# Modeling Dynamic Structure of Human Verbal and Nonverbal Communication

Takashi Matsuyama and Hiroaki Kawashima
Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo, Kyoto 606-8501, Japan
{tm,kawashima}@i.kyoto-u.ac.jp

## Abstract

*In human communication, dynamics of communication, i.e. timing structure of utterances, nodding, gesture, pause and so on, plays a crucial role to realize smooth natural communication. We proposed a computational scheme named Interval-based Linear Hybrid Dynamical System (ILHDS, in short) for modeling complex dynamic events and conducted several experiments to explore characteristics of dynamic structures of human verbal and nonverbal communication based on ILHDS. In the paper, we describe the theoretical scheme of ILHDS followed by its practical applications.*

## 1. Introduction

Understanding the meaning of user commands and presenting appropriate information to a user is one of the primary objectives of human-machine interaction systems. Most of the existing approaches, therefore, set the goal to realize interaction systems that understand semantic information specified by a user and generate attractive presentation to a user using multimedia data such as text, graphs, pictures, video, sound, and so on. Now advanced systems are being developed that can understand spoken words and gestures as well as generate 3D images.

While such multimedia interaction systems are important, users sometimes feel frustration when the systems get out of human interaction protocols. That is, the systems often ignore dynamic features such as acceleration patterns, pause lengths, tempo speed, and rhythms, which convey rich nonverbal and non-semantic information in human communication.

In this paper, we attempt to model such dynamic features or temporal structures in verbal and nonverbal communication based on a novel computational model, named Interval-based Linear Hybrid Dynamical System (ILHDS, in short). A hybrid dynamical system is the integration of two types of dynamical systems: one described by differential equations,

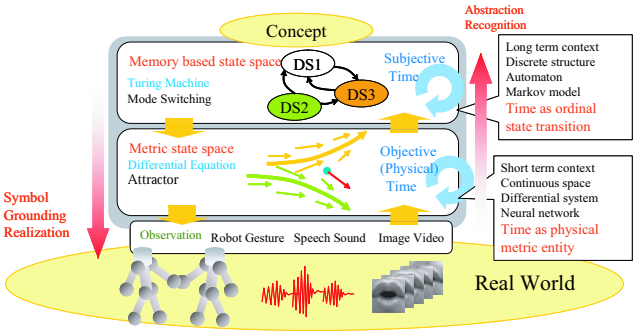


**Figure 1. Architecture of hybrid dynamical systems**

which is suitable for describing physical phenomena (consider time as physical metric entity), and a discrete-event system, which is suitable for describing human subjective or intellectual activities (consider time as ordinal state transition) (Figure 1).

We developed ILHDS based on the following rationale. Firstly, we assume that a complex human behavior consists of dynamic primitives, which are often referred to as motion elements, movemes, visemes, and so on. For example, a cyclic lip motion can be described by a cyclic sequence of simple lip motions such as “open”, “close”, and “remain closed”. Once the set of dynamic primitives is determined, a complex behavior can be partitioned into “temporal intervals”, each of which is characterized by a dynamic primitive and its temporal duration.

Secondly, we assume that not only temporal orders of motion elements but also their duration lengths or temporal differences among beginning and ending timing of the temporal intervals convey rich information in human communication. For example, some psychological experiments suggest that duration lengths of facial actions play an important role for human judgments of basic facial expression categories.

Based on the assumptions above, we proposed ILHDS for modeling dynamic events in terms of temporal intervals. The system has a two-layer architecture consisting of a fi-

nite state automaton and a set of linear dynamical systems. In this architecture, each linear dynamical system represents the dynamics of a motion primitive and corresponds one to one to a discrete state of the automaton. In other words, the automaton controls the activation order and timing of the linear dynamical systems. Thus, ILHDS can model and generate multimedia signals that represent complex human behaviors.

Applying ILHDS to various human verbal and nonverbal communication behaviors, we can successfully extract dynamic features of the behaviors based on relations of temporal intervals, classify fine-grained facial expressions (Section 4), and analyze synchronization/delay mechanisms between mouth motion and speech utterance (Section 5).

## 2. Interval-Based Hybrid Dynamical System

### 2.1. System Architecture

ILHDS has a two-layer architecture (Figure 2). The first layer (the top of Figure 2) records a finite state automaton as a discrete-event system that models stochastic transitions between discrete events. The second layer (not explicitly described in Figure 2) consists of a set of linear dynamical systems $\mathcal{D} = \{D_1, ..., D_N\}$. To integrate these two layers, we introduce *intervals* (the second top of Figure 2): each interval is described by $< q_i, \tau >$, where $q_i$ denotes a state in the automaton and $\tau$ the physical temporal duration of the interval. Each state in the automaton corresponds to a unique linear dynamical system recorded at the second layer: $q_i$ denotes the label of the corresponding linear dynamical system as well as a state in the automaton. Note that the number of states in the automaton is not greater than that of intervals; multiple different intervals can correspond to the same state in the automaton, i.e. their dynamics are described/controlled by the same linear dynamical system.

When a temporal sequence of observed signal data, which is represented by a multivariate vector sequence (the bottom in Figure 2), is given, it is first transformed into a sequence of internal states (the second and third bottom in Figure 2). Then, that sequence is partitioned into a sequence of intervals (the second top in Figure 2). That is, the internal state sequence is partitioned into a group of sub-sequences so that the dynamic state variation in each sub-sequence can be described by a linear dynamical system, which is denoted by $q_i$ recorded in the interval covering that sub-sequence.

Once ILHDS has been constructed by learning as will be described in Section 3, it can generate a multivariate signal sequence by activating the automaton: the activated automaton first generates a sequence of intervals, each of which then generates a signal sequence based on its corresponding linear dynamical system. Note that the activation

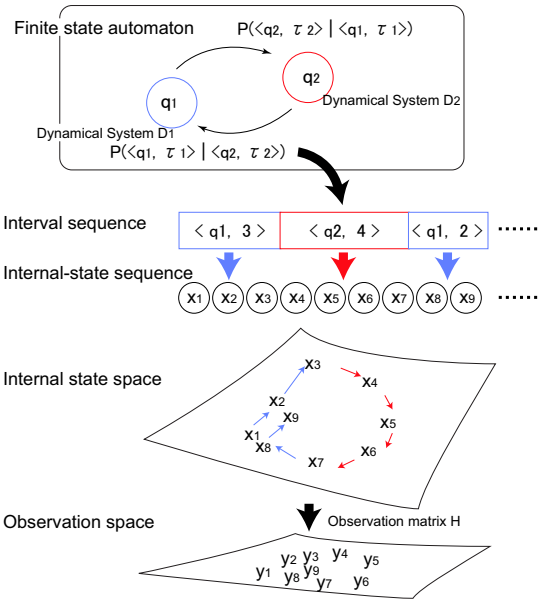


**Figure 2. Interval-based hybrid dynamical system**

timing and period of the linear dynamical system are controlled by the duration length recorded in the interval.

We define some terms and notations for later discussions. Firstly, we simply use the term "dynamical systems" to denote linear dynamical systems.

**Internal state:** All the constituent dynamical systems are assumed to share an $n$-dimensional internal state space. Each activated dynamical system can generate sequences of real valued internal state vector $x \in \mathbf{R}^n$, which can be mapped onto the observation space by a linear function. We assume such linear transformation function is also shared by all the dynamical systems.

**Observation:** An observation sequence is described by a multivariate vector $y \in \mathbf{R}^m$ sequence in a $m$-dimensional observation space.

**Discrete state:** The finite state automaton has a discrete state set $Q = \{q_1, ..., q_N\}$. Each state $q_i \in \mathcal{Q}$ corresponds to the dynamical system $D_i$, respectively.

**Duration length of an interval:** The duration that an interval continues is described by a positive integer; to reduce parameter size, we set a minimum duration $l_{\min}$ and a maximum duration $l_{\max}$ and the duration is defined by $\tau \in \mathcal{T} \triangleq \{l_{\min}, l_{\min} + \delta, l_{\min} + 2\delta, ..., l_{\max}\}$.

**Interval:** An interval generated by the automaton is defined as a combination of a discrete state and a duration length. We use notation $< q_i, \tau > \in \mathcal{Q} \times \mathcal{T}$ to represent the interval that has state $q_i$ and duration $\tau$.

### 2.2. Linear Dynamical Systems

The state transition of dynamical system $D_i$ in the internal state space, and the mapping from the internal state space to the observation space is modeled by the following linear equations:

$$\begin{aligned} x_t &= F^{(i)}x_{t-1} + g^{(i)} + \omega_t^{(i)} \\ y_t &= Hx_t + \upsilon_t, \end{aligned} \quad (1)$$

where $F^{(i)}$ is a transition matrix and $g^{(i)}$ is a bias vector. $H$ is an observation matrix that defines linear projection from the internal state space to the observation space. $\omega^{(i)}$ and $\upsilon$ is the process noise and the observation noise, which are modeled by Gaussian distributions respectively. Note that each dynamical system is defined by $F^{(i)}$, $g^{(i)}$, and $\omega^{(i)}$.

### 2.3. Interval-Based State Transition

In this section, we define the transition of discrete states in the automaton that generate interval sequences. Here, we assume first-order Markov property for the generated intervals. A major difference from conventional state transition models, such as hidden Markov models, is that the automaton models the correlation between duration lengths of adjacent intervals as well as the transition of discrete states.

Let $\mathcal{I} = I_1, ..., I_K$ be an interval sequence generated by the automaton. To simplify the model, we assume that adjacent intervals have no temporal gaps or overlaps. Here, the interval $I_k$ depends only on the previous interval $I_{k-1}$ because of the Markov property assumption. Then, the Markov process of intervals can be modeled by the following conditional probability:

$$P(I_k =< q_j, \tau > | I_{k-1} =< q_i, \tau_p >),$$

which denotes the probability that interval $< q_j, \tau >$ occurs after interval $< q_i, \tau_p >$.

The computation of probability $P(I_k =< q_j, \tau > | I_{k-1} =< q_i, \tau_p >)$ requires a large parameter set, which does not only increase computational cost but also incur the problem of over-fitting during a training phase. We therefore use a parametric model for the duration length distribution. That is, for each state transition in the automaton, we record $P(q_j|q_i)$ together with a parametric distribution for $P(\tau|\tau_p, q_i, q_j)$.

## 3. Learning Process for ILHDS

### 3.1. Difficulties in Learning

Let us assume that only a group of multivariate signal sequences is given as training data. Then, in most of hybrid dynamical systems, the system identification process that estimates system parameters becomes difficult because of its paradoxical nature. That is, the system consists of a set of subsystems (in our case, linear dynamical systems) and the parameter estimation of each subsystem requires partitioned training data to be modeled by that subsystem, while the segmentation process of training data requires a set of identified subsystems. Moreover, the number of subsystems is also unknown in general.

The expectation-maximization (EM) algorithm [4] is one of the most common approaches to solve this kind of paradoxical problems. The algorithm estimates parameters based on the iterative calculation. In each step, the algorithm conducts model fitting to training data using the model parameters that were updated in the previous step. Then, the parameters are updated based on the result of the current model fitting process.

However, the EM algorithm-based parameter estimation method involves two problems:

1. Initialization of the EM algorithm
2. Estimation of the number of subsystems

To solve these problems, we propose a two-step learning method.

### 3.2 Two-Step Learning Method

The key idea of our learning method is that we divide the estimation process into two steps: clustering of dynamical systems to estimate a set of required dynamical systems and parameter refinement of the estimated dynamical systems.

We here assume that internal-state sequences have been estimated from observation sequences, i.e. an observation matrix $H$ and distribution parameters of observation noise $\upsilon$ have been estimated based on prior knowledge or system-identification techniques [7].

**[Step 1] Clustering of Dynamical Systems:** The first step is a clustering process that finds a set of dynamical systems required to describe training data: the number of the systems and their parameters. This step employs a typical data sequence as training data. Then, an agglomerative hierarchical clustering is applied to the training data to estimate a set of dynamical systems required to model the data (Figure 3):

1. Partition the training sequence into a group of very short sub-sequences and estimate a dynamical system that can model each sub-sequence respectively.
2. Compute the distance between each pair of estimated dynamical systems.

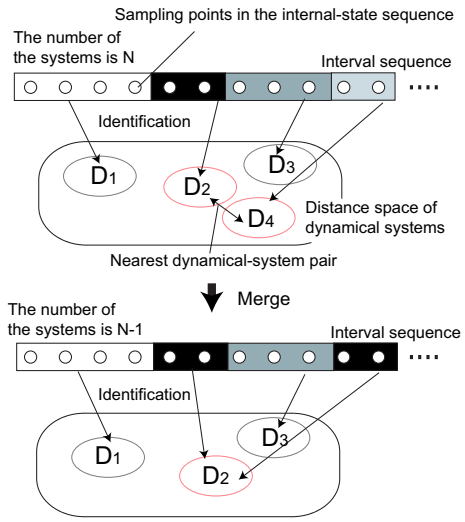


**Figure 3. Hierarchical clustering of dynamical systems.**

3. Integrate the closest pair of dynamical systems: compute parameters of the integrated dynamical system based on such sub-sequences that were modeled by the pair of dynamical systems to be integrated.

4. Iterate the above integration process until the closest distance between a pair of dynamical systems becomes greater than a pre-specified value.

After this process, we get the number of required dynamical systems $N$ and approximate parameters of the dynamical systems.

**[Step 2] Refinement of the Parameters:** The second step is a refinement process of the system parameters based on the EM algorithm. The process is applied to all training data, whereas the clustering process is applied to a selected typical training sequence. While the EM algorithm strongly depends on its initial parameters, the clustering step provides an initial parameter set that is relatively close to the optimum.

Once the system parameters have been identified, each sequence in the training data set can be described by a sequence of intervals respectively, which then is used to estimate parameters of the automaton. Firstly note that a set of discrete states have been determined uniquely from the set of dynamical systems obtained by the clustering process. Then, for each pair of discrete states, the transition probability and the duration length distribution associated with the state transition are computed. Thus, ILHDS is identified.

# 4. Analysis of Timing Structures in Facial Expressions

## 4.1. Timing Structure in Facial Expression

Facial expression plays an important role in human communication; it can express emotion and intention to others.

Many systems developed so far describe facial expressions based on "action units" of the Facial Action Coding System (FACS) developed by Ekman and Friesen [5]. However, since FACS cannot describe dynamic characteristics of facial expressions, its descriptive capability is confined to rather stereotype ones such as happiness, surprise, fear, anger, disgust, and sadness. We believe, on the other hand, in human communication, facial expressions carry more fine-grained emotional and intentional information; human facial expression can be considered as being generated based on two mechanisms: (1) emotional expression produced by spontaneous muscular action and (2) intentional display to convey some intention to others. To recognize human emotion and intention from facial expressions, therefore, the analysis of their dynamic structures is required.

To describe dynamic characteristics of facial expressions, we apply ILHDS to video data of human faces (see the bottom row of Figure 4):

1. First extract and track each facial part: eyes, eyebrows, mouth, and nose.

2. The motion of each part is described by a multivariate vector sequence: each vector represents a shape of the part at time $t$.

3. Then, apply ILHDS to each sequence to obtain an interval sequence, which describes the dynamic structure of that part motion. Note that we assume ILHDS has been identified beforehand using training data.

## 4.2. Facial Score

Aligning along the common temporal axis a group of interval sequences obtained by the above process, we have what we call a *facial score* (the top right of Figure 4 and Figure 5), where for each facial part, intervals with the same mode (i.e. modeled by the same linear dynamical system) are given the same color and aligned at the same row. The facial score is similar to a musical score, which describes the timing of notes in music. Using the score, we can describe facial expressions as spatio-temporal combination of the intervals.

Figure 4 depicts the overall flow of our facial expression recognition and generation system:

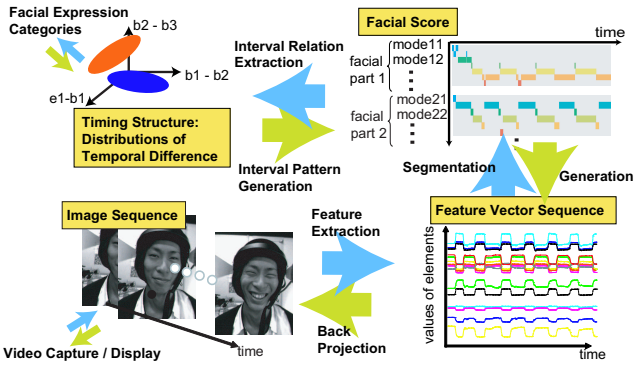


**Figure 4. Overall flow of facial expression recognition and generation using the facial score.**

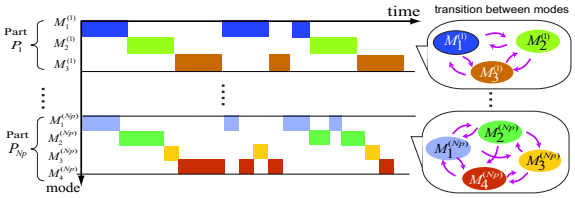


**Figure 5. Facial score**

**Facial image generation** : Once a facial score is obtained by the learning process described in Section 3.2, we can activate ILHDS to generate facial expression video just like as playing music according to a musical sore (down arrow at the right column in Figure 4).

**Facial expression recognition** : Comparing onset and termination timing of intervals of different facial parts, we can extract various temporal features that can be used to classify facial expressions (top left in Figure 4).

## 4.3. Experiments

To evaluate the effectiveness of ILHDS and the descriptive power of the facial score for facial expression recognition, we compare the timing structure of intentional smiles with that of spontaneous smiles; in human communication it is useful to make a distinction between these two smiles, while most previous systems classified them into the same category.

**Acquisition of facial scores:** We tracked feature points in facial image sequences using the active appearance model (AAM) [3]. The sequence of features (x,y-coordinates) in each facial part was converted to an interval sequence using ILHDS. The upper graph in Figure 6 shows

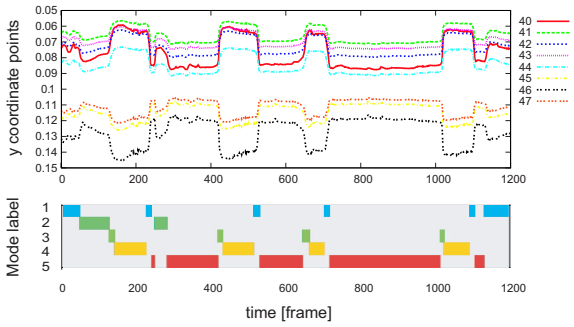


**Figure 6. Feature sequence of the mouth part (top, only y-coordinate variations are depicted) and the computed mouth part score (bottom).**

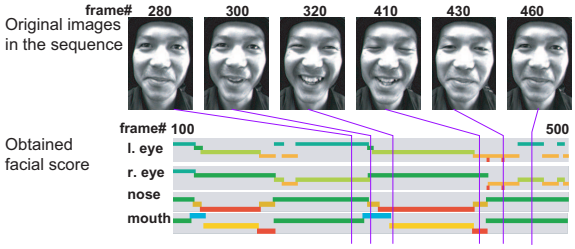


**Figure 7. Facial score of intentional smile.**

a group of temporal sequences of y-coordinates of feature points in a mouth part and the lower the computed mouth part score.

Figure 7 shows an example of the full set of the facial score that describes dynamic characteristics of all facial parts during intentional smiles. This figure suggests that the movement of each smile can be segmented into the following four modes: two stationary modes ("neutral" and "smiling") and two dynamic modes ("onset" and "offset" of smiling).

**Discrimination of intentional and spontaneous smiles:** To evaluate the descriptive power of the facial score, we prepared a pair of training data sets: one for intentional smiles and the other for spontaneous ones and applied the leaning method described in Section 3.2 to each set respectively. Then, we have ILHDS1 and ILHDS2 to describe dynamic structures of intentional and spontaneous smiles respectively.

In the experiments, we used a facial score that consists of three facial parts: left eye, nose, and mouth. In addition, since the duration lengths of stationary modes such as "neutral" and "smiling" closely depend on the context of the expression, we focus on the two dynamic modes: "onset" motion $M_b$ (from neutral to smiling), and "offset" motion

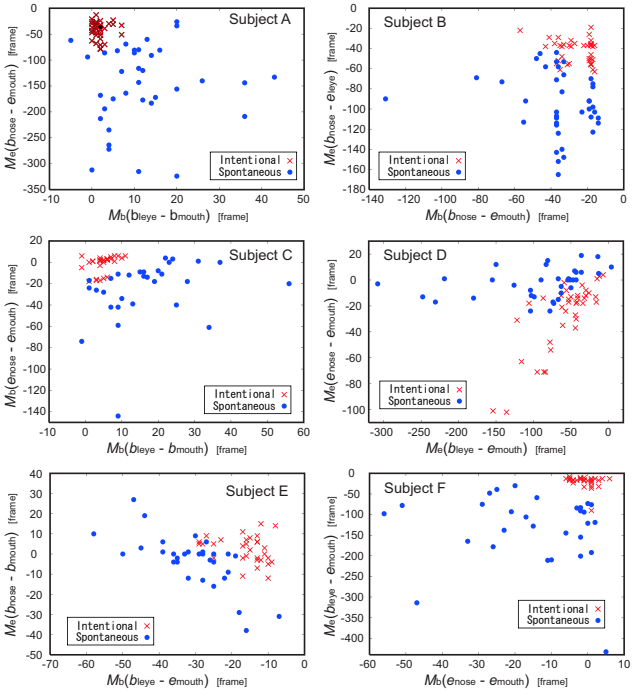


**Figure 8. Extracted timing structure distributions for intentional and spontaneous smiles.**

$M_e$ (from smiling to neutral).

Let $b_{\mathrm{leye}}$ and $e_{\mathrm{leye}}$ be the begin and end timing points of the left eye motion in its facial score. Similarly, let $b_{\mathrm{nose}}$ and $e_{\mathrm{nose}}$ be those of the nose motion, and $b_{\mathrm{mouth}}$ and $e_{\mathrm{mouth}}$ be those of the mouth motion, respectively. Then we extract temporal differences between such timing points; for example, we use $M_b(b_{\mathrm{nose}} - b_{\mathrm{mouth}})$, which denote the temporal difference between the beginning of nose motion and that of mouth motion during the onset of smile.

Since preliminary experiments showed that any single temporal difference cannot discriminate the two smile categories (i.e., intentional and spontaneous), we employed a pair of temporal differences as a distinguishing feature. That is, a feature to characterize each shot of smile is represented by a point in the two-dimensional space whose axes denote a selected pair of temporal differences. Since there exists many possibilities for the combination of temporal differences, for each pair of temporal differences, we calculated the Maharanobis generalized distance between a pair of distributions of two smile categories, and selected such pair of temporal differences that the two distributions took the largest distance. Note that since smiling actions may differ from person to person, we extracted a distinguishing feature for each subject person.

Figure 8 shows the experimental results for six persons, from which we observe while distinguishing features vary from person to person, we can discriminate intentional and spontaneous smiles using their dynamic features. In fact, the performance evaluation of the smile discrimination showed that the rate of the correct discrimination ranges from 79.4% to 100% depending on subjects.

# 5. Modeling Cross-Media Timing Structures in Multimedia Signals

## 5.1. Cross-Media Timing Structures in Multimedia Signals

Measuring dynamic human actions such as speech and music performance with multiple sensors, we can obtain multimedia signal data. We human usually sense/feel cross-modal dynamic structures fabricated by multimedia signals such as synchronization and delay. For example, it is well-known that the simultaneity between auditory and visual patterns influences human perception.

The cross-modal timing structure is also important to realize multimedia systems such as human computer interfaces (e.g., audio-visual speech recognition systems [6]) and computer graphics techniques that generate some media signal from another (e.g., lip sync to input speech [1]).

Dynamic Bayesian networks, such as coupled hidden Markov models [2, 6], are proposed to describe relations between cooccurrent or adjacent states of different media data. They are often used as media integration methods.

While such methods enable us to represent short-term cross-media relations, they are not well suited to describe systematic and long-term cross-media relations. For example, an opening lip motion is strongly synchronized with an explosive sound /p/, while the lip motion is loosely synchronized with a vowel sound /e/.

To represent such systematic and long-term synchronization/delay and mutual dependency among multimedia signals, here we propose a novel model based on ILHDS. For each media signal sequence in multimedia data, we first apply ILHDS to obtain the interval sequence respectively. Then, by comparing intervals of different media signals, we construct a *cross-media timing-structure model*, which is a stochastic model to describe temporal structures across multimedia signals.

## 5.2 Modeling Cross-Media Timing Structures

Applying ILHDS to each media signal sequence in multimedia data, we obtain a group of interval sequences (the top in Figure 9). Let $I_k$ be an interval of mode $M_i$ in one of the obtained interval sequences and $I'_{k'}$ an interval of mode $M'_p$ in another interval sequence overlapping with $I_k$. Note that modes $M_i$ and $M'_p$ specify the linear dynamical systems that describe dynamics in intervals $I_k$ and $I'_{k'}$ respec-

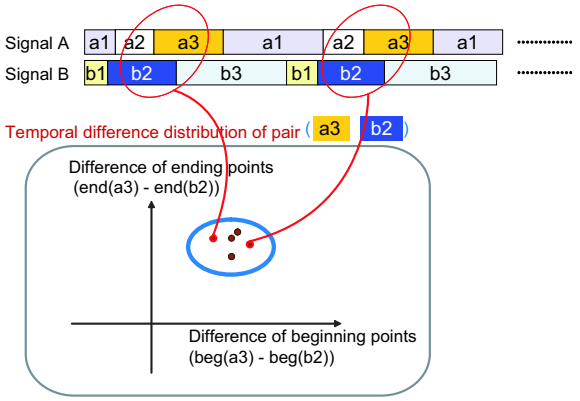


**Figure 9. Learning cross-media timing-structure model.**

tively. Let $b_k(e_k)$ and $b'_{k'}(e'_{k'})$ denote the beginning (ending) points of intervals $I_k$ and $I'_{k'}$, respectively.

To model the cross-media relation between modes $M_i$ and $M'_p$, we collect all pairs of overlapping intervals that satisfy the same temporal relation as that between $I_k$ and $I'_{k'}$ and compute

$$P(b_k - b'_{k'}, e_k - e'_{k'} | m_k = M_i, m'_{k'} = M'_p), \quad (2)$$

where $m_k$ and $m'_{k'}$ are the modes of interval $I_k$ and $I'_{k'}$ ( the bottom in Figure 9). We refer to this distribution as a *temporal difference distribution*. This distribution represents rich cross-media synchronization structures between a pair of different media signals. For example, if the peak of the distribution comes to the origin, the two modes tend to be synchronized each other at both beginning and ending points, while if $b_k - b_{k'}$ has large variance, the two modes loosely synchronized at their onset timing.

Note that we compute temporal difference distributions for all possible mode pairs and record them as fundamental characteristics of the cross-media timing structure of a given multimedia signal data. In addition to a set of such temporal difference distributions, we also model which mode pair tends to overlap with each other across different media (co-occurrence probabilities of modes), and which mode pair tends to appear in neighboring intervals in each media signal data (mode-transition probabilities). The cross-media timing structure is defined by these mutual dependency relations between modes.

## 5.3. Media Conversion Based on Timing Structures

Once the cross-media timing structure model is learned from simultaneously captured multimedia signal data, we can exploit the model for generating one media signal from another related media signal. The overall flow of the media conversion from signal $S'$ to $S$ is as follows:

1. A reference (input) signal $S'$ is partitioned into an interval sequence $\mathcal{I}' = \{I'_1, ..., I'_{K'}\}$.
2. An interval sequence $\mathcal{I} = \{I_1, ..., I_K\}$ is generated from $\mathcal{I}'$ based on the cross-media timing structure model. ($K$ and $K'$ is the number of intervals in $\mathcal{I}$ and $\mathcal{I}'$, and note that $K \neq K'$ in general.)
3. Signal $S$ is generated from $\mathcal{I}$.

The key process of this media conversion lies in step 2. Let $\Phi$ be the cross-media timing structure model that is learned in advance. Then, the problem of generating an interval sequence $\mathcal{I}$ from $\mathcal{I}'$ can be formulated by the following optimization:

$$\hat{\mathcal{I}} = \arg\max_{\mathcal{I}} P(\mathcal{I}|\mathcal{I}', \Phi). \quad (3)$$

In the equation above, we have to determine the number of intervals $K$ and their properties, which can be described by triples $< b_k, e_k, m_k > (k = 1, ..., K)$, where $b_k, e_k \leq T$ and $m_k \in \mathcal{M}$. Here, $T$ is the length of signal $S'$, and $\mathcal{M}$ is the set of modes of intervals, i.e. set of linear dynamical systems, which was fixed at the learning process. If we searched for all possible interval sequences $\{\mathcal{I}\}$, the computational cost would increase exponentially as $T$ becomes longer. We therefore use a dynamic programming method to solve Equation (3), where we assume that generated intervals have no gaps or overlaps; thus, pairs $< e_k, m_k > (k = 1, ..., K)$ are required to be estimated under this assumption.

## 5.4. Experiments

To evaluate the descriptive power of the proposed cross-media timing structure model and the performance of the media conversion method, we conducted experiments on the lip video generation from an input audio signal.

**Feature extraction:** A continuous utterance of five vowels /a/,/i/,/u/,/e/,/o/ (in this order) was captured using mutually synchronized camera and microphone. The utterance was repeated nine times (18 sec.). A lip region in each video image was extracted by AAM, which was used in Section 4. Filter bank analysis was used for the audio feature extraction and the principal component analysis (PCA) was used for visual feature extraction of the lip motion. The extracted features were used as observed data to train ILHDS.

**Learning the cross-media timing structure model:** Using the extracted audio and visual feature vector sequences as signal $S'$ and $S$, we estimated the number of modes and parameters of each mode, partitioned each signal into an interval sequence, and then computed the cross-media timing

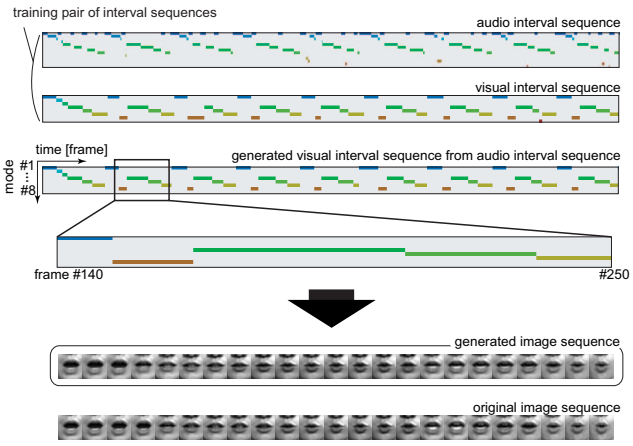


**Figure 10. Media conversion.**

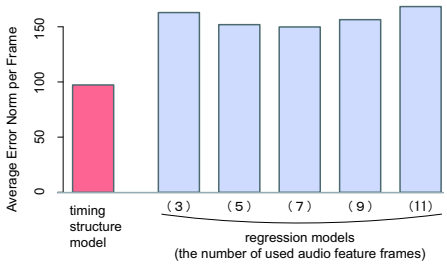


**Figure 11. Average error norm per frame between generated and original sequences.**

structure according to the method described in Section 5.2. The estimated number of modes was 13 and 8 for audio and visual modes, respectively. The segmentation results are shown in Figure 10 (the first and second rows). Because of the noise, some vowels were divided into several different audio modes.

**Evaluation of timing generation:** Based on the estimated cross-media timing-structure, we applied the media conversion method in Subsection 5.3: we used an audio signal interval sequence included in the training data of ILHDS as an input (source) media (top row in Figure 10) and converted it into a video signal interval sequence (third row in Figure 10).

Then, to verify the performance of the media conversion method, we first compared the converted interval sequence with the original one which was generated from the video data measured simultaneously with the input audio data (second row in Figure 10). Moreover, we also compared the pair of video data: one generated from the converted interval sequence (second bottom row in Figure 10) and the originally captured one (bottom row in Figure 10). From these data, the media conversion method seemed to work very well.

To quantitatively compare our method with others, we generated feature vector sequences based on several regression models. Seven regression models were constructed, each of which estimated visual feature vector $y_t$ from $2a + 1$ frames of audio feature vectors $y_{t-a}, y_{t-a+1}, ..., y_t, ..., y_{t+a}$, where $a = 1, 2, ..., 5$. Figure 11 shows the average error norm per frame of each model. We see that our method provide the smallest error compared to the regression models.

## 6. Conclusion

We proposed ILHDS as a novel computational model to represent dynamic events and structures. Applying ILHDS to human behavior analysis, we can successfully extract dynamic features based on the relation of temporal intervals, classify fine-grained facial expressions, and analyze the synchronization/delay mechanism between mouth motion and speech utterance.

In this paper, we concentrated on modeling a single human behavior rather than multiparty interaction, because our first concern is to see the effectiveness of ILHDS for modeling and learning dynamic events and structures from multimedia signals. Currently we are extending the proposed scheme to model multiparty interaction and to realize natural human-machine interaction systems.

This study is supported by Grand in Aid for Scientific Research No.18049046 and 21st Century COE Program on Informatics Research Center for Development of Knowledge Society Infrastructure of the Ministry of Education, Culture, Sports, Science and Technology. Research efforts by members of our laboratory are greatly acknowledged.

## References

[1] M. Brand. Voice puppetry. *Proc. SIGGRAPH*, pages 21–28, 1999.

[2] M. Brand, N. Oliver, and A. Pentland. Coupled hidden Markov models for complex action recognition. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 994–999, 1997.

[3] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance model. *Proc. European Conference on Computer Vision*, 2:484–498, 1998.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Statist. Soc. B*, 39:1–38, 1977.

[5] P. Ekman and W. V. Friesen. *Unmasking the Face*. Prentice Hall, 1975.

[6] A. V. Nefian, L. Liang, X. Pi, X. Liu, and K. Murphy. Dynamic Bayesian networks for audio-visual speech recognition. *EURASIP Journal on Applied Signal Processing*, 2002(11):1–15, 2002.

[7] P. V. Overschee and B. D. Moor. A unifying theorem for three subspace system identification algorithms. *Automata*, 31(12):1853–1864, 1995.