

Modeling and Recognition of Human Driving Behavior based on Stochastic Switched ARX model

Tatsuya Suzuki, Shogo Sekizawa, Shinkichi Inagaki, Soichiro Hayakawa,
Nuio Tsuchida, Taishi Tsuda, Hiroaki Fujinami

Abstract—This paper presents a development of the modeling of the human driving behavior based on the expression as Stochastic Switched ARX model (SS-ARX) focusing on the driver's collision avoidance behavior. First, the parameter estimation technique for the SS-ARX model is introduced based on the EM algorithm. This can be achieved by extending the parameter estimation technique for conventional Hidden Markov Model (HMM). Second, the parameter estimation technique is applied to the collected driving data, and find parameter set for each driving data. The driving data are collected by using the three-dimensional driving simulator based on CAVE, which provides stereoscopic immersive vision. Finally, the performance of the SS-ARX model in the case of using as the recognizer is examined. The results show the high potential ability of the SS-ARX model as the behavior recognizer.

I. INTRODUCTION

Recently, the modeling of driving behavior has attracted much attention by many researchers[1][2][3]. In order to model the driver's driving maneuver, the conventional techniques such as the nonlinear regression models, the neural network and fuzzy systems have been used. These techniques, however, have some problems as follows: (1) the obtained model often results in too complicated model, (2) this makes it impossible to understand the physical meaning of the driver's maneuver. When we look at the driving behavior, it is often found that the driver appropriately switches the simple control laws instead of adopting the complex nonlinear control law. From this viewpoint, the authors have proposed the modeling of the driving behavior based on the piecewise linear model[4]. Although this work is quite promising for the design of the human assist control system, the behavior recognition problem must be addressed at the same time.

In this paper, we try to build the model of the human driving behavior based on the expression as Stochastic Switched ARX model (SS-ARX) focusing on the driver's collision avoidance behavior. Then the most likely SS-ARX model is used to classify and recognize the human driving behavior.

T.Suzuki, S.Sekizawa and S.Inagaki are with the Department of Mechanical Science and Engineering, Graduate School of Engineering, Nagoya University, Furo-cho, Chikusa-ku, Nagoya, Japan t_suzuki@nuem.nagoya-u.ac.jp

S.Hayakawa and N.Tsuchida are with Toyota Technological Institute, Hisakata, Tempaku-ku, Nagoya, Japan s_hayakawa@toyota-ti.ac.jp

T.Tsuda and H.Fujinami are with Toyota Motor Corporation, Toyota-cho 1, Toyota, Japan taishi@tsuda.tec.toyota.co.jp

The parameter estimation technique for the SS-ARX model shown in this paper can be considered as a natural extension of the parameter estimation in the Hidden Markov Model (HMM) which shows great performance in the field of speech recognition[8]. First, an ARX model is assigned to each discrete state of HMM. Then, the parameter estimation algorithm is derived based on the EM algorithm.

The driving data are collected by using the three-dimensional driving simulator based on CAVE, which provides stereoscopic immersive vision. In our modeling, the relationship between the sensory information such as the range between cars, range rate and lateral displacement between cars and the output of driver such as the steering amount are expressed by the ARX model.

This paper is organized as follows. In Section 2, the ARX model and HMM are briefly reviewed. In Section 3, the parameter estimation technique for the SS-ARX model is developed. The configuration of the developed DS based on the CAVE is introduced in Section 4. In Section 4, the scenario of our examination is also described. The modeling and recognition results are shown in Sections 5 and 6.

II. BRIEF REVIEW OF ARX MODEL AND HMM

As a preliminary for the SS-ARX model, the conventional ARX model and HMM are briefly reviewed.

A. ARX Model

The standard ARX model is described by the following difference equation:

$$y_t = c_1 y_{t-1} + c_2 y_{t-2} + \cdots + c_n y_{t-n} + d_0 u_t + d_1 u_{t-1} + \cdots + d_m u_{t-m} + e_t \quad (1)$$

where y_t and u_t are an output and an input of the system at t . Also, n and m are order of the ARX model, and $c_1, c_2, \dots, c_n, d_0, d_1, \dots, d_m$ are parameters. e_t is called an equation error, and is supposed to have a Gaussian distribution with variance σ .

By using the following vector form,

$$\theta = (c_1, c_2, \dots, c_n, d_0, d_1, \dots, d_m)^T \quad (2)$$

$$\psi_t = (y_{t-1}, y_{t-2}, \dots, y_{t-n}, u_t, u_{t-1}, \dots, u_{t-m})^T \quad (3)$$

Equation (1) is rewritten as follows:

$$y_t = \psi_t^T \theta + e_t \quad (4)$$

B. Hidden Markov Model (HMM)

HMM is an extension of the standard Markov model, in which the multiple output symbol can be generated at each state according to the output probability.

HMM is specified by the following parameters and has a graphical representation as shown in Fig.1 (In Fig.1, the number of discrete state is 3).

- Set of discrete states $S [= S_i, (i=0, 1, \dots, N)]$
- Set of output symbols $V [= v_k, (k=0, 1, \dots, M)]$
- a_{ij} : State transition probability ($i=0, 1, \dots, N; j=0, 1, \dots, N$)
- $b_i(v_k)$: Output probability at each state ($i=0, 1, \dots, N; k=0, 1, \dots, M$)
- π_i : Initial state probability ($i=0, 1, \dots, N$)

$N + 1$ and $M + 1$ denote number of states and output symbols, respectively. HMM switches the discrete state according to the state transition probability a_{ij} , and an output symbol v_k is generated according to the output probability $b_i(v_k)$ at each state S_i .

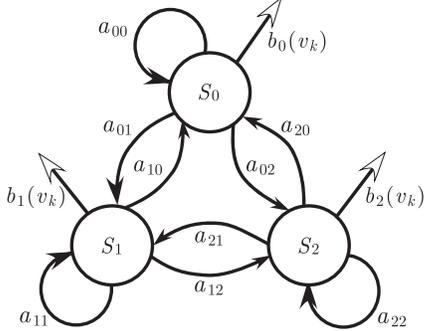


Fig. 1. HMM (3states)

III. PARAMETER ESTIMATION FOR SS-ARX MODEL

SS-ARX model can be regarded as the model in which the ARX model and HMM are combined. SS-ARX is defined as the system in which an ARX model is switched to other one according to the state transition probability as shown in Fig.2. The parameters in ARX model assigned to the discrete state S_i is designated by subscript i .

A. Parameters in SS-ARX model

The parameters in SS-ARX model are specified as follows:

- Set of discrete states $S [= S_i, (i=0, 1, \dots, N)]$
- a_{ij} : State transition probability ($i=0, 1, \dots, N; j=0, 1, \dots, N$)
- π_i : Initial state probability ($i=0, 1, \dots, N$)
- θ_i : Parameters in ARX model assigned to S_i ($i=0, 1, 2, \dots, N$)
- σ_i : Variances of equation error in ARX model assigned to S_i ($i=0, 1, 2, \dots, N$)

$N + 1$ denotes the number of discrete states.

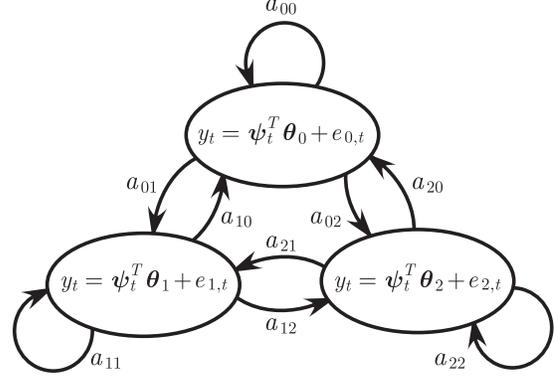


Fig. 2. SS-ARX Model (3states)

B. Definition of output symbol in SS-ARX model

In order to derive the parameter estimation law, the output symbol and output probability are defined for the SS-ARX model as follows: By this definition, the parameter estimation algorithm for HMM can be naturally extended to the one for SS-ARX model. First of all, an output symbol o_t at time t is defined as the combination of the output y_t and the regressor ψ_t , that is, $o_t = (y_t, \psi_t)$. Then, the output probability $b_i(o_t)$ is defined by the assumption of the Gaussian distribution of the equation error.

$$b_i(o_t) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp \left\{ -\frac{(\psi_t^T \theta_i - y_t)^2}{2\sigma_i^2} \right\}. \quad (5)$$

By these definitions, the parameter estimation problem for SS-ARX model can be addressed as the similar problem for the HMM with continuous output probability.

C. Parameter Estimation

1) *EM algorithm*: For the ARX model, in which its parameters are denoted by λ , we consider an unobservable state sequence $\mathbf{s} = (s_0, s_1, \dots, s_t, \dots, s_T)$ and an observable output sequence $\mathbf{O} = (o_0, o_1, \dots, o_t, \dots, o_T)$. The maximization of the likelihood value of the \mathbf{s} and \mathbf{O} , $L(\mathbf{s}, \mathbf{O}; \lambda) = P(\mathbf{s}, \mathbf{O} | \lambda)$ is achieved by introducing the EM algorithm. EM algorithm can locally maximize the likelihood value $L(\mathbf{s}, \mathbf{O}; \lambda) = P(\mathbf{s}, \mathbf{O} | \lambda)$ by iterative procedure.

Generally, the EM algorithm tries to find the parameter λ' which maximizes the following Q function:

$$Q(\lambda, \lambda') = E[\log \{P(\mathbf{s}, \mathbf{O} | \lambda')\} | \mathbf{O}, \lambda] \quad (6)$$

$$= \sum_{\mathbf{s}} P(\mathbf{s} | \mathbf{O}, \lambda) \log \{P(\mathbf{s}, \mathbf{O} | \lambda')\} \quad (7)$$

by executing following procedures iteratively.

- 1) Specify an initial parameter λ .
- 2) Find the λ' which maximizes the $Q(\lambda, \lambda')$.
- 3) Substitute λ' for λ , and iterate 2) until $\lambda' = \lambda$ holds.

2) *Parameter estimation algorithm*: The parameters of SS-ARX model before and after the update are given by $\lambda = (\pi_i, a_{ij}, \theta_i, \sigma_i)$, and $\lambda' = (\pi'_i, a'_{ij}, \theta'_i, \sigma'_i)$. From (7),

$$Q(\lambda, \lambda') = \sum_{\mathbf{s}} P(\mathbf{s}|\mathbf{O}, \lambda) \log \{P(\mathbf{s}, \mathbf{O}|\lambda')\} \quad (8)$$

$$= \frac{1}{P(\mathbf{O}|\lambda)} \sum_{\mathbf{s}} P(\mathbf{s}, \mathbf{O}|\lambda) \log \{P(\mathbf{s}, \mathbf{O}|\lambda')\}. \quad (9)$$

Since $\frac{1}{P(\mathbf{O}|\lambda)}$ is constant, the maximization of $Q(\lambda, \lambda')$ implies the maximization of $\tilde{Q}(\lambda, \lambda')$ given by

$$\tilde{Q}(\lambda, \lambda') = \sum_{\mathbf{s}} P(\mathbf{s}, \mathbf{O}|\lambda) \log \{P(\mathbf{s}, \mathbf{O}|\lambda')\}. \quad (10)$$

By using the definition

$$P(\mathbf{s}, \mathbf{O}|\lambda) = \pi_{s_0} b_{s_0}(o_0) \times a_{s_0 s_1} b_{s_1}(o_1) \times \dots \times a_{s_{T-1} s_T} b_{s_T}(o_T), \quad (11)$$

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

$$\tilde{Q}(\lambda, \lambda') = \tilde{Q}_1(\lambda, \lambda') + \tilde{Q}_2(\lambda, \lambda') + \tilde{Q}_3(\lambda, \lambda') \quad (12)$$

where

$$\tilde{Q}_1(\lambda, \lambda') = \sum_{s_0=0}^N \sum_{s_1=0}^N \dots \sum_{s_T=0}^N \pi_{s_0} b_{s_0}(o_0) \times a_{s_0 s_1} b_{s_1}(o_1) \times \dots \times a_{s_{T-1} s_T} b_{s_T}(o_T) \log \{\pi'_{s_0}\}, \quad (13)$$

$$\tilde{Q}_2(\lambda, \lambda') = \sum_{s_0=0}^N \sum_{s_1=0}^N \dots \sum_{s_T=0}^N \pi_{s_0} b_{s_0}(o_0) \times a_{s_0 s_1} b_{s_1}(o_1) \times \dots \times a_{s_{T-1} s_T} b_{s_T}(o_T) \times \sum_{t=1}^T \log \{a'_{s_{t-1} s_t}\}, \quad (14)$$

$$\tilde{Q}_3(\lambda, \lambda') = \sum_{s_0=0}^N \sum_{s_1=0}^N \dots \sum_{s_T=0}^N \pi_{s_0} b_{s_0}(o_0) \times a_{s_0 s_1} b_{s_1}(o_1) \times \dots \times a_{s_{T-1} s_T} b_{s_T}(o_T) \times \sum_{t=0}^T \log \{b'_{s_t}(o_t)\} \quad (15)$$

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

$$\alpha(i, t) = \sum_{s_0=0}^N \sum_{s_1=0}^N \dots \sum_{s_{t-1}=0}^N \pi_{s_0} b_{s_0}(o_0) \times a_{s_0 s_1} b_{s_1}(o_1) \times \dots \times a_{s_{t-1} s_t} b_{s_t}(o_t) \quad (16)$$

$$\beta(i, t) = \sum_{s_{t+1}=0}^N \sum_{s_{t+2}=0}^N \dots \sum_{s_T=0}^N a_{s_t s_{t+1}} b_{t+1}(o_{t+1}) \times a_{s_{t+1} s_{t+2}} b_{s_{t+2}}(o_{t+2}) \times \dots \times a_{s_{T-1} s_T} b_{s_T}(o_T) \quad (17)$$

The meaning of $\alpha(i, t)$ is the probability for the SS-ARX model λ to generate the output sequence $\mathbf{O} = (o_0, o_1, \dots, o_t)$ until t and reach the state S_i at t (i.e.

$s_t = S_i$). Also, the meaning of $\beta(i, t)$ is the probability for the SS-ARX model λ to generate the output sequence $\mathbf{O} = (o_{t+1}, o_{t+2}, \dots, o_T)$ starting from S_i at t (i.e. $s_t = S_i$) and reach the final state at T .

By using eqs. (16) and (17), eqs. (13), (14) and (15) are rewritten as follows:

$$\tilde{Q}_1(\lambda, \lambda') = \sum_{i=0}^N \pi_i \log \{\pi'_i\} b_i(o_0) \beta(i, 0) \quad (18)$$

$$\tilde{Q}_2(\lambda, \lambda') = \sum_{t=1}^T \sum_{i=0}^N \sum_{j=0}^N \log \{a'_{ij}\} \alpha(i, t-1) a_{ij} b_i(o_t) \beta(j, t) \quad (19)$$

$$\tilde{Q}_3(\lambda, \lambda') = \sum_{t=0}^T \sum_{i=0}^N \log \{b'_i(o_t)\} \alpha(i, t) \beta(i, t) \quad (20)$$

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

$$\pi'_i = \frac{\pi_i b_i(o_0) \beta(i, 0)}{\sum_{k=0}^N \pi_k b_k(o_0) \beta(k, 0)} \quad (21)$$

$$a'_{ij} = \frac{\sum_{t=1}^T \alpha(i, t-1) a_{ij} b_j(o_t) \beta(j, t)}{\sum_{k=0}^N \sum_{t=1}^T \alpha(i, t-1) a_{ik} b_k(o_t) \beta(k, t)} \quad (22)$$

$$\theta'_i = \left\{ \sum_{t=0}^T \left\{ \psi_t \psi_t^T \alpha(i, t) \beta(i, t) \right\} \right\}^{-1} \times \left\{ \sum_{t=0}^T \left\{ \psi_t y_t \alpha(i, t) \beta(i, t) \right\} \right\} \quad (23)$$

$$\sigma_i'^2 = \frac{\sum_{t=0}^T \left\{ |\psi_t^T \theta'_i - y_t|^2 \alpha(i, t) \beta(i, t) \right\}}{\sum_{t=0}^T \left\{ \alpha(i, t) \beta(i, t) \right\}} \quad (24)$$

By iterating three steps in the EM algorithm together with (21), (22), (23) and (24), the parameter λ is locally maximized.

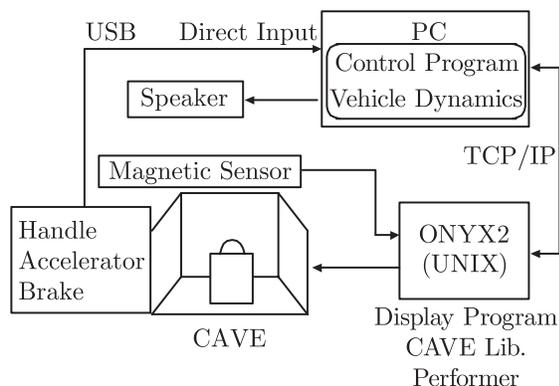
Note that eq. (23) can be regarded as the weighted least mean square solution in which the weight parameters are specified by the sojourn probability at the state S_i .

IV. APPLICATION TO DRIVERS BEHAVIOR MODELING

A. Configuration of Driving Simulator

The configuration and appearance of the developed DS based on the CAVE are shown in Figs. 3(a) and (b). The display unit in the CAVE system provides the 3D virtual environment, and it is controlled by ONYX2. The display program was developed by making use of 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 on the driver's output to the handle, accelerator and brake is transferred to the PC through the USB terminal, and the vehicle position and motion are calculated based on these inputs and vehicle dynamics implemented on the PC using the CarSim software. The results of the calculation

are transferred to ONYX2 through the Internet (TCP/IP), and the 3D visual image based on the position and motion of the vehicle is displayed.



(a)



(b)

Fig. 3. The developed driving simulator based on CAVE.

B. Data acquisition in collision avoidance

1) *Experimental environment and conditions:* In this paper, we focus on the driver's collision avoidance behavior at the instance of the sudden stopping of the preceding vehicle when the examinee looks aside from the road ahead. 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 outputs of drivers are also specified as follows:

- 1) Steering amount (y_t)

The selection of sensory information was motivated by the result in [2]. One may argue that the braking operation should be taken into account the model. However, thanks to the enough range between cars, the influence of the braking is smaller than that of steering.

In our modeling, the regressor vector in (4) was specified as follows:

$$\psi_t = (u_{1,t-m}, u_{2,t-m}, u_{3,t-m})^T \quad (25)$$

where m represents the time delay from the perception to the operation in human behavior. In the analyses shown in the later sections, m was set to be twelve. Since the sampling interval was 16[msec], this implies that we have specified about 200[msec] time delay, which is known as a reasonable value in the field of human engineering. Also, the number of discrete states in the SS-ARX model was set to be three.

The configuration of the experimental environment is shown in Fig.4.

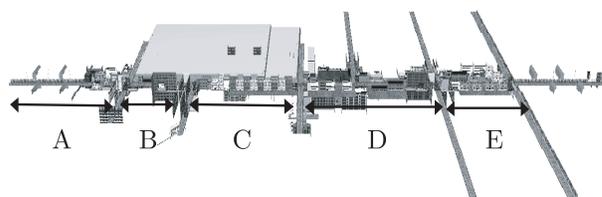


Fig. 4. Road environment for experiments

It has four intersections and two T-type junctions. The road has 1km length and 7m wide, and the pedestrian way is 1.5m wide. The friction coefficient of the road was set to be 0.8. The scene around the start point is shown in Fig.5. The vehicle moves from the left side to the right side in Fig.4. The scene at 940m point is same as one at the start point. After 940m point, the same environment from the start point shows up again. Therefore, the driver feels that this virtual environment is straight road without any end point. There exist only two vehicles in this environment. One of them is a sedan-type car driven by the examinee. The other one is the big truck, which runs in front of the examinee and is controlled by the operator. The car used in the simulator has an engine with 3000cc displacement. Also, the car is supposed to have an ABS. The truck in front of the examinee runs at the constant speed of 50[km/h]. The maximum deceleration of the truck was supposed to be 7[m/s²].

C. Procedure of experiment

The examinee drives the car with keeping constant distance to the preceding truck. The operator set the red or green parking vehicles on the right side at the intersection. The examinee are supposed to take a look at the right side at each intersection, then the examinee answers what colors the parking vehicle is and/or whether there are any parking vehicles. When the examinee looks right side at some intersection, the preceding truck is supposed to stop suddenly with maximum deceleration. Then, the collision avoidance behavior of the examinee is measured.



Fig. 5. Road image at start point

V. MEASURED DATA AND MODELING RESULTS

Based on the setup described in previous section, three drivers carried out the experiment under virtual environments. First of all, the collision avoidance behavior, which is characterized by the profile between the beginning of the steering operation and the stopping of the steering operation, are measured and analyzed. The profiles of the measured steering and the estimated steering calculated from the obtained parameters of the corresponding SS-ARX model are depicted in Fig.6. The obtained parameters are listed in Table I. Note that the two vertical lines in Fig.6 represent the most likely switching points calculated by using the following Viterbi algorithm (Recall that the number of discrete states is three).

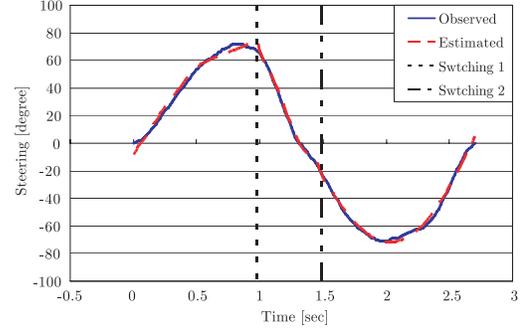
$$\delta(i, t) = \max_{0 \leq j \leq N} \{\delta(j, t-1) a_{ji} b_i(o_t)\} \quad (26)$$

$$\gamma(i, t) = \arg \max_{0 \leq j \leq N} \{\delta(j, t-1) a_{ji} b_i(o_t)\} \quad (27)$$

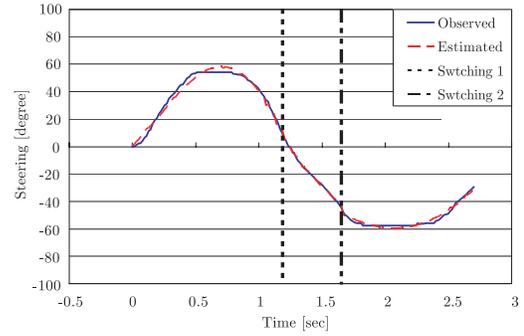
where $\gamma(i, t)$ represents the most likely state S_i at time t . The estimated steering profile in Fig.6 is generated by regarding the SS-ARX model as a deterministic switched ARX model specified by θ_i and switching points (given by $\gamma(i, t)$). From Fig.6, we can see that the parameter estimation algorithm works well. It is also interesting to see the difference in the identified parameters of each driver.

VI. DRIVERS BEHAVIOR RECOGNITION

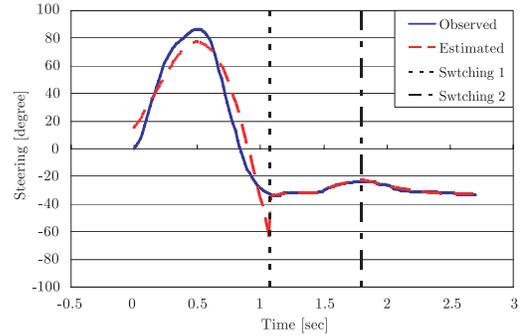
Now, the obtained SS-ARX model for each driver is used as the ‘behavior recognizer’. The list of Logarithm of the likelihood value of each SS-ARX model for all measured data are listed in Table II. In Table II, SS-ARX(*) denotes the SS-ARX model obtained from the corresponding driver’s data profile. Also, O_* denotes the measured output sequence of the corresponding driver. As shown in this table, the likelihood values for other drivers data are smaller than that for corresponding drivers data (designated by underlines). These results clearly show the potential ability of the SS-ARX model to be used as the behavior recognizer of human being.



(a)



(b)



(c)

Fig. 6. Steering profiles of three drivers (measured and estimated)

VII. CONCLUSIONS

In this paper, we have developed a new modeling technique of the human driving behavior based on the expression as Stochastic Switched ARX model (SS-ARX) with focusing on the driver’s collision avoidance behavior. First of all, the parameter estimation technique for the SS-ARX model has been introduced based on the EM algorithm. This can be achieved by extending the parameter estimation technique for conventional Hidden Markov Model (HMM). Second, the parameter estimation technique has been applied to the measured driving data, and have found the parameter set for each driving data. The driving data was collected by using the three-dimensional driving simulator based on CAVE, which provides stereoscopic immersive

TABLE I
IDENTIFIED PARAMETERS

SS-ARX(A)		
Parameters in ARX model	Variances	
$\theta_0 = (-0.824773, 2.244757, 0.603667)$	$\sigma_0 = 0.037493$	
$\theta_1 = (-3.483122, 4.816041, -5.470960)$	$\sigma_1 = 0.011330$	
$\theta_2 = (-3.936077, 4.499318, -4.582369)$	$\sigma_2 = 0.024018$	
A matrix		
$a_{ij} = \begin{bmatrix} 0.999716 & 0.000284 & 0.000000 \\ 0.000000 & 0.977819 & 0.022181 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$		
SS-ARX(B)		
Parameters in ARX model	Variances	
$\theta_0 = (-1.581261, 1.887177, -4.802167)$	$\sigma_0 = 0.032215$	
$\theta_1 = (1.878792, -1.140327, 0.043192)$	$\sigma_1 = 0.008611$	
$\theta_2 = (-1.100865, 0.516427, -1.362143)$	$\sigma_2 = 0.018346$	
A matrix		
$a_{ij} = \begin{bmatrix} 0.994427 & 0.005573 & 0.000000 \\ 0.000000 & 0.998804 & 0.001196 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$		
SS-ARX(C)		
Parameters in ARX model	Variances	
$\theta_0 = (-2.523657, 2.362553, -7.386183)$	$\sigma_0 = 0.149673$	
$\theta_1 = (0.609914, -1.271813, 1.165948)$	$\sigma_1 = 0.004568$	
$\theta_2 = (-0.066241, 0.187735, -0.645891)$	$\sigma_2 = 0.007644$	
A matrix		
$a_{ij} = \begin{bmatrix} 0.999743 & 0.000257 & 0.000000 \\ 0.000000 & 0.992420 & 0.007580 \\ 0.000000 & 0.000000 & 1.000000 \end{bmatrix}$		

TABLE II
RECOGNITION RESULTS

	O_A	O_B	O_C
SS-ARXmodel	Log likelihood	Log likelihood	Log likelihood
SS-ARX(A)	-731.673661	-4600.403127	-4910.527209
SS-ARX(B)	-2887.603213	-726.257733	-3711.502708
SS-ARX(C)	-3397.165848	-2780.210061	-725.213853

vision. Finally, the performance of the SS-ARX model in the case of using as the recognizer, has been examined. The results have shown the high potential ability of the SS-ARX model as the behavior recognizer.

VIII. ACKNOWLEDGMENTS

This work was supported by the Space Robotic Center of the Toyota Technological Institute, where CAVE is installed. The authors would like to thank all researchers involved at the Space Robotic Center for their helpful suggestions.

REFERENCES

- [1] M.C. Nechyba and Y. Xu : Human Control Strategy: Abstraction, Verification and Replication, IEEE Control Systems Magazine, vol. 17, no. 5, pp. 48-61(1997)
- [2] Kim Jong-Hae, Yoshimitchi Matsui and Soichiro Hayakawa et al.: Acquisition and Modeling of Driving skill using Three-Dimensional Virtual System, Proceeding of ICASE/SICE Joint Workshop, Vol. 1, pp. 49-53, October(2002)
- [3] Hiroshi Uno, Kaneo Hiramatsu :Aged Driver's Avoidance Capabilities in an Emergent Traffic situation, SAEJ, Vol.32, No.1, 113-118, January(2001)
- [4] Kim Jong-Hae, Soichiro Hayakawa, Tatsuya Suzuki et al.: Modeling of Driver's Collision Avoidance Behavior based on Piecewise Linear Model, Proceedings of IEEE CDC2004, pp.2310-2315
- [5] Lecture Notes in Computer Science; Hybrid Systems: Computation and Control, Springer
- [6] Alberto Bemporad and Jacob Roll, Lennart Ljung : Identification of Hybrid Systems via Mixed-Integer Programming, Proceedings of The 40th IEEE Conference on Decision and Control, pp. 786-792, December(2001)
- [7] Giancarlo Ferrari-Trecate et al., A clustering technique for the identification of piecewise affine system, automatica39, pp.205-217, (2003)
- [8] Lawrence R. Rabiner; A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, IEEE, Vol.77, 1989
- [9] 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, 31(12), pp. 1691-1724(1995)
- [10] M.Akamatsu, S.Doi, M.Okuwa, K.Takuguchi : Behavior Based Human Environment Creation Technology, Project International Workshop on ITS Human Interface, Japan (ITS HMI symposium 2000)
- [11] S. Doi, Sueharu Nagiri et al.:Research and Development of Driving Support System for Individual Driving Ability, Toyota Central R and D Labs. Report, 71-107(1998)