Music-Driven Dance Synthesis by Multimodal Dance Performance Analysis

by

Yasemin Demir

A Thesis Submitted to the
Graduate School of Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Electrical & Computer Engineering

Koç University

December 2008

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Yasemin Demir

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Committee Members:

Assist. Prof. Engin Erzin

Assist. Prof. Yücel Yemez

Prof. A. Murat Tekalp

Asst. Prof. Oğuz Sunay

Asst. Prof. Çağatay Başdoğan

Date:

To my family

ABSTRACT

We present a framework for audio-visual analysis of dance performances towards the goal of music-driven dance synthesis. Dance figures, which are performed synchronously with the musical rhythm, can be analyzed through the audio spectra using spectral and chromatic musical features. In the proposed multimodal dance performance analysis system, dance figures are manually labeled over the video stream and modeled by employing HMMs. The music segments, which correspond to beat and meter boundaries, are used to train hidden Markov model (HMM) structures to learn meter related temporal audio patterns which are correlated with the dance figures. Bi-gram based co-occurences of temporal audio patterns and dance figures are calculated. and bi-gram based co-occurrence performances for two different audio feature streams are evaluated. In our evaluations, mel-scale cepstral coefficients (MFCC) with their first and second derivatives and chroma features are used as our candidate audio feature set. The proposed framework in this thesis, can be used towards analysis and synthesis of audio-driven human body animation.

ÖZETÇE

Bu tezde çoklu modelli dans performans analizi ile müzikle sürülen dans sentezinin yapılabilmesi amaçlanmıştır. Dans figürleri, müzik harmonisi ile uyumlu, müzik ritmiyle eşzamanlidir ve işitsel öznitelikler kullanılarak analiz edilebilir. Dans figürleriyle ilintili işitsel örüntüleri belirlemek amacıyla, işitsel veri spektrumuna ait öznitelikler, SMM yapıları kullanılarak modellenmiştir. Elde edilen işitsel örüntüler ve dans figürleri arasındaki ilinti, eşzamanlı gerçekleşme performansları hesaplanarak değerlendirilmiştir. İşitsel veri olarak mel frekansı sepstral katsayıları ve renksel parlaklığı temsil eden kroma öznitelikleri ele alınmıştır. Bu tezde sunulan sistem kullanılarak, işitsel veriyle sürülen vücut animasyonu sentezi çalışmalarının geliştirilmesi amaçlanmaktadır.

ACKNOWLEDGMENTS

There are several people I wish to thank for the precious support they provided to my research during this study, which contributed to my personal and professional development. First and foremost, I would like to thank my advisor and co-supervisors Prof. Murat Tekalp, Prof. Engin Erzin and Prof. Yucel Yemez for having given me the opportunity to join their group and the inspiration of doing research in an exciting project with both theoretical and experimental aspects. Thank you for your great support throughout this research project. Your truly scientist intuition has inspired my growth as a student. I also would like to acknowledge TUBITAK for their full support during this research project.

I owe my great thanks to Ferda Ofli for his support and encouragement in every step of this research project.

Special thanks go to my friends Gokhan Uzunbas, Selda Yildiz, Zeynep Kaymak, Bahar Ondul, Besray Unal, Erinc Dikici, Barlas Oguz, Kayhan Eritmen, Kevser Aydemir, Gizem Karsli, Ozge Engin, Nurcan Tuncbag, Gozde Kar, Sefer Baday, Osman Yogurtcu, Gokay Ertem, Selcuk Bucak, Cagdas Tuna, Ulas Bagci, Cagri Akargun, Mert Gur, Sina Ocal, Umut Ozcan, Hakan Dogan, Ilgar Veryeri, Ekin Tuzun, Cengiz Ulubas, Beyza Morgan, Koray Ozdemir, Tugce Karaca, Basri Cavusoglu. This paper would not have been possible without their endless support and encouragement that was always by my side. The burden of writing this thesis was lessened by the support and humor of all these lovely friends.

Last and most importantly, I am eternally grateful to my parents, my brother and my sister who raised me, taught me, supported me, and loved me. I am sincerely thankful to my brother and sister for their support and encouragement in various ways throughout all my studies. To my family I dedicate this thesis.

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

Installing human perception into a robot using audio-visual observations is one of the exciting problems in humanoid robot studies [1]. Human emotion recognition through audio-visual observations is a challenging problem in computer animation studies, in which animations are synchronized with an appropriate audio/speech in order to influence the spectators sensually [2, 3, 4]. Another challenging problem is audio-visual body analysis which has been increasingly studied in the last ten years. Modeling human locomotion (walking, running) using Hidden Markov Models (HMMs) is studied in [5]. The resulting HMM structure is observed to be insufficient to recognize more complex motion such as dancing.

The audio-visual analysis of dance performances to build audio driven body animation can be viewed as a special case of the general audio-visual body analysis problem [6, 7, 8], which can be investigated under four main tasks: i) tracking and estimation of motion parameters over the visual stream, ii) extracting temporal descriptors of audio signals iii)correlation analysis of audio and visual descriptors, and iv) building joint audio-visual models to define and synthesize dance figures of a body animation. In a recent study, we analyzed dance figures of a dancing person over an audio-visual database [6]. The 3D positions of body joints were first extracted by tracking markers on the human body. HMM structures were then used to model motion dynamics. Feature level audiovisual correlation analysis was finally performed using mel-scale cepstral co-efficients (MFCCs) for audio and 3D positions of body joints. Although we obtained encouraging results for basic dance sequences, an investigation and evaluation of candidate spectral audio features is needed to analyze complex dance figures. In order to find a solution for analyzing such kind of complex motion structures, human rhythm perception should be taken into account which is also a popular research area in psychology. Thus, much research has been done on modeling the human understanding on rhythmic patterns from a given audio signal [9, 10, 11, 12]. In

addition to these recent studies, in [13], dance motion is detected through musical analysis under the same assumption that dance motion primitives are synchronized with the musical rhythm. Dance motion primitives are extracted along with the melody and the musical rhythm structure. The experimental results confirmed that the primitive motions are in accordance with the musical rhythm for the given dance sequence. Likewise, there exist other studies for estimation of dancing skills based on the same rhythmic factors [14, 15].

In particular cases, such as traditional dances relying on non-rhythmic musical structures, rhythm information, itself, becomes insufficient to synthesize dance figures. We need extra audio features to best describe different dance figures. Computational Auditory Scene Analysis, which is a common name for audio analysis methods, would serve a key point for the particular cases by clarifying these special audio features that human able to perceive [16, 17].

Extracting meaningful patterns automatically is one of the main studies in music summarization. One of the first works of these area is proposed by Logan and Chu, who describe an agglomerative clustering for determining the repeating parts and a hidden Markov model (HMM) solution for generating the key phrases [17]. Mel- frequency cepstral coefficients (MFCCs) extracted for each audio frame serve as the audio features. Frames are clustered together according to the clustering method they use. Ergodic HMMs are trained with only a few states under the assumption that each state represent a musical structure to which Viterbi decoding applied. In another approach, spectral envelope serves as the audio feature [18]. These studies come up with a result that the number of states used in HMMs is not sufficient to extract the temporal patterns. An additional solution [19] which increase the number of states of HMMs and frame size for each audio feature set. They calculated a state histogram over 15 frames after training HMMs and used these histograms to cluster each frame according to a cost function optimization process. Different control parameters to determine a meaningful in-state duration, and the clustering method is adopted a context-aware method called fuzzy C-means in [20, 21].

A alternative solution to find similarities between frames over an audio signal is constructing a self-similarity matrix [22]. The resulting similarity matrix include visible areas which belongs to similar timbral characteristics in an audio signal. These visible area boundaries is used in clustering of similar patterns in [23]. In [24], a spectral clustering method is

applied to meaningful patterns. Audio features describing the tonal content of the signal, e.g, chroma features instead of general timbre related features as MFCCs are used for the evaluations. The similarity results are located in off-diagonal lines instead of rectangular areas demonstrating the similar areas in the previous similarity evaluations. These two approaches are compared [25]. The comparison results show that HMM approach is more robust than the similarity matrix method in clustering the similar parts.

Audio rhythm and spectral structure of each rhythm segment are the key point for synthesis of dance figures. Humans are capable of perceiving rhythmic structures of audio, even a baby has the ability of waving his body harmonically with the music. These perceivable rhythmic structures reoccur through the audio and help researchers for modeling the dance-music correlation. For the purpose of modeling the music and extracting the repeating rhythmic structures spectral audio features are evaluated in [26, 27]. Audio tracks are comprised of repeating pieces that are recognizable. Evaluation of audio features for audio-visual analysis aims to extract these repeating patterns for a given musical excerpt. The temporal repeating patterns within beat locations, and the repeating meaningful combinational sets of these repeating patterns within measure boundaries can be determined as the result of evaluations.

In this thesis, we investigate the correlation of audio and visual descriptors to generate a music-driven multimodal dance performance. Although considerable research has been devoted to audio rhythmic pattern extraction, rather less attention has paid to consubstantiate different audio features and extract temporal information by employing HMMs which are trained different number of state and branch combinations. Occurrences of temporal repeating audio patterns and dance figures is investigated based on bi-grams of dance figures.

In Chapter 2, multimodal dance is defined and a system overview is given. In Chapter 3, audio analysis blocks are described. In Section 3.1, we introduce audio beat extraction, measure and meter detection algorithm. In Section 3.2, we describe our audio features. In the following sections, Section 3.3 and Section 3.4 we give details of beat and meter clustering respectiveley. In Chapter 4, we describe audio-figure mapping which is as multimodal correlation analysis. In Section 4.1 we describe manual labeling of training videos. In Section 4.2, we describe tracking of feature points and extraction of elementary dance figure

related video patterns is given in Section 4.3. In Section 4.4 we introduce joint meter-figure mapping in detail. We present experimental results in Chapter 5. Finally, in Chapter 6 we give our conclusions and suggestions for future work.

Chapter 2

MULTIMODAL DANCE MODEL AND SYSTEM OVERVIEW

2.1 Multimodal Dance Model

A comprehensive formal language that describes contra dances and similar folk dances which can be read and understood by both software and humans has been proposed in [28]. In this thesis, our eventual goal is to carry out such audio-driven dance animation by temporal audio analysis. In our scenario, actor performs traditional dances and specifically generates different number of dance figures synchronous with a audio for each dance type.

2.2 System Overview

Block diagram of the multimodal audio-visual analysis and synthesis system is given in Fig. 2.1.

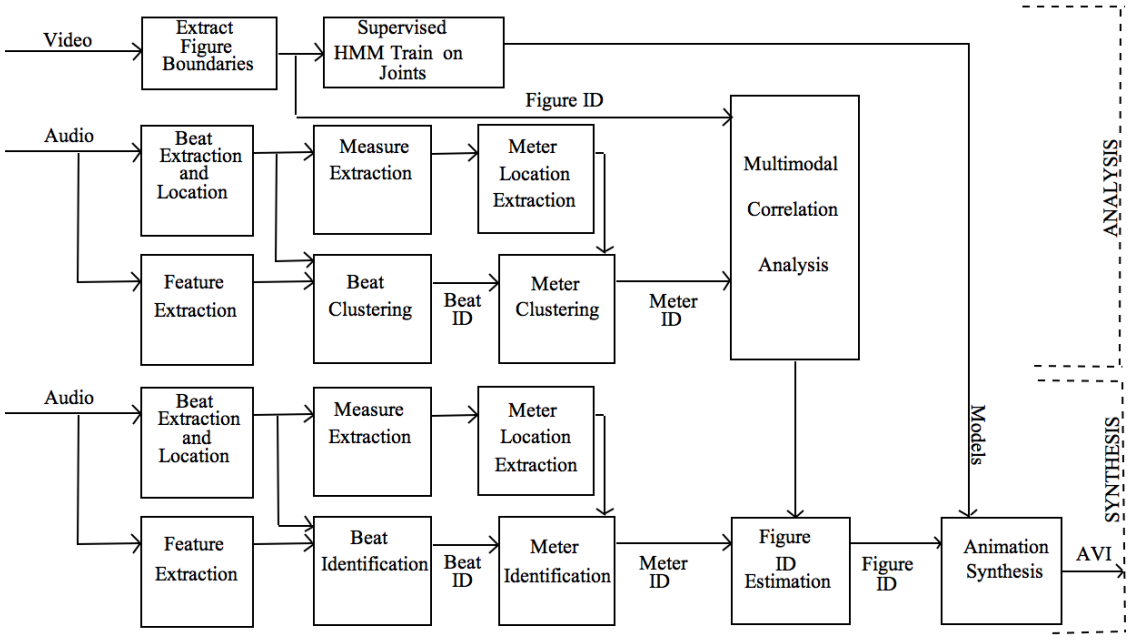


Figure 2.1: System overview

Audio and video are analyzed separately depicted in Fig. 2.1. Video is analyzed through figure boundary extraction and supervised HMM training blocks. Figure boundary extraction, dance figure boundaries are determined with human supervision and HMMs are trained under the supervision of figure identities, finally repeating dance figures are modeled.

Audio is analyzed in two stages. In beat extraction, audio beats are tracked and beat boundaries are located. In measure extraction and meter location, audio measure is estimated and meter boundaries located. In audio feature extraction, audio features, MFCC and chroma features, are extracted. In beat and meter clustering, these extracted features are clustered in between extracted beat locations and meter locations, eventually beat and meter identities are determined. Towards our eventual goal, co-occurence of extracted figure identities and meter identities are evaluated in multimodal correlation analysis. In the synthesis, beat and meter boundaries of the audio extracted. Audio features are extracted and identified in beat and meter identification blocks. Then the corresponding figure identities of each meter identity determined through the given co-occurence of multimodal correlation analysis. Finally, a dancing avatar is synthesized by employing the resulting HMMs of video analysis blocks using estimated figure identities.

Chapter 3

MUSIC ANALYSIS

The act of dancing is the natural response of the body to the musical tempo and beat sequence. According to these responses the movements of the body is shaped and dance figures are generated. Many research has been done since 1970s established on the basic music psychology theory in which human and machine have to perform synchronously. Based on the previous research pre-analysis of audio features, e.g. extracting temporal patterns related with dance figures, would expedite the synchronization of audio and video patterns in an audio-visual dance analysis and synthesis scenario [29, 30].

3.1 Audio Beat Tracking, Measure and Meter Detection

Beat tracking, tempo extraction and meter estimation are related problems. The underlying speed of a musical excerpt can be estimated by tempo extraction and individual beat locations are calculated as a result of beat tracking. Measure estimation gives an inference about the number of beats existing in between a measure in musical excerpt. The primary challenge in these tasks is the fluctuating tempi which complicates estimation of these parameters especially in beat tracking problem while tempo extraction does not depend on the individual beats at all. On the other hand, there are some musical excerpts with a constant tempo where beat extraction, tempo estimation and meter detection problems presents similar information. The problems has different applications in the literature. Extracting tempo helps classifying audio excerpts while beat tracking is useful for synchronization of audio-visual parameters, e.g. dance figures and temporal audio features in a dance performance.

3.1.1 Beat Detection

Some general approaches developed for detecting the beats and tempo based on onset detection functions which was first discovered by [31]. Then this onset detection method

developed by other researchers. Alonso et. al, applied a pulse train which is capable of detecting the onsets and and in the second step of method a periodicity detector designed [32]. Possible periodicity is estimated by autocorrelation, spectral sum and spectral product and then a dynamic programming employed in order to calculate optimal path in between the derived periodicity. Antonopoulos et al. also proposed an algorithm based on audio self-similarity measurement [33]. This measurement was estimated from spectral audio features like MFCCs. First order intervals between local minima are calculated in the audio signal and served as the beat period. These intervals are assumed to correspond to the beat period in the music and thus the largest peaks in the histograms are taken as the most salient beat periods. Davies and Plumbley (2007) used three different onset detection functions rooted from spectral difference, phase deviation and complex domain to extract the beat locations and tempo [34]. By using a weighted shift-invariant comb filter bank, the autocorrelation function of each onset detection function is filtered and the resulting strongest peaks showed the fluctuating tempo. The correlation of these tempo-dependent impulse train with each onset detection functions gives the beat locations. Ellis also developed an algorithm based on real-valued onset detection function [35]. Periodicity is detected by using a autocorrelation function and the resulting signal smoothed with the onset detection function. Then a dynamic programming employed to extract the beat locations. In another solution, Klapuri et al. applied four different onset detection functions in four different frequencies [36]. In order to estimate tempo, a bank of comb filter resonators and a probabilistic model inferring the basic music information are driven to the signal in order to extract beat locations, tatum, tactus and measure boundaries respectively. All these different solutions for more general cases were compared in a recent article [37]. The comparison result showed that Davies and Plumbley is one of the two best algorithms for detecting beat and tempo for different genre and different tempo values. We employed this algorithm in order to track beats which is explained in Appendix A in details.

3.1.2 Measure and Meter Detection

In musical notation, measure (or bar) is the metric unit between two bars on the staff which is defined as a given number of beats of a given duration. Moreover, meter is a specific rhythm determined by the number of beats and the time value assigned to each note in a

measure.

The first study of estimating the meter and tempo through the figures of Bach by using note descriptions was presented in [38]. Later, a rhythm based approach was defined by [39] which was capable of finding metric structure of a given audio. A further study performed an oscillator based solution under the assumption that oscillators response to the incoming rhythm and the periodicity causing maximum amplitude enables tempo and meter estimation [40]. In 90s, a symbolic music representation method used to follow fluctuations of tempo by using a synthetic database composed of MIDI-based excerpts [41]. In this thesis we implemented Davies and Plumbley meter and measure detection algorithm which is is explained in Appendix A.

3.2 Audio Feature Extraction

In order to extract temporal patterns through the audio signal, we analyze spectral and chromatic audio features that would be helpful when used with temporal features that are proposed in [42, 43]. Although spectral features are capable of discriminating the instruments and and their arrangements, they are deficient for representing melodic or harmonic content like chromatic features. Besides, in most of the musical audio signals, chorus is the most easily observed repeating pattern which can be an alternative similarity level instead of the previous studies. Much research has been done for extracting these chroma dependent repeating pattern similarities. In one of the studies chroma features were extracted through beat-synchronized audio frames and dynamic time warping (DTW) approach was applied in order to extract similarity matrix within measure boundaries, finally a manually parameterized HMM is used to come up with a high level repeating pattern structure [29]. Another different method was to combine beat-synchronized pitch related features with chroma features by using HMMs under the assumption that similarity depends on matching repeated parts durations with beat-synchronized segments [30].In this thesis, we investigate both spectral and chromatic features to evaluate the ability of combinations of these features to model the musical signals.

3.2.1 MFCC Features

In our evaluations mel-scale cepstral coefficients (MFCC) with their first and second derivatives, spectral centroid, spectral flux and spectral roll-off are used as candidate audio features. Short-time analysis of overlapping audio segments of size 25ms are performed for the extraction of audio features for each 10ms frame. Hamming window of size 25ms is applied to the analysis audio segment to remove edge effects. The MFCC feature vector is constructed with 13 cepstral coefficients including the energy coefficient, F_M. The first and second derivatives of the MFCC features are considered as dynamic spectral features, and denoted as $F_{\Delta M}$ and $F_{\Delta\Delta M}$.

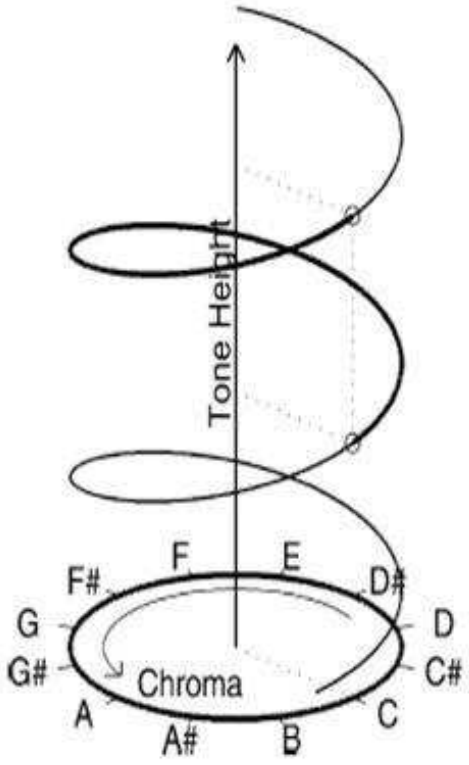


Figure 3.1: Illustration of helix of pitch perception. The vertical dimension and the angular dimension are tone height and chroma respectively.

3.2.2 Chroma features

Chroma features are used to recognize melody-based similarity within audio segments by evaluating the similarity of existing chord patterns. In 1964, the human auditory perceptual structure of pitch represented with two dimensions instead of one by using a helix in [44]. In this representation, when the pitch, p, of a note increases, its location moves along the helix, rotating chromatically through all of the pitch classes before it returns to the initial pitch class one cycle above the starting point. These representation is depicted in Fig. 3.1. This representation implies that the distance between two pitches depends on both chroma, c, and tone height, h, rather than on p alone which is given in equation below where $c \in [01)$. This representation shows that linear changes in c yields as logarithmic changes in the pitch. By dividing the interval between 0 and 1 into 12 equal parts, the 12 pitches of the equal-tempered chromatic scale can be obtained as follows where

$$p = 2^{h+c}. \tag{3.1}$$

Later in 1980s, this representation is generalized to a frequency interpretation that is structurally equivalent to Shepard's pitch decompositon where frequency is calculated as

$$f = 2^{h+c}. \tag{3.2}$$

Intuitively chroma is calculated from a given frequency as follows,

$$c = log_2 f - \lfloor f \rfloor \tag{3.3}$$

In this work, 12-dimensional chroma vector was extracted from the DFT of the audio signal and a pattern matching method is performed. We use a chroma feature extraction method similar to that in [45] for chord detection with the idea of extracting a chroma feature within overlapping audio segments of size 25ms for each 10ms frame which was first applied in [46].

In this algorithm a 12-element chroma feature vector c_t with elements $c_{t,k}$ is calculated for each frame where t and k are frame number and chroma bin number $1, ..., 12$ and where c_t is defined as

$$c_t = [c_{t,1}c_{t,2}c_{t,3}\cdots c_{t,12}]. \tag{3.4}$$

The chroma feature vector elements for each frame are calculated using the equation,

$$c_{t,k} = \frac{\sum_{n\in S_k} F_t(n)}{N_k} \qquad k \in 1, 2, ..., 12, \tag{3.5}$$

where $F_t(n)$ is the logarithmic magnitude of discrete Fourier transform (DFT) of the t^{th} frame and S_k defines a subset of the discrete frequency space for each pitch class and N_k is the number of elements in each pitch class S_k. The DFT length equal to the first power of 2 greater than or equal to the length of the frame segment used in calculations. The details about the algorithm can be found in [47].

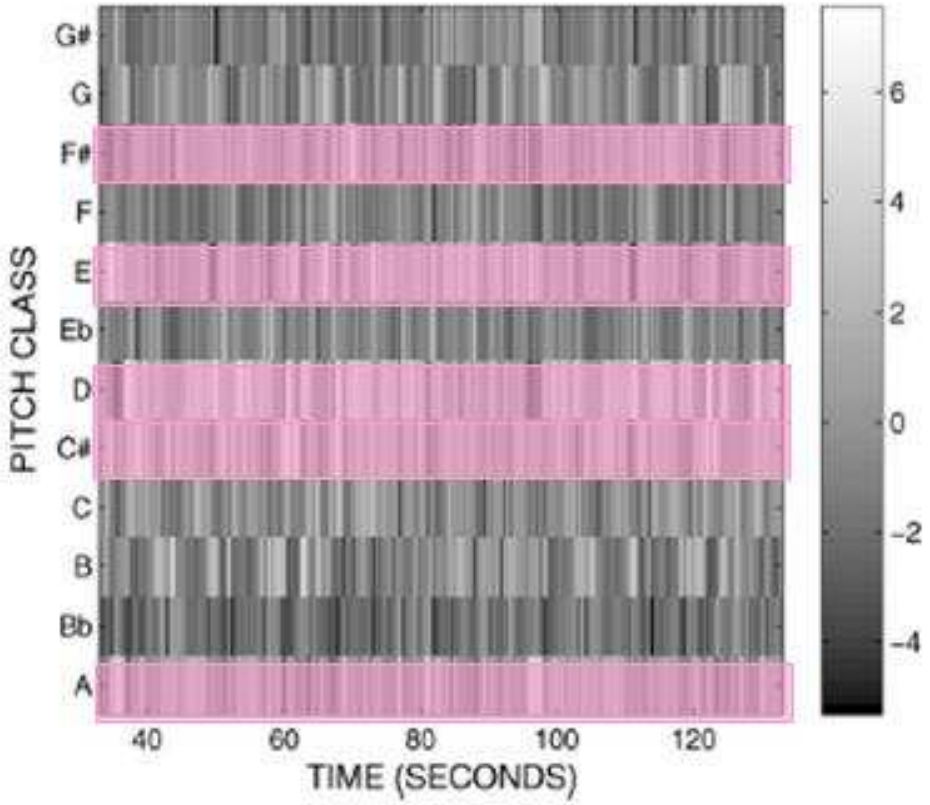


Figure 3.2: Part of the chroma features for "Zeybekiko" showing the energy located in each pitch class for each frame.

A part of calculated chroma features for "Zeybekiko" is depicted in Fig. 3.2. Each

chroma feature has been labeled with its corresponding pitch class.

3.3 Beat Clustering

The beat clustering defines recurrent elementary beat related patterns using unsupervised temporal clustering over individual feature streams. The beat related feature streams $\boldsymbol{F}^b$ is used to train the HMM structure Λ_b, which capture recurrent beat related segments $\boldsymbol{\varepsilon}^b$. For ease of notation, we use a generic notation to represent the HMM structure which is identical for the audio stream. As proposed in [48, 49], the HMM structure Λ_b, which is used for unsupervised temporal segmentation, has M parallel branches and N states as shown in Fig. 3.3. The states labeled as s_s and s_e are non emitting start and end states of the parallel HMM structure. Fig. 3.3 clearly illustrates that the parallel HMM Λ is composed of M parallel left-to-right HMMs, $\{\lambda_1, \lambda_2, \ldots, \lambda_M\}$, where each λ_m is composed of N states, $\{s_{m,1}, s_{m,2}, \ldots, s_{m,N}\}$. The state transition matrix A_{λ_m} of each λ_m is associated with a sub-diagonal matrix of A_Λ. The feature stream is a sequence of feature vectors, $\boldsymbol{F}^b = \{\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_T\}$, where $\boldsymbol{f}_t$ denotes the feature vector at frame t. Unsupervised temporal segmentation using HMM model Λ_b yields L number of segments $\boldsymbol{\varepsilon} = \{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_L\}$. The l-th temporal segment is associated with the following sequence of feature vectors,

$$\varepsilon_l = \{\boldsymbol{f}_{t_l}, \boldsymbol{f}_{t_l+1}, \ldots, \boldsymbol{f}_{t_{l+1}-1}\} \quad l = 1, 2, \ldots, L \tag{3.6}$$

where $\boldsymbol{f}_{t_1}$ is the first feature vector $\boldsymbol{f}_1$ and $\boldsymbol{f}_{t_{L+1}-1}$ is the last feature vector $\boldsymbol{f}_T$.

The segmentation of the feature stream is performed using Viterbi decoding to maximize the probability of model match, which is the probability of feature sequence $\boldsymbol{F}^b$ given the trained parallel HMM Λ_b,

$$\begin{aligned} \mathrm{P}(\boldsymbol{F}^b|\Lambda) &= \max_{t_l, m_l} \prod_{l=1}^{L} \mathrm{P}(\{\boldsymbol{f}_{t_l}, \boldsymbol{f}_{t_l+1}, \ldots, \boldsymbol{f}_{t_{l+1}-1}\}|\lambda_{m_l}) \\ &= \max_{\varepsilon_l, m_l} \prod_{l=1}^{L} \mathrm{P}(\varepsilon_l|\lambda_{m_l}) \end{aligned} \tag{3.7}$$

where ε_l is the l-th temporal segment, which is modeled by the m_l-th branch of the parallel HMM Λ_b. One can show that λ_{m_l} is the best match for the feature sequence ε_l, that is,

$$m_l = \operatorname*{argmax}_m \mathrm{P}(\varepsilon_l|\lambda_m) \tag{3.8}$$

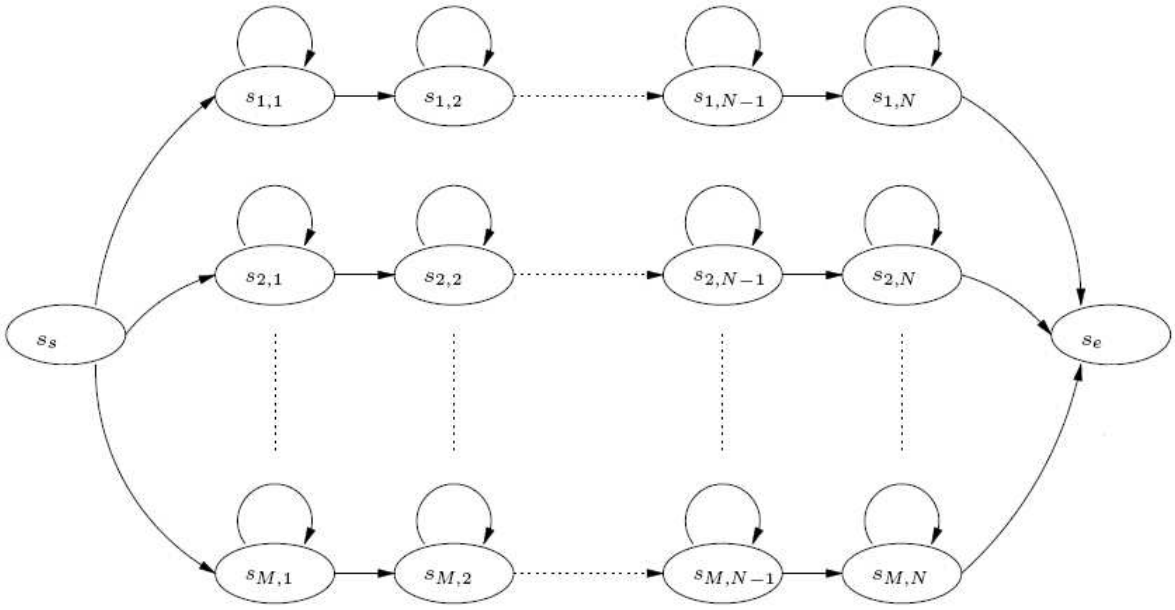


Figure 3.3: Parallel HMM structure

Since, the temporal segment ε_l from frame t_l to $(t_{l+1} - 1)$ is associated with segment label m_l, we define the sequence of frame labels based on this association as,

$$\ell_t = m_l \qquad \text{for } \; t = t_l, t_l + 1, \ldots, t_{l+1} - 1 \tag{3.9}$$

where ℓ_t is the label of the t-th frame and we have a label sequence $\boldsymbol{\ell} = \{\ell_1, \ell_2, \ldots, \ell_T\}$ corresponding to the feature sequence $\boldsymbol{F}^b$. Beat clustering extracts the beat related label sequence $\boldsymbol{\ell}^b$ given the beat related audio feature stream $\boldsymbol{F}^b$.

3.4 Meter Clustering

In the meter clustering, unsupervised temporal clustering over the beat related discrete label stream is performed within meter boundaries to detect recurrent meter related label patterns. For this task, the same parallel HMM structure which is given in Fig. 3.3 is used with discrete single-stream HMM branches. In single-stream HMMs, all streams share the

same state transition structure.

The beat related discrete label stream, denoted by $\boldsymbol{\ell}^b$, is defined such that for every frame k, $\ell_k^b = [\ell_k^b]^T$. We represent the discrete single-stream parallel HMM structure with Γ_b and its m-th branch with γ_m^b. The discrete HMM Γ_b is trained over the beat related discrete label stream. Each branch γ_m^b, associated with a meter related temporal label pattern, is then described with a state transition matrix $\boldsymbol{A}_{\gamma_m^b}$, a discrete observation probability distribution $\boldsymbol{B}_{\gamma_m^b}$ and an initial state probability matrix $\boldsymbol{\Pi}_{\gamma_m^b}$,

$$\gamma_m^b = (\boldsymbol{A}_{\gamma_m^b}, \boldsymbol{B}_{\gamma_m^b}, \boldsymbol{\Pi}_{\gamma_m^b}) \tag{3.10}$$

The discrete observation probability distribution $\boldsymbol{B}_{\gamma_m^b}$ defines the probability of observing a meter related audio frame label at state s and frame k, $P(\ell_k^b|s)$.

The parallel HMM structure has two important parameters to set before training the models Λ_b and Γ_b. The first parameter is the number of states in each branch, N, which would be selected by considering the minimum duration of temporal patterns. The second parameter is the number of branches, M. In order to select N and M, a supervised temporal classification is applied over the labeled audio feature stream, $\boldsymbol{F}$, in both beat and meter clustering stages, and the same parallel HMM structure is trained. By using Viterbi decoding the feature stream is classified . The number of states in each branch of the HMM N and the number of branch of the HMM M are selected by using a performance analysis method which reads in a set of label les in the output from the supervised classification and compares them with the corresponding reference transcription files in the output from the unsupervised clustering. The comparison is based on a Dynamic Programming-based string alignment procedure. The basic output of DP procedure is recognition statistics for the whole le set including parameters such as H is the number of correct labels, I is the number of insertions and N is the total number of labels in the dening transcription les. The percentage number of labels correctly recognized, $\%Accuracy$, is given by

$$\%Accuracy = \frac{H - I}{N} X 100\% \tag{3.11}$$

Chapter 4

AUDIO-FIGURE MAPPING

In this chapter, audio-figure mapping process is defined. The process of manual labeling of training videos is presented in Section 4.1. Tracking of video feature points and video figure analysis is presented in Section 4.2 and Section 4.3 respectively. Finally, the multimodal correlation analysis block is described in Section 4.4.

4.1 Manual Labeling of Training Videos

We manually label the start and end frames of each dance figure throughout the dance recordings in our database.

4.2 Tracking of Feature Points

A marker based approach is employed for motion tracking which is presented in detail in [50]. In this work, a set of color markers are attached at or around the joints of the dancer where each the original images are processed in the YCrCb color space in order to have a flexibility over intensity variations in each image captured by the cameras from different views. The proposed method tracks the 3D positions of the joints of the body based on the markers 2D projections on each cameras image plane.

4.3 Extraction of Elementary Dance Figure Related Video Patterns

In this section, dance figures in the dance recordings are modeled using HMMs. Typically, dance figures always contain a very concrete sequence of movements hence a left-right HMM structure is employed. Each different dance figure is modeled separately by employing a different HMM. The tracked and normalized 3D positions of 16 landmarks defined on the body (typically the body joints) serve as the input feature stream to train the HMMs. A supervised temporal classification is applied over the input feature stream $\boldsymbol{F}^v$ and the same HMM structure Λ_v is trained given the manuel labeled dance figure labels $\boldsymbol{\ell}^m$. Each of the

parameters is represented by a single Gaussian function and one full covariance matrix is computed for each state which is explained in detail in [50].

4.4 Joint Meter-Figure Mapping

In this section, multimodal correlation analysis block that is shown in Fig. 2.1 is defined. First, dance figure boundaries are extracted by human supervision as defined in Sec. 4.1. Then, repeating dance figures are modeled by employing HMMs to the tracked 3D positions of the joints of the body, which is described in Sec. 4.3 in detail. Second, audio features, $\boldsymbol{F}$, are extracted from the input audio signal as described in Sec. 3.2 and temporal clustering of audio feature sequence $\boldsymbol{F}$ is performed in between beat and meter boundaries, which is given in Sec. 3.3 and Sec. 3.4 consecutively. Finally, co-occurences of the dance figure labels and meter related audio labels are calculated for each different dance excerpt and for each two different feature stream by employing a bi-gram based dance modeling approach.

Bi-grams are very commonly used in one of the most successful language models for speech recognition [51] . In speech recognition, bi-grams provide the conditional probability of a word given the preceding word. In our scenario, words are replaced with dance figures, our figure related parameters are the conditional dance figure occurrences. We evaluate the co-occurrence of a dance figure label given the preceding dance figure label, $l_i^v|l_j^v$, and corresponding meter related audio labels. This evaluation is done for each different combination of dance figures in each different dance scenario. The resulting bi-gram based co-occurence matrices associate temporal meter related label patterns and dance figure label patterns which are valuable for the audio-driven human dance figure synthesis task.

Chapter 5

RESULTS

In this section, we present experimental results and evaluation of the proposed system. Section 5.1 describes the audio database which is used in the experimental evaluation of audio beat tracking, tempo extraction and measure estimation algorithm and the audio-visual database, which is used in the experimental evaluation of audio-visual pattern analysis to generate objective and subjective results and presents the evaluation of audio beat tracking, tempo extraction and measure estimation. The evaluation of the beat and meter clustering is presented in Section 5.1.1 and Section 5.1.2 respectively , and the performance results for both single and mix tune bi-gram synthesis are presented in Section 5.2.2 and Section 5.2.2 consecutively.

5.1 Music Analysis Results

The beat detection algorithm is evaluated through the MIREX'06 beat tracking database containing 20 audio excerpts, each with 40 different annotations of 40 different people.

Our training audiovisual dataset includes multiview video recordings of four different dance performances for dances *zeybekiko*, *kasap*, belly and salsa. Each dance recording has a duration of approximately 5 minutes and consists of a set of dance figures repeated successively during the whole performance.

The evaluation criteria for beat tracking, which exists in the same environment and used in the music journal [37], is considered as the evaluation criteria for our system which is based on a correlation score, "P" score. This score is calculated as,

$$P = \frac{1}{S}\sum_{s=1}^{40}\frac{1}{NP}\sum_{m=-W_s}^{W_s}\sum_{n=1}^{N} y[n]a_s(n-m) \tag{5.1}$$

where $y[n]$ is the impulse train generated from the result of beat tracking algorithm, while $a_s(n)$ is the impulse train generated from annotated beat sequence. N is the length of beat

sequences and NP is the corresponding maximum length in either impulse train,

$$NP = max(\sum y[n], \sum a_s(n)). \tag{5.2}$$

W_s is the error window. The window size W_s is chosen to be to be 20% of the beat period throughout the experiment. Note that the value of P lies between 0 and 1.

P score for our beat tracking results is calculated as 0.6167 which is consistent with the beat tracking evaluation scores in [37]. Furthermore, we evaluated the same algorithm for measure estimation over an 612 excerpts in ballroom database. The excerpts consist of measures with 3/4 and 4/4. The resulting correct esitimates were 77.13% which is a considerable result for such a large and selected database.

5.1.1 Beat Clustering Results

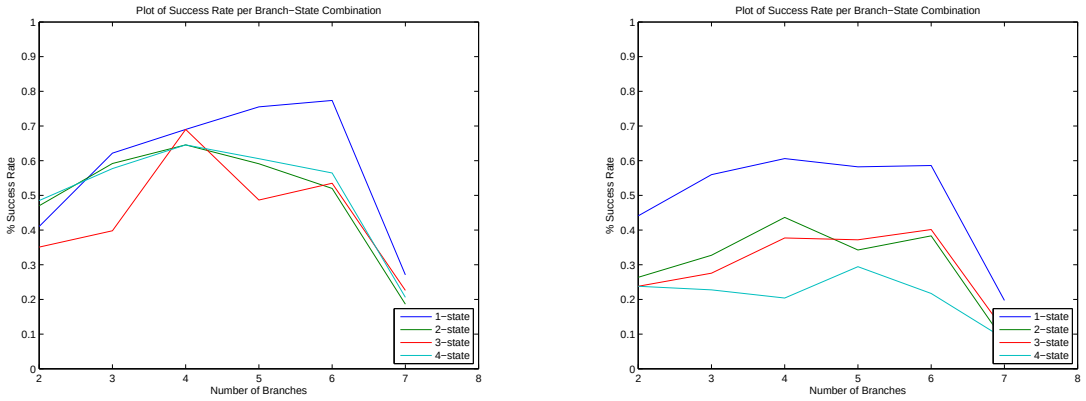
The parallel HMM Λ_b is trained with features extracted from the audio database using Expectation-Maximization (EM) algorithm. The resulting HMM structure provides a probabilistic cluster model for unsupervised segmentation of beat features into recurring elementary patterns.

The number of states, N, and the number of branches, M, is a critical model parameter. In order to determine N and M the parallel HMM is trained for varying number of states from 1 to 20 and branches from 2 to 7. For each state and branch combination, the percentage number of labels correctly recognized, $\%Accuracy$, which is described in Section 3.4, is calculated. Hence, we set the best N and M combination for Λ_b to the measures that clearly maximizes $\%Accuracy$ in plots in Fig. 5.1. The best N and M is set to 2 and 6 respectively, as a result of recognition accuracy of beat clustering of belly with MFCC features for varying number of branches. Moreover, the best N and M is set to 2 and 4 respectively as a result of recognition accuracy of beat clustering of belly with chroma features.

Consequently, when the training beat related feature sequence is segmented using Λ_b, the segments belonging to the same beat patterns are observed to be alike.

5.1.2 Meter Clustering Results

The parallel HMM Γ_b is trained with beat related discrete label stream to obtain unsupervised temporal segmentation of the audio stream. In order to determine N and M the



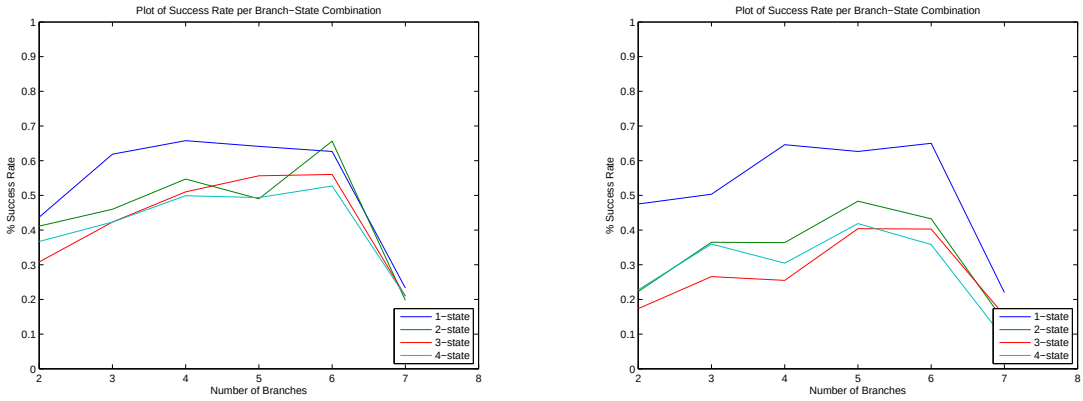
(a) (b)

Figure 5.1: (a) Recognition accuracy of beat clustering of belly with MFCC features for varying number of branches, (b) Recognition accuracy of beat clustering of belly with Chroma features for varying number of branches

parallel HMM, Γ_b, is trained for varying number of states from 1 to 20 and branches from 2 to 7 and as a result N and M combination for Γ_b is set to the number which maximize the percentage number of labels correctly recognized, $\%Accuracy$, in plots in Fig. 5.2. The best N and M is set to 2 and 4 respectively, as a result of recognition accuracy of meter clustering of belly with MFCC features for varying number of branches. Moreover, the best N and M is set to 2 and 4 respectively as a result of recognition accuracy of meter clustering of belly with chroma features.

5.2 Dance Synthesis Results

In this section, we address audio-driven dance synthesis using the proposed audio-visual dance model. The system takes audio as input and produces a sequence of dance figures, which are naturally correlated with the input audio. Audio-driven dance synthesis generates a 3D position sequence which corresponds to the 16 different joints of human body. This resulting sequence is naturally correlated to a driven audio excerpt. These 3D position sequence is animated using the graphics library OpenGL and synthesis dance videos generated for each kind of dance performance which are "kasap", "belly" and "salsa".



(a) (b)

Figure 5.2: (a) Recognition accuracy of meter clustering of belly with MFCC features for varying number of branches, (b)Recognition accuracy of meter clustering of belly with Chroma features for varying number of branches

5.2.1 Single Tune bi-gram synthesis results

This section presents the bi-gram based co-occurence analysis of dance models for each different dance performance. In order to synthesize dance figures, co-occurrence of a dance figure label given the preceding dance figure label, l_i^v—l_j^v, and corresponding meter related audio labels, l^m, is calculated. These co-occurrences for each dance excerpt is used to synthesize dance figures according to meter related audio patterns of the driven music.

The co-occurences of dance figure label stream and meter related audio label stream which is recognition output of Γ_b using training data, are given below in Table. 5.1 − Table. 5.8. And the co-occurrences of dance figure label stream and meter related label stream which is the recognition output of the Γ_b using test data are given below in Table. 5.9 − Table. 5.16. In these tables, each column corresponds to a different figure dependency and letters in each row represents different meter cluster. Each dance is modeled with both MFCC and chroma features separately.

These co-occurences associate temporal meter related patterns and dance figure patterns which are valuable for the audio-driven human dance figure synthesis task. Dance choreography can be examined as a result of these bi-gram based co-occurences where each zero

co-occurence means no existence of a dance figure label given the preceding dance figure label, l_i^v—l_j^v, together with meter related audio label l^m.

Table 5.1: Co-occurence Results for "Zeybekiko" with Chroma Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$	$l_1^v\|l_2^v$	$l_1^v\|l_3^v$	$l_2^v\|l_1^v$	$l_2^v\|l_2^v$	$l_2^v\|l_3^v$	$l_3^v\|l_1^v$	$l_3^v\|l_2^v$	$l_3^v\|l_3^v$
a	0	5	7	12	0	0	0	6	9
b	0	1	3	3	0	0	0	4	1
c	0	2	4	6	0	0	0	3	0

Table 5.2: Co-occurence Results for "Zeybekiko" with MFCC Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$	$l_1^v\|l_2^v$	$l_1^v\|l_3^v$	$l_2^v\|l_1^v$	$l_2^v\|l_2^v$	$l_2^v\|l_3^v$	$l_3^v\|l_1^v$	$l_3^v\|l_2^v$	$l_3^v\|l_3^v$
a	0	1	3	3	0	0	0	4	6
b	0	7	5	11	0	0	0	7	5
c	0	3	2	9	0	0	0	1	1

Table 5.3: Co-occurence Results for "Kasap" with MFCC Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$	$l_1^v\|l_2^v$	$l_1^v\|l_3^v$	$l_2^v\|l_1^v$	$l_2^v\|l_2^v$	$l_2^v\|l_3^v$	$l_3^v\|l_1^v$	$l_3^v\|l_2^v$	$l_3^v\|l_3^v$
a	0	0	13	9	0	0	0	7	0
b	0	0	1	1	0	0	0	3	0
c	0	0	3	0	0	0	0	1	0
d	0	0	3	2	0	0	0	2	0
e	0	0	1	1	0	0	0	1	0

5.2.2 Mix Tune bi-gram synthesis results

In mix tune bigram synthesis, a mixture of all dance choreographies are modeled by a set of HMMs which is consist of different HMMs that are trained for each different dance model.

Table 5.4: Co-occurence Results for "Kasap" with Chroma Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$	$l_1^v\|l_2^v$	$l_1^v\|l_3^v$	$l_2^v\|l_1^v$	$l_2^v\|l_2^v$	$l_2^v\|l_3^v$	$l_3^v\|l_1^v$	$l_3^v\|l_2^v$	$l_3^v\|l_3^v$
a	0	0	6	13	0	0	0	5	0
b	0	0	3	3	0	0	0	1	0
c	0	0	4	0	0	0	0	1	0
d	0	0	2	4	0	0	0	1	0
e	0	0	2	3	0	0	0	1	0

Table 5.5: Co-occurence Results for "Salsa" with MFCC Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$
a	21
b	33
c	18
d	6

Table 5.6: Co-occurence Results for "Salsa" with Chroma Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$
a	14
b	43
c	13
d	4

For a given audio, this HMM is employed in order to generate appropriate dance figures according to the driven audio. The synthesis results can be downloaded from [52].

Table 5.7: Co-occurence Results for "Belly" with MFCC Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$
a	17
b	25
c	18
d	16

Table 5.8: Co-occurence Results for "Belly" with Chroma Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$
a	18
b	42
c	8
d	7

Table 5.9: Co-occurence Results for "Zeybekiko" with Chroma Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$	$l_1^v\|l_2^v$	$l_1^v\|l_3^v$	$l_2^v\|l_1^v$	$l_2^v\|l_2^v$	$l_2^v\|l_3^v$	$l_3^v\|l_1^v$	$l_3^v\|l_2^v$	$l_3^v\|l_3^v$
a	0	7	6	11	0	0	0	6	9
b	0	2	3	2	0	0	0	3	2
c	0	2	3	8	0	0	0	2	0

Music Feature Selection

In this section, we defined an Euclidian distance between the co-occurence results of dance figure labels and meter related audio labels resulting of unsupervised clustering and the co-occurence results of dance figure labels and meter related audio labels, which is the recognition output of the Γ_b, that is unsupervised trained with the determined best state

Table 5.10: Co-occurence Results for "Zeybekiko" with MFCC Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$	$l_1^v\|l_2^v$	$l_1^v\|l_3^v$	$l_2^v\|l_1^v$	$l_2^v\|l_2^v$	$l_2^v\|l_3^v$	$l_3^v\|l_1^v$	$l_3^v\|l_2^v$	$l_3^v\|l_3^v$
a	0	4	2	2	0	0	0	3	6
b	0	5	7	13	0	0	0	6	4
c	0	2	3	7	0	0	0	2	1

Table 5.11: Co-occurence Results for "Kasap" with MFCC Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$	$l_1^v\|l_2^v$	$l_1^v\|l_3^v$	$l_2^v\|l_1^v$	$l_2^v\|l_2^v$	$l_2^v\|l_3^v$	$l_3^v\|l_1^v$	$l_3^v\|l_2^v$	$l_3^v\|l_3^v$
a	0	0	10	11	0	0	0	8	0
b	0	0	2	2	0	0	0	1	0
c	0	0	1	0	0	0	0	3	0
d	0	0	1	3	0	0	0	3	0
e	0	0	1	1	0	0	0	1	0

Table 5.12: Co-occurence Results for "Kasap" with Chroma Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$	$l_1^v\|l_2^v$	$l_1^v\|l_3^v$	$l_2^v\|l_1^v$	$l_2^v\|l_2^v$	$l_2^v\|l_3^v$	$l_3^v\|l_1^v$	$l_3^v\|l_2^v$	$l_3^v\|l_3^v$
a	0	0	5	14	0	0	0	5	0
b	0	0	2	3	0	0	0	2	0
c	0	0	4	0	0	0	0	1	0
d	0	0	3	2	0	0	0	2	0
e	0	0	2	2	0	0	0	2	0

and branch combination, as given,

$$P = \sum_{j=1}^{m} \sum_{i=1}^{n} [a_{ij} - b_{ij}]^2 \tag{5.3}$$

where m is the number of columns in the table, n is the number of rows in the table, a is element of co-occurence results of dance figure labels and meter related audio labels

Table 5.13: Co-occurence Results for "Salsa" with MFCC Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$
a	18
b	36
c	16
d	8

Table 5.14: Co-occurence Results for "Salsa" with Chroma Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$
a	15
b	40
c	15
d	5

Table 5.15: Co-occurence Results for "Belly" with MFCC Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$
a	15
b	23
c	17
d	21

resulting of unsupervised clustering and b is element of co-occurence results of dance figure labels and meter related audio labels resulting of supervised segmentation. These calculated Euclidian distance tables are given below in Table. 5.17 where minimum distance for each dance model with different audio feature set defines the best feature set for modeling each different dance.

According to these results Chroma feature set would be the best feature set with the

Table 5.16: Co-occurence Results for "Belly" with Chroma Features

$l^m/(l_i^v\|l_j^v)$	$l_1^v\|l_1^v$
a	22
b	40
c	7
d	6

Table 5.17: Euclidian Distance for Each Different Feature Set and Dance Excerpt

	Zeybekiko	Kasap	Salsa	Belly
$MFCC$	33	34	26	34
$Chroma$	16	12	15	22

minimum distance for each different dance choreography.

Chapter 6

CONCLUSIONS

In this thesis, multi-modal dance performance analysis and synthesis of audio and video is investigated under different dance scenarios. The multi-modal dance performance analysis and synthesis system clarifies four basic aspects in depth; i) beat and meter sync audio analysis, ii) meter sync genre recognition, iii) bi-gram based dance figure modeling and iv) meter sync dance figure synthesis.

First, audio beat is extracted, meter is estimated and measure locations are investigated. Then, audio is analyzed in between extracted beat and measure locations, repeating audio patterns are modeled and video is analyzed in order to extract repeating dance figure patterns. Finally, the bi-gram based co-occurence analysis of audio and video modality for the synthesis of dance figures associated with the driven audio is studied. Employing the results of multi-modal dance performance analysis of audio and video, we concentrate on the dance figure pattern and audio pattern bi-gram based co-occurence in order to automatically synthesize dance figures according to the driven single-tune/mix-tune audio.

In the audio analysis, beat and meter clustering is applied respectively which is a new beat and meter synchronized method in audio analysis sense. The beat clustering deals with the HMM based unsupervised detection of elementary beat related audio patterns of an audio through both extracted MFCCs and chroma feature sets separately. Then, in the meter clustering elementary meter related audio patterns of an audio is extracted through the discrete feature sequence that represents the recognition output of the beat clustering. Employing HMM based analysis gives us the advantage of flexibility in modeling variations within audio patterns. As a result of the HMM based audio analysis, we can make a comparison between MFCCs and chroma features, where chroma features with smaller euclidian distance for each different genre outperforms MFCCs. Furthermore, beat and meter synchronized audio analysis improved the audio genre recognition performance for our audio database. However, since our audio database is composed of a limited number

of genres, more research would be done with a larger audio database in order to generalize this conclusion.

In the video analysis, repeating dance figures are modeled by employing a supervised HMM based analysis approach. 3-D positions of human body joints, which serve as the feature set for video analysis, are modeled in between measure boundaries where each measure boundary corresponds to a separate dance figure. The HMM based approach enables us to model variations of each dance figure which yields realistic avatar synthesis.

Finally, in the multi-modal synthesis, bi-gram based co-occurence analysis is applied over the extracted audio and video patterns. The bi-gram based co-occurence analysis gives us the possibility to understand the dance choreography and synthesize the right combination of dance figures for each different dance according to the driven audio excerpt. We come up with deterministic dance choreographies regarding of our audio-visual database, where dance figures are independent from the audio patterns, and this multi-modal synthesis reduces to meter synchronized dance figure synthesis concerning the extracted dance choreographies. Since the dance choreographies used in this research deterministic, more research with a larger and non-deterministic database is left for future work. The HMM-based analysis and synthesis system helps us to define meaningful elementary audio and video patterns and observes variational patterns which are useful for generating realistic animations and bi-gram based co-occurence analysis helps us to define the dance choreography of a given dance excerpt.

Appendix A

AUDIO BEAT TRACKING ALGORITHM

The beat tracking approach that we implemented is composed of three states : i) Input representation state, ii) General state and iii) Context-Dependent State.

A.1 A) Input Representation State

In the first state analysis, the input signal is transformed in to a more purposive representation in order to extract beat locations. This purposive input representation function is determined as complex spectral difference onset detection function in order to cover a wide range of audio signals including drum hits and violin tonal. The onset detection function, $O(n)$ is calculated, at each frame n, by measuring the Euclidean distance between the observed spectrum $S_k(n)$ and the estimated spectrum $\tilde{S}_k(n)$, including all frequency bins,

$$O(n) = \sum_{k=1}^{K} (S_k(n) - \tilde{S}_k(n))^2 \tag{A.1}$$

where k is the frequency bin in range $[1, K]$. A more detailed derivation of detection function is given in Appendix B. The onset detection function is then partitioned in to overlapping analysis frames to be sensitive to the variations of tempo and phase of beats,

$$O_i(n) = \begin{cases} O(n) & n = 1 + (i-1)B_f, ..., B_f + (i-1)B_h \\ 0 & \text{otherwise} \end{cases} \tag{A.2}$$

where $B_f = 512$ detection function samples with a overlapping window size of $B_f/4$.

The purpose of the general state analysis is to capture the beat period and beat alignment from the given onset detection function. According to the Davies approach the beat period is identified and then the beat alignment is found out. Thus, the autocorrelation function of the complex spectral difference onset detection function is employed and by this way the phase-related information is excluded. First the detection function is processed to get

well defined peaks in the resulting autocorrelation function. Therefore an adaptive mean threshold of the detection function is calculated,

$$\overline{O}_i(n) = mean\{O_i(q)\} \qquad m - \frac{Q}{2} \leq Q \leq m + \frac{Q}{2} \tag{A.3}$$

where $Q = 16$ detection function samples. Then this calculated mean is extracted from the detection function and the result is half-wave rectified. Furthermore all the negative values equalized to zero which yields a resulting detection function for each frame as

$$\widetilde{O}_i(n) = HWR(O_i(n) - \overline{O}_i(n)) \tag{A.4}$$

where HWR denotes half-wave rectification, $HWR(x) = (x + |x|)/2$ and the autocorrelation function, $A(k)$ is calculated and normalized to linearize due to the increasing lags as follows.

i) Beat Period Extraction

$$A(k) = \frac{\sum_m^(B_f)\widetilde{O}_i(m)\widetilde{O}_i(m-i)}{|l - B_f|} \qquad l = 1, ..., B_f \tag{A.5}$$

Tempo is estimated by the following relationship,

$$bpm = \frac{60}{l \times t_{DF}} \tag{A.6}$$

where t_{DF} is determined as 11.6 ms which is used in a recent study [53]which makes the beat tracking process independent from sampling frequency of the input. Then a similar method employed to find the beat period in which comb filter-type structures are used. Integer multiples of periodicity, τ, of the corresponding weighted delta functions are added to generate a comb template $C_\tau(k)$ which is able to reflect periodicity at different metrical levels.

$$C_\tau(k) = \sum_{p=1}^{4} \sum_{v=1-p}^{p-1} \frac{\delta(k - \tau p + v)}{2p - 1} \tag{A.7}$$

A number of comb templates are combined with a weighting Rayleigh distribution function into a shift-invariant comb filterbank $F_R(k, \tau)$. The weighting function, $\omega_R(\tau)$, is determined in Rayleigh distribution in order to decrease the short lags strongly and the long lags slowly,

$$\omega_R(\tau) = \frac{\tau}{\beta^2} e^{-r^2/2\beta^2} \tag{A.8}$$

where β has a range between 40 and 50 detection function samples. The shift-invariant comb filterbank $F_R(k,\tau)$ is represented in matrix form as follows.

$$F_R(k,\tau) = \omega_R(\tau) C_\tau(k) \tag{A.9}$$

$F_R(k,\tau)$ is multiplied by A(k) and the index for the corresponding maximum value of the resulting function yields as the beat period, τ_R.

$$b_R(\tau) = \sum_{k=1}^{B_f} A(k) F_R(k,\tau) \tag{A.10}$$

$$\tau_R = \underset{\tau}{\text{argmax}}(b_R(\tau)) \tag{A.11}$$

A.2 ii) Beat Alignment Extraction

The second purpose of the general state of this algorithm is to find the beat locations given the calculated beat period. A comb filter matrix $H_R(n,\alpha)$ is calculated from the onset detection function containing all possible shifts of beat period. An alignment comb template $\psi_\alpha(n)$ is constructed from an impulse sequence at beat period intervals. A linearly decreasing weighting function $\nu(n)$ is multiplied by each comb filter under the assumption that tempo is not constant through the whole audio,

$$\psi_\alpha(n) = \sum_{m=1}^{\lfloor B_f/\tau_R \rfloor} \nu(n)\delta(n - m\tau_R + \alpha) \qquad n = 1, ..., B_f \tag{A.12}$$

where α represents beat period intervals offset and

$$\nu(n) = \frac{B_f - n}{B_f} \qquad n = 1, ..., B_f. \tag{A.13}$$

The comb filer matrix $H_R(n,\alpha)$ is calculated as follows:

$$H_R(n,\alpha) = \psi_\alpha(n) \qquad 1 \leq \alpha \leq \tau_R. \tag{A.14}$$

The α_R is determined as the index for the corresponding maximum value of the multiplication of the comb filter matrix $H_R(n,\alpha)$ with the detection function as a similar approach in beat period calculation.

$$z_R(\alpha) = \sum_{n=1}^{B_f} O_i(n) H_R(n, \alpha) \tag{A.15}$$

$$\alpha_R = \underset{\boldsymbol{\alpha}}{\text{argmax}}(z_R(\alpha)) \tag{A.16}$$

Since the beat period and the alignment offset α_R is calculated the beat locations with in a frame is calculated as follows:

$$\gamma_{i,b} = (t_i + \alpha_R) + (b-1)\tau_R \qquad b = 1, ..., B \tag{A.17}$$

where t_i denoted the starting time of he detection function frame, $O_i(m)$.

A.3 C) Context-Dependent State

The General State is a memoryless process which estimates beat alignment by applying a recursive algorithm with a limitation of estimating beat alignment in between frames that are independent of previous frames. This may cause meter boundary and on and off-beat switching. In order to eliminate these errors, Context-Dependent State is generated incorporates tempo and beat alignment information extracted in General State.

A.4 i) Meter Estimation

The Context-Dependent State differs from the General State during the process of the beat period extraction. The Context-Dependent State, first calculates the meter of the given audio signal and the resulting meter value is applied in to comb template in which the element number is set to the meter value. Then a Gaussian weighting localized to the prior beat period is used instead of the Rayleigh weighting curve. In order to extract meter of the input a similar method proposed in [54] is employed which relies on extracting a strong peak in the meter level periodic autocorrelation function. Then a similar method described in [**?**] is applied in order to classify the input meter as being binary or a ternary. Given a beat period from the General State analysis, meter value is determined by comparing the odd and even multiples of the beat period energy through the autocorrelation function by employing the condition given below.

$$A(2\tau_R) + A(4\tau_R) > A(3\tau_R) + A(6\tau_R) \tag{A.18}$$

A.5 ii) Beat Period Extraction

If the given condition satisfied, the meter value T is set to $T = 4$, otherwise $T = 3$. Furthermore, the number of elements in each comb template, $C_\tau(k)$, is set to T and the resulting comb filterbank matrix, $F_c(k, \tau)$, is recalculated by the given comb template below.

$$C_\tau(k) = \sum_{p=1}^{T} \sum_{v=1-p}^{p-1} \delta(k - \tau p + v) \tag{A.19}$$

The beat period derived from the General State helps limiting the range of possible beat periods by a context-dependent Gaussian function localized around τ_R instead of employing the Rayleigh weighting.

$$\omega_C(\tau) = e^{-(\tau-\tau_R)^2/2\sigma_w^2} \tag{A.20}$$

where σ_w determines the beat period range. Then the shift-invariant comb filterbank for the Context- Dependent State, $F_C(k, \tau)$ is formulated as follows, by setting each comb template $C_\tau(k)$ as the τth column of the matrix and weighting each template by the Gaussian weighting.

$$F_C(k, \tau) = omega_C(\tau) C_\tau(k) \tag{A.21}$$

As in General State, the beat period tau_C is extracted from the product of the autocorrelation function,$A(k)$ and comb filterbank matrix as follows,

$$y_C(\tau) = sum_{k=1}^{B_f} A(k) F_C(k, \tau), \tag{A.22}$$

$$\tau_C = \underset{\boldsymbol{\tau}}{\operatorname{argmax}}(y_C(\tau)). \tag{A.23}$$

A.6 iii) Beat Alignment Extraction

Beat alignment extraction in Context-Dependent State is similar as in General State. The Context-Dependent State aims to extract a consistent beat alignment.

The offset from the location of the first beat within the current detection function frame to the start of that frame is calculated by beat alignment. α_C is predicted by adding τ_CDF samples to the last beat $\gamma_{i-1,B}$ from the previous detection function frame as follows,

$$\alpha_C \approx \gamma_{i-1,B} + \tau_C. \tag{A.24}$$

To allow flexibility to the beat locations, an weighting function $\rho_C(\alpha)$ is created for beat alignment localized around the α_C,

$$\rho_C(\alpha) = e^{-(\alpha-(\gamma_{i-1,B}+\tau_C))^2/2\sigma_p^2}. \tag{A.25}$$

Then a context-dependent comb filter matrix $H_C(m,\alpha)$ is formulated, with an alignment comb $\psi_\alpha(m)$ at an offset on each column,

$$\psi_\alpha(m) = sum_{k=1}^{\lfloor B_f - \tau_C \rfloor} \nu(m)\delta(m - k\tau_C + \alpha), \tag{A.26}$$

where

$$H_C(m,\alpha) = \rho_C(\alpha)\psi_\alpha(m). \tag{A.27}$$

The α_C is determined as the index for the corresponding maximum value of the multiplication of the comb filter matrix $H_C(n,\alpha)$ with the detection function as a similar approach in beat period calculation.

$$z_C(\alpha) = \sum_{n=1}^{B_f} O_i(n)H_C(n,\alpha) \tag{A.28}$$

$$\alpha_C = \underset{\boldsymbol{\alpha}}{\operatorname{argmax}}(z_C(\alpha)) \tag{A.29}$$

Since the beat period and the alignment offset α_C is calculated the beat locations with in a frame is calculated as follows:

$$\gamma_{i,b} = (t_i + \alpha_C) + (b-1)\tau_C \qquad b = 1, ..., B \tag{A.30}$$

where t_i denoted the starting time of he detection function frame, $O_i(m)$.

A.7 C) Two-State Model

The proposed Two-State Model is depicted in Fig. A.1. The General State and Context-Dependent States are able to work in discrete modes in order to extract beat alignment but yields unsatisfactory results in order to maintain continuity of extracting correct beat

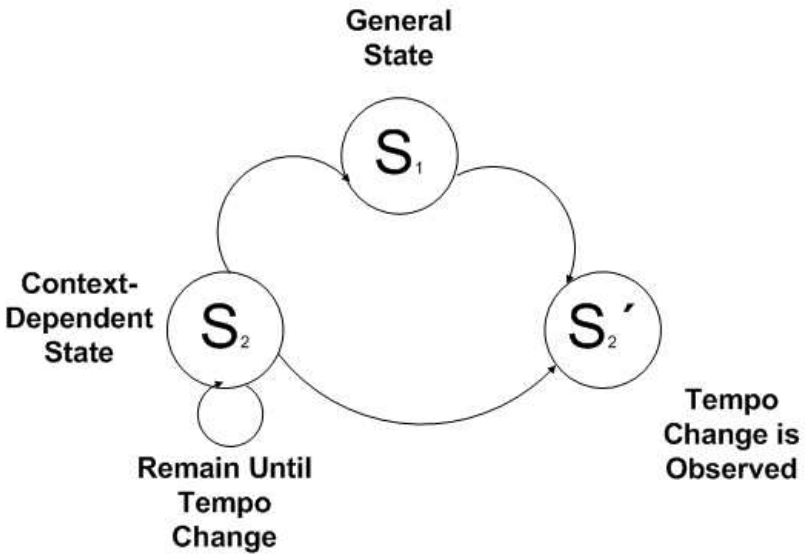


Figure A.1: Two-State switching model: Analysis begins in General State S_1 . The observation of a consistent beat period generates a Context-Dependent State S_2 that is replaced by $S_2^{'}$ if a tempo change is observed.

locations. The General State is unable to switch metrical levels. The Context-Dependent State is able to prevent these errors by limiting the range of beat periods and alignments by using the prior knowledge from the General State, but it is unable to adopt changing beat periods. In order to use both state advantages Two-State Model is generated. The General State is employed in order to estimate the initial beat period and detect changing beat periods, and the Context-Dependent State is added as the second state which maintains the continuity within a given beat period.

The General State and the Context-Dependent State are labeled as S_1 and S_2 respectively. The current state variable is notated with q and assigned $q = S_1$ at first. The initial beat locations are extracted employing the General State.

The Context-Dependent State is activated when beat periods of three consecutive frame stay consistent satisfying the condition below.

$$|2\tau_R(i) - \tau_R(i-1) - \tau_R(i-2)| < \eta_1 \tag{A.31}$$

where $\tau_R(i)$ is ith frame beat period and η_1 is the threshold which is defined $\eta_1 = \sigma_w$ at 4 DF samples that stands in between the range of Gaussian weighting. If the given condition

is satisfied $q = S_2$ and the Context-Dependent state is activated. Meanwhile the General State continues to extracting beat period. τ_R and τ_G are compared at each frame and if has a small difference the Context-Dependent State continues to operate. The condition is given as follows where $\eta_2 = \eta_1$ where η_2 stands outside of the Gaussian weighting range.

$$|\tau_R(i) - \tau_C(i)| < \eta_2 \tag{A.32}$$

If this condition is not satisfied, the Context-Dependent State S_2 is replaced by a new Context-Dependent State $S_2^{'}$. Before passing to $q = S_2^{'}$, α_R is recalculated by the General State and new reference locations are extracted for the second stage analysis. For the following frames, the new Context-Dependent State $S_2^{'}$ is employed to extract beat locations till a consistent beat period is observed.

This two-state model works properly especially in resolving tempo discontinuities and eliminating errors caused by odd beats. Further details of the complete beat tracking system can be found in Davies and Plumbley [34].

Appendix B

ONSET DETECTION FUNCTION

The complex spectral difference onset detection function is used for calculations as described in[55]. Overlapping short-term audio frames are analyzed in order to derive onset detection function. A fixed time-resolution for each detection function, independent of the input sampling frequency, the hop size, h, is set to a sampling frequency dependent constant: $h = f_s \times t_{DF}$ which represents the resolution of each sample of detection function. We then apply a $N = 2h$ long of the analysis frame which results as 50% overlapping frames which at each iteration. For an audio sampled at 44.1 kHz, we calculate a frame size of $N = 1024$ with a hop size of $h = 512$ audio samples.

We take short-term Fourier transform (STFT) of an input audio signal $s(n)$ which is windowed by an N length Hanning window $w(m)$ and find the kth bin of the short term spectral frame $S_k(n)$ as

$$S_k(m) = \sum_{n=-\infty}^{\infty} s(n)w(mh - n) \exp^{-2\pi nk/N} \tag{B.1}$$

and the magnitude, $R_k(m)$, of short term spectral frame is calculated as

$$R_k(m) = |S_k(m)| \tag{B.2}$$

where $S_k(m) \in \mathbb{C}$ and $S_k(m) \in \mathbb{R}$.

By applying the assumption that during steady-state regions of the signal the magnitude spectrum should remain approximately constant, we can make a prediction of the magnitude spectrum $\widetilde{R}_k$ at frame m, given the magnitude of the previous frame

$$\widetilde{R}_k(m) = R_k(m - 1) \tag{B.3}$$

Then according to the assumption that during steady state regions of the signal, the phase velocity at the kth bin ideally be constant is estimated as

$$\phi_k(m) - \phi_k(m - 1) \approx \phi_k(m - 1) - \phi_k(m - 2). \tag{B.4}$$

A short-hand notation is replaced with the left-hand side of phase velocity estimation.

$$\Delta\phi_k(m) = \phi_k(m) - \phi_k(m-1)] \tag{B.5}$$

By substituting the short-hand notation into phase velocity estimation equation, we get

$$\phi_k = princarg[\phi_k(m) - \Delta\phi_k(m-1)] \tag{B.6}$$

where *princarg* maps the phase value into the range $[-\pi, \pi]$.

The magnitude spectrum $\widetilde{R}_k$ and the phase spectrum $\phi_k(m)$ can be represented in polar form resulting a spectral prediction in the complex domain

$$\widetilde{S}_k(m) = \widetilde{R}_k(m) \exp^{j\phi_k(m)} \tag{B.7}$$

which is compared to the observed complex spectrum

$$S_k(m) = R_k(m) \exp^{j\phi_k(m)} \tag{B.8}$$

In order to derive the complex spectral difference detection function $O(m)$ at frame m , the sum of the Euclidean distance between the predicted and observed spectra for all bins is calculated which is given below,

$$O(n) = \sum_{k=1}^{K} (S_k(n) - \tilde{S}_k(n))^2. \tag{B.9}$$

BIBLIOGRAPHY

[1] Y. Kuniyoshi, M. Inaba, and H. Inoue, "Learning by watching: Extracting reusable task knowledge from visual observation of human performance," *T-RA*, vol. 10, pp. 799–822.

[2] Randy Pausch, Jon Snoddy, Robert Taylor, Scott Watson, and Eric Haseltine, "Disney's aladdin: first steps toward storytelling in virtual reality," in *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, New York, NY, USA, 1996, pp. 193–203, ACM.

[3] K. Laybourne, *The Animation Book*, Three Rivers Press, 1998.

[4] Justine Cassell, Hannes Högni Vilhjálmsson, and Timothy Bickmore, "Beat: the behavior expression animation toolkit," in *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, New York, NY, USA, 2001, pp. 477–486, ACM.

[5] Michael Isard and Andrew Blake, "A mixed-state condensation tracker with automatic model-switching," in *ICCV '98: Proceedings of the Sixth International Conference on Computer Vision*, Washington, DC, USA, 1998, p. 107, IEEE Computer Society.

[6] F. Ofli, Y. Demir, E. Erzin, Y. Yemez, and A. M. Tekalp, "Multicamera audio-visual analysis of dance figures," *Multimedia and Expo, 2007 IEEE International Conference on*, pp. 1703–1706, 2-5 July 2007.

[7] Tae hoon Kim, Sang Il Park, and Sung Yong Shin, "Rhythmic-motion synthesis based on motion-beat analysis," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 392–401, 2003.

[8] A. Ikeuchi K. Shiratori, T. Nakazawa, "Dancing-to-music character animation," *COMPUTER GRAPHICS FORUM*, vol. 25, no. 3, pp. 449–458, 2006.

[9] D. Eck, M. Gasser, and R. Port, "Dynamics and embodiment in beat induction," 2000.

[10] P. J. Essens and D. Povel, *Metrical and Nonmetrical Representation of Temporal Patterns*, Perception and Psychophysics, 1985.

[11] M. R. Jones and M. Boltz, "Dynamic attending and responses to time," *Psychological Review*, vol. 96, pp. 459–491, 1989.

[12] E. Large and J. Kolen, "Resonance and the perception of musical meter," 1994.

[13] Masahide Naemura and Masami Suzuki, "Extraction of rhythmical factors on dance actions thorough motion analysis," in *WACV-MOTION '05: Proceedings of the Seventh IEEE Workshops on Application of Computer Vision (WACV/MOTION'05) - Volume 1*, Washington, DC, USA, 2005, pp. 407–413, IEEE Computer Society.

[14] A. S. Bregman, *Auditory Scene Analysis: The Perceptual Organization of sound*, The MIT Press, 1990.

[15] C. Bregler and J. Malik, "Tracking people with twists and exponential maps," in *CVPR '98: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, 1998, p. 8, IEEE Computer Society.

[16] Guy J. Brown and Martin Cooke, "Computational auditory scene analysis:exploiting principles of perceived continuity," *Computer Speech and Language*, vol. 8, pp. 297–336, 1994.

[17] B. Logan and S. Chu, "Music summarization using key phrases," in *ICASSP '00: Proceedings of the Acoustics, Speech, and Signal Processing, 2000. on IEEE International Conference*, Washington, DC, USA, 2000, pp. II749–II752, IEEE Computer Society.

[18] J. Aucouturier and M. Sandler, "Segmentation of musical signals using hidden markov models," 2001.

[19] M. Sandler M. Casey and C.Rhodes S. Abdallah, K. Nolad, "Theory and evaluation of a bayesian music structure extractor," *In Proc. of 6th International Conference on Music Information Retrieval, London, UK, Sept.*, 2005.

[20] C. Rhodes, M. Casey, S. Abdallah, and M. Sandler, "A markov-chain monte-carlo approach to musical audio segmentation," *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 5, pp. V–V, 14-19 May 2006.

[21] M. Levy, M. Sandier, and M. Casey, "Extraction of high-level musical structure from audio data and its application to thumbnail generation," *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 5, pp. V–V, 14-19 May 2006.

[22] Jonathan Foote, "Visualizing music and audio using self-similarity," in *MULTIMEDIA '99: Proceedings of the seventh ACM international conference on Multimedia (Part 1)*, New York, NY, USA, 1999, pp. 77–80, ACM.

[23] J. Foote, "Automatic audio segmentation using a measure of audio novelty," *Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on*, vol. 1, pp. 452–455 vol.1, 2000.

[24] J. T. Foote and M. L. Cooper, "Media segmentation using self-similarity decomposition," *In Proc. of The SPIE Storage and Retrieval for Multimedia Databases*, vol. 5021, pp. pages 167175 vol.5021, 2003.

[25] G. Peeters, "Deriving musical structures from signal analysis for music audio summary generation: sequence and state approach," in *Computer Music Modeling and Retrieval*, 143-166, Ed. Springer Berlin / Heidelberg, 2004.

[26] Lie Lu, Muyuan Wang, and Hong-Jiang Zhang, "Repeating pattern discovery and structure analysis from acoustic music data," in *MIR '04: Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, New York, NY, USA, 2004, pp. 275–282, ACM.

[27] Y. Wang M. Skankanhalli X. Shao, C. Xu, "Automatic music summarization in compressed domain," *In Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing*, 2004.

[28] "Contra/country dance markup language project," `http://www.farmdale.com/cdml/`.

[29] H. Jeong Y. Shiu and C.-C. J. Kuo, "Musical structure analysis using similarity matrix and dynamic programming," in *In Proc. of SPIE on Multimedia Systems and Applications VIII*, 2005, vol. 6015.

[30] Namunu C. Maddage, Changsheng Xu, Mohan S. Kankanhalli, and Xi Shao, "Content-based music structure analysis with applications to music semantics understanding," in *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, New York, NY, USA, 2004, pp. 112–119, ACM.

[31] E.D. Scheirer, "Tempo and beat analysis of acoustical musical signals," *Journal of the Acoustical Society of America*, vol. 103, pp. 588 601, 1998.

[32] Richard G. Alonso, M. and B. David, "Tempo estimation for audio recordings," *Journal of New Music Research*, vol. 36, pp. 17–25, 2007.

[33] Pikrakis A. Antonopoulos, I. and S. Theodoridis, "Self-similarity analysis applied on tempo induction from music recordings," *Journal of New Music Research*, vol. 36, pp. 27 – 38, 2007.

[34] Matthew E. P. Davies and Mark D. Plumbley, "Context-dependent beat tracking of musical audio," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 3, pp. 1009–1020, March 2007.

[35] D.P.W. Ellis, "Beat tracking with dynamic programming," *Journal of New Music Research*, vol. 36, pp. 51 60, 2007.

[36] A.P. Klapuri, A.J. Eronen, and J.T. Astola, "Analysis of the meter of acoustic musical signals," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 1, pp. 342–355, Jan. 2006.

[37] Moelants D. Davies M. E. P. McKinney, M. F. and A. Klapuri, "Evaluation of audio beat tracking and music tempo extraction algorithms," *Journal of New Music Research*, vol. 36, pp. 1–16, 2008.

[38] H.C. Longuet-Higgins and M.J. Steedman, *On interpreting Bach*, Edinburgh University Press.

[39] Dirk-Jan Povel and Peter Essens, "Perception of temporal patterns," *Music Perception*, vol. 2, no. 4, pp. 411–440, 1985.

[40] J. D. McAuley, "Finding metrical structure in time," in *Proceedings of the 1993 Connectionist Models Summer School*, 1993, pp. 219–227.

[41] R. B. Dannenberg and B. Mont-Reynaud, "Following an improvisation in real-time," in *Proceedings of the International Computer Music Conference, San Francisco*. International Computer Music Association, 1987, pp. 241–248.

[42] George Tzanetakis, "Automatic musical genre classification of audio signals.," in *ISMIR : International Conference on Music Information Retrieval*, 2001.

[43] Ulas Bagci and Engin Erzin, "Boosting classifiers for music genre classification," in *International Symposium on Computer and Information Sciences (ISCIS)*, 2005, pp. 575–584.

[44] R. N. Shepard, "Circularity in judgements of relative pitch," *Journal of the Acoustical Society of America*, vol. 36, pp. 2346–2353, 1964.

[45] D.P.W. Ellis and G.E. Poliner, "Identifying 'cover songs' with chroma features and dynamic programming beat tracking," *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 4, pp. IV–1429–IV–1432, 15-20 April 2007.

[46] H.-M. Yu W.-H. Tsai and H.-M. Wang, "A query-by-example technique for retrieving cover versions of popular songs with similar melodies," *In Proc. Int. Conf. on Music Info. Retr. ISMIR-05, London*, p. 183190, 2005.

[47] G.H Wakefield M.A. Bartsch, "Audio thumbnailing of popular music using chroma-based representations," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 96– 104, 2005.

[48] M.E. Sargin, Y. Yemez, E. Erzin, and A.M. Tekalp, "Analysis of head gesture and prosody patterns for prosody-driven head-gesture animation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1330– 1345, 2008.

[49] M.E. Sargin, Y. Yemez, E. Erzin, and A.M. Tekalp, "Audiovisual synchronization and fusion using canonical correlation analysis," *IEEE Transactions on Multimedia*, vol. 9, no. 7, pp. 1396– 1403, 2007.

[50] F. Ofli, Y. Demir, Y. Yemez, E. Erzin, A. M. Tekalp, K. Balc, I. Kzoglu, L. Akarun, C. Canton-Ferrer, J. Tilmanne, E. Bozkurt, and A. T. Erdem, "An audio-driven dancing avatar," *Journal on Multimodal Interfaces*, vol. 2, no. 2, pp. 93–103, 2008.

[51] Michael John Collins, "A new statistical parser based on bigram lexical dependencies," pp. 184–191, 1996.

[52] "Synthesis demos," `portal.ku.edu.tr/~ydemir/thesisDemos`, 2008.

[53] J.P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M.B. Sandler, "A tutorial on onset detection in music signals," *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 5, pp. 1035–1047, Sept. 2005.

[54] J. C. Brown, "Determination of the meter of musical scores by autocorrelation," *Journal of the Acoustical Society of America*, vol. 94, no. 4, pp. 1953–1957, 1993.

[55] J.P. Bello, C. Duxbury, M. Davies, and M. Sandler, "On the use of phase and energy for musical onset detection in the complex domain," *Signal Processing Letters, IEEE*, vol. 11, no. 6, pp. 553–556, June 2004.

VITA

YASEMIN DEMIR was born in Konya, Turkey on May 21, 1984. She received her B.Sc. degree in Telecommunications Engineering from Istanbul Technical University, Istanbul, Turkey, in 2006. From August 2006 to July 2008, she worked as a teaching and research assistant in Koç University, Istanbul, Turkey. At Koç University, she focused on Evaluations of Audio Features for Audio-Visual Analysis and Synthesis of Dance Figures, which has been supported by TUBITAK BIDEB and European FP6 Network of Excellence SIMILAR project. She attended the SIMILAR NoE Summer Workshop on Multi-Modal Interfaces, Istanbul, Turkey on July-August 2007.

She has submitted a journal publication and published several papers about Multimodal Signal Analysis and Synthesis in the following conferences: SIU2007 (Eskişehir, Turkey), eNTERFACE2007 (Istanbul, Turkey), EUSIPCO2007 (Poznań, Poland), EUSIPCO2008 (Lausanne, Switzerland), SIU2008 (Didim, Turkey), ICASSP2008 (Las Vegas, USA), ICME2007 (Beijing, China).