lane change (time = 0), the system already detects 65% of lane changes in the simulator data and 37% in the vehicle data; thus, to some extent, the system can detect the behavior that characterizes the onset of a lane-change maneuver (most notably, changes in steering wheel angle) before the vehicle has made any significant lateral movement. For the simulator data, the system reaches 82% accuracy within 0.5 s of maneuver onset, 93% within 1.0 s, and 96% within 1.5 s. For the vehicle data, the system achieves 61% accuracy within 0.5 s, 77% within 1.0 s, and 85% within 1.5 s. The average time from maneuver onset to lane crossing (i.e., when the vehicle's lateral position crosses over into the destination lane) is 1.64 s for the simulator data and 1.32 s for the vehicle data; thus, by the time of lane crossing, the system detects 97% and 83% (respectively) of all lane changes.

Another way to analyze accuracy is as a function of lateral movement – that is, a function of how far the vehicle has traveled laterally toward the destination lane from the start of the maneuver. This measure is plotted in Figure 3. Again, the system detects lane changes before almost any significant movement, achieving 75% and 56% accuracy for the first data points in the simulator and vehicle results (respectively). The primary reason for such early detection is the reliance of the system on behavioral observables, most importantly steering angle, in addition to environmental observables such as lateral position. By the time the vehicle has traveled one fourth of the

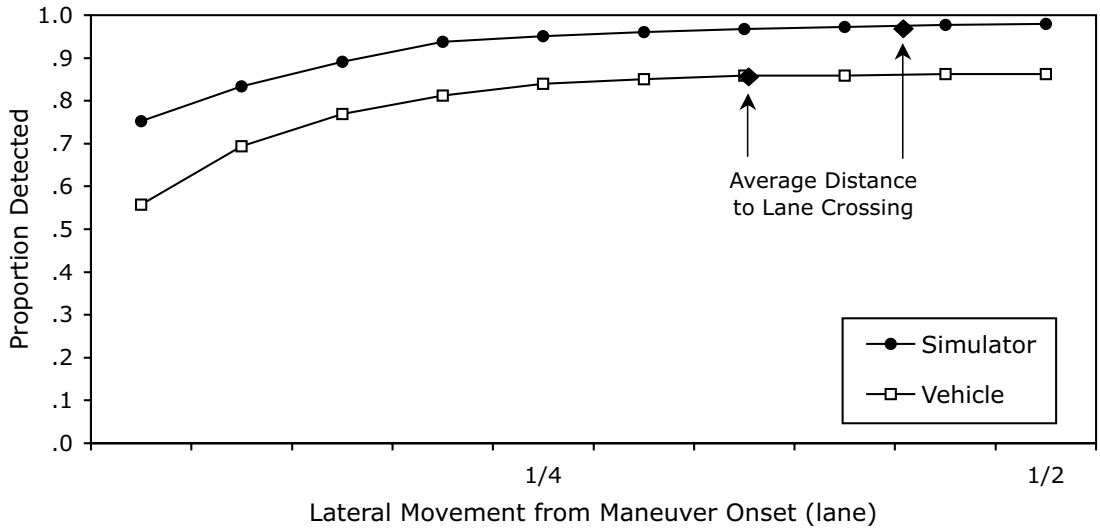

Figure 3. Lane changes detected by lateral movement at a 5% false positive rate for the simulator and vehicle data. Average distance to lane crossing indicates the average lateral distance between the vehicle's position at maneuver onset and the lane boundary.

total lane width, the system reaches 95% and 84% for the simulator and vehicle data (respectively). The average lateral movement to the point of lane crossing is also plotted in Figure 3: The average movement is .43 of the lane width for the simulator (1.56 m for a 3.66-m lane on the simulated U.S. roadway) and .35 for the vehicle (1.22 m for a 3.50-m lane on the Japanese roadway). Thus, although the average vehicle position at maneuver onset is slightly biased off of lane center toward the destination lane (because the average movements are less than half a lane width), the system exhibits high accuracy even in the short time needed for the vehicle to transition to the destination lane.

GENERAL DISCUSSION

As demonstrated, the model-tracing methodology combined with a fairly simple computational driver model allows for rapid, accurate detection of a driver's lane changes. To compare our results with those in previous studies, Pentland and Liu (1999) reported an accuracy of 95% on simulator data 1.5 s into the maneuver; however, the only other data point between 0 and 1.5 s is at the very start of the maneuver (0 s), with an accuracy of roughly 17%. Oliver and Pentland (2000) reported a maximum accuracy of 29% on instrumented vehicle data using various configurations of their statistical approach (although, in fairness, their system recognized additional maneuvers other than lane changes, such as turning). Kuge et al. (2000) achieved 98% accuracy on sample-by-sample detection of emergency lane changes (with large steering angle changes) but did not report an analogous accuracy for normal lane changes, citing problems with distinguishing between normal lane-change and lane-keeping data points. None of these studies analyzed detection accuracy with respect to lateral movement, and none addressed the issue of false alarms in a rigorous way. We have also recently begun exploring the use of a data-driven statistical framework called support vector machines for lane-change detection; initial results on real vehicle data (Mandalia & Salvucci, 2005) have demonstrated almost 98% accuracy at a 5% false alarm rate and 87% accuracy within 0.3 s of maneuver onset, but this study lacked more detailed results in terms of a full ROC analysis and performance analysis over time and lateral movement. Thus, the model-tracing system is the first system to demonstrate high sample-by-sample accuracy at low false alarm rates as well as high accuracy over the course of a lane change with respect to time and lateral movement.

The differences in system performance for simulator and real-vehicle data arise primarily from two major differences between the data sources. First, the vehicle data unavoidably incorporate

noise from sensors and lower level analysis algorithms (e.g., computer-vision analysis of the roadway), whereas the simulator data provide perfect information about the external environment. Second, current technology in real-vehicle sensors does not allow for complete knowledge about the surrounding environment – for example, current sensors can more easily acquire information about the lead vehicle than about adjacent vehicles – whereas, of course, the simulator allows for as much information as desired. Both differences affect our system: For the simulator data, the system has fully accurate data on lead and adjacent vehicles and knows of the presence or absence of adjacent lanes (which facilitates deciding when to spawn new lane-change models), whereas for the vehicle data, sensor noise and lack of knowledge of adjacent lanes make the detection problem more challenging. Nevertheless, system performance even for the real-vehicle data remains quite high and, we believe, indicates a high potential for successful integration with real-world vehicle systems.

One very interesting direction for future study of the proposed method is the incorporation of other types of input data. Arguably, the most obvious potential source of information is the driver's use of turn signals (or blinkers): When the driver turns on this signal, it seems reasonable to favor classification as a lane change. However, even this seemingly clear binary signal can have associated ambiguities. For example, a driver switching on the turn signal might be indicating either the decision to change lanes immediately or the intent to change lanes eventually (the latter to signal other vehicles that a lane change is desired). In addition, drivers are not always consistent in their use of turn signals – for instance, both the real-vehicle study by Lee et al. (2004, p. 84) and the simulator study by Salvucci and Liu (2002) found that drivers activated their turn signals only half of the time (or less) at the onset of a lane-change maneuver. Also, Olsen (2003, p. 96) found that drivers failed to use turn signals at all during approximately one third (35.8%) of all lane changes. For the model-tracing system, the computational driver model could simply switch on its turn signal when initiating a lane change, but given the complexities of real driver turn signal usage, it is likely that a more complex model of driver turn signal use would be required to incorporate these data in a useful manner.

Another promising source of input data involves analysis of a driver's focus of visual attention, either at a fine-grained level with eye gaze data or at a coarser grained level with head pose data. Acquisition and analysis of eye gaze data within a real vehicle is a very challenging problem, not only in terms of acquiring adequate images of the driver's eye or eyes but also in developing a sufficient model of the surrounding environment such that eye gaze can be associated with particular external objects. A less powerful but likely more feasible alternative is the use of head pose data in sensing the general direction of driver gaze (through the front windshield, at the side mirror, etc.). Again, use of such data would require that the driver model account for distributions of visual attention, as have been studied in the decision phase (e.g., Tijerina et al., 2005) and execution phase (e.g., Salvucci & Liu, 2002) of lane changes. The ACT-R integrated driver model (Salvucci, 2006) begins to account for these types of gaze distributions, primarily in the execution phase, but much more work is needed to model the decision phase and to integrate the full complex model into a model-tracing framework.

Ultimately, we hope that the lane-change detection system can be successfully integrated with related lower level machine-vision systems (e.g., McCall, Wipf, Trivedi, & Rao, in press) and, in turn, be utilized by a fuller intelligent vehicle system. For example, lane-change detection could potentially improve an existing lane-departure warning system by informing the system of likely lane changes: If the vehicle begins to depart from the current lane but the system detects an intended lane-change maneuver, the warning system could change its response (e.g., not produce an audible warning, or change the force feedback on the steering wheel) to the driver and avoid annoying and distracting false alarms. The warning system could also detect intended lane changes in dangerous situations (e.g., a vehicle in the blind spot leading to a "lane-change crash": see Chovan, Tijerina, Alexander, & Hendricks, 1994; Hetrick, 1997) and provide warnings based on these detections. Whereas the current system focuses on the detection of lane changes in the execution phase, any work toward improving detection in the decision-making phase, as described previously, could facilitate prediction of lane changes much earlier than is possible with the current system.

Although this paper instantiates one application of the model-tracing methodology for lane-change detection, the approach is by no means limited to lane changing – indeed, the basic ideas in the methodology generalize well to other intentions, such as turning, stopping, and starting. We are now exploring the application of the methodology to other intentions, including the development of further computational models of driver behavior during these intentions and integration of these models into the model-tracing process to enable fuller real-time driver intent inference. A more general intent inference system could have significant potential benefits to intelligent vehicle systems designed to help with other types of driving situations beyond lane keeping and changing, such as parking assistance or adaptive cruise control systems.

ACKNOWLEDGMENTS

This work was supported by National Science Foundation Grant #IIS-0133083 and by Nissan Motor Company. We thank Erwin Boer and the other Nissan team members for valuable guidance on this work and John Lee and three anonymous reviewers for many helpful suggestions.

REFERENCES

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review, 111,* 1036–1060.

Anderson, J. R., Boyle, C. F., Corbett, A. T., & Lewis, M. W. (1990). Cognitive modeling and intelligent tutoring. *Artificial Intelligence, 42,* 7–49.

Anderson, J. R., Corbett, A. T., Koedinger, K., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. *Journal of the Learning Sciences, 4,* 167–207.

Barfield, W., & Dingus, T. (Eds.). (1997). *Human factors in intelligent transportation systems.* Mahwah, NJ: Erlbaum.

Chovan, J. D., Tijerina, L., Alexander, G., & Hendricks, D. L. (1994). *Examination of lane change crashes and potential IVHS countermeasures* (Tech. Rep. No. DOT-HS-808-701/DOT-VNTSC-NHTSA-93-2). Cambridge, MA: U.S. Department of Transportation, Volpe National Transportation Systems Center.

Corbett, A. T., & Anderson, J. R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction, 4,* 253–278.

Donges, E. (1978). A two-level model of driver steering behavior. *Human Factors, 20,* 691–707.

Frederiksen, J. R., & White, B. Y. (1990). Intelligent tutors as intelligent testers. In N. Frederiksen, R. Glaser, A. Lesgold, & M. G. Shafto (Eds.), *Diagnostic monitoring of skill and knowledge acquisition* (pp. 1–25). Hillsdale, NJ: Erlbaum.

Hetrick, S. (1997). *Examination of driver lane change behavior and the potential effectiveness of warning onset rules for lane change or "side" crash avoidance systems.* Unpublished master's thesis, Virginia Polytechnic Institute and State University, Department of Industrial Systems and Engineering, Blacksburg, VA.

Horvitz, E., Breese, J., Heckerman, D., Hovel, D., & Rommelse, K. (1998). The Lumiere project: Bayesian user modeling for inferring the goals and needs of software users. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence* (pp. 256–265). San Francisco, CA: Morgan Kaufmann.

Jaeger, S., Manke, S., Reichert, J., & Waibel, A. (2001). Online handwriting recognition: The NPen++ recognizer. *International Journal on Document Analysis and Recognition, 3,* 169–180.

Kuge, N., Yamamura, T., Shimoyama, O., & Liu, A. (2000). A driver behavior recognition method based on a drive model framework. In *Proceedings of the Society of Automotive Engineers World Congress 2000* (pp. 469–476). New York: Society of Automotive Engineers.

Land, M., & Horwood, J. (1995). Which parts of the road guide steering? *Nature, 377,* 339–340.

Lee, S. E., Olsen, E. C. B., & Wierwille, W. W. (2004). *A comprehensive examination of naturalistic lane changes* (Tech. Rep. DOT-HS-809-702). Washington, DC: U.S. Department of Transportation, National Highway Traffic Safety Administration.

Mandalia, H. M., & Salvucci, D. D. (2005). Using support vector machines for lane change detection. In *Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting* (pp. 1965–1969). Santa Monica, CA: Human Factors and Ergonomics Society.

McCall, J., Wipf, D., Trivedi, M. M., & Rao, B. (in press). Lane change intent analysis using robust operators and sparse Bayesian learning. *IEEE Transactions on Intelligent Transportation Systems.*

Michon, J. A. (Ed.) (1993). *Generic intelligent driver support system.* London: Taylor & Francis.

Oliver, N., & Pentland, A. P. (2000). Graphical models for driver behavior recognition in a SmartCar. In *Proceedings of the IEEE Intelligent Vehicles Symposium 2000* (pp. 7–12). Piscataway, NJ: IEEE.

Olsen, E. C. B. (2003). *Modeling slow lead vehicle lane changing.* Unpublished doctoral dissertation, Virginia Polytechnic Institute and State University, Department of Industrial and Systems Engineering, Blacksburg, VA. Retrieved January 17, 2007 from http://scholar.lib.vt.edu/theses/available/etd-12032003-152916/

Pentland, A., & Liu, A. (1999). Modeling and prediction of human behavior. *Neural Computation, 11,* 229–242.

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

Rubin, K. S., Jones, P. M., & Mitchell, C. M. (1988). OFMspert: Inference of operator intentions in supervisory control using a blackboard architecture. *IEEE Transactions on Systems, Man, and Cybernetics, 18,* 618–637.

Salvucci, D. D. (2004). Inferring driver intent: A case study in lane-change detection. In *Proceedings of the Human Factors Ergonomics Society 48th Annual Meeting* (pp. 2228–2231). Santa Monica, CA: Human Factors and Ergonomics Society.

Salvucci, D. D. (2006). Modeling driver behavior in a cognitive architecture. *Human Factors, 48,* 362–380.

Salvucci, D. D., Boer, E. R., & Liu, A. (2001). Toward an integrated model of driver behavior in a cognitive architecture. *Transportation Research Record, 1779,* 9–16.

Salvucci, D. D., & Gray, R. (2004). A two-point visual control model of steering. *Perception, 33,* 1233–1248.

Salvucci, D. D., & Liu, A. (2002). The time course of a lane change: Driver control and eye-movement behavior. *Transportation Research Part F, 5,* 123–132.

Salvucci, D. D., & Siedlecki, T. (2003). Toward a unified framework for tracking cognitive processes. In *Proceedings of the 25th Annual Conference of the Cognitive Science Society* (pp. 1023–1028). Mahwah, NJ: Erlbaum.

Tappert, C. C., Suen, C. Y., & Wakahara, T. (1990). The state of the art in on-line handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 12,* 787–808.

Tijerina, L. (1999). Operational and behavioral issues in the comprehensive evaluation of lane change crash avoidance systems. *Journal of Transportation Human Factors, 1,* 159–176.

Tijerina, L., Garrott, W. R., Stoltzfus, D., & Parmer, E. (2005). Van and passenger car driver eye glance behavior during the lane change decision phase. In *Proceedings of the 84th Annual Meeting of the Transportation Research Board* [CD-ROM]. Washington, DC: Transportation Research Board.

Waibel, A., & Lee., K. (Eds.) (1990). *Readings in speech recognition.* San Mateo, CA: Morgan Kaufmann.

Dario D. Salvucci is an assistant professor of computer science at Drexel University. He received his Ph.D. in computer science from Carnegie Mellon University in 1999.

Hiren M. Mandalia is a senior software engineer at Motorola Inc., Schaumburg, Illinois. He received his M.S. in computer science from Drexel University in 2005.

Nobuyuki Kuge is an engineer at Nissan Motor Company, Ltd. He received his M.A. in behavioral science from Hokkaido University in 1991.

Tomohiro Yamamura is a manager at Nissan Motor Company, Ltd. He received his B.E. in aeronautics from the University of Tokyo in 1987.

Date received: June 9, 2005
Date accepted: March 3, 2006