Chapter 5

Modeling Timing Structures in Multimedia Signals

In this chapter, we propose a model to represent timing structures in multimedia signals, and exploit the model to generate a media signal from another related signal. The difference from the previous chapter is that we here show a general framework for modeling and utilizing mutual dependency among media signals based on the temporal relations among hybrid dynamical systems rather than only apply the system to each media signal and analyze the temporal structures among dynamic events.

5.1 Timing Structures in Multimedia Signals

Measuring dynamic human actions such as speech and musical performance with multiple sensors, we obtain multiple media signals across different modalities. We human usually sense and feel cross-modal dynamic features fabricated by multimedia signals such as synchronization and delay. For example, it is well-known fact that the simultaneity between auditory and visual patterns influences human perception (e.g., the McGurk effect [MM76]), and we can find some psychological studies about the audio-visual simultaneity (e.g., [FSKN04]).

On the other hand, modeling cross-modal structures is also important to realize multimedia systems (Figure 5.1); for example, human computer interfaces such as audio-visual speech recognition systems [NLP+02] and computer graphic techniques such as generating a media signal from another related signal (e.g., lip motion generation from input audio signals [Bra99]). Articulated motion model-

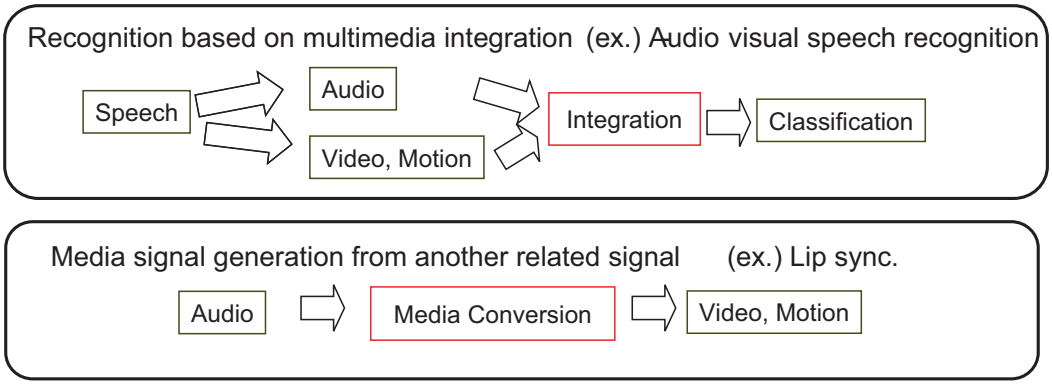


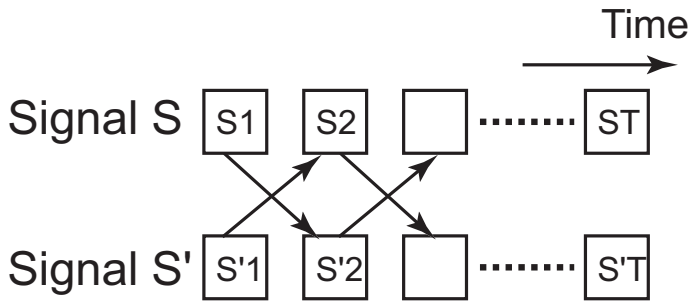
Figure 5.1: Applications of modeling cross-modal structures.

ing can also exploit this kind of temporal structures because motion timing among each different part plays an important role to realize natural motion generation.

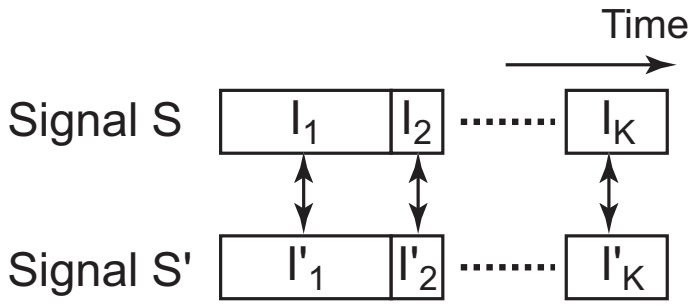
Dynamic Bayesian networks, such as coupled hidden Markov models (HMMs) [BOP97], are one of the most well-known methods to integrate multiple media signals [NLP+02]. These models describe relations between concurrent (co-occurred) or adjacent states of different media data (Figure 5.2(a) and 5.2(b)). A coupled HMM can be categorized into a frame-wise method because it models the frequency of state pairs that occur in adjacent frames. Although this frame-wise representation enables us to model short term relations or interaction among multiple processes, 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/; in addition, the motion always precedes the sound (Figure 5.3 left). We can see such an organized temporal difference in music performances also; performers often make preceding motion before the actual sound (Figure 5.3 right).

In this chapter, we propose a novel model that directly represents this important aspect of temporal relations, what we refer to as *timing structure*, such as synchronization and mutual dependency with organized temporal difference among multiple media signals (Figure 5.2(c)).

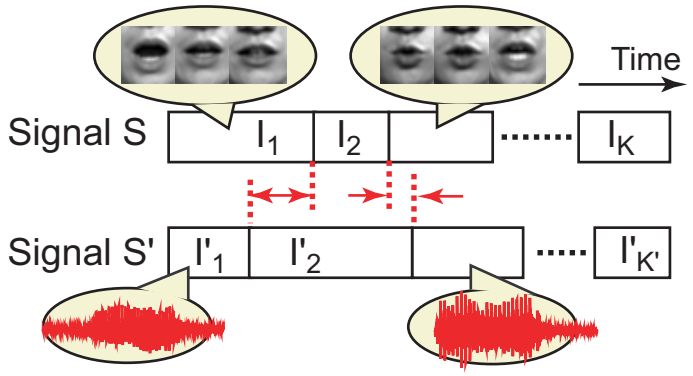
First, we assume that each media signal is described by a finite set of "modes" (i.e., primitive temporal patterns) similar to the previous chapter; we apply an interval-based hybrid dynamical system (interval system) to represent signal patterns in each media based on the modes. Then, we introduce a *timing structure model*, which is a stochastic model for describing temporal structure among in-



(a) Frame-wise modeling (adjacent time relations)



(b) Frame-wise modeling (co-occurrence)



(c) Timing based modeling

Figure 5.2: Temporal structure representation in multimedia signals.

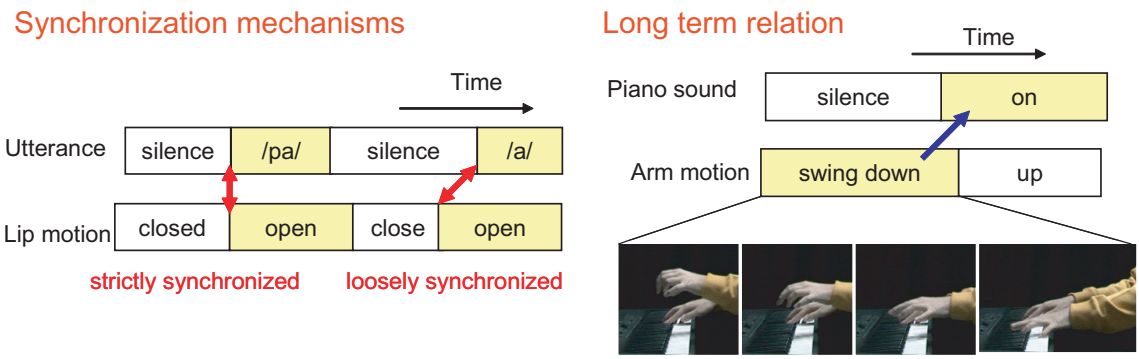


Figure 5.3: Open issues of existing multimedia co-occurrence models.

tervals in different media signals. The model explicitly represents temporal difference among beginning and ending points of intervals, it therefore provides a framework of integrating multiple interval systems across modalities as we will see in the following sections. Consequently, we can exploit the timing structure model to wide area of multimedia systems including human machine interaction systems in which media synchronization plays an important role. In the experiments, we verified the effectiveness of the method by applying it to media signal conversion that generates a media signal from another media signal.

As we described in Chapter 1, segment models [ODK96] can also be candidate models. Despite we use interval systems for experiments in this chapter, the timing structure model, which is proposed in this chapter, can be applicable for every model that provides an interval-based representation of media signals, where each interval is a temporal region labeled by one of the modes.

5.2 Modeling Timing Structures in Multimedia Signals

5.2.1 Temporal Interval Representation of Media Signals

To define timing structure, we assume that each media signal is represented by a single interval system, and the parameters of the interval system are estimated in advance (see [ODK96, LWS02], for example). Then, each media signal is described by an interval sequence. In the following paragraphs, we introduce some terms and notations for the structure and the model definition.

Media signals. Multimedia signals are obtained by measuring dynamic event with N_s sensors simultaneously. Let S_c be a single media signal. Then, multimedia signals become $\mathcal{S} = \{S_1, \cdots, S_{N_s}\}$. We assume that S_c is a discrete signal that is sampled by rate ΔT_c.

Modes and Mode sets. Mode $M_i^{(c)}$ is the property of temporal variation occurred in signal S_c (e.g., "opening mouth" and "closing mouth" in a facial video signal). We define a mode set of S_c as a finite set: $\mathcal{M}^{(c)} = \{M_1^{(c)}, \cdots, M_{N_c}^{(c)}\}$. Each mode is represented by a sub model of the interval system (i.e., linear dynamical systems).

Intervals. Interval $I_k^{(c)}$ is a temporal region that a single mode represents. Index k denotes a temporal order that the interval appeared in signal S_c. Interval $I_k^{(c)}$ has properties of beginning and ending time $b_k^{(c)}, e_k^{(c)} \in \mathbf{N}$ (the natural number set), and mode label $m_k^{(c)} \in \mathcal{M}^{(c)}$. Note that we simply refer to the indices of sampled order as "time". We assume signal S_c is partitioned into interval sequence $\mathcal{I}^{(c)} = \{I_1^{(c)}, ..., I_{K_c}^{(c)}\}$ by the interval system, where the intervals have no gaps or (i.e., $b_{k+1}^{(c)} = e_k^{(c)} + 1$ and $m_k^{(c)} \neq m_{k+1}^{(c)}$).

Interval representation of media signals. Interval representation of multimedia signals is a set of interval sequences: $\{\mathcal{I}^{(1)}, ..., \mathcal{I}^{(N_s)}\}$.

5.2.2 Definition of Timing Structure in Multimedia Signals

In this chapter, we concentrate on modeling timing structure between two media signals S and S'. (We use the mark " ' " to discriminate between the two signals.)

Let us use notation $I_{(i)}$ for an interval I_k that has mode $M_i \in \mathcal{M}$ in signal S (i.e.,$m_k = M_i$), and let $b_{(i)}, e_{(i)}$ be its beginning and ending time points, respectively. (We omit index k, which denotes the order of the interval.) Similarly, let $I'_{(p)}$ be an interval that has mode $M'_p \in \mathcal{M}'$ in the range $[b'_{(p)}, e'_{(p)}]$ of signal S'. Then, the temporal relation of two modes becomes the quaternary relation of the four temporal points $R(b_{(i)}, e_{(i)}, b'_{(p)}, e'_{(p)})$. If signal S and S' has different sampling rate, let the cycles be ΔT and $\Delta T'$, we have to consider the relation of continuous time such as $b_{(i)}\Delta T$ and $b'_{(i)}\Delta T'$ on behalf of $b_{(i)}$ and $b'_{(i)}$. In this subsection, we just use $b_{(i)} \in \mathbf{R}$(the real number set) for both continuous time and the indices of discrete time to simplify the notation.

Similar to Subsection 4.3.1 in the previous chapter, we can define timing structure as the relation R that can be determined by the combination of four binary

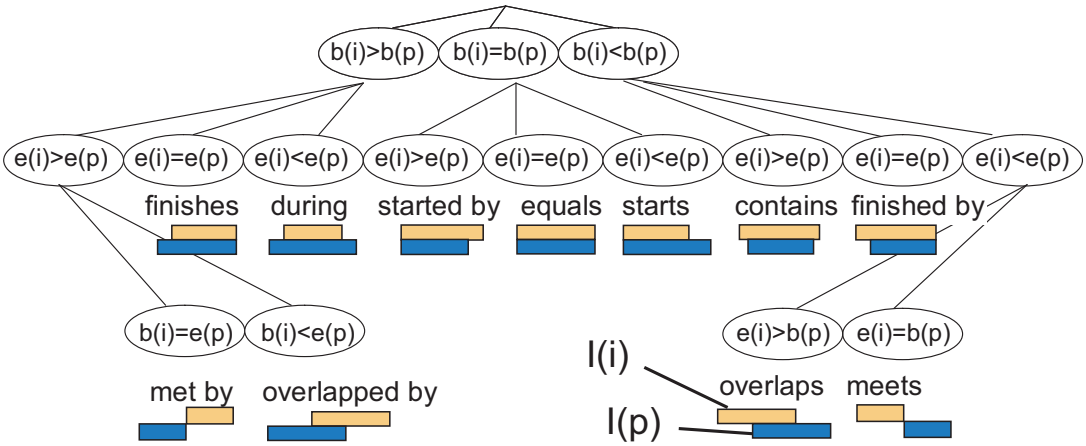


Figure 5.4: Overlapped interval relations (Subset of Figure 4.3(a)).

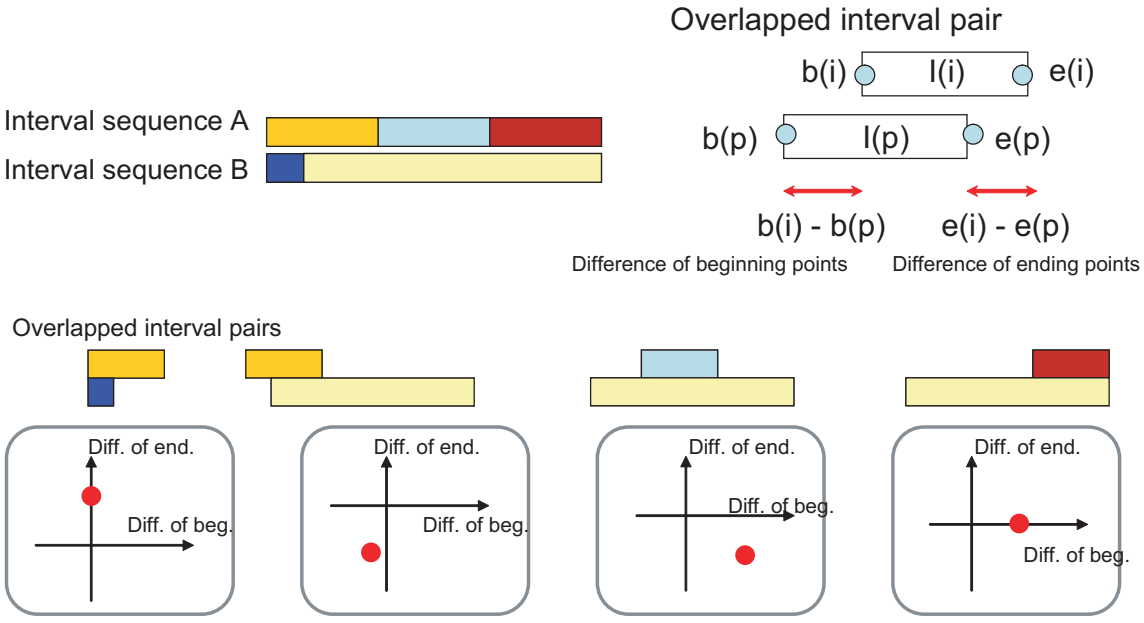


Figure 5.5: Examples of the metric relations. Four points represent temporal relations of two modes that appeared in overlapped intervals.

relations: $R_{bb}(b_{(i)}, b'_{(p)})$, $R_{be}(b_{(i)}, e'_{(p)})$, $R_{eb}(e_{(i)}, b'_{(p)})$, $R_{ee}(e_{(i)}, e'_{(p)})$. In the following, we specify the four binary relations that is suitable for modeling temporal structure in media signals (e.g., temporal difference between sound and motion).

We first introduce metric relations for R_{bb} and R_{ee} by assuming that R_{be} and R_{eb} is $R_{\leq}$ and $R_{\geq}$, respectively (i.e., the two modes have overlaps), as shown in Figure 5.4. This assumption is natural when the influence of one mode to the other modes with long temporal distance can be ignored. For the metric of R_{bb} and R_{ee}, we use temporal difference $b_{(i)} - b'_{(p)}$ and $e_{(i)} - e'_{(p)}$, respectively; the relation is represented by a point $(D_b, D_e) \in \mathbf{R}^2$ (see also Figure 4.3(b)).

Figure 5.5 shows some examples of the relations. There are three modes in interval sequence A, and two modes in interval sequence B. The four figures below represent the relations of mode pairs that appear in the overlapped interval pairs.

In the next subsection, we model this type of temporal metric relation using two-dimensional distributions. As a result, the model provides framework to represent synchronization and co-occurrence.

5.2.3 Modeling Timing Structures

Temporal Difference Distribution of Mode Pairs

To model the metric relations that described in the previous subsection, we introduce the following distribution for every mode pair $(M_i, M'_p) \in \mathcal{M} \times \mathcal{M}'$:

$$P(b_k - b'_{k'} = D_b, e_k - e'_{k'} = D_e | m_k = M_i, m'_{k'} = M'_p, [b_k, e_k] \cap [b'_{k'}, e'_{k'}] \neq \phi). \quad (5.1)$$

We refer to this distribution as a *temporal difference distribution* of the mode pair. As we described in Subsection 5.2.2, the domain of the distribution is $\mathbf{R}^2$.

Because the distribution explicitly represent the frequency of the metric relation between two modes (i.e., temporal difference between beginning points and the difference between ending points), it provides significant temporal structures for two media signals. For example, if the peak of the distribution comes to the origin, the two modes tend to be synchronized each other at the beginning and ending points, while if $b_k - b_{k'}$ has large variance, the two modes loosely synchronized at their onset timing.

To estimate the distribution, we collect all pairs of overlapping intervals that have the same mode pairs (Figure 5.6). Since training data is usually finite when

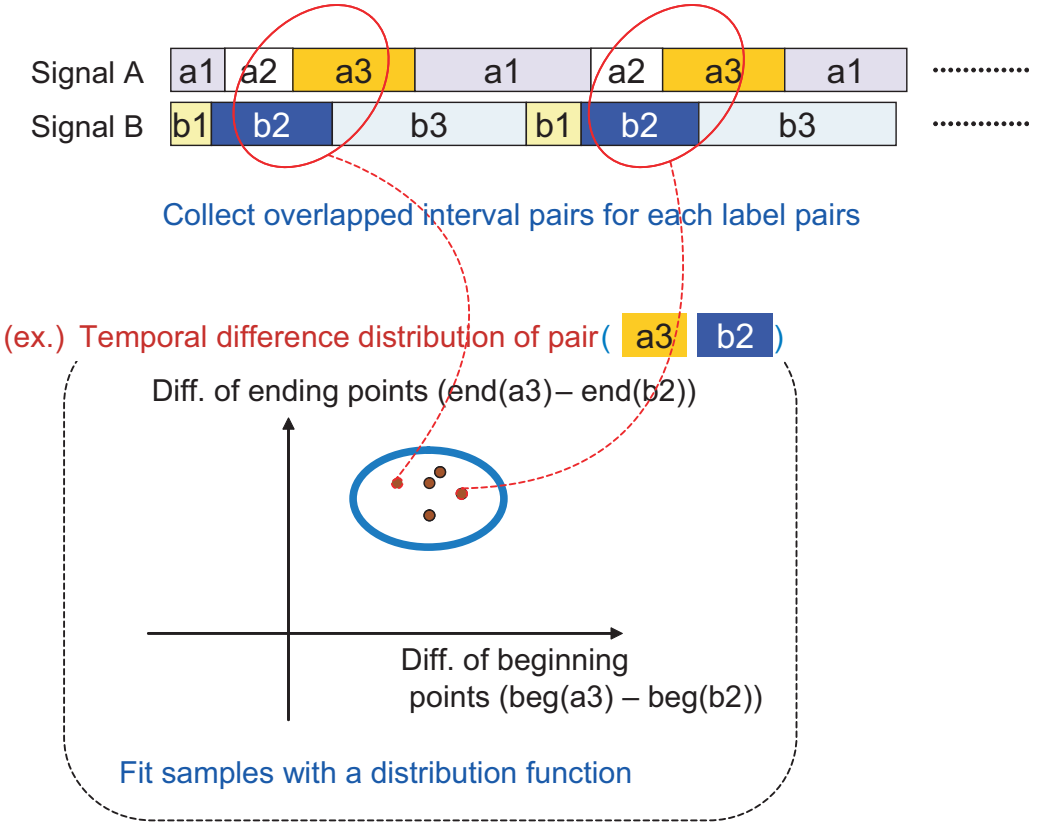


Figure 5.6: Learning of a timing structure model.

we use the model in real applications, we fit a density function such as Gaussian or its mixture models to the samples.

Co-occurrence Distribution of Mode Pairs

As we see in Equation (5.1), the temporal difference distribution is a probability distribution under the condition of the given mode pair. To represent frequency that each mode pair appears in the overlapped interval pairs, we introduce the following distribution:

$$P(m_k = M_i, m_{k'} = M'_p \mid [b_k, e_k] \cap [b'_{k'}, e'_{k'}] \neq \phi). \tag{5.2}$$

We refer to this distribution as *co-occurrence distribution* of mode pairs. The distribution can be easily estimated by calculating a mode pair histogram from every overlapped interval pairs.

Transition Probability of Modes

Using Equation (5.1) and (5.2), we can represent timing structure that is defined in Subsection 5.2.2. Although timing structure models temporal metric relations between media signals, temporal relation in each media signal is also important. Therefore, similar to previously introduced interval systems, we use the following transition probability of adjacent modes in each signal:

$$P(m_k = M_j | m_{k-1} = M_i) \, (M_i, M_j \in \mathcal{M}). \tag{5.3}$$

5.3 Media Signal Conversion Based on Timing Structures

Once we estimated the timing structure model that introduced in Section 5.2 from simultaneously captured multimedia data, we can exploit the model for generating one media signal from another related signal. We refer to the process as *media signal conversion*, and introduce the algorithm in this section.

The overall flow of media signal conversion from signal S' to S is as follows (see also Figure 5.7):

1. A reference (input) media signal S' is partitioned into an interval sequence $\mathcal{I}' = \{I'_1, ..., I'_{K'}\}$.

2. A media interval sequence $\mathcal{I} = \{I_1, ..., I_K\}$ is generated from a reference interval sequence $\mathcal{I}'$ based on the trained timing structure model. (K and K' is the number of intervals in $\mathcal{I}$ and $\mathcal{I}'$, and $K \neq K'$ in general.)

3. Signal S is generated from $\mathcal{I}$.

The key process of this media conversion lies in step 2. Since the methods of step 1 and 3 have been already introduced in Chapter 2, we here propose a novel method for step 2: a method that generates one media interval sequence from another related media interval sequence based on the timing structure model. In the following subsections, we assume that the two media signals S, S' have the same sampling rate to simplify the algorithm.

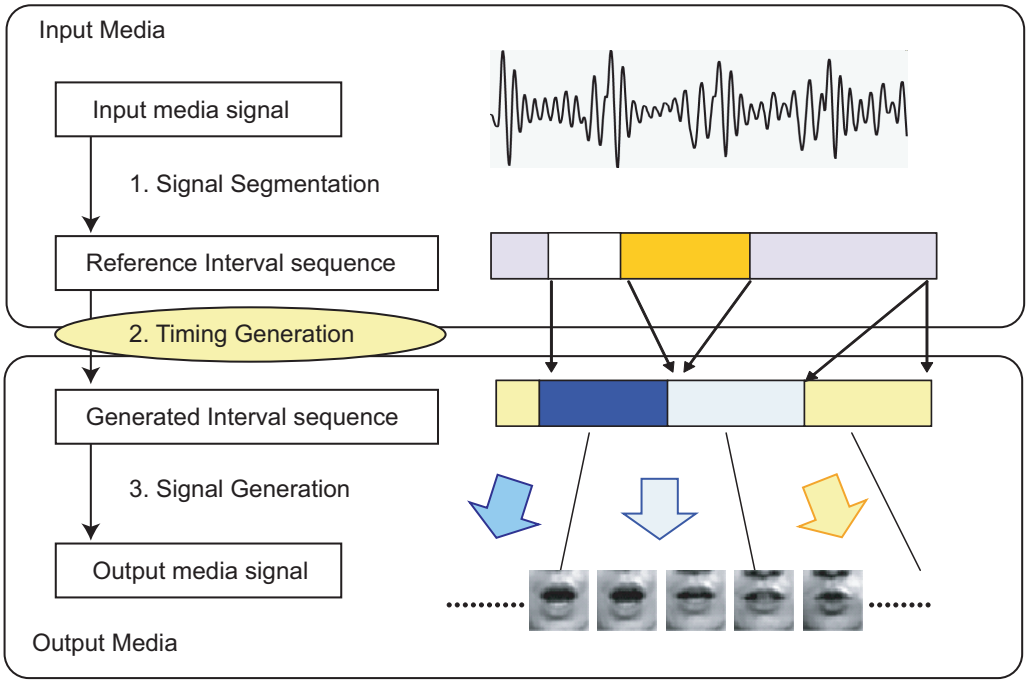


Figure 5.7: The flow of media conversion.

5.3.1 Formulation of Media Signal Conversion Problem

Let Φ be the timing structure model that is learned in advance (i.e., all the parameters described in Subsection 5.2.3 is estimated). Then, the problem of generating an interval sequence $\mathcal{I}$ from a reference interval sequence $\mathcal{I}'$ can be formulated by the following optimization:

$$\hat{\mathcal{I}} = \arg\max_{\mathcal{I}} P(\mathcal{I}|\mathcal{I}', \Phi). \tag{5.4}$$

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 $b_k, e_k \in [1, T]$ and $m_k \in \mathcal{M}$. Here, T is the length of signal S', and $\mathcal{M}$ is the mode set, which is estimated simultaneously with the signal segmentation. If we search for all the 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 (5.4), 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 (see Subsection 5.3.2 for details).

We currently do not consider online media signal conversion because it re-

quires a trace back step that finds partitioning points of intervals from the final to the first frame of the input signal. If online processing is necessary, one of the simplest method is dividing input stream comparatively longer range than the sampling rate and apply the following method to each of the divided range.

5.3.2 Interval Sequence Generation via Dynamic Programming

To simplify the notation, we omit the model parameter variable Φ in the following equations. Let us use notation $f_t = 1$ that denotes the interval "finishes" at time t, which is similar to the notation that we introduced in Subsection 2.3.2. Then, $P(m_t = M_j, f_t = 1|\mathcal{I}')$, which is the probability when an interval finishes at time t and the mode of time t becomes M_j in the condition of the given interval sequence $\mathcal{I}'$, can be calculated by the following recursive equation:

$$\begin{aligned} & P(m_t = M_j, f_t = 1|\mathcal{I}') \\ = & \sum_{\tau} \sum_{i\,(\neq j)} \left\{ \begin{array}{r} P(m_t = M_j, f_t = 1, l_t = \tau | m_{t-\tau} = M_i, f_{t-\tau} = 1, \mathcal{I}') \\ \times P(m_{t-\tau} = M_i, f_{t-\tau} = 1|\mathcal{I}') \end{array} \right\}, \end{aligned} \quad (5.5)$$

where l_t is a duration length of an interval (i.e., it continues l_t at time t) and m_t is a mode label at time t. The lattice in Figure 5.8 depicts the path of the above recursive calculation. Each pair of arrows from each circle denotes whether the interval "continues" or "finishes", and every bottom circle sums up all the finishing interval probabilities.

The following dynamic programming algorithm is deduced directly from the recursive equation (5.5):

$$\begin{aligned} E_t(j) \;=\; & \max_{\tau} \max_{i\,(\neq j)} \underline{P(m_t = M_j, f_t = 1, l_t = \tau | m_{t-\tau} = M_i, f_{t-\tau} = 1, \mathcal{I}')} E_{t-\tau}(i), \\ & \text{where} \quad E_t(j) \triangleq \max_{m_1^{t-1}} P(m_1^{t-1}, m_t = M_j, f_t = 1|\mathcal{I}'). \end{aligned} \quad (5.6)$$

$E_t(j)$ denotes the maximum probability when the interval of mode M_j finishes at time t, and is optimized for the mode sequence from time 1 to $t-1$ under the condition of given $\mathcal{I}'$. The probability with underline denotes that interval I_k with a triple $(b_k = t - \tau + 1, e_k = t, m_k = M_j)$ occurs just after the interval I_{k-1} that has mode $m_{k-1} = M_i$ and ends at $e_{k-1} = t - \tau$. We refer to this probability as an *interval transition probability*.

We recursively calculate the maximum probability for every mode that fin-

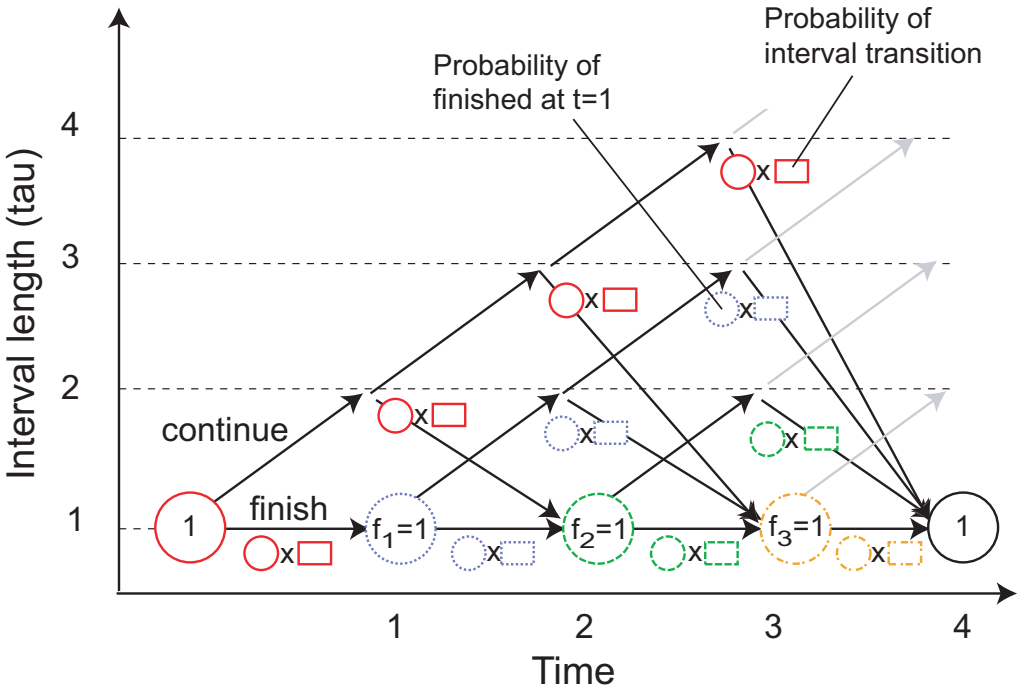


Figure 5.8: Lattice to search optimal interval sequence (num. of mode =2). We assume that $\sum_j P(m_T = M_j, f_T = 1|\mathcal{I}') = 1$

ishes at time $t(t = 1, ..., T)$ using Equation (5.6). After the recursive calculation, we find the mode index $j^* = \arg\max_j E_T(j)$. Then, we can get the duration length of the interval that finishes at time T with mode label M_{j*}, if we preserve τ that gives the maximum value at each recursion of Equation (5.6). Repeating this trace back, we finally obtain the optimized interval sequence and the number of intervals.

The remaining problem for the algorithm is the method of calculating the interval transition probability. As we see in the next subsection, this probability can be estimated from a trained timing structure model.

5.3.3 Calculation of Interval Transition Probability

As we described in previous subsection, the interval transition probability appeared Equation (5.6) is the transition from interval I_{k-1} to I_k. To simplify the notation, we here replace $t - \tau + 1$ with B_k. Let $e_{\min} = B_k$ and $e_{\max} = \min(T, B_k + l_{\max} - 1)$ be the minimum and maximum values of e_k, where $l_{\max}$ is the maximum length of the intervals. Let $I'_{k'}, ..., I'_{k'+R} \in \mathcal{I}'$ be reference inter-

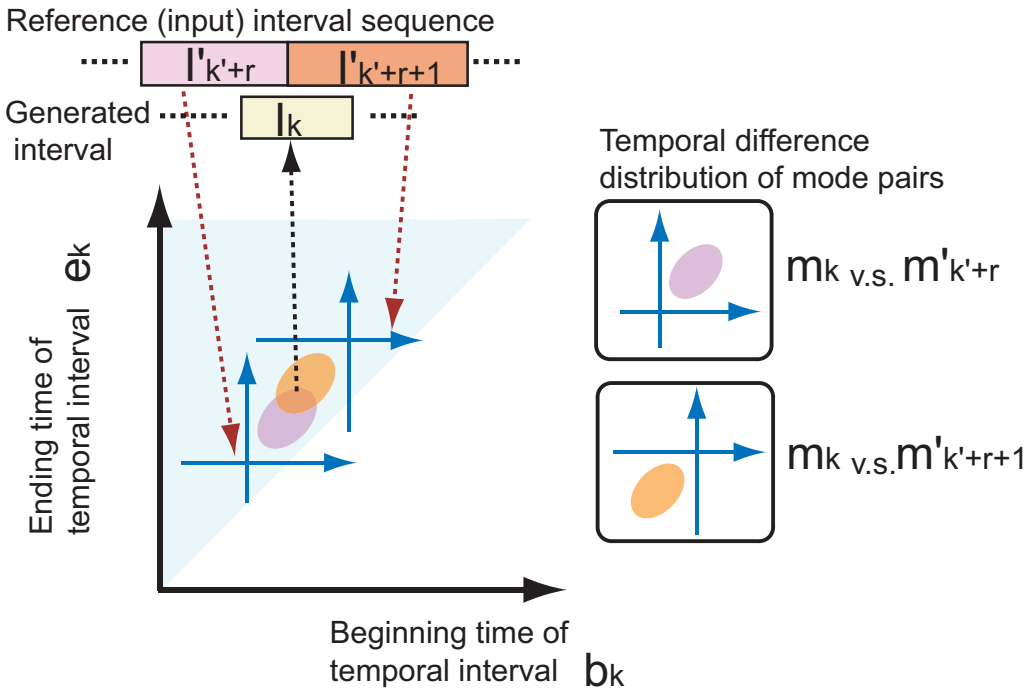


Figure 5.9: An interval probability calculation from the trained timing structure model.

vals that are possible to overlap with I_k. Assuming that the reference intervals are independent of each other (this assumption empirically works well), the interval transition probability can be calculated by the following equation:

$$\begin{aligned} & P(m_t = M_j, f_t = 1, l_t = \tau | m_{t-\tau} = M_i, f_{t-\tau} = 1, \mathcal{I}') \\ = \; & P(m_k = M_j, e_k, e_k \in [e_{\min}, e_{\max}] | m_{k-1} = M_i, b_k = B_k, I'_{k'}, ..., I'_{k'+R}) \\ = \; & \prod_{r=0}^{R} \{ \mathrm{Rect}(e_k, e_k \in [e_{\min}, b'_{k'+r} - 1]) \\ & + \kappa_r P(m_k = M_j, e_k, e_k \in [b'_{k'+r}, e_{\max}] | m_{k-1} = M_i, b_k = B_k, I'_{k'+r}) \} , \end{aligned} \tag{5.7}$$

where $\mathrm{Rect}(e, e \in [a, b]) = 1$ in the range $[a, b]$; else 0. Since the domain of e_k is $[e_{\min}, e_{\max}]$, Rect is out of range when $r = 0$, and $b'_{k'} = e_{\min}$. κ is a normalizing factor: $\kappa_r = 1$ $(r = 0)$ and

$$\kappa_r = P(m_k = M_j, e_k, e_k \in [b'_{k'+r}, e_{\max}] | b_k = B_k, m_{k-1} = M_i)^{-1} \quad (r = 1, ..., R).$$

In the experiments, we assume κ_r is uniform for (m_k, e_k); thus, $\kappa_r = N(e_{\max} -$

$e_{\min}+1$) (N is the number of modes).

Using some assumptions that we will describe later, we can decompose the probability in Equation (5.7) as follows:

$$\begin{aligned} & P(m_k = M_j, e_k, e_k \in [b'_{k'+r}, e_{\max}] \mid m_{k-1} = M_i, b_k = B_k, I'_{k'+r}) \\ = \; & P(e_k \mid e_k \in [b'_{k'+r}, e_{\max}], m_k = M_j, b_k = B_k, I'_{k'+r}) \\ & \times P(m_k = M_j | e_k \in [b'_{k'+r}, e_{\max}], m_{k-1} = M_i, b_k = B_k, I'_{k'+r}) \\ & \times P(e_k \in [b'_{k'+r}, e_{\max}] \mid m_{k-1} = M_i, b_k = B_k) \end{aligned}$$

The first term is the probability of e_k under the condition that I_k overlaps with $I'_{k'+r}$. We assume that it conditionally independent of m_{k-1}. This probability can be calculated from Equation (5.1). Here, we omit the details of the deduction, and just make an intuitive explanation using Figure 5.9. First, an overlapped mode pair in I_k and $I'_{k'+r}$ provides a relative distribution of $(b_k - b'_{k'+r}, e_k - e'_{k'+r})$. Since $I'_{k'+r}$ is given, the relative distribution is mapped to the absolute time domain (the upper triangle region). Normalizing the distribution of (b_k, e_k) for $e_k \in [b'_{k'+r}, e_{\max}]$, we obtain the probability of the first term. The second term can be calculated using Equation (5.2) and (5.3). For the third term, we assume that the probability of $e_k \geq b'_{k'+r}$ is independent of $I'_{k'+r}$. Then, this term can be calculated by modeling temporal duration length l_t. In the experiments, we assumed uniform distribution of e_k and used $(e_{\max} - b'_{k'+r})/(e_{\max} - e_{\min} + 1)$.

The computational cost of interval transition probabilities strongly depends on the maximum interval length $l_{\max}$. If we successfully estimate the modes, $l_{\max}$ becomes comparatively small (i.e., balanced among modes) than the total input length. Thus, the cost becomes reasonable.

5.4 Experiments

To evaluate the descriptive power of the proposed timing structure model and the performance of the media conversion method, we first used simulated data for the verification of the interval generation algorithm described in Subsection 5.4.1. We then conducted the experiment that examines the overall media conversion flow shown in Figure 5.7 using audio and video data, and evaluated the precision of lip video generation from an input audio signal in Subsection 5.4.2.

5.4.1 Evaluation of Interval Sequence Generation Algorithm Using Simulated Data

To verify the interval generation algorithm described in Subsection 5.3.2, we input an interval sequence that comprised two modes $\mathcal{M}' = \{M'_1, M'_2\}$ and attempted to generate another related interval sequence that comprised $\mathcal{M} = \{M_1, M_2, M_3\}$ based on manually given temporal difference distributions.

Each of the temporal difference distribution was assumed to be a Gaussian function. Let $\mu_{i,p}$ be a mean vector of the temporal difference distribution of mode pair (M_i, M'_p) where $M_i \in \mathcal{M}$ (i.e., mode set for generating interval sequences) and $M'_p \in \mathcal{M}'$ (i.e., mode set for input interval sequences). Mean vectors $\mu_{i,p}$ $(i = 1, 2, 3, p = 1, 2)$ were manually decided as follows:

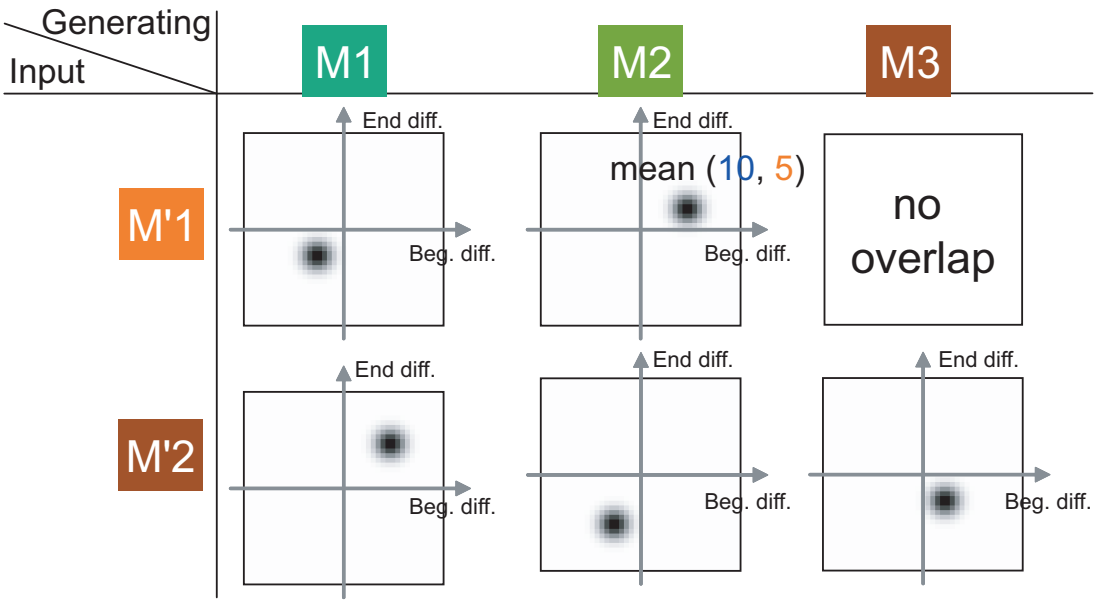
$$\begin{array}{lll} \mu_{1,1} = (-5, -5), & \mu_{2,1} = (10, 5), & \mu_{3,1} : \text{not available}, \\ \mu_{1,2} = (10, 10), & \mu_{2,2} = (-5, -10), & \mu_{3,2} = (5, -5), \end{array} \tag{5.8}$$

where mode pair (M_3, M'_1) was assumed to have no overlap. All the variances were set to be 4 and all the covariances were assumed to be zero. Figure 5.10 (a) shows the assumed temporal difference distributions.

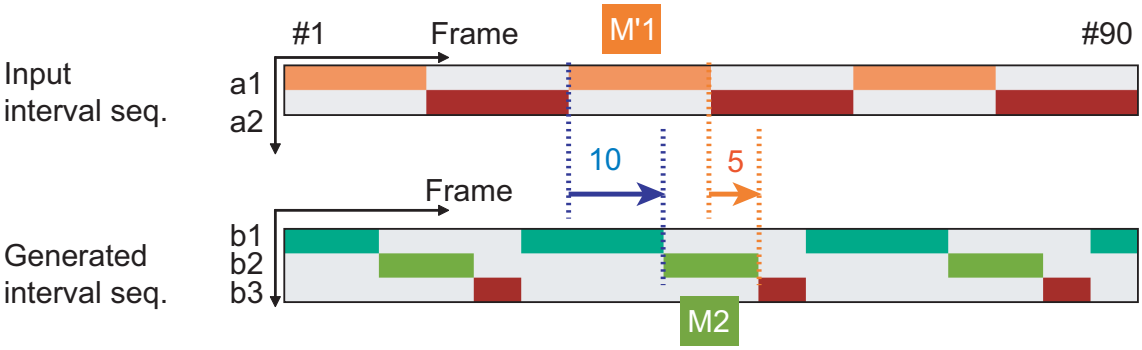
As for co-occurrence distribution defined in Equation (5.2), uniform distribution was assumed. That is, the probabilities were set to be 0.2 for all the mode pairs except pair (M_3, M'_1). As for mode transition probabilities, transition probabilities from M_1 to M_2, M_2 to M_3, and M_3 to M_1 were set to be one, and the remaining were set to be zero for generating cyclic transition of modes.

Then, the interval sequence shown in Figure 5.10 (b) (upper) was used as input of the interval generation algorithm described in Subsection 5.3.2. Figure 5.10 (b) (bottom) shows the generated interval sequence using the algorithm. We see that the temporal differences between beginning and ending points correspond to the elements of mean vectors $\mu_{i,p}$ of Gaussian distributions. For example, we see that the mode M_2 always begins ten frames after the beginning point of M'_1 begins, and finishes five frames after the ending point of M'_1.

We also examined other simulated data using different conditions such as the number of input modes is larger than that of generating modes. In those several experiments, we checked that the proposed algorithm always generated interval sequences in which each temporal interval of modes was determined so as to maximize the probability in Equation (5.4) with respect to given parameters. Consequently, the proposed timing structure model, especially temporal differ-



(a) Manually given temporal difference distributions (Gaussian).



(b) The input and generated interval sequences.

Figure 5.10: Verification of interval sequence generation from another related interval sequence described in Subsection 5.3.2 using manually given timing distributions.

ence distributions, successfully determines the temporal relation among modes appeared in input and generated interval sequences.

5.4.2 Evaluation of Image Sequence Generation from an Audio Signal

We applied the media conversion method described in Section 5.3 to the application that generates image sequences from an audio signal.

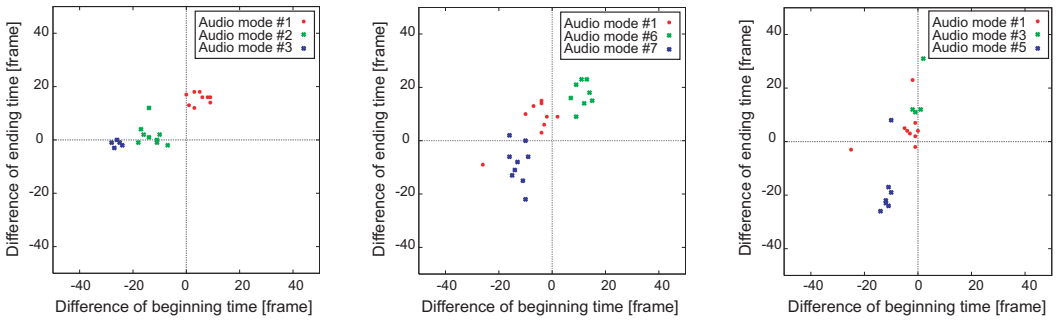
Feature Extraction

A continuous utterance of five vowels /a/,/i/,/u/,/e/,/o/ (in this order) was captured using mutually synchronized camera and microphone. This utterance was repeated nine times (18 sec.). The resolution of the video data was 720×480 and the frame rate was 60 fps. The sampling rate of the audio signal was 48 kHz (downsampled to 16 kHz in the analysis). Then, we applied short-term Fourier transform to the audio data with the window step of 1/60msec; thus, the frame rate corresponds to the video data.

Filter bank analysis was used for the audio feature extraction. We obtained 1134 frames of audio feature vectors each of which had dimensionality of 25, which corresponded to the number of filter banks. As for the video feature, a lip region in each video image was extracted by the active appearance model (AAM) [CET98] described in Section 4.4 (see also Appendix D for details). Then, the lip regions were downsampled to 32×32 pixels and the principal component analysis (PCA) was applied to the downsampled lip image sequence. Finally, we obtained 1134 frames of video feature vectors each of which had dimensionality of 27, which corresponded to the number of used principal components.

Learning the Timing Structure Model

Using the extracted audio and visual feature vector sequences as signal S' and S, we estimated the number of modes, parameters of each mode, and the temporal partitioning of each signal. We used linear dynamical systems for the models of modes. To estimate the parameters, we exploited hierarchical clustering of the dynamical systems described in Section 3.3. The estimated number of modes was 13 and 8 for audio and visual modes, respectively. The segmentation results are



(a) visual mode #1 (b) visual mode #5 (c) visual mode #7

Figure 5.11: Scattering plots of temporal difference between overlapped audio and visual modes. Visual mode #1, #5, and #7 corresponds to lip motion /o/ → /a/, /e/ → /o/, and /a/ → /i/, respectively

shown in Figure 5.12 (the first and second rows). Because of the noise, some vowels were divided into several different audio modes.

Temporal difference distributions of Equation (5.1), co-occurrence distributions of Equation (5.2), and mode transition probabilities of Equation (5.3) were estimated from the two interval sequences obtained in the segmentation process. Figure 5.11 shows the scattered plots of the samples that are temporal difference between beginning points and ending points of the overlapped modes appeared in the two interval sequences. Each chart shows samples of one visual mode to typical (two or three) audio modes. We see that the beginning motion from /a/ to /i/ synchronized with the actual sound (right chart) compared to the motion from /o/ to /a/ (left) and from /e/ to /o/ (middle). Applying Gaussian mixture models to these distributions, we estimated the temporal difference distributions. The numbers of the mixtures were manually determined.

Evaluation of Timing Generation

Based on the estimated cross-media timing-structure, we applied the media conversion method in Section 5.3. We used an audio signal interval sequence included in the training data of the interval system as an input (reference) media data (top row in Figure 5.12) and converted it into a video signal interval sequence (third row in Figure 5.12).

Then, to verify the performance of the media conversion method, we first compared the converted interval sequence with the original one, which was generated

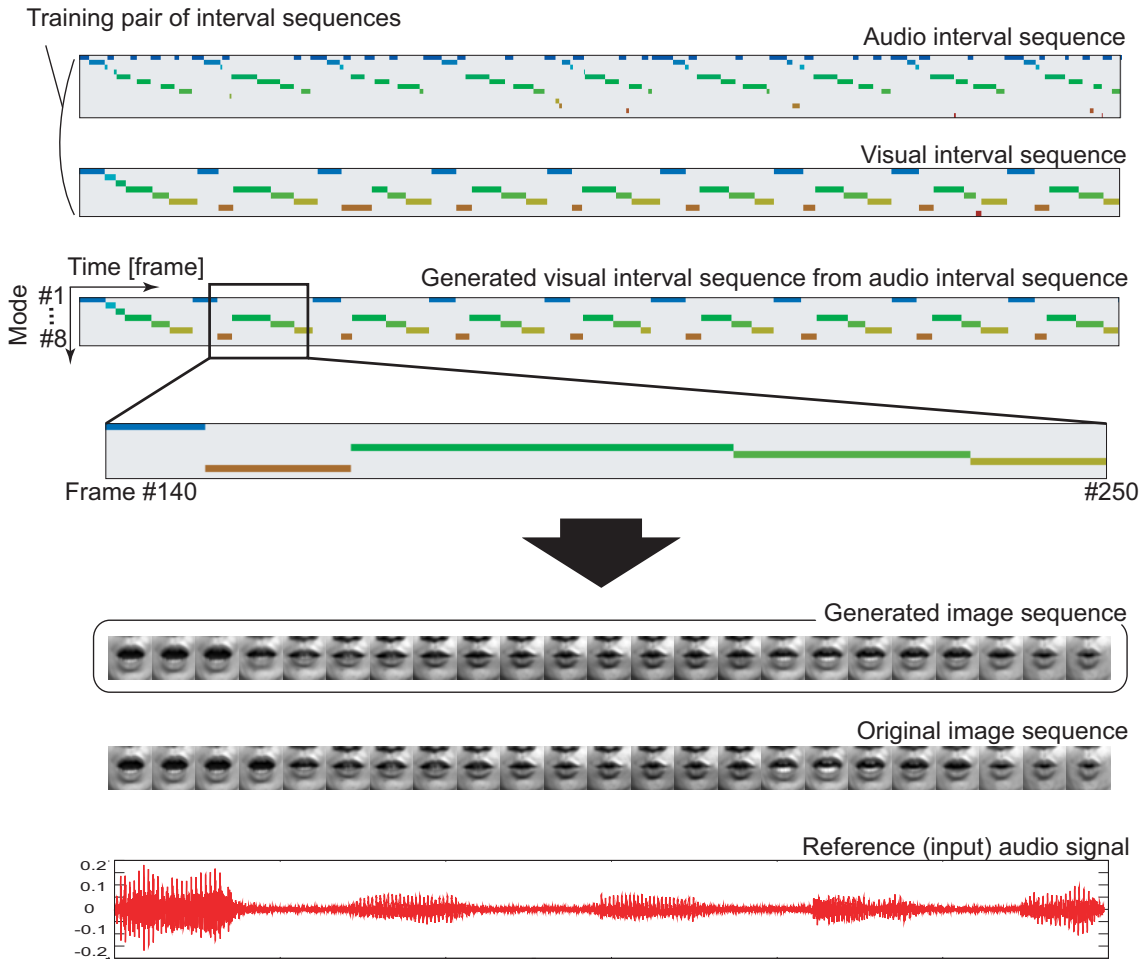


Figure 5.12: Generated visual interval sequence and an image sequence from the audio signal.

from the video data measured simultaneously with the input audio data (second row in Figure 5.12). Moreover, we also compared the pair of video data: one generated from the converted interval sequence (third bottom row in Figure 5.12) and the originally captured one (second bottom row in Figure 5.12), where images from frame #140 to #250 were shown in Figure 5.12. As for step 3, these image sequences were decoded from the visual feature vectors by the linear combination of principal axes (eigenvectors of PCA) and feature vectors (principal components). We also see the visual motion precedes the actual sound by comparing to the wave data (in the bottom row). From these data, the media conversion method seemed to work very well.

To quantitatively compare our method with others using a cross-validation method, we generated feature vector sequences based on several regression

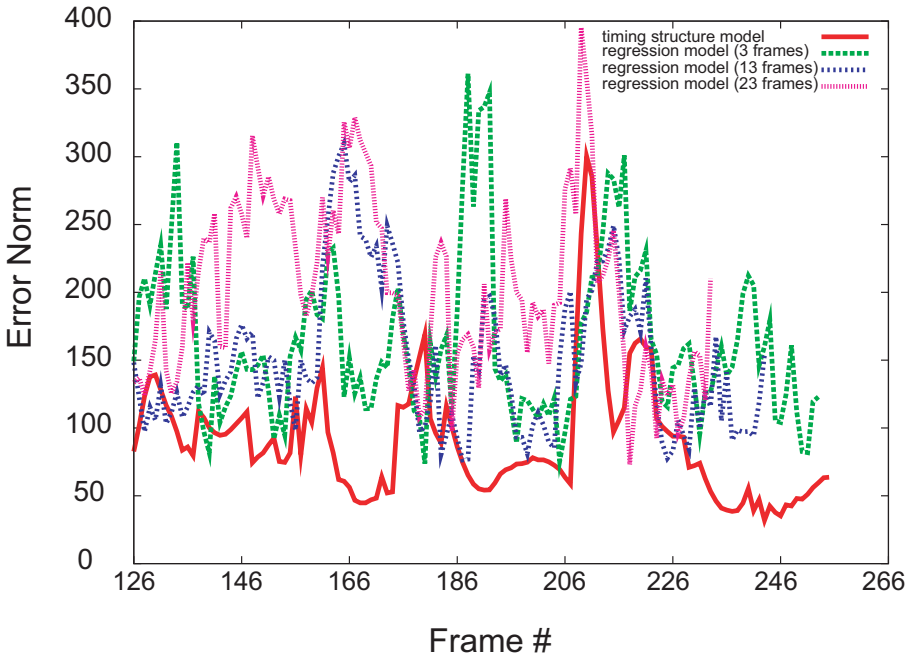
models. Seven regression models were constructed; each of the models estimated visual feature vector y_t from $2a+1$ frames of audio feature vectors $y_{t-a}, y_{t-a+1}, ..., y_t, ..., y_{t+a}$, where $a = 1, 3, 5, 6, 7, 9, 11$. For the cross validation, we used eight sequences in the nine utterance sequences for the training and one for the test. We tested all the possible combinations and averaged the errors. Figure 5.13 (a) shows the error norm of each frame in the range of frame # 126 to #255. We see that the generated sequence based on the learned timing structure model has small error values compared to other regression models in most part except some ranges such as around frame #170. One of the reasons the error of the timing-based method was larger than regression models is that these regions corresponded to such as vowel /i/, so the sound and visual motion might be synchronized well. Figure 5.13 (b) shows the average error norm per frame of each model. All the generated frames were used to calculate the average values. We see that the timing-based model provide the smallest error compared to the regression models [1].

5.5 Discussion

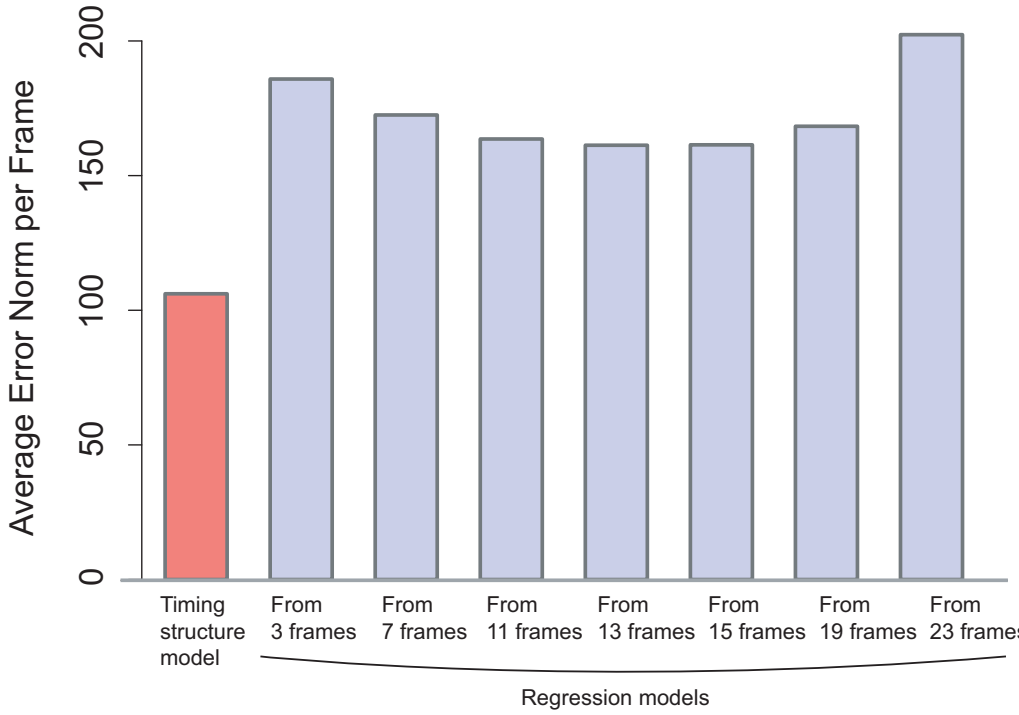
We proposed a timing structure model that explicitly represents dynamic features in multimedia signals using temporal metric relations among intervals. The experiment shows that the model can be applied to one media signal from another signal across the modalities.

Although this is a preliminary result of evaluating the proposed timing models, its basic ability for representing temporal synchronization is expected to be useful for wide variety of areas. For example, human machine interaction systems including speaker tracking and audio-visual speech recognition, computer graphics such as generating motion from another related audio signals, and robotics such as calculating motion of each joint based on input events. We will discuss these points in Chapter 6 as feature work.

[1] Correction has been made for bug fix in this paragraph and Fig.5.13.



(a) Error norm of each frame in the range of frame #140 to #250.



(b) Average error norm (per frame).

Figure 5.13: Error norm of each frame and its average per frame between generated sequences and original sequence.