Modeling and Recognition of Driving Behavior Based on Stochastic Switched ARX Model

Shogo Sekizawa, Shinkichi Inagaki, Tatsuya Suzuki, *Member, IEEE*, Soichiro Hayakawa, Nuio Tsuchida, Taishi Tsuda, and Hiroaki Fujinami

***Abstract*—This paper presents the development of the modeling and recognition of human driving behavior based on a stochastic switched autoregressive exogenous (SS-ARX) model. First, a parameter estimation algorithm for the SS-ARX model with multiple measured input–output sequences is developed based on the expectation–maximization algorithm. This can be achieved by extending the parameter estimation technique for the conventional hidden Markov model. Second, the developed parameter estimation algorithm is applied to driving data with the focus being on driver's collision avoidance behavior. The driving data were collected using a driving simulator based on the cave automatic virtual environment, which is a stereoscopic immersive virtual reality system. Then, the parameter set for each driver is obtained, and certain driving characteristics are identified from the viewpoint of switched control mechanism. Finally, the performance of the SS-ARX model as a behavior recognizer is examined. The results show that the SS-ARX model holds remarkable potential to function as a behavior recognizer.**

***Index Terms*—Behavior recognition, driving behavior, expectation–maximization (EM) algorithm, hidden Markov model (HMM), stochastic switched autoregressive exogenous (SS-ARX) model.**

I. Introduction

RECENTLY, the demand for vehicle control systems is shifting from the realization of high-performance vehicles to the development of human-friendly vehicle control systems. This implies that the emphasis is shifting toward design and manufacturing vehicles that meet each customer's specific requirements. Although considered important, this matter has not yet been addressed entirely due to the wide variety of customer demands. To formally address this problem, it is necessary to develop a mathematical model of the driving behavior of different drivers and to exploit it to design a control system. This viewpoint is of particular importance when the vehicle is to be driven by the elderly and/or the disabled.

Manuscript received November 9, 2005; revised May 2, 2006, December 20, 2006, and June 7, 2007. This work was supported by the Space Robotic Center of the Toyota Technological Institute, where CAVE is installed. The Associate Editor for this paper was T. A. Dingus.

S. Sekizawa, S. Inagaki, and T. Suzuki are with the Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Nagoya 464-8603, Japan (e-mail: t_suzuki@nuem.nagoya-u.ac.jp).

S. Hayakawa and N. Tsuchida are with Toyota Technological Institute, Nagoya 468-8511, Japan (e-mail: s_hayakawa@toyota-ti.ac.jp).

T. Tsuda and H. Fujinami are with the Development Department No. 3, Integrated System Engineering Division, Vehicle Engineering Group, Toyota Motor Corporation, Toyota 471-8571, Japan (e-mail: taishi@tsuda.tec.toyota.co.jp).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2007.903441

Due to the aforementioned circumstances, driving behavior modeling has attracted much attention from several researchers [1]–[4], [8], [9]. In order to model driving behavior, the conventional techniques such as the nonlinear regression models, neural networks (NNs), and fuzzy systems have been used [5], [6]. The usage of these techniques, however, poses certain problems. 1) The obtained model is often too complicated, and 2) this, in turn, makes it impossible to understand the physical meanings of the driver's maneuver. When we consider driving behavior, it is often found that a driver appropriately switches between certain simple primitive skills instead of adopting a complex nonlinear control law. This switching between primitive skills can be attributed to the driver's decision-making process. This consideration strongly motivates us to model the human driving behavior as a hybrid dynamical system (HDS) [10]–[12]. HDSs comprise both continuous dynamics and discrete mode changes. The former are typically associated with differential (or difference) equations and the latter with automata, logic, etc. By regarding the driver's primitive skills and switching scenario as the continuous and discrete parts of an HDS, respectively, the understanding of the human driving behavior can be recast as a parameter estimation problem in the HDS framework. In the control and computer science communities, many studies have reported the expression, stability analysis, control, verification, and identification [13], [14] of the HDSs. However, the application of the HDS model to the analysis of human behavior has not yet been fully discussed.

HDSs can be broadly classified into two classes: HDSs wherein the transition between discrete states (modes) is specified in terms of deterministic logic and HDSs wherein this transition is specified by transition probabilities. In [8] and [9], we applied the HDS with deterministic mode change to the modeling of driving behavior. Although this paper could capture the motion and decision-making aspects of human behavior, it is not considered suitable for real-time behavior recognition due to its high computational cost. This is because mixed integer programming and the data clustering technique were used for the parameter estimation in [8] and [9], respectively.

In this paper, a stochastic switched autoregressive exogenous (SS-ARX) model, which is an HDS model with stochastic mode change, is proposed and applied to human behavior modeling. The proposed model can be regarded as a natural extension of the standard hidden Markov model (HMM) [15], [16], wherein different models are allocated to each discrete state of the HMM. Furthermore, a concrete parameter estimation algorithm is derived based on the expectation–maximization

(EM) algorithm. Due to the similarity between behavioral data and speech signals, it is considered that the SS-ARX model holds remarkable potential to function as a robust behavior recognizer. The significant advantages of using SS-ARX as a behavior model are described as follows. 1) It can calculate behavior likelihood with reasonable computational cost. 2) It can explicitly capture the mathematical relationship between the input and output signals of human behavior in each discrete state (mode), which cannot be realized in a standard HMM application [16]. 3) It can express the stochastic nature inherent in human behavior.

As a typical application, we focus on the driver's collision avoidance behavior, and the proposed model and parameter estimation technique is applied. The driving data are collected using a 3-D driving simulator (DS) based on the cave automatic virtual environment (CAVE), which can provide stereoscopic immersive vision [7]. In the proposed modeling technique, the SS-ARX model expresses the relationship between sensory information (such as the distance between cars, range rate, and lateral displacement between cars) and driver output (such as steering amount). Finally, the performance of the SS-ARX model as a behavior recognizer is demonstrated through certain experimental results.

This paper is organized as follows: The ARX model is briefly reviewed in Section II. The parameter estimation technique for the SS-ARX model is introduced in Section III. The configuration of the developed DS based on the CAVE is introduced in Section IV, along with the scenario of the examination conducted in this paper. The modeling and recognition results are shown in Sections V and VI, respectively.

II. Brief Review of the ARX Model

This section provides a brief review of the conventional ARX model, which is the predecessor of the SS-ARX model.

The standard ARX model is described by the following differential equations:

$$y_t = c_1 y_{t-1} + c_1 y_{t-2} + \cdots + c_n y_{t-n} + d_0 u_t + d_1 u_{t-1} + \cdots + d_m u_{t-m} + e_t \quad (1)$$

where y_t and u_t are the output and input of the system at time t. They are assumed to be scalar-valued signals. Note that this assumption will be eliminated in Section III-D. Furthermore, n and m are the orders of the ARX model, and $c_1, c_2, \ldots, c_n$, $d_0, d_1, \ldots, d_m$ are the parameters. e_t is referred to as the equation error and is considered to exhibit Gaussian distribution with variance σ.

By using the following vector form:

$$\boldsymbol{\theta} = (c_1, c_2, \ldots, c_n, d_0, d_1, \ldots, d_m)^T \quad (2)$$

$$\boldsymbol{\psi}_t = (y_{t-1}, y_{t-2}, \ldots, y_{t-n}, u_t, u_{t-1}, \ldots, u_{t-m})^T \quad (3)$$

(1) is rewritten as follows:

$$y_t = \boldsymbol{\theta}^T \boldsymbol{\psi}_t + e_t. \quad (4)$$

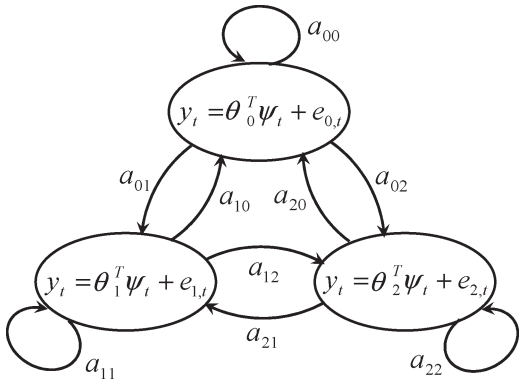


Fig. 1. SS-ARX model (three states).

III. Parameter Estimation for the SS-ARX Model

The SS-ARX model is defined as the system in which an ARX model is switched to another based on the state transition probability, as shown in Fig. 1. It can be regarded as the combination of several ARX models and HMM. The parameters of the ARX model assigned to a discrete state S_i are designated by subscript i.

A. Parameters of the SS-ARX Model

The parameters of the SS-ARX model are specified as follows:

1) set of discrete states $S[= S_i, (i = 0, 1, \ldots, N)]$;
2) a_{ij}: state transition probability $(i = 0, 1, \ldots, N; j = 0, 1, \ldots, N)$;
3) π_i: initial state probability $(i = 0, 1, \ldots, N)$;
4) $\boldsymbol{\theta}_i$: parameters of the ARX model assigned to $S_i (i = 0, 1, 2, \ldots, N)$;
5) σ_i: variance in the equation error in the ARX model assigned to $S_i (i = 0, 1, 2, \ldots, N)$.

$N + 1$ denotes the number of discrete states. In the subsequent sections, we denote the set of parameters in the SS-ARX model by $\lambda = (\pi_i, a_{ij}, \boldsymbol{\theta}_i, \sigma_i)$.

B. Definition of the Measured Signal and Three Fundamental Problems

To address the three problems stated later in this section, the measured signal and its occurrence probability are defined for the SS-ARX model as follows: First, a measured signal o_t at time t is defined as the combination of the output y_t and the regressor $\boldsymbol{\psi}_t$, i.e., $o_t = (y_t, \boldsymbol{\psi}_t)$. Subsequently, its occurrence probability $b_i(o_t)$ is defined by assuming the Gaussian distribution of the equation error and is given by

$$b_i(o_t) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left\{ -\frac{\left(\boldsymbol{\theta}_i^T \boldsymbol{\psi}_t - y_t\right)^2}{2\sigma_i^2} \right\}. \quad (5)$$

Based on these definitions, the following three fundamental problems can be addressed for the SS-ARX model:

1) Evaluation problem

In the evaluation problem, the probability that the measured signal sequence $\boldsymbol{O} = (o_0, o_1, \ldots, o_t, \ldots, o_T)$

originates from the model $\lambda = (\pi_i, a_{ij}, \boldsymbol{\theta}_i, \sigma_i)$ is calculated. This problem can be solved by applying the forward algorithm [15].

2) Decoding problem

In the decoding problem, the most likely underlying state sequence $\boldsymbol{s} = (s_0, s_1, \ldots, s_t, \ldots, s_T)$ yielding the measured signal sequence $\boldsymbol{O} = (o_0, o_1, \ldots, o_t, \ldots, o_T)$ is found for the model $\lambda = (\pi_i, a_{ij}, \boldsymbol{\theta}_i, \sigma_i)$ and measured signal sequence $\boldsymbol{O}$. This state estimation can be realized by applying the Viterbi algorithm [15].

3) Estimation problem

In the estimation problem, the model parameter $\lambda = (\pi_i, a_{ij}, \boldsymbol{\theta}_i, \sigma_i)$, providing the highest occurrence probability for the measured signal sequence $\boldsymbol{O} = (o_0, o_1, \ldots, o_t, \ldots, o_T)$, is estimated.

The solution for problems 1) and 2) are identical to that for the standard HMM. However, the parameter estimation algorithm for the SS-ARX model requires some extension to the one for the standard HMM. In the following section, a concrete parameter estimation algorithm for the SS-ARX model is derived.

C. Parameter Estimation

Here, we assume that L measured signal sequences are collected for the parameter estimation of the SS-ARX model.

1) EM Algorithm: First, we consider an immeasurable state sequence $\boldsymbol{s} = (s_0, s_1, \ldots, s_t, \ldots, s_T)$ and measurable signal sequences $\boldsymbol{O_l} = (o_{l,0}, o_{l,1}, \ldots, o_{l,t}, \ldots, o_{l,T})$ (l represents the index of the measured signal sequence). The maximization of the likelihood value of the $\boldsymbol{s}$ and $\boldsymbol{O_l}$, $\sum_{l=1}^{L} L(\boldsymbol{s}, \boldsymbol{O_l}; \lambda) = \sum_{l=1}^{L} P(\boldsymbol{s}, \boldsymbol{O_l}|\lambda)$ is achieved using EM algorithm. The EM algorithm can locally maximize the likelihood value $\sum_{l=1}^{L} L(\boldsymbol{s}, \boldsymbol{O_l}; \lambda) = \sum_{l=1}^{L} P(\boldsymbol{s}, \boldsymbol{O_l}|\lambda)$ by iterative procedure.

Typically, the EM algorithm attempts to find the parameter λ' such that it maximizes the following Q function:

$$Q(\lambda, \lambda') = \sum_{l=1}^{L} E\left[\log\left\{P(\boldsymbol{s}, \boldsymbol{O_l}|\,\lambda')\right\}|\boldsymbol{O_l}, \lambda\right] \tag{6}$$

$$= \sum_{l=1}^{L} \sum_{\boldsymbol{s}} P(\boldsymbol{s}|\boldsymbol{O_l}, \lambda) \log\left\{P(\boldsymbol{s}, \boldsymbol{O_l}|\lambda')\right\} \tag{7}$$

by executing following procedures iteratively.

1) Specify the initial parameter λ.
2) Find the λ' such that it maximizes the $Q(\lambda, \lambda')$.
3) If $\lambda' = \lambda$, stop, and if $\lambda' \neq \lambda$, substitute λ' for λ and go to 2).

2) Parameter Estimation Algorithm: The parameters of the SS-ARX model before and after the update are considered to be given by $\lambda = (\pi_i, a_{ij}, \boldsymbol{\theta}_i, \sigma_i)$ and $\lambda' = (\pi_i', a_{ij}', \boldsymbol{\theta}_i', \sigma_i')$, respectively. From (7)

$$Q(\lambda, \lambda') = \sum_{l=1}^{L} \left\{ \sum_{\boldsymbol{s}} \frac{1}{P(\boldsymbol{O_l}|\lambda)} P(\boldsymbol{s}, \boldsymbol{O_l}|\lambda) \times \log\{P(\boldsymbol{s}, \boldsymbol{O_l}|\lambda')\} \right\}. \tag{8}$$

Now, we replace $(1/P(\boldsymbol{O_1}|\lambda)), (1/P(\boldsymbol{O_2}|\lambda)), \ldots, (1/P(\boldsymbol{O_L}|\lambda))$ in (8) by $k_1, k_2, \ldots, k_L$. Then, the maximization of $Q(\lambda, \lambda')$ implies the maximization of $\tilde{Q}(\lambda, \lambda')$ given by

$$\tilde{Q}(\lambda, \lambda') = \sum_{l=1}^{L} \left\{ \sum_{\boldsymbol{s}} k_l P(\boldsymbol{s}, \boldsymbol{O_l}|\lambda) \log\left\{P(\boldsymbol{s}, \boldsymbol{O_l}|\lambda')\right\} \right\}. \tag{9}$$

By using the definition

$$P(\boldsymbol{s}, \boldsymbol{O_l}|\lambda) = \pi_{s_0} b_{s_0}(o_{l,0}) \times a_{s_0 s_1} b_{s_1}(o_{l,1}) \times a_{s_1 s_2} b_{s_2}(o_{l,2}) \times \cdots \times a_{s_{T-1} s_T} b_{s_T}(o_{l,T}) \tag{10}$$

$\tilde{Q}(\lambda, \lambda')$ can be decomposed as follows:

$$\tilde{Q}(\lambda, \lambda') = \tilde{Q}_1(\lambda, \lambda') + \tilde{Q}_2(\lambda, \lambda') + \tilde{Q}_3(\lambda, \lambda') \tag{11}$$

where

$$\tilde{Q}_1(\lambda, \lambda') = \sum_{l=1}^{L} \sum_{s_0=0}^{N} \sum_{s_1=0}^{N} \cdots \sum_{s_T=0}^{N} k_l \pi_{s_0} b_{s_0}(o_{l,0}) \times a_{s_0 s_1} b_{s_1}(o_{l,1}) \times \cdots \times a_{s_{T-1} s_T} b_{s_T}(o_{l,T}) \times \log\left\{\pi_{s_0}'\right\} \tag{12}$$

$$\tilde{Q}_2(\lambda, \lambda') = \sum_{l=1}^{L} \sum_{s_0=0}^{N} \sum_{s_1=0}^{N} \cdots \sum_{s_T=0}^{N} k_l \pi_{s_0} b_{s_0}(o_{l,0}) \times a_{s_0 s_1} b_{s_1}(o_{l,1}) \times \cdots \times a_{s_{T-1} s_T} b_{s_T}(o_{l,T}) \times \sum_{t=1}^{T} \log\left\{a_{s_{t-1} s_t}'\right\} \tag{13}$$

$$\tilde{Q}_3(\lambda, \lambda') = \sum_{l=1}^{L} \sum_{s_0=0}^{N} \sum_{s_1=0}^{N} \cdots \sum_{s_T=0}^{N} k_l \pi_{s_0} b_{s_0}(o_{l,0}) \times a_{s_0 s_1} b_{s_1}(o_{l,1}) \times \cdots \times a_{s_{T-1} s_T} b_{s_T}(o_{l,T}) \times \sum_{t=0}^{T} \log\left\{b_{s_t}'(o_{l,t})\right\}. \tag{14}$$

Next, the forward $\alpha(l, i, t)$ and backward $\beta(l, i, t)$ probabilities are defined as follows:

$$\alpha(l, i, t) = \sum_{s_0=0}^{N} \sum_{s_1=0}^{N} \cdots \sum_{s_{t-1}=0}^{N} \pi_{s_0} b_{s_0}(o_{l,0}) \times a_{s_0 s_1} b_{s_1}(o_{l,1}) \times \cdots \times a_{s_{t-1} s_t} b_{s_t}(o_{l,t}) \tag{15}$$

$$\beta(l, i, t) = \sum_{s_{t+1}=0}^{N} \sum_{s_{t+2}=0}^{N} \cdots \sum_{s_T=0}^{N} a_{s_t s_{t+1}} b_{t+1}(o_{l,t+1}) \times a_{s_{t+1} s_{t+2}} b_{s_{t+2}}(o_{l,t+2}) \times \cdots \times a_{s_{T-1} s_T} b_{s_T}(o_{l,T}) \tag{16}$$

where $\alpha(l, i, t)$ is the probability that the SS-ARX model λ generates the lth measured signal subsequence $\boldsymbol{O_l} = (o_{l,0}, o_{l,1}, \ldots, o_{l,t})$ until t and reaches S_i at t (i.e., $s_t = S_i$). Furthermore, $\beta(l, i, t)$ is the probability that the SS-ARX

model λ generates the lth measured signal subsequence $\boldsymbol{O_l} = (o_{l,t+1}, o_{l,t+2}, \ldots, o_{l,T})$ starting from S_i at t (i.e., $s_t = S_i$) and reaches the final state at T.

By using (15) and (16), (12)–(14) can be rewritten as follows:

$$\tilde{Q}_1(\lambda, \lambda') = \sum_{l=1}^{L}\sum_{i=0}^{N} k_l \pi_i b_i(o_{l,0}) \log\{\pi_i'\}\, \beta(l,i,0) \tag{17}$$

$$\tilde{Q}_2(\lambda, \lambda') = \sum_{l=1}^{L}\sum_{t=1}^{T}\sum_{i=0}^{N}\sum_{j=0}^{N} k_l \log\{a_{ij}'\} \times \alpha(l,i,t-1) a_{ij} b_j(o_{l,t}) \beta(l,j,t) \tag{18}$$

$$\tilde{Q}_3(\lambda, \lambda') = \sum_{l=1}^{L}\sum_{t=0}^{T}\sum_{i=0}^{N} k_l \log\{b_i'(o_{l,t})\}\, \alpha(l,i,t)\beta(l,i,t). \tag{19}$$

By maximizing $\tilde{Q}_1(\lambda, \lambda')$, $\tilde{Q}_2(\lambda, \lambda')$, and $\tilde{Q}_3(\lambda, \lambda')$, $\tilde{Q}(\lambda, \lambda')$ can be maximized. Therefore, λ', which maximizes $\tilde{Q}(\lambda, \lambda')$, can be obtained as follows:

$$\pi_i' = \frac{\sum_{l=1}^{L} k_l \pi_i b_i(o_{l,0}) \beta(l,i,0)}{\sum_{i=0}^{N}\sum_{l=1}^{L} k_l \pi_i b_i(o_{l,0}) \beta(l,i,0)} \tag{20}$$

$$a_{ij}' = \frac{\sum_{t=1}^{T}\sum_{l=1}^{L} k_l \alpha(l,i,t-1) a_{ij} b_j(o_{l,t}) \beta(l,j,t)}{\sum_{j=0}^{N}\sum_{t=1}^{T}\sum_{l=1}^{L} k_l \alpha(l,i,t-1) a_{ij} b_j(o_{l,t}) \beta(l,j,t)} \tag{21}$$

$$\boldsymbol{\theta}_i' = \left\{\sum_{t=0}^{T}\sum_{l=1}^{L} k_l \boldsymbol{\psi}_{l,t} \boldsymbol{\psi}_{l,t}^T \alpha(l,i,t)\beta(l,i,t)\right\}^{-1} \times \left\{\sum_{t=0}^{T}\sum_{l=1}^{L} k_l \boldsymbol{\psi}_{l,t} y_{l,t} \alpha(l,i,t)\beta(l,i,t)\right\} \tag{22}$$

$$\sigma_i'^2 = \frac{\sum_{t=0}^{T}\sum_{l=1}^{L} k_l \left|\boldsymbol{\theta}_i'^T \boldsymbol{\psi}_{l,t} - y_{l,t}\right|^2 \alpha(l,i,t)\beta(l,i,t)}{\sum_{t=0}^{T}\sum_{l=1}^{L} k_l \alpha(l,i,t)\beta(l,i,t)} \tag{23}$$

λ can be locally optimized by executing the three steps in the EM algorithm iteratively along with (20)–(23).

Note that (22) can be regarded as the weighted least mean-square solution in which the weighted parameters are specified by $\alpha(l,i,t)\beta(l,i,t)$, i.e., the probability that $o_{l,t}$ is generated from S_i.

D. Parameter Estimation for Multiple Input and Output Cases

The SS-ARX model and its parameter estimation algorithm can be easily extended to cover the case wherein y_t and u_t in (1) are vector-valued signals. In this case, the ARX model (1) is rewritten as follows:

$$\boldsymbol{y}_t = \boldsymbol{c}_1 \boldsymbol{y}_{t-1} + \boldsymbol{c}_1 \boldsymbol{y}_{t-2} + \cdots + \boldsymbol{c}_n \boldsymbol{y}_{t-n} + \boldsymbol{d}_0 \boldsymbol{u}_t + \boldsymbol{d}_1 \boldsymbol{u}_{t-1} + \cdots + \boldsymbol{d}_m \boldsymbol{u}_{t-m} + \boldsymbol{e}_t \tag{24}$$

where the dimensions of $\boldsymbol{y}_t$ and $\boldsymbol{u}_t$ are considered to be q and p, respectively. Then, $\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_n$ and $\boldsymbol{d}_0, \boldsymbol{d}_1, \ldots, \boldsymbol{d}_m$ are $q \times q$ and $q \times p$ matrices, respectively. Furthermore, $\boldsymbol{e}_t$ is a q-dimensional vector that is considered to exhibit a mixed Gaussian distribution with covariance matrix $\boldsymbol{\Sigma}$.

Subsequently, the measured signal and its occurrence probability are redefined as follows:

$$b_i(\boldsymbol{o}_t) = \frac{1}{(2\pi)^{\frac{q}{2}} |\boldsymbol{\Sigma}_i|^{\frac{1}{2}}} \times \exp\left\{-\frac{1}{2}\left(\boldsymbol{\theta}_i^T \boldsymbol{\psi}_t - \boldsymbol{y}_t\right)^T \boldsymbol{\Sigma}_i^{-1} \left(\boldsymbol{\theta}_i^T \boldsymbol{\psi}_t - \boldsymbol{y}_t\right)\right\}. \tag{25}$$

Now, by applying the same procedure for the single input–output case, λ', which maximizes the $\tilde{Q}(\lambda, \lambda')$, can be obtained as follows:

$$\pi_i' = \frac{\sum_{l=1}^{L} k_l \pi_i b_i(\boldsymbol{o}_{l,0}) \beta(l,i,0)}{\sum_{i=0}^{N}\sum_{l=1}^{L} k_l \pi_i b_i(\boldsymbol{o}_{l,0}) \beta(l,i,0)} \tag{26}$$

$$a_{ij}' = \frac{\sum_{t=1}^{T}\sum_{l=1}^{L} k_l \alpha(l,i,t-1) a_{ij} b_j(\boldsymbol{o}_{l,t}) \beta(l,j,t)}{\sum_{j=0}^{N}\sum_{t=1}^{T}\sum_{l=1}^{L} k_l \alpha(l,i,t-1) a_{ij} b_j(\boldsymbol{o}_{l,t}) \beta(l,j,t)} \tag{27}$$

$$\boldsymbol{\theta}_i'^T = \left\{\sum_{t=0}^{T}\sum_{l=1}^{L} k_l \boldsymbol{y}_{l,t} \boldsymbol{\psi}_{l,t}^T \alpha(l,i,t)\beta(l,i,t)\right\} \times \left\{\sum_{t=0}^{T}\sum_{l=1}^{L} k_l \boldsymbol{\psi}_{l,t} \boldsymbol{\psi}_{l,t}^T \alpha(l,i,t)\beta(l,i,t)\right\}^{-1} \tag{28}$$

$$\boldsymbol{\Sigma}_i' = \frac{\sum_{t=0}^{T}\sum_{l=1}^{L} k_l \boldsymbol{z}_{l,i,t} \boldsymbol{z}_{l,i,t}^T \alpha(l,i,t)\beta(l,i,t)}{\sum_{t=0}^{T}\sum_{l=1}^{L} k_l \alpha(l,i,t)\beta(l,i,t)} \tag{29}$$

where $\boldsymbol{z}_{l,i,t} = \boldsymbol{\theta}_i'^T \boldsymbol{\psi}_{l,t} - \boldsymbol{y}_{l,t}$.

IV. Application to Driver Behavior Modeling

A. Configuration of the DS

The configuration and projected image of the developed DS [7] based on the CAVE are shown in Fig. 2(a) and (b). The display unit in the CAVE system provides a 3-D virtual environment, and it is controlled by ONYX2. The display program is developed using the CAVE library and the Performer. The cockpit is built by installing a handle, an accelerator, and a brake in the CAVE system. The information regarding the driver's output to the steering, accelerator, and brake is transferred to a PC via a USB terminal. The vehicle position and motion are calculated based on these inputs and the vehicle dynamics implemented on the PC using the CarSim software. The results of the calculation are transferred to ONYX2 via the Internet (TCP/IP), and a 3-D visual image based on the position and motion of the vehicle is displayed.

Although our DS cannot provide any real motion, it enables us to capture much more reliable information about the driving behavior, owing to its stereoscopic immersive vision, than the other standard DS. Moreover, in the Carsim software, the parameters of real vehicle were used. We have already verified that the behavior observed through this DS has similar characteristics with the ones observed in real driving situation [7].

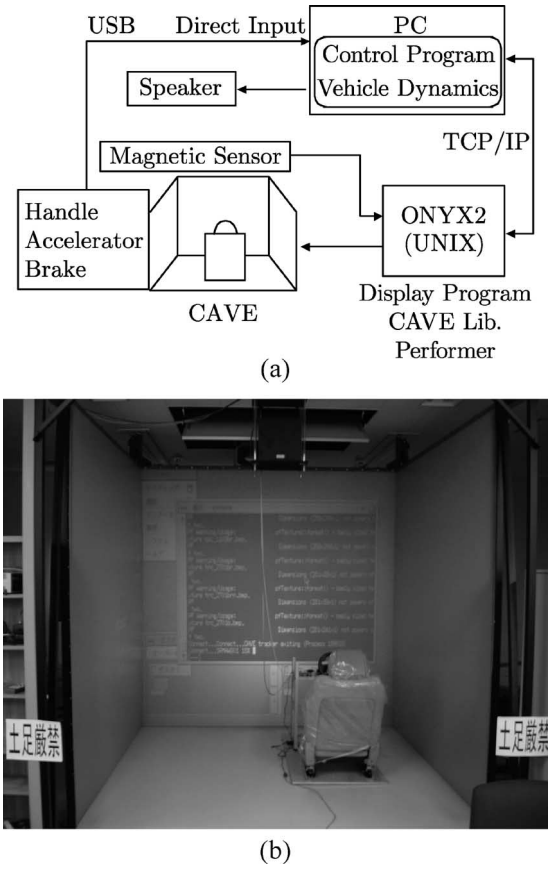


Fig. 2. Developed DS based on the CAVE. (a) Configuration of the DS. (b) CAVE system.

B. Data Acquisition in Collision Avoidance

1) Experimental Environment and Condition: In this paper, we focus on the driver's collision avoidance behavior at the instance when the preceding vehicle is brought to a sudden halt and the examinee is looking away from the road. In order to model the driver's collision avoidance behavior, the following sensory information is captured as the inputs:

1) range between cars $(u_{1,t})$;
2) range rate $(u_{2,t})$;
3) lateral displacement between cars $(u_{3,t})$.

The output of the driver is also specified as steering amount (y_t).

The selection of sensory information was based on our empirical knowledge. One may argue that braking operation should be taken into account in the model. However, as sufficient distance was maintained between the vehicles, the influence of braking is less than that of steering. In fact, in our experimental setup, the examinees were directed to drive the car while maintaining an almost constant distance (about 25 m) between their own vehicles and the preceding truck. This range was determined based on a parameter referred to as time to collision (TTC); this parameter can be defined as follows:

$$\text{TTC} = \frac{\text{range between cars } (u_{1,t})}{\text{velocity of vehicle}}.$$

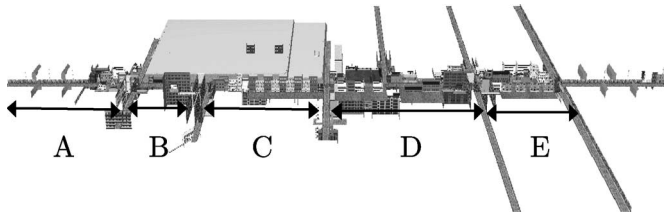


Fig. 3. Road environment for the experiments.

Fig. 4. Scene at the start point.

In this case, the TTC was approximately 2 s (the velocity of the car was set to about 50 km/h), and this value is typically recognized to be the "sufficient range" to avoid a collision without braking or with a small amount of braking. Therefore, we did not consider the braking operation as driver output. If we consider the situation with a smaller TTC, it is obvious that the braking operation must be included in the model explicitly. It may not be easy to provide a clear understanding of the model obtained by including both the steering and braking operations. However, from the viewpoint of computation, this can be realized by simply applying the update law (26) to (29) instead of (20) to (23) to the measured data.

In the considered task, the regressor vector in (4) was specified as follows:

$$\boldsymbol{\psi}_t = (u_{1,t-m}, u_{2,t-m}, u_{3,t-m})^T \tag{30}$$

where m represents the time delay between the perception and the operation in the human behavior [18]. In the analyses shown in the subsequent sections, m was set to be 12. The sampling interval of 16 ms implies that the specified time delay is about 200 ms, which is considered to be a reasonable value in the field of human engineering.

The configuration of the experimental environment is shown in Fig. 3. The road environment in this paper comprises four intersections and two T-type junctions. It was 1 km long, 7 m wide, and the pedestrian walk was 1.5 m wide. The friction coefficient of the road was set to be 0.8. The scene around the start point is shown in Fig. 4. The vehicle moves from the left to the right in Fig. 3. The scene at 940 m is identical to that at the start point. The environment appearing after 940 m is identical to that appearing after the start point. Therefore, to the driver, this virtual environment appears to be a straight continuous road. Only two vehicles exist in this environment.

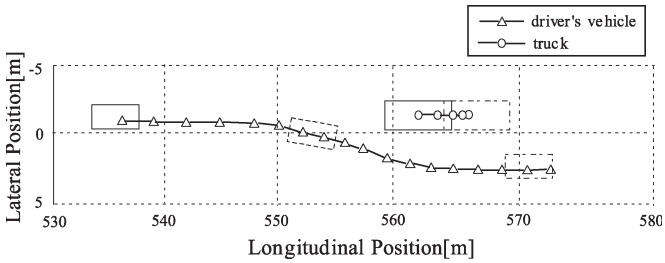


Fig. 5. Typical relationship between the positions of the two vehicles in the experiment.

TABLE I
PERSONAL INFORMATION OF THE DRIVERS

Examinee	Age [Years old]	Mileage per year [km]	Driving career [years]
A	23	1000	4
B	23	1000	3
C	22	1000	4
D	22	8000	3
E	22	5000	4
F	29	20000	10
G	23	5000	4

One of them is a sedan that is driven by the examinee. The other is a big truck that is driven in front of the examinee's vehicle by an operator. The car used in the simulator has a 3000-cc engine and an antilock breaking system. The truck in front of the examinee's vehicle travels at a constant speed of 50 km/h. The maximum deceleration of the truck was set to be 7 m/s^2.

C. Experimental Procedure

The examinee drives the car while maintaining a constant distance between his/her own car and the preceding truck. The operator sets red or green parked vehicles on the right side at the intersections. The examinees are supposed to take a look to the right at each intersection; then, they answer what color the parked vehicle was and/or whether there were any parked vehicles. At one of the intersections, when the examinee looks to the right, the preceding truck is brought to a sudden halt with maximum deceleration. Then, the collision avoidance behavior of the examinee is measured. The typical relationship between the positions of the examinee's vehicle and the truck in the experiment is shown in Fig. 5.

V. MEASURED DATA AND MODELING RESULTS

A. Measured Data

Based on the setup described in the previous section, seven drivers (A, B, C, D, E, F, and G) conducted the driving experiment in the virtual environment. The personal information of all the seven drivers is listed in Table I. First, four trial data sets of the collision avoidance behavior of each driver, which are characterized by the profile between the beginning and end of the steering operation, are measured and analyzed.

The measured profiles of the sensory information and steering operation for the seven drivers are shown in Figs. 6–9. The Ai, Bi, and Ci in each figure denote the corresponding driver's trial index. In Figs. 6–9, we can see that each driver has unique characteristics. In particular, large variations are observed between driver behaviors with respect to the lateral displacement between cars and steering profiles.

B. Parameter Estimation

We adopted the left-to-right model with three states as the structure of the SS-ARX model, as shown in Fig. 10. This type of model is expected to work well for the analysis of the time series data [15]. Moreover, the selection of the number of modes was determined from the results of our previous study based on a deterministic HDS model [8].

The parameters of the SS-ARX models corresponding to each driver's behavior are estimated by using the following profiles: (A1, A2), (B1, B2), (C1, C2), (D1, D2), (E1, E2), (F1, F2), and (G1, G2). The remaining profiles are used for the verification of recognition performance in the next section. Prior to the estimation, all measured variables were normalized.

In the EM algorithm, the locally optimal solutions of the parameters are obtained by updating the parameters to improve their log likelihood values. Therefore, the proposed strategy cannot always find the globally optimal parameters, and the accuracy of parameter estimation depends on the initial values of the parameters specified in the EM algorithm. In order to overcome this problem, we have tested several initial values of the parameters and determined the optimal parameters by comparing their resulting log likelihood values. The obtained parameters of the corresponding SS-ARX model are listed in Table II.

Next, in order to verify the accuracy of the proposed parameter estimation procedure, the measured and reproduced steering profiles of driver B are compared, as shown in Fig. 11. The reproduced steering profiles, as shown in Fig. 11, are generated by regarding the SS-ARX model as a deterministic switched ARX model specified by $\boldsymbol{\theta}_i$ and estimated switching points. The estimated switching points are obtained by applying the Viterbi algorithm to calculate the following two variables recursively [15] (note that the number of discrete states is three; the subsequently obtained two switching points for each profile are shown in Fig. 11):

$$\delta(l,i,t) = \max_{0\le j\le N} \{\delta(l,j,t-1)a_{ji}b_i(o_{l,t})\} \tag{31}$$

$$\gamma(l,i,t) = \arg\max_{0\le j\le N} \{\delta(l,j,t-1)a_{ji}b_i(o_{l,t})\} \tag{32}$$

where $\gamma(l,i,t)$ represents the most likely state S_i at time t. This algorithm enables us to estimate the most likely switching points by calculating (31) and (32). The two vertical lines in Fig. 11 in each profile represent the most likely switching points that are estimated using the Viterbi algorithm. This figure shows that both the profiles agree well with each other. This result validates the proposed parameter estimation procedure.

One may argue that the switching points must be associated with the measured signals, i.e., the switching condition expressed as the function of measured signals must be found. We have addressed this problem in a previous study [8]. The strategy developed in [8], however, requires a large amount of computation time and does not meet our requirement of real-time behavior recognition. The strategy proposed in this paper can provide an alternative technique for estimating the switching condition expressed as the function of the measured

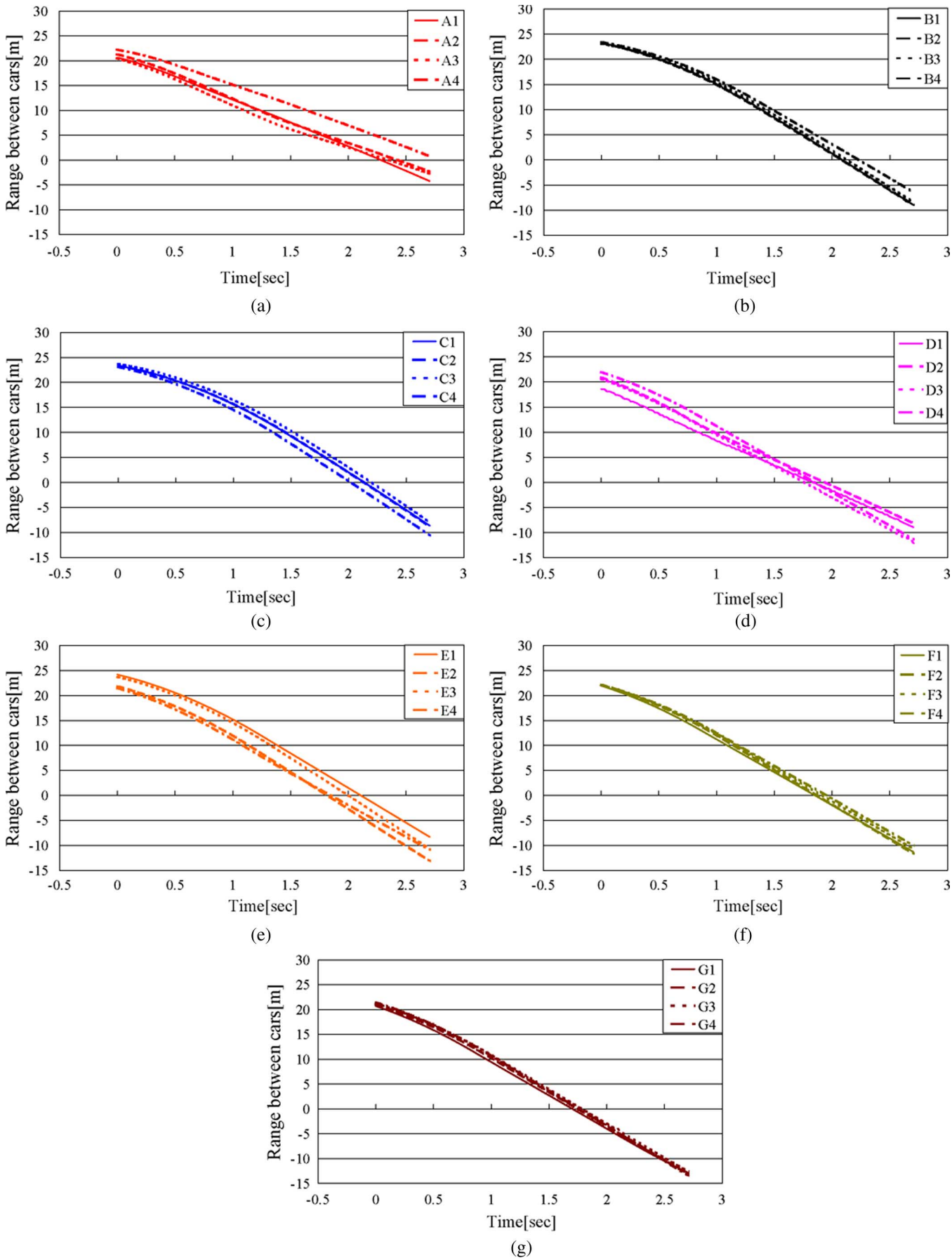


Fig. 6. Range between the cars of the seven drivers.

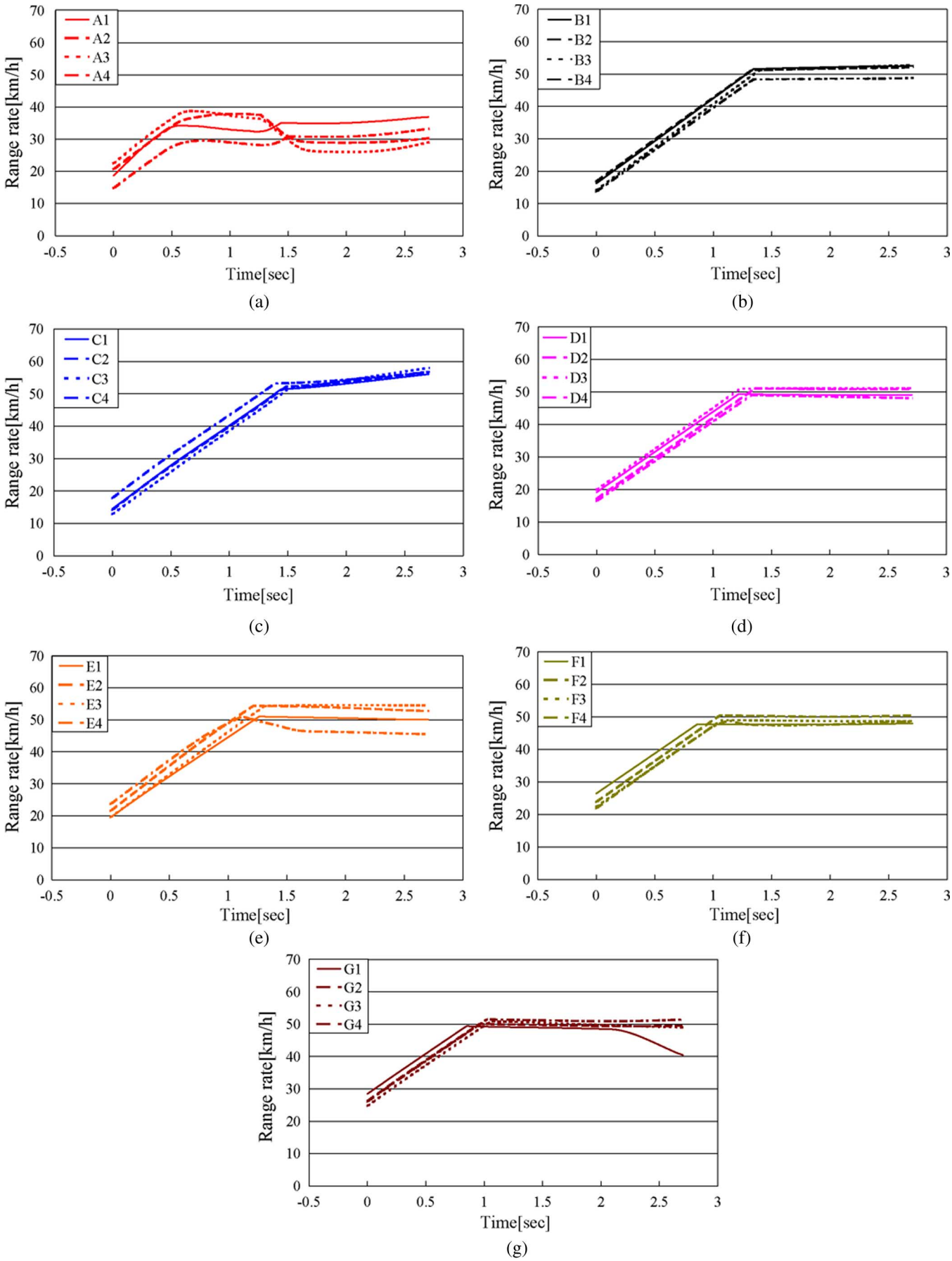


Fig. 7. Range rate of the seven drivers.

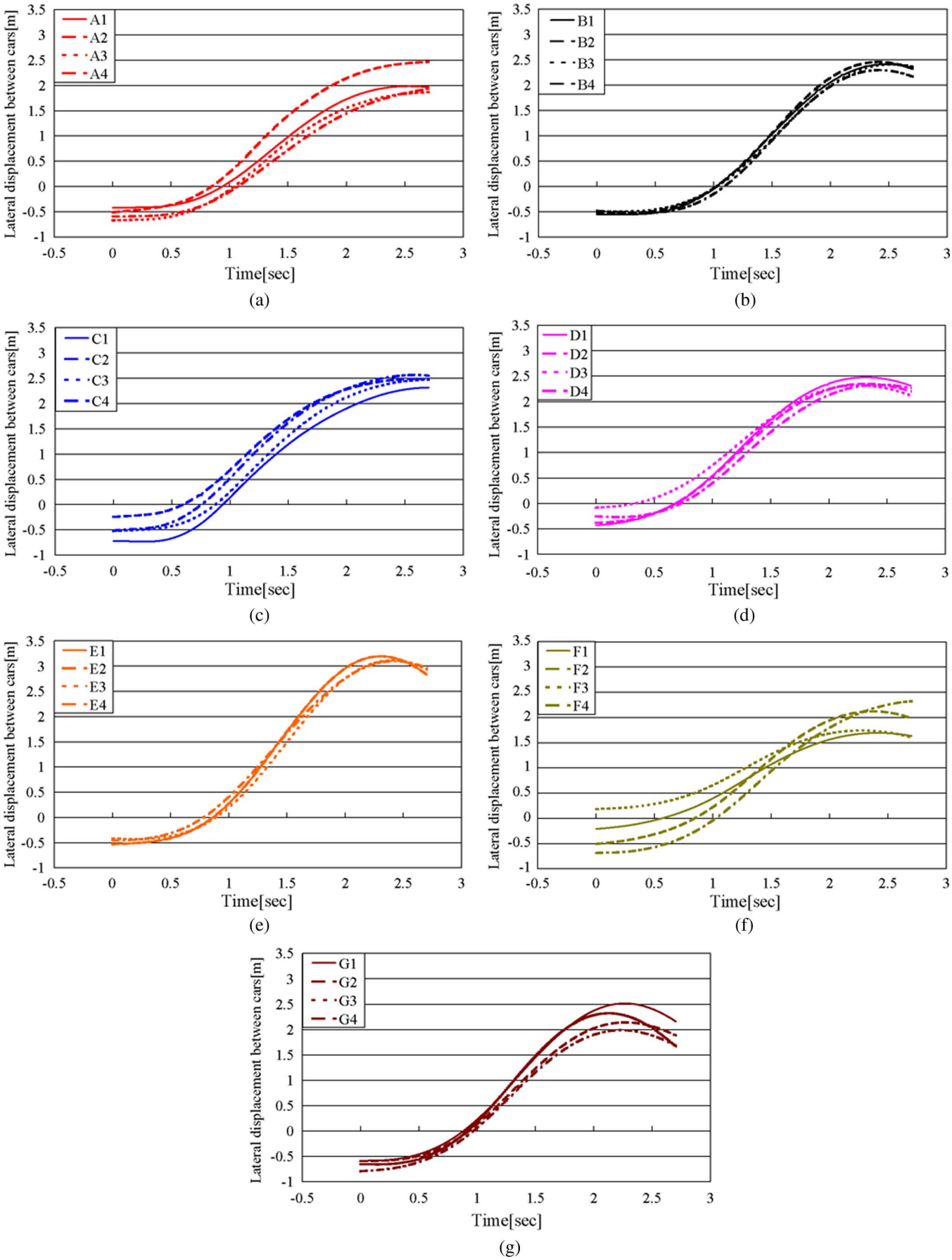


Fig. 8. Lateral displacement between the cars of the seven drivers.

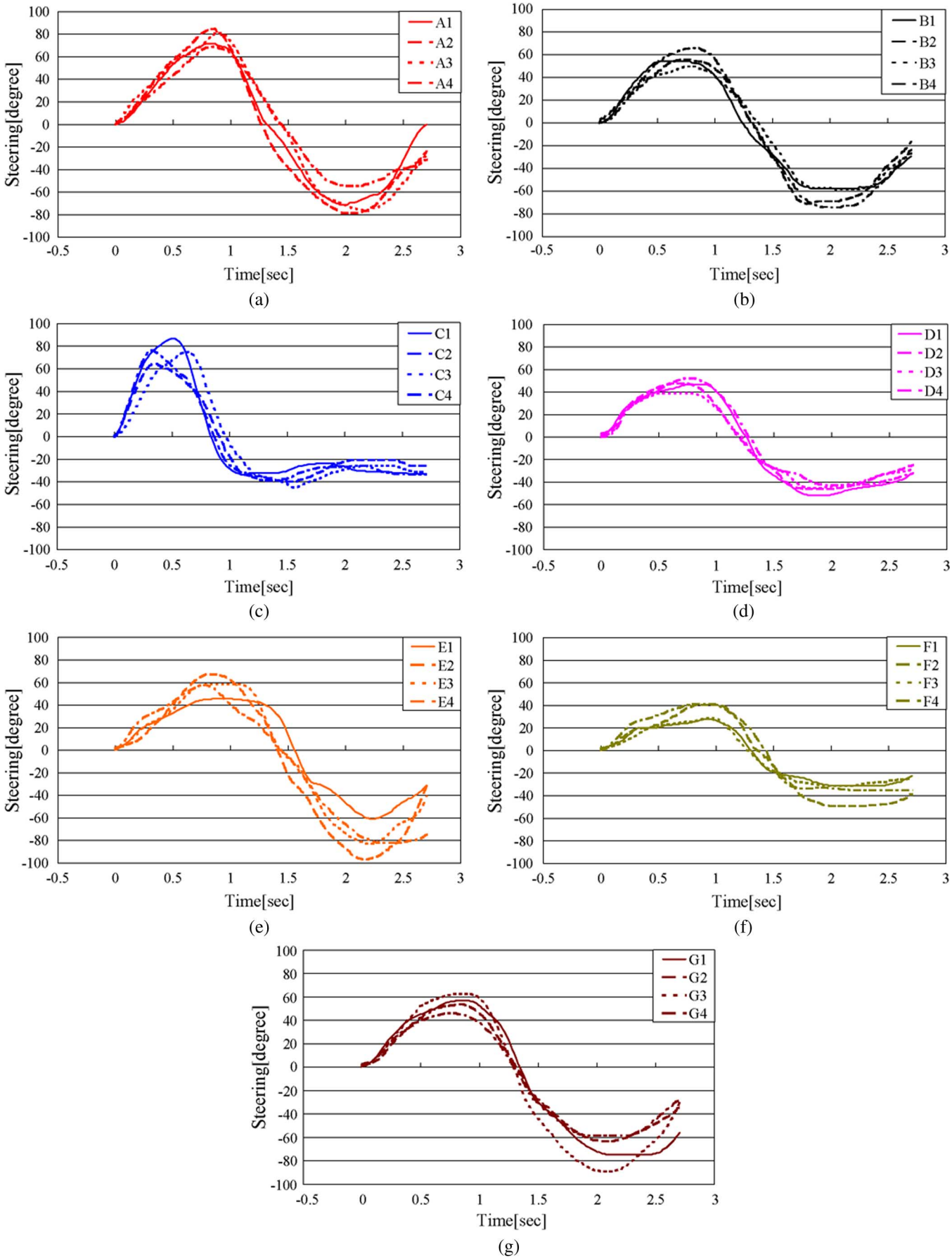


Fig. 9. Steering profiles of the seven drivers.

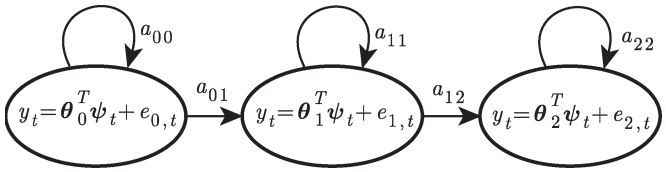


Fig. 10. Left-to-right model with three states.

signals by collecting the several estimated switching points and investigating the relationship between them and the measured signals. Also, it seems possible to develop a stochastic switched model with a more complex switching mechanism (i.e., by considering mode transition probabilities to be the functions of measured signals). These issues are currently being investigated and will be reported in our future study.

The investigation of the identified parameters of each driver listed in Table II could yield interesting results. θ_i in Table II can be considered to represent the "primitive driving skill" of each driver in each mode. Broadly speaking, modes 0, 1, and 2 can be considered to express "avoidance," "overtaking," and "recovery," respectively. Although the estimated θ_i varies for each driver, a few common characteristics are observed. These are listed below.

1) Comparison among modes: Since all measured signals are normalized, the magnitude of $\theta_{i,j}$ [$\theta_{i,j}$ denotes the jth element of θ_i, i.e., j is a type of sensory information ($j = 1, 2, 3$)] expresses the importance of the corresponding sensory information for each driver. In the case of drivers A, B, C, D, and E, the magnitudes of the second and third elements in mode 1, $|\theta_{1,2}|$ and $|\theta_{1,3}|$ are smaller than that in mode 0, $|\theta_{0,2}|$ and $|\theta_{0,3}|$, respectively (also, in the case of driver G, $|\theta_{1,2}|$ is smaller than $|\theta_{0,2}|$). This implies that the range rate and lateral displacement are less significant in mode 1 as compared to mode 0. Also, in the case of drivers A, B, C, F, and G, the magnitude of the first element in mode 2 $|\theta_{2,1}|$ is smaller than that in mode 1 $|\theta_{1,1}|$. This implies that the range is less significant in mode 2 as compared to mode 1. Thus, essentially, the driving behavior can be comprehended by regarding the switching of control skills as follows: First, the range rate and/or lateral displacement oriented skill (mode 0) is adopted. Then, the driver switches to range-oriented skill (mode 1) and finally switches back to the range-rate-and/or lateral-displacement-oriented skill (mode 2) as the mode progresses.
2) Comparison among elements: In the case of drivers B, C, D, E, and F, the magnitude of the third element $|\theta_{i,3}|$ is greater than that of the first element $|\theta_{i,1}|$ (also, $|\theta_{1,3}|$ and $|\theta_{2,3}|$ are greater than $|\theta_{1,1}|$ and $|\theta_{2,1}|$, respectively, in the driver G). These results indicate that the drivers tend to lend more importance to lateral information rather than the longitudinal information throughout the behavior.
3) Comparison of signs: The signs of each element in mode 0 are the same: negative, positive, and negative. On the other hand, the signs in modes 1 and 2 are not consistent. Although we were unable to establish concrete significance of these signs, this phenomenon may represent a type of diversity in driving skills in each mode.

TABLE II
ESTIMATED PARAMETERS

SS-ARX model for driver A

Parameters in ARX model	Variances
$\theta_0 = (-1.806131, 2.948160, -2.383352)$	$\sigma_0 = 0.478417$
$\theta_1 = (3.883612, -1.609900, -1.561564)$	$\sigma_1 = 0.419417$
$\theta_2 = (-3.257210, 3.675345, -2.637144)$	$\sigma_2 = 0.673647$

A matrix

$$a_{ij} = \begin{bmatrix} 0.986560 & 0.013440 & 0.000000 \\ 0.000000 & 0.889899 & 0.110101 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$$

SS-ARX model for driver B

Parameters in ARX model	Variances
$\theta_0 = (-1.475241, 1.815917, -4.479522)$	$\sigma_0 = 0.075641$
$\theta_1 = (-1.614410, 1.253208, -2.181464)$	$\sigma_1 = 0.127688$
$\theta_2 = (-1.208364, 0.810375, -1.678744)$	$\sigma_2 = 0.130666$

A matrix

$$a_{ij} = \begin{bmatrix} 0.988574 & 0.011426 & 0.000000 \\ 0.000000 & 0.978157 & 0.021843 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$$

SS-ARX model for driver C

Parameters in ARX model	Variances
$\theta_0 = (-2.775854, 2.487336, -8.077736)$	$\sigma_0 = 1.725615$
$\theta_1 = (1.439090, -2.089830, 2.009599)$	$\sigma_1 = 0.078102$
$\theta_2 = (0.020965, -0.212446, -0.189831)$	$\sigma_2 = 0.073008$

A matrix

$$a_{ij} = \begin{bmatrix} 0.987346 & 0.012654 & 0.000000 \\ 0.000000 & 0.987649 & 0.012351 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$$

SS-ARX model for driver D

Parameters in ARX model	Variances
$\theta_0 = (-1.411337, 2.804954, -3.760746)$	$\sigma_0 = 0.166029$
$\theta_1 = (-0.000116, 0.997163, -2.103864)$	$\sigma_1 = 0.105283$
$\theta_2 = (1.131279, 1.882750, -1.934112)$	$\sigma_2 = 0.096696$

A matrix

$$a_{ij} = \begin{bmatrix} 0.978749 & 0.021251 & 0.000000 \\ 0.000000 & 0.980625 & 0.019375 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$$

SS-ARX model for driver E

Parameters in ARX model	Variances
$\theta_0 = (-1.615877, 2.116206, -2.769467)$	$\sigma_0 = 0.221731$
$\theta_1 = (-0.876601, 1.554333, -2.221077)$	$\sigma_1 = 0.262589$
$\theta_2 = (-1.199592, 1.931455, -2.575547)$	$\sigma_2 = 0.173846$

A matrix

$$a_{ij} = \begin{bmatrix} 0.987845 & 0.012155 & 0.000000 \\ 0.000000 & 0.972538 & 0.027462 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$$

SS-ARX model for driver F

Parameters in ARX model	Variances
$\theta_0 = (-0.726869, 1.109808, -1.188360)$	$\sigma_0 = 0.560780$
$\theta_1 = (-1.770390, 1.947067, -3.207973)$	$\sigma_1 = 0.042469$
$\theta_2 = (-0.273634, 0.254668, -1.052316)$	$\sigma_2 = 0.059613$

A matrix

$$a_{ij} = \begin{bmatrix} 0.981439 & 0.018561 & 0.000000 \\ 0.000000 & 0.988588 & 0.011412 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$$

SS-ARX model for driver G

Parameters in ARX model	Variances
$\theta_0 = (-2.653897, 4.489167, -1.278389)$	$\sigma_0 = 0.827850$
$\theta_1 = (-0.111077, 2.629970, -2.698000)$	$\sigma_1 = 0.220705$
$\theta_2 = (-0.008396, 1.925053, -2.064376)$	$\sigma_2 = 0.036096$

A matrix

$$a_{ij} = \begin{bmatrix} 0.990359 & 0.009641 & 0.000000 \\ 0.000000 & 0.984540 & 0.015460 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$$

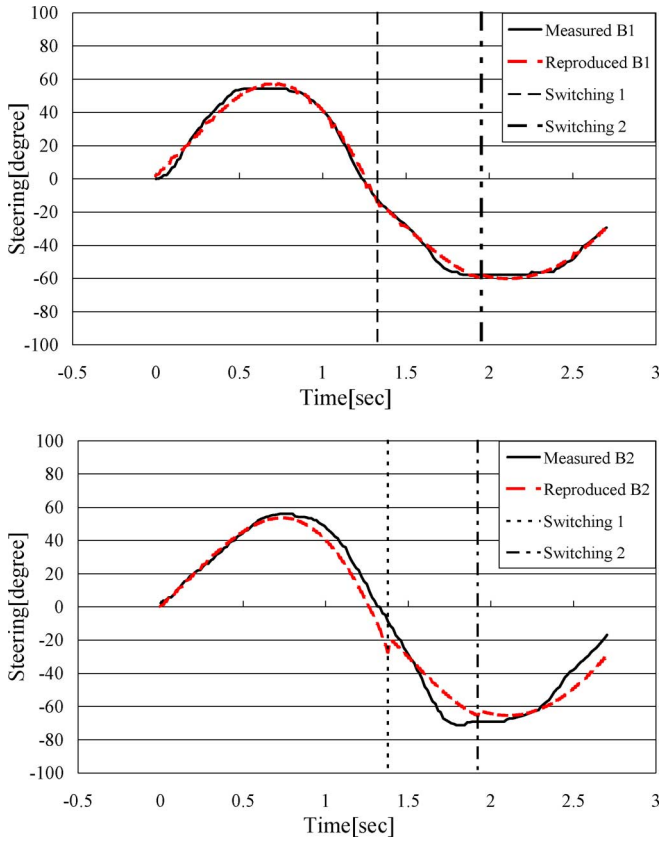


Fig. 11. Estimated switching points and steering profiles of driver B (measured and reproduced).

It is not easy to obtain a global understanding of these parameters. In fact, it is observed that several different characteristics exist among drivers. However, these differences motivate us to develop a behavior recognizer that is based on the SS-ARX model, as investigated in the next section. The analyses shown in this section reveal the advantage of using the SS-ARX model, which cannot be realized using a standard HMM.

VI. Driver Recognition and Discussion

The obtained SS-ARX model for each driver is used as a "behavior recognizer." The objective of this recognition is to identify the driver by calculating the log likelihood values of the measured profiles over the SS-ARX model.

The log likelihood values of the 14 measured profiles [(A3, A4), (B3, B4), (C3, C4), (D3, D4), (E3, E4), (F3, F4), and (G3, G4)] over the seven SS-ARX models obtained in the previous section are calculated and listed in Table III. In this table, SS-ARX ($*1$-$*2$) denotes the SS-ARX model estimated by applying the corresponding driver's data profile. Note that profiles $*3$ and $*4$ (unknown profiles) were not used for parameter estimation. The greatest likelihood values for profiles $*3$ and $*4$ are underlined. We can see that almost all maximum likelihood values originate from the corresponding driver's SS-ARX model, except for driver A. This implies that the rate of successful recognition is 12/14. Furthermore, even in the case of unsuccessful recognition (driver A), it might be possible to correct the result by comparing the estimated parameter θ

TABLE III
Recognition Results

	$A3$	$A4$
SS-ARXmodel	Log likelihood	Log likelihood
SS-ARX(A1-A2)	-844.393963	-822.843722
SS-ARX(B1-B2)	-1496.72515	-1540.50566
SS-ARX(C1-C2)	-933.49619	-948.062464
SS-ARX(D1-D2)	-1163.937829	-908.367879
SS-ARX(E1-E2)	-874.285055	-869.124728
SS-ARX(F1-F2)	-879.361149	-878.527970
SS-ARX(G1-G2)	-831.703758	-818.87472
	$B3$	$B4$
SS-ARXmodel	Log likelihood	Log likelihood
SS-ARX(A1-A2)	-935.370440	-868.925945
SS-ARX(B1-B2)	-780.431040	-777.993461
SS-ARX(C1-C2)	-901.905799	-926.085292
SS-ARX(D1-D2)	-1004.802049	-1024.875554
SS-ARX(E1-E2)	-832.931733	-842.592334
SS-ARX(F1-F2)	-886.340131	-883.962263
SS-ARX(G1-G2)	-1073.899792	-1048.044208
	$C3$	$C4$
SS-ARXmodel	Log likelihood	Log likelihood
SS-ARX(A1-A2)	-913.869748	-958.360681
SS-ARX(B1-B2)	-1604.326991	-1415.680406
SS-ARX(C1-C2)	-782.308665	-794.411785
SS-ARX(D1-D2)	-1023.656168	-1095.030933
SS-ARX(E1-E2)	-860.092267	-855.754494
SS-ARX(F1-F2)	-880.981143	-880.549985
SS-ARX(G1-G2)	-915.124784	-928.807654
	$D3$	$D4$
SS-ARXmodel	Log likelihood	Log likelihood
SS-ARX(A1-A2)	-1439.583913	-1431.475072
SS-ARX(B1-B2)	-2927.403174	-2936.204836
SS-ARX(C1-C2)	-812.788815	-823.574500
SS-ARX(D1-D2)	-784.707235	-763.646089
SS-ARX(E1-E2)	-822.654834	-825.622672
SS-ARX(F1-F2)	-885.079074	-899.863200
SS-ARX(G1-G2)	-967.063878	-1019.214726
	$E3$	$E4$
SS-ARXmodel	Log likelihood	Log likelihood
SS-ARX(A1-A2)	-953.757583	-908.632927
SS-ARX(B1-B2)	-1406.394738	-1185.842025
SS-ARX(C1-C2)	-1116.019478	-1152.620423
SS-ARX(D1-D2)	-1017.709337	-1208.381576
SS-ARX(E1-E2)	-838.853416	-844.593149
SS-ARX(F1-F2)	-881.133588	-849.292104
SS-ARX(G1-G2)	-904.641383	-854.481182
	$F3$	$F4$
SS-ARXmodel	Log likelihood	Log likelihood
SS-ARX(A1-A2)	-1093.530389	-919.910273
SS-ARX(B1-B2)	-2276.448465	-886.560424
SS-ARX(C1-C2)	-806.855871	-974.816573
SS-ARX(D1-D2)	-838.350542	-2172.188937
SS-ARX(E1-E2)	-838.232064	-808.328295
SS-ARX(F1-F2)	-788.455825	-792.388328
SS-ARX(G1-G2)	-924.622051	-913.368778
	$G3$	$G4$
SS-ARXmodel	Log likelihood	Log likelihood
SS-ARX(A1-A2)	-1046.646399	-973.22811
SS-ARX(B1-B2)	-2074.580191	-2144.572605
SS-ARX(C1-C2)	-1030.087645	-999.644306
SS-ARX(D1-D2)	-814.073056	-857.294936
SS-ARX(E1-E2)	-894.057906	-878.926715
SS-ARX(F1-F2)	-1800.755561	-1299.65956
SS-ARX(G1-G2)	-786.553466	-839.169413

at each mode. These results show that the SS-ARX model holds remarkable potential to function as a recognizer of human driving behavior.

Finally, the comparison with other typical strategies is stated below. Some of the typical alternative concepts that can realize behavior modeling are NNs and standard HMM. Although the quantitative comparison of the performances of NN and HMM could yield interesting results, it is not easy to realize this comparison as there exist several variations in both NN and HMM. In order to execute a fair comparison, very carefully determined structure and parameter settings are required in each model. The biggest advantage of the proposed SS-ARX model, however, can be clearly stated in a qualitative manner as follows: When NN is applied to behavior modeling, the significance of the obtained parameters is usually unintelligible. Furthermore, the standard HMM cannot provide information regarding the primitive skills of human behavior. On the other hand, the SS-ARX model enables us to explicitly capture the parameters of primitive skills, together with their switching probabilities. Thus, we can obtain more information to recognize the behavior in addition to the likelihood values of the measured data over the model.

VII. Conclusion

In this paper, we have developed a new technique for the modeling and recognition of human driving behavior based on the SS-ARX model. First, a parameter estimation algorithm for the SS-ARX model has been derived based on the EM algorithm. This was achieved by extending the parameter estimation technique for standard HMM. Second, the developed parameter estimation algorithm was applied to the measured driving data with focus on the collision avoidance behavior. The driving data were collected using a 3-D DS based on the CAVE, which provides stereoscopic immersive vision. Then, the parameter set for each driver was obtained, and certain driving characteristics were identified from the viewpoint of the switched control mechanism. Finally, the performance of the SS-ARX model as a behavior recognizer was examined. The obtained results showed that the SS-ARX model holds remarkable potential to function as a behavior recognizer.

Acknowledgment

The authors would like to thank all of the researchers involved at the Space Robotic Center for their helpful suggestions.

References

[1] D. McRuer and D. Weir, "Theory of manual vehicular control," *Ergonomics*, vol. 12, no. 4, pp. 599–633, Jul. 1969.

[2] C. MacAdam, "Application of an optimal preview control for simulation of closed-loop automobile driving," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-11, no. 6, pp. 393–399, Jun. 1981.

[3] A. Modjtahedzadeh and R. Hess, "A model of driver steering control behavior for use in assessing vehicle handling qualities," *Trans. ASME, J. Dyn. Syst. Meas. Control*, vol. 15, pp. 456–464, Sep. 1993.

[4] T. Pilutti and A. G. Ulsoy, "Identification of driver state for lane-keeping tasks," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 29, no. 5, pp. 486–502, Sep. 1999.

[5] J. Sjoberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P. Y. Glorenner, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: A unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, Dec. 1995.

[6] K. S. Narendra and K. Pathasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Netw.*, vol. 1, no. 1, pp. 4–27, Mar. 1990.

[7] J. H. Kim, Y. Matsui, S. Hayakawa, T. Suzuki, S. Okuma, and N. Tsuchida, "Acquisition and modeling of driving skills by using three dimensional driving simulator," *IEICE Trans. Fundam.*, vol. E88-A, no. 3, pp. 770–778, Mar. 2005.

[8] J. H. Kim, S. Hayakawa, T. Suzuki, K. Hayashi, S. Okuma, N. Tsuchida, M. Shimizu, and S. Kido, "Modeling of driver's collision avoidance maneuver based on controller switching model," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 6, pp. 1131–1143, Dec. 2005.

[9] T. Suzuki, S. Yamada, S. Hayakawa, N. Tsuchida, T. Tsuda, and H. Fujinami, "Modeling of drivers collision avoidance behavior based on hybrid system model—An approach with data clustering," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2005, pp. 3817–3822.

[10] O. Maler and A. Puneli, *Lecture Notes in Computer Science 2623; Hybrid Systems: Computation and Control*. New York: Springer-Verlag, 2003.

[11] R. Alur and G. J. Pappas, *Lecture Notes in Computer Science 2993; Hybrid Systems: Computation and Control*. New York: Springer-Verlag, 2004.

[12] M. Morari and L. Thiele, *Lecture Notes in Computer Science 3414; Hybrid Systems: Computation and Control*. New York: Springer-Verlag, 2005.

[13] A. Bemporad, J. Roll, and L. Ljung, "Identification of hybrid systems via mixed-integer programming," in *Proc. 40th IEEE Conf. Decision Control*, 2001, pp. 786–792.

[14] G. Ferrari-Trecate, M. Musellic, D. Liberatid, and M. Morari, "A clustering technique for the identification of piecewise affine system," *Automatica*, vol. 39, no. 2, pp. 205–217, Feb. 2003.

[15] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

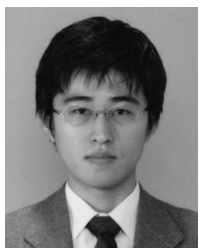
[16] M. C. Nechyba and Y. Xu, "Human control strategy: Abstraction, verification and replication," *IEEE Control Syst. Mag.*, vol. 17, no. 5, pp. 48–61, Oct. 1997.

[17] D. Mitrovic, "Reliable method for driving events recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 198–205, Jun. 2005.

[18] S. Inaba, *Traffic Accident and Human-Factors Engineering*. San Antonio, TX: Corona, 1988. (in Japanese).

Shogo Sekizawa was born in Hokkaido, Japan, in 1982. He received the B.S. degree in mechanical and aerospace engineering from Nagoya University, Nagoya, Japan, in 2005.

He is currently with the Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University. His research interests include the modeling of drivers' behavior and the design of a driving assist system.



Shinkichi Inagaki was born in Mie, Japan, in 1975. He received the Ph.D. degree in precision engineering from the University of Tokyo, Tokyo, Japan, in 2003.

Since 2003, he has been an Assistant Professor with the Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Nagoya, Japan. His research interests include autonomous decentralized systems, robotics, and hybrid dynamical systems.

Dr. Inagaki is a member of the Society of Instrument and Control Engineers, Robotics Society of Japan, and Japan Society of Mechanical Engineers.

Tatsuya Suzuki (S'89–M'91) was born in Aichi, Japan, in 1964. He received the B.E., M.E., and Ph.D. degrees in electronic mechanical engineering from Nagoya University, Nagoya, Japan, in 1986, 1988, and 1991, respectively.

From 1998 to 1999, he was a Visiting Researcher with the Mechanical Engineering Department, University of California, Berkeley. Currently, he is a Professor with the Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University. His current research interests include hybrid dynamical systems and human behavior analysis.

Dr. Suzuki is a member of the Institute of Electrical Engineers of Japan (IEEJ), Society of Instrument and Control Engineers, Institute of Systems, Control and Information Engineers, Robotics Society of Japan, and Japan Society of Mechanical Engineers. He received the paper award from IEEJ in 1995.

Soichiro Hayakawa was born in Aichi, Japan, in 1968. He received the Ph.D. degree in electrical engineering from Nagoya University, Nagoya, Japan, in 1996.

Since 1996, he has been an Assistant Professor with Toyota Technological Institute, Nagoya. His research interests include motion control systems, robotics, virtual reality, and hybrid dynamical systems.

Dr. Hayakawa is a member of the Institute of Electrical Engineers of Japan, Society of Instrument and Control Engineers, Institute of Electronics, Information, and Communication Engineers, and Robotics Society of Japan.

Nuio Tsuchida was born in Aichi, Japan, in 1943. He received the Ph.D. degree in electrical and electronics engineering from Nagoya University, Nagoya, Japan, in 1972.

From 1972 to 1982, he was an Assistant Professor with the Department of Electrical and Electronics Engineering, Nagoya University. Since 1992, he has been a Professor with Toyota Technological Institute, Nagoya. His research interests include actuators, sensors, and robotics.

Dr. Tsuchida is a member of the Institute of Electrical Engineers of Japan, Institute of Electronics, Information, and Communication Engineers, and Institute of Electrostatics Japan.

Taishi Tsuda received the B.S. and M.S. degrees in mechanical engineering from the University of Tokyo, Tokyo, Japan, in 1998 and 2000, respectively.

He is currently engaged in the development of a driving support system in the Development Department No. 3, Integrated System Engineering Division, Vehicle Engineering Group, Toyota Motor Corporation, Toyota, Japan. His research interests also include vehicle control.

Mr. Tsuda is a member of the Society of Automotive Engineers of Japan and Japan Society for Precision Engineering.

Hiroaki Fujinami received the B.S. degree in mechanical engineering from Tokyo University of Science, Tokyo, Japan, in 1987.

He is currently the Group Manager of the Development Department No. 3, Integrated System Engineering Division, Vehicle Engineering Group, Toyota Motor Corporation, Toyota, Japan. His interests include vehicle driving support systems and control.