

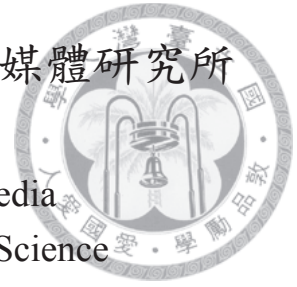
國立臺灣大學電機資訊學院資訊網路與多媒體研究所

博士論文

Graduate Institute of Networking and Multimedia
College of Electrical Engineering and Computer Science

National Taiwan University

Doctoral Dissertation



基於銜接技術之音樂改作

Concatenative Audio Music Re-composition

林映孜

Yin-Tzu Lin

指導教授：吳家麟博士、張智星博士

Advisor: Ja-Ling Wu, Ph.D.,

Jyh-Shing Roger Jang, Ph. D.

中華民國 104 年 1 月

January, 2015



國立臺灣大學博士學位論文
口試委員會審定書

基於銜接技術之音樂改作

Concatenative Audio Music Re-composition

本論文係林映孜君（學號 D98944002）在國立臺灣大學資訊網路與多媒體研究所完成之博士學位論文，於民國一百零三年十二月卅一日承下列考試委員審查通過及口試及格，特此證明

口試委員：

<u>吳宗麟</u>	<u>張紹星</u> (簽名)
(指導教授)	
<u>陳恆石</u>	<u>王新民</u>
<u>鄭文皇</u>	<u>楊奕軒</u>
_____	_____
_____	_____

所長：

逢愛君





誌謝

很高興在 2014 年的最後一天通過我的博士論文學位考試，記得在口試的前一週吳老師要我把學術生涯中所有的發表做一個整理，一邊整理一邊真的覺得很感謝老師們跟學長姐學弟妹們平時對我的幫助與鼓勵，許多的 project 如果不是大家的幫助與討論，也許不會完成也不會寫成 paper 投出去，裡面有好幾篇我還覺得，結果不怎樣好像也沒甚麼好寫的，但老師跟學長姐一直鼓勵我，一個 conference 投不上，老師就會推薦我改一改再投下一個，直到最後終於投上。在投的當時有時我還天真的覺得不以為然，因為覺得明明就不夠好，但是就在這一次一次的修改中，不知不覺越來越完整，最後回顧起來，真是覺得還好有投。

這個學位的完成，要感謝許許多多的人，首先要感謝我的指導教授吳家麟老師一直以來對我的鼓勵跟包容，謝謝老師給了我這樣一個讓我自行探索的空間，卻又常常不著痕跡的提醒催促著我要趕快不要拖，在我們老是拖拖拉拉寫很慢的時候，老師卻總是好快就把論文改好並給予我們許多建議，在老師身上學到的東西好多好多，像是面對研究的態度、處事的方法、present 的技巧等等，真的不能用文字能夠表達完。再來也要感謝我的第二位指導教授張智星老師，在我博士生涯的最後幾年來到台大，我在張老師身上學到了許多跟吳老師關注的不同面向的東西，感謝老師在我常常無限上綱要複雜化事情的時候拉著我將它簡化，也感謝老師花了好多的時間幫我修改 journal，還有給我機會擔任 MSAR 課程助教跟著學習 handle 大班級的課程，也謝謝老師幫助我蒐集實驗資料。謝謝鄭文皇老師，從碩班起便推著帶著我發了好幾篇 paper，又教了我許多初做研究的基本功夫，而且總是以搞笑的方式化解我的緊張，最終也撥冗擔任了 proposal 跟 defense 的口試委員。謝謝陳祝嵩老師，給了我機會擔任 DSP 課程助教，在課程上學了許多，也謝謝老師在 Proposal 口試時給予的建議，後來朝這方向發展還在 ISM 得了獎。謝謝 proposal 的口試委員陳文進老師、defense 的口試委員：王新民老師、陳恆佑老師、楊奕軒老師撥冗來參與口試，並給予我許多建議與肯定，未來我會繼續努力改進。謝謝 CMLab 的老師群：歐陽明老師、陳文進老師、莊永裕老師、陳炳宇老師、周承復老師、徐宏民老師，網媒所所長逢愛君老師、網媒所導師洪一平老師，時常與我聊天，關心我的狀況並給予我建議。謝謝陳彥仰老師在我博一時給我機會讓我參與了 HCI 的 project，顛覆了我對於 UI 設計的概念，也帶給我寫 App 的體驗。謝謝洪士灝老師在我博二修課時帶給我平行程式的觀念，老師臉書的文字也時常讓我思考不同面向的事情。

感謝我的好同學裕訓，總是告訴我許多新訊息，常常跟我聊天化解我許多的莫名自我糾結跟疑慮，以及口試上的各種幫忙。謝謝嘉祐學長在我博士生涯最後這年回 lab 博後與我們討論切磋，在好多次我又想放棄時，說服我一定要投出 paper，還有許多其他待人處事的建議，十分感謝學長。謝謝仲毅學長最後這幾個月回來 lab 跟我們一起 meeting，在博士論文最後 shaping 的階段與我討論辯論，給了我許多很棒的建議，在我悲觀自卑糾結的小劇場時拉了我好幾把。謝謝敏君學姊，教了我許多帶學弟妹的技巧，也建立了一套給新進學弟妹的訓練

模式，在撰寫國科會報告上傳承了我許多的經驗，才讓我在實驗室事務運作上更加地順利，也總是很正面又很 high 的鼓勵我。謝謝官順暉學長，在太極合作的那段日子，很感謝學長時常跟我討論並鼓勵我，跟我聊許多的新資訊，特別謝謝學長邀請我加入優人的案子，才讓我的博士生涯有了第一篇 first author 的 paper，建立了一些信心。謝謝大鈞學弟，在我忙於論文時幫忙 cover 了許多實驗室的事務。謝謝賴瑞欣學長，在我博班生涯的中段來到 lab 博後一年多，與我討論促成了我想法上的成長，學長正向的態度也影響了我許多。謝謝育慈學姊，給了我機會去嘗試教課，並磨練了我的表達。謝謝 DSP 組博班學長們：謝致仁、沈允中、黃俊翔、郭晉豪、朱威達，每次有遇到我時都會鼓勵我，告訴我說一切都會沒問題。

謝謝與我合作 project 的學弟妹林恆毅、林君毅、林霓苗、劉怡廷、陳柏年、聞浩凱、李泉龍，感謝有你們跟我一起討論、容忍我的猶豫不決，謝謝你們超棒的執行力，這些 project 才能完成。謝謝 DSP 組所有的碩班學弟妹：張銘修、張炳傑、陳鴻銘、江明哲、黃彥霖、林恆毅、顏芷妤、程瀚平、邱柏叡、胡傳姓、林志宏、白育姍、葉容瑜、陳祺文、周瑋慈、林弘偉、王品翔、蘇則仲、曾筱雲、曾翊寧、張宇蓓、連奕婷、蔡明宏、陳群元、羅際巧、傅承堯、顧宗浩、李明璋、劉怡廷、繆昕、王舜玄、楊竣宇、陳厚凱、陳柏年、周于荃、孫家豪、聞浩凱、吳政陽、謝宗廷、何哲廷、江宛峯、李泉龍、梁振鋒、邱靖詠、姚尊仁、張祐榕、江嫚書、林杰鴻、林育辰、蘇兆為、彭奕嘉、游宗霖，謝謝你們在實驗室的事務上的幫忙，才能減輕我的負擔。謝謝 CMLab 實驗室的成員：陳美鑿、陳心怡、張明旭、許智成、翁明昉、葉哲華、鄭鎧尹、黃子魁、黃子桓、李根逸、陳冠婷、郭盈希、張哲瀚、黃群凱、蘇彥禎、吳昱霆、梁容豪、羅聖傑、陳銘宏、林靖茹、胡俊彥，對我的鼓勵，平時時的聊天與在實驗室事務上的互相幫助。謝謝 MIR-Lab 實驗室的成員劉怡芬、葉子雋、陳亮宇、范哲誠、呂俊宏、王崇喆、蘇昭宇、顏明祺、周思瑜，謝謝你們在 meeting 時給予我的建議，以及跟我聊天排解苦悶。謝謝 MSAR 兩屆的修課同學，跟你們切磋也讓我學到許多，謝謝你們幫忙我實驗 data 的蒐集。

謝謝實驗室助理賴怡嘉、賴紫晴、周如虹，對於報帳的幫助，以及器材借用，實驗室各項事務上的協助，還有平時跟我聊天，排解我的各種糾結。謝謝網媒所打工的老闆雅琳姐，讓我有機會幫忙所務運作，謝謝你在各式畢業行政上的支援與幫忙！謝謝網媒所打工的好同事林宛諭、邱德泉、陳永祥、廖文慶、周承滿、林芳而，謝謝大家平時互相 cover、代班，以及互相鼓勵聊天排解苦悶，特別謝謝婉瑜在口試事務上的幫忙。

謝謝我高中大學的好姊妹瑄慧、士甄、又慈、皮皮、珏珏、筱雨、耀萱、信聰，謝謝大家在我博士之路時常鼓勵我，聽我 murmur，也謝謝幫我評了線上的問卷，謝謝我的聲樂老師何欣蘋，每次聲樂課跟老師聊完天都覺得好有能量。謝謝青韻的眾多好友與學長姐學弟妹：特別謝謝子苙姊姊、韶純姊姊、瑋芳姊姊、禹任、紋菱、星兒、亭汝、小陽光、為之、小綠、博文、庭毅等平時對我的鼓勵，每次的練唱都讓我的心靈得到釋放，也謝謝大家包容我，幫忙我寫了好多的線上問卷評分。最後謝謝我的家人，謝謝老公、公公、婆婆、爸爸、媽媽、大姑、大伯、哥哥、嫂嫂對我各方面的支持，在許多我想要放棄的時候，謝謝你們支撐著我，特別謝謝阿嬤，在 journal revise 階段幫我詢問了專業人士，給予我很有建設性的建議。由衷感謝我所有的長輩與朋友們，在我博士生涯中對我的鼓勵與幫助，未來期許自己也能貢獻自己的能量給身旁的人。

林映孜 謹誌
民國 104 年 1 月

Yin-Tzu LIN 林映孜



EDUCATION

- Sep. 2009 – NATIONAL TAIWAN UNIVERSITY (NTU) Taipei, Taiwan
Jan. 2015 *Ph.D., Graduate Institute of Networking and Multimedia (GINM)*
- Thesis Topic: Concatenative Audio Music Re-composition
 - Advisor: Ja-Ling Wu, Jyh-Shing Roger Jang
- Feb. 2006 – NATIONAL TAIWAN UNIVERSITY (NTU) Taipei, Taiwan
Feb. 2008 *M.S., Department of Computer Science and Information Engineering (CSIE)*
- Thesis Topic: Cadence Detection for Music Structure Analysis
 - Advisor: Ja-Ling Wu
- Sep. 2002 – NATIONAL TSING HUA UNIVERSITY (NTHU) HsinChu, Taiwan
Feb. 2006 *B.S., Department of Computer Science (CS)*
- GPA 3.94

EXPERIENCE

- July 2010 – INTERNSHIP
June 2013 *Digimax Inc.*
- Feb. 2008 – RESEARCH ASSISTANT
Sep. 2009 *Communications and Multimedia Laboratory (CML), CSIE, NTU*

AWARD & HONORS

- **BEST STUDENT PAPER AWARD**,
IEEE International Symposium on Multimedia (ISM 2014)
- **EXCELLENT TEACHING ASSISTANT AWARD (優良助教獎)**
CSIE, National Taiwan University (Spring 2013)

PUBLICATIONS

INTERNATIONAL JOURNAL

1. **Yin-Tzu Lin**, I-Ting Liu, Jyh-Shing Roger Jang, and Ja-Ling Wu, "**Audio Musical Dice Game: A User-preference-aware Musical Medley Generating System**," accepted by *ACM Transactions on Multimedia Computing, Communications and Applications*.
2. Wen-Huang Cheng, Yung-Yu Chuang, **Yin-Tzu Lin**, Chi-Chang Hsieh, Shao-Yen Fang, Bing-Yu Chen, and Ja-Ling Wu, "**Semantic Analysis for Automatic Event Recognition and Segmentation of Wedding Ceremony Videos**," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1639-1650, November 2008. (SCI, EI, impact factor=3.022)



INTERNATIONAL CONFERENCE

1. **Yin-Tzu Lin**, Chuan-Lung Lee, Jyh-Shing Roger Jang and Ja-Ling Wu, "**Bridging Music via Sound Effects**", accepted by *IEEE International Symposium on Multimedia (ISM2014)*, Taichung, Taiwan, Dec. 10-12, 2014. (full paper, acceptance rate 22.47%, *best student paper award*)
2. Hao-Kai Wen, Wei-Che Chang, Chia-Hu Chang, **Yin-Tzu Lin**, Ja-Ling Wu, "**Event Detection for Broadcast Halfpipe Sports Video**", accepted by *the 22nd ACM International Conference on Multimedia (ACM MM 2014)*, Orlando, Florida, USA, Nov. 3-7, 2014. (technical demo)
3. **Yin-Tzu Lin**, Po-Nien Chen, Chia-Hu Chang, Ja-Ling Wu, "**MSVA: Musical Street View Animator: An Effective and Efficient Way to Enjoy the Street Views of Your Journey**", accepted by *the 22nd ACM International Conference on Multimedia (ACM MM 2014)*, Orlando, Florida, USA, Nov. 3-7, 2014. (short paper)
4. **Yin-Tzu Lin**, I-Ting Liu, Jyh-Shing Roger Jang, and Ja-Ling Wu, "**Audio Musical Dice Game: A Demonstration for Personalized Medley Creation System**," in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014)*, Taipei, Taiwan, Oct. 27-31, 2014. (late breaking demo)
5. **Yin-Tzu Lin**, Tsung-Hung Tsai, Min-Chun Hu, Wen-Huang Cheng, Ja-Ling Wu, "**Semantic based Background Music Recommendation for Home Videos**," in *Proceedings of the 20th International Conference on Multimedia Modeling (MMM 2014)*, Dublin, Ireland, Jan. 8-10, 2014. (short paper) (EI)
6. I-Ting Liu, **Yin-Tzu Lin**, Ja-Ling Wu, "**Music Cut and Paste: A Personalized Musical Medley Generating System**," in *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013)*, Curitiba, PR, Brazil, Nov. 4-8, 2013.
7. **Yin-Tzu Lin**, Shuen-Huei Guan, Yuan-Chang Yao, Wen-Huang Cheng, and Ja-Ling Wu, "**U-Drumwave: An Interactive Performance System for Drumming**," in *Proceedings of the 18th International Conference on Multimedia Modeling (MMM 2012)*, Klagenfurt, Austria, Jan. 4-6, 2012 (full paper) (EI).
8. **Yin-Tzu Lin**, Wen-Huang Cheng, and Ja-Ling Wu, "**Submission to MIREX AMS Task 2011–Sparse Coding Similarity Learning Method**," participated the seventh of the Music Information Retrieval Evaluation eXchange (MIREX 2011), 2011.
9. Heng-Yi Lin, **Yin-Tzu Lin**, Ming-Chun Tien, and Ja-Ling Wu, "**Music Paste: Concatenating Music Clips Based on Chroma and Rhythm Feature**," in *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR 2009)*, Kobe, Oct. 26-30, 2009.
10. Ming-Chun Tien, Wei-Ta Chu, **Yin-Tzu Lin**, and Ja-Ling Wu, "**Sports Wizard: Sports Video Browsing Based on Semantic Concepts and Game Structure**," in *Proceedings of the 17th ACM International Conference on Multimedia (MM 2009)*, Beijing Hotel, Beijing, China, October 19-24, 2009. (EI)
11. Che-Hua Yeh, Pei-Ruu Shih, **Yin-Tzu Lin**, Kuan-Ting Liu, Huang-Ming Chang, and Ming Ouhyoung, "**A Comparison of Three Methods of Face Recognition for Home Photos**", in *Proceedings of SIGGRAPH '09: Posters*, Article 30 , 1 pages, Aug. 2009.
12. Wen-Huang Cheng, Yung-Yu Chuang, Bing-Yu Chen, Ja-Ling Wu, Shao-Yen Fang, **Yin-Tzu Lin**, Chi-Chang Hsieh, Chen-Ming Pan, Wei-Ta Chu, and Min-Chun Tien, "**Semantic-Event Based Analysis and Segmentation of Wedding Ceremony Videos**," in *Proceedings of the 9th ACM SIGMM International Workshop on Multimedia Information Retrieval (MIR 2007)*, 28-29 September, 2007, Augsburg, Germany.
13. Yu-Chien Kao, Huang-Chih Kuo, **Yin-Tzu Lin**, Chia-Wen Hou, Yi-Hsien Li, Hao-Tin Huang, Youn-Long Lin, "**A High-Performance VLSI Architecture for Intra Prediction and Mode Decision in H.264/AVC Video Encoding**," in *Proceedings of the IEEE Asia Pacific Conference on Circuits and Systems (APCCAS 2006)*, 4-7 Dec, 2006, Grand Copthorne Waterfront, Singapore.



摘要

利用既有的音訊音樂相銜接而產生新的音樂，我們稱作「基於銜接技術之音樂改作 (concatenative audio music re-composition)」。本論文針對此類音樂改作發展了一系列的技術。這些改作的音樂，可以應用在個人影片或是幻燈片 (slideshow)，或是不間斷的舞曲集錦。基於內容分析技術，樂理，以及心理聲學理論，我們提出了多種編作與選取素材的方式。首先我們可以依照相似性，句子結尾，或是小節的資訊來決定兩段音樂的接點。接著為了使音樂的節拍能夠順暢，我們提出以心理聲學為基礎的音樂速度調整方法。而為了處理節奏跟音量相差太多的素材，我們亦相對應的提出考慮兩倍節拍的速度調整法以及音量的正規化方法。在素材的選擇方面，我們提出了兩種選擇方式。一種是直接法，先利用成對的比較去除極端的音樂素材，接著利用接點的相似度來排序。而圖形法則是先將音樂的素材都處理成為樂句，藉著巧妙的內容分析技術，我們生成了一個我們稱之為音樂骰子圖 (music dice graph) 的 graph。利用這張圖，我們便可提供個人化的什錦歌生成服務，依照使用者指定的條件，例如結構、一定要用的音樂素材等等，產生悅耳的什錦歌。此外，我們亦開發了可供使用者選歌、設定參數、修改接點的圖形化程式介面。實驗證明了各個步驟的有效性，呈現了方法之間的比較，並可協助使用者進行適切地參數選擇。





Abstract

In this dissertation, systematic techniques have been developed for helping users to make new music by concatenating existing audio materials, i.e. concatenative audio music re-composition. The re-composed music can be used as the background music for personal films and slideshows or for non-stop dance suites. Based on the content analysis techniques, music theory, and psychoacoustics, various composition and selection schemes have studied in detail. We could locate appropriate connecting positions on the basis of similarity values, phrase boundaries or bar information. Besides, psychoacoustics-based tempo adjustment methods are used to smooth the tempo of concatenated music pieces. For cases of distinct tempo or volume, effective dual tempo adjustment and volume normalization schemes have been proposed and investigated, respectively. Two different schemes are proposed for selecting materials from music collections: The straightforward scheme filtered out unfitting clips by pair wise comparison and ordered the clips by similarity values at the found connecting points. The graph-assisted scheme, first, constructed a musical dice graph from pre-processed clips based on the results of music signal analyses. Then, with the graph, we can provide personalized medley creation service, which will generate various pleasing medleys conform to the specified conditions, such as the medley structure or must-use clips. We also provide an GUI for the users to choose music clips, specify parameters and adjust concatenation boundaries. Experiment results showed the effectiveness of individual components, comparisons among methods, and provide guidelines for users to choose parameters.

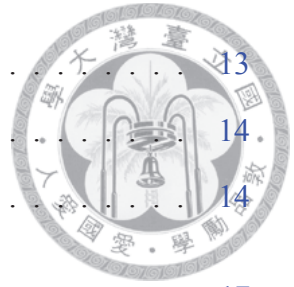


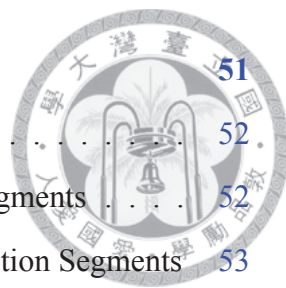


Contents

口試委員會審定書	iii
誌謝	v
Curriculum Vitae	vii
摘要	ix
Abstract	xi
1 Introduction	1
1.1 Background and Motivation	1
1.1.1 Media Re-composition	1
1.1.2 Types of Music Re-composition	3
1.1.3 Concatenative Audio Music Re-composition	5
1.2 Problem Statement	5
1.3 Summary of Contribution	7
1.3.1 Thorough Investigation of Material Concatenation Methods	7
1.3.2 Personalized Material Selection Scheme	9
1.4 Thesis Organization	10
2 Review of the Literature	11
2.1 Music Re-composition in Symbolic Domain	11
2.2 Self Re-composition – Audio Retargetting	12
2.3 Short Material Re-composition – Concatenative Synthesis	12

2.4	Overlaid Material Re-composition – Mashup Creation	13
2.5	Material Selection – Playlist Generation	14
2.6	Material Concatenation – Automatic DJ tools	14
3	Domain Knowledge and Audio Music Features	17
3.1	Temporal Related Factors	17
3.2	Pitch Related Factors	20
3.3	Dynamics Factors	21
3.4	Timbre Factors	23
4	Concatenation Methods	25
4.1	Transition Segments Locating Process	25
4.1.1	At the Most Similar Position	25
4.1.2	At the Phrase Boundary	27
4.1.3	With Bar Alignment	30
4.2	Tempo Adjustment Process	31
4.2.1	Transition Duration Determination	31
4.2.2	Dual Tempo Adjustment	34
4.3	Synthesis Process	35
4.3.1	Volume Normalization	35
4.3.2	Crossfading	36
5	Material Selection	37
5.1	Straightforward Scheme	37
5.1.1	Filtering	38
5.1.2	Ordering	40
5.2	Graph-assisted and Personalized Scheme	42
5.2.1	Musical Dice Graph Construction	43
5.2.2	Medley Generation	47
5.3	User Interface	48





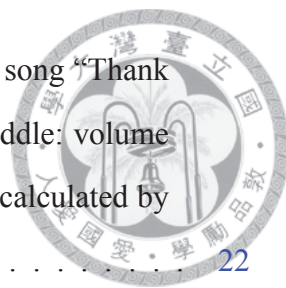
6 Experiments	51
6.1 Evaluations on Concatenation Methods	52
6.1.1 Overlap Duration of Similarity-based Transition Segments	52
6.1.2 Similarity Measurements in Similarity-based Transition Segments	53
6.1.3 Effectiveness of Phrase Detection	54
6.1.4 Comparison Between Similarity-based and Phrase-based Transition Segments Locating Methods	58
6.1.5 The Just Noticeable Difference of Tempo	59
6.1.6 Effectiveness Bar Alignment and Dual Tempo Adjustment	60
6.2 Evaluations on Selection Schemes	63
6.2.1 Effectiveness of Clustering Criteria	63
6.2.2 Effectiveness of Path Finding	65
6.3 Overall Performance	66
6.4 Discussion	68
6.4.1 The Influence of Accompanied with Visual Content	68
6.4.2 The Influence of User Familiarity with the Songs	69
6.4.3 Other Criteria that Might Contribute to Better Clip Selection	70
6.4.4 Comparison with Human Created Medley	70
7 Conclusions and Future Work	73
7.1 Conclusions	73
7.2 Future Work	74
Bibliography	75



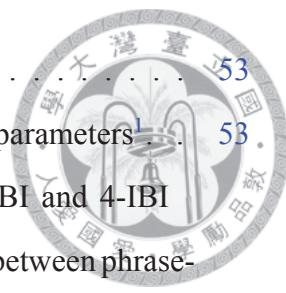


List of Figures

1.1	(a–c) Examples of media re-composition: (a) A student extended the portrait of Guan Hanqing (關漢卿) on a Chinese textbook, and painted a motorcycle for him. (b) Some netizens modified the poster of the famous Taiwanese romance drama “In Time with You (Chinese: 我可能不會愛你; literally: “I might not love you”)", and satirically changed the leading roles’ faces to those of the presidential election candidates of Taiwan in 2012. The title of the poster was also satirically changed to “I might not ‘vote for’ you”. (c) Screenshot of the results of tiling slideshow[1], an automatic photo slideshow creation system. (d) Example of the appropriation art: <i>Monroe in Warhol Style</i> . Andy Warhol (1967).	2
1.2	Functional blocks of a general music re-composition process by human. . .	6
1.3	General framework for automatic concatenative audio music re-composition. . .	7
1.4	Two proposed schemes for concatenative audio music re-composition. . .	8
1.5	Operational scenario of the proposed system. Upper part: the user specified medley structure and must-used clips. Bottom part: the completed medley produced by our system. The result of this example is available at http://www.cmlab.csie.ntu.edu.tw/\nobreakspace{}known/medley/results/scenario.wav	10
3.1	Example of beat and onset’s position	18
3.2	Illustration that indicates the 5 dimensions in the rhythm feature.	19
3.3	Schematic diagram for calculating the chroma feature. (Images are taken from [2])	21



3.4	Example result of the volume levels of an excerpt from the song “Thank you for the Music (by Abba)”. Top: the origin waveform; middle: volume levels calculated by Equation (3.2); bottom: volume levels calculated by Equation (3.3).	22
3.5	Waveform and spectrum of flute, piano, trumpet in middle C. (Images are taken from [3])	23
3.6	Steps for extracting MFCC.	24
4.1	Distance matrix of two clips chosen from Chinese pop songs: “Real man (《大丈夫》)” (clip <i>a</i>) and “Let’s move it” (clip <i>b</i>), respectively.	27
4.2	Proposed bar alignment method.	31
4.3	Diagram for changing tempi within a duration of K IBIs.	32
4.4	Schematic diagram of finding the transition duration.	33
4.5	Tempi changes in the transition duration of the clips of “Let’s move it” and “Real man.”	33
4.6	Proposed bar alignment method (with consideration of dual tempo adjustment).	34
5.1	System Framework for our first project: “Music Paste [4]”s	38
5.2	How to find the tempo dissimilarity.	39
5.3	An example of the ordering matrix for 4 clips.	41
5.4	Example of a musical dice graph.	42
5.5	System Framework for our second project: “Audio Musical Dice Game [5, 6].” Purple stars marks the function blocks that can take user preference into account.	44
5.6	Screenshot of our graphical user interface for medley creation, where users can specify the medley structure, must-use clips, and other parameters . . .	49
5.7	Screenshot of our graphical user interface for medley generation, where the user can adjust the phrase boundaries manually.	49
6.1	Comparisons among various overlap durations.	52



6.2	User preference comparisons of similarity measurements.	53
6.3	The results of singing voice detection with different HMM parameters	53
6.4	Results of user evaluation for clip concatenation with 1-IBI and 4-IBI crossfades, in which the relevant p-values of pairwise t-tests between phrase-based and similarity-based methods are displayed under the corresponding bars of each experiment.	59
6.5	Percentages of evaluators who can recognized the tempi difference of the samples	60
6.6	We use the similarity between “the latter clip” and “the phrase after the former clip in the original song” as the metric to measure the suitability of the consecutive clips for connection.	62
6.7	Mean scores of each one of the temporal adjustment methods for total data and for each similarity type of the test samples, in which the relevant p-values of paired Wilcoxon signed rank test on “BD vs. PT”, “BD vs. Echonest”, and “PT vs. Echonest” are displayed above the corresponding bars of each experiment, respectively.	63
6.8	Results of user evaluation on clip selection with the proposed clustering criteria, in which the relevant p-value of pairwise t-test of the proposed and the lower bound methods is displayed under the corresponding bars of each experiment.	64
6.9	Results of user evaluation on the proposed path-finding scheme based on the Viterbi algorithm, in which the relevant p-value of pairwise t-test is displayed under the corresponding bars of each experiment.	66
6.10	Results of user evaluation on the overall performance of “audio musical dice game” [6] when compared with “music paste” [4], in which the relevant p-value of pairwise t-test is displayed under the corresponding bars of each experiment.	67
6.11	Percentages of the how many evaluators scored lower, the same, and higher when they listen to the sample at the 2nd time, respectively	69





List of Tables

2.1	Related studies in playlist generation and automatic DJ tools.	16
5.1	Different substitution scores of the edit-algorithm according to the intervals between chord roots.[7]	45
6.1	The phrase detection results.	55





Chapter 1

Introduction

*Music has Charms to sooth a savage Breast,
To soften Rocks, or bend a knotted Oak.*

– William Congreve (The Mourning Bride, 1697)

1.1 Background and Motivation

The development of digital music gives people convenient ways to access their favourite music pieces. The prevalence of digital capture devices also make people start to create their own media, such as photo, video, audio, etc.. With the help of media editing tools, people can combine existing media, organize them to make new media, and then share the creations on social websites like Facebook, Youtube, etc.. There are more and more such kinds of creations spread over the internet. Besides entertaining the masses (c.f. [Figure 1.1\(a\)](#)), these creations also play the major role in satirizing society (c.f. [Figure 1.1\(b\)](#)) or commemorating personal experiences (c.f. [Figure 1.1\(c\)](#)), and become the sustenance of our daily life.

1.1.1 Media Re-composition

We define the process of using existing media as materials to make new media as “media re-composition.” It is related to the technique of “appropriation” in modern art, which means using existing elements and re-contextualising them with little or no transformation



(a)



(b)



(c)



(d)

Figure 1.1: (a–c) Examples of media re-composition: (a) A student extended the portrait of Guan Hanqing (關漢卿) on a Chinese textbook, and painted a motorcycle for him. (b) Some netizens modified the poster of the famous Taiwanese romance drama “In Time with You (Chinese: 我可能不會愛你; literally: “I might not love you”)", and satirically changed the leading roles’ faces to those of the presidential election candidates of Taiwan in 2012. The title of the poster was also satirically changed to “I might not ‘vote for’ you”. (c) Screenshot of the results of tiling slideshow[1], an automatic photo slideshow creation system. (d) Example of the appropriation art: *Monroe in Warhol Style*. Andy Warhol (1967).

in the creation of a new work. Figure 1.1(d) show an example of appropriation art by Andy Warhol (1967), which contains a photo of Marilyn Monroe with different colors. In this study, we will focus on the media re-composition for music, the so-called music re-composition.



1.1.2 Types of Music Re-composition

Music re-composition can be categorised from three aspects: the type of source music, the type of content usage, and the type of composition method.

Categorised by the type of source material

In general, there are two types of source materials for music re-composition: symbolic and audio domains. The first type refers to using symbolic representation of music as materials for re-composition, e.g. taking existing melodies, rhythms, or styles into account. One example is a sonata¹ re-composed by David Cope's EMI System [8], which mimics the melody and style of Beethoven's works. Another example is the Chinese pop song "Shen Qi Bai Ma (Riding on a white horse)" by Lala Hsu (徐佳瑩《身騎白馬》²). In that song, Hsu inserted a famous melody from Taiwanese opera. Music re-composition in symbolic domain causes no obvious audible artifacts in the auditory aspect. However, to be listened to, the re-composed music should be performed by performers or be synthesized.

The second type of source refers to using the audio representation as materials for re-composition, i.e. using the recordings of existing music pieces. An example is the Chinese pop song "Long Live Punk'n'Funk" by Jutoupi (豬頭皮《中華民國萬萬歲》³), in which the audio recording of Jutopi's rap has been overlaid with the audio of Lenny Kravitz's "Are You Gonna Go My Way"⁴. In contrast to symbolic music re-composition, there is no need to re-perform the used music materials. However, we need to infer the content in the audio materials and deal with possible discontinuity artifacts in the resultant music.

¹<ftp://arts.ucsc.edu/pub/cope/beet2.mp3>

²http://www.youtube.com/watch?v=VzXOT26_Da8

³<https://www.youtube.com/watch?v=wIrFFtgElnY>

⁴<https://www.youtube.com/watch?v=uAcAuuLNEHY>

Categorised by the type of content usage

According to the type of content usage, we can categorize the music re-composition into three types: use of single music piece, use of both existing music and new music piece, and use of multiple existing music pieces. For the first type, only single music piece is used, we name it self re-composition. In the audio domain, this type of usage is also called audio retargeting [9, 10]. The goal is often to lengthen or shorten the input music while preserving the characteristics of the original music. Common approaches would be to identify the near-identical parts in the music and then delete or repeat those parts to make the duration of music match the user's need. For the second type, composers will compose new music to make it match with the existing music. As a result, the re-composed music will be more natural and with less artifacts. For the third type, multiple existing music pieces are used as materials to re-compose music. The music materials are relatively fixed, so we need to find out proper method to compose these pieces euphoniously.



Categorised by the type of composition method

Based on the composition methods, there are three different types: overlaid, concatenated, and hybrid. For the first type, the music pieces are composed to play concurrently. For example, the mashup song composed by FAROFF⁵ is made by the vocal track of Beatles' "Let it be" overlaid with the instrumental track of Bob Marley's "No Cry". The second composition type is to concatenate the input music, that is, the music pieces are played successively. An example is the Taiwanese pop song "Red Line" by Jody Chiang (江蕙 《紅線》⁶), the Taiwanese folk song "Bāng Chhun-hong (《望春風》)" is inserted and concatenated with other part of "Red Line." The third type is to use both the aforementioned two composition methods. For example, some netizens concatenated short words in President Obama's speech recordings with each other and overlaid them with an instrumental track of the song "Jingle Bells", resulting in an interest rap song⁷.

⁵ <http://www.youtube.com/watch?v=Ac1X16K5X1U>

⁶ <http://www.youtube.com/watch?v=uf7Ame9RtM>

⁷ http://www.youtube.com/watch?v=HW_hvI1bYAw

1.1.3 Concatenative Audio Music Re-composition



In this dissertation, we focus on the music re-composition in audio domain (audio music re-composition), that is, composing of multiple existing music pieces on the basis of concatenating methods. We name this type of re-composition as “concatenative audio music re-composition”. The term “concatenative”, “concatenate”, or “concatenation” could also be described with other terms like, “justapose”, “segue”, “strung together”, “combine”, “connect”, “link to form a single piece”, “arranged so that the end of one merges into the start of the next”. And the resultant music piece of concatenative audio music re-composition can also be called a medley or a megamix.

A musical medley is a piece of music composed from parts of existing music pieces [11]. As stated in [12], “the term was first used by 16th-century composers,..., for a piece that strings together several favourite tunes.” In light operas and musicals, the overtures that are composed of the most prominent melodies in the associated work are also called medley overtures [13]. In the digital audio era, the term “medley” indicates the remix composed of parts of tracks of a particular artist or popular songs of a specific genre⁸. In the rest of this dissertation, we will mainly use the term “concatenate” to describe the relationship between music materials, and use the term “medley” to describe the result of the concatenated audio music.

1.2 Problem Statement

In the past, medleys were usually edited by professional audio engineers and distributed by music production companies, e.g. The Beatles Movie Medley⁹. Currently, more and more music hobbyists create their own medleys from their favorite songs with the help of newly-developed audio technologies and publish the results on websites like Youtube. The resultant medleys can be used as background music for personal films and slideshows or for non-stop dance suites. If each individual track only appears as a partial sample (usually less than 30 seconds), users are allowed to include their favorite songs while avoiding

⁸[http://en.wikipedia.org/wiki/Medley_\(music\)](http://en.wikipedia.org/wiki/Medley_(music))

⁹<https://www.youtube.com/watch?v=pKOiculk5tA>

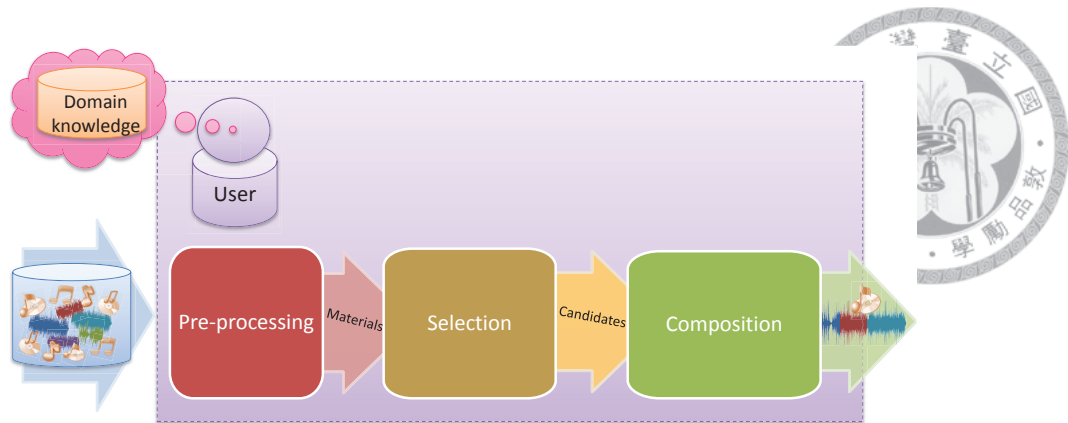


Figure 1.2: Functional blocks of a general music re-composition process by human.

copyright infringement issues.

Existing editing tools like Goldwave¹⁰ and Audition¹¹ enable users to cut and connect audio clips at manually-specified positions. However, these tools still require users to have the knowledge and skills necessary to (i) select suitable materials to put together and (ii) euphoniously connect the materials. As shown in Figure 1.2, the process of manual music re-composition can be divide into three steps, pre-processing, selection and composition. During the process, people may do these steps iteratively. Professionals may reduce the number of iterations with the help of their domain knowledge or experiences. But for general people, the process can be just trial-and-error. Given the vast amount of digital music currently available¹², finding suitable songs which can be sequenced to produce a cohesive and pleasant medley is a time- and labor-intensive process. In addition, once the user decides which music materials should be adjoined, they still need to listen to each of them to determine the positions for cutting the audio files into clips and connecting them together. Furthermore, users may need to manually adjust the tempi and volume levels of the clips to smoothly connect them.

As a result, automatic schemes would help to reduce human efforts in facing of time- and labor-intensive tasks. A general automatic scheme for concatenative audio music re-composition can be illustrated in Figure 1.3. First, the system analyses the input music based on certain domain knowledge by extracting features and detecting basic music components, such as, beats, chord, etc.. Then, according to the analysed results, the system

¹⁰<http://www.goldwave.com/>

¹¹<http://www.adobe.com/products/audition.html>

¹²There are 26 million songs on the iTunes store[14].

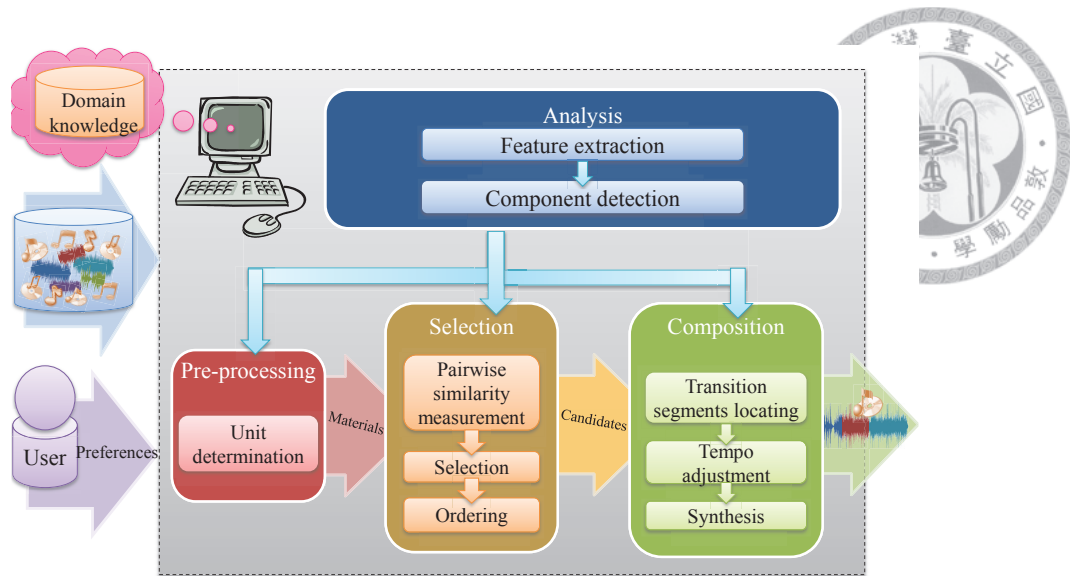


Figure 1.3: General framework for automatic concatenative audio music re-composition.

can determine the unit for re-composition (pre-processing), select and order the music materials based on the result of pairwise similarity comparison, and finally compose the materials. Besides, user preferences should also be taken into account to enhance the quality of re-composed music. For professional users, one could maximize the flexibility of the system to adapt to users' targets. For general users, one could minimize their efforts at creating new media.

1.3 Summary of Contribution

This dissertation is devoted to develop systematic techniques for concatenative audio music re-composition by exploiting content-based music signal analysis. Most of the results are outcomes of several projects I have explored during my PhD study. The main contributions of this thesis in solving the faced problems are twofold, as summarized in the following two sections.

1.3.1 Thorough Investigation of Material Concatenation Methods

During the projects, numerous concatenation methods have been proposed and investigated. The adventure of music concatenation started from the idea illustrated in [Figure 1.4\(a\)](#). We knew that a listener will anticipate the follow-up music based the on current

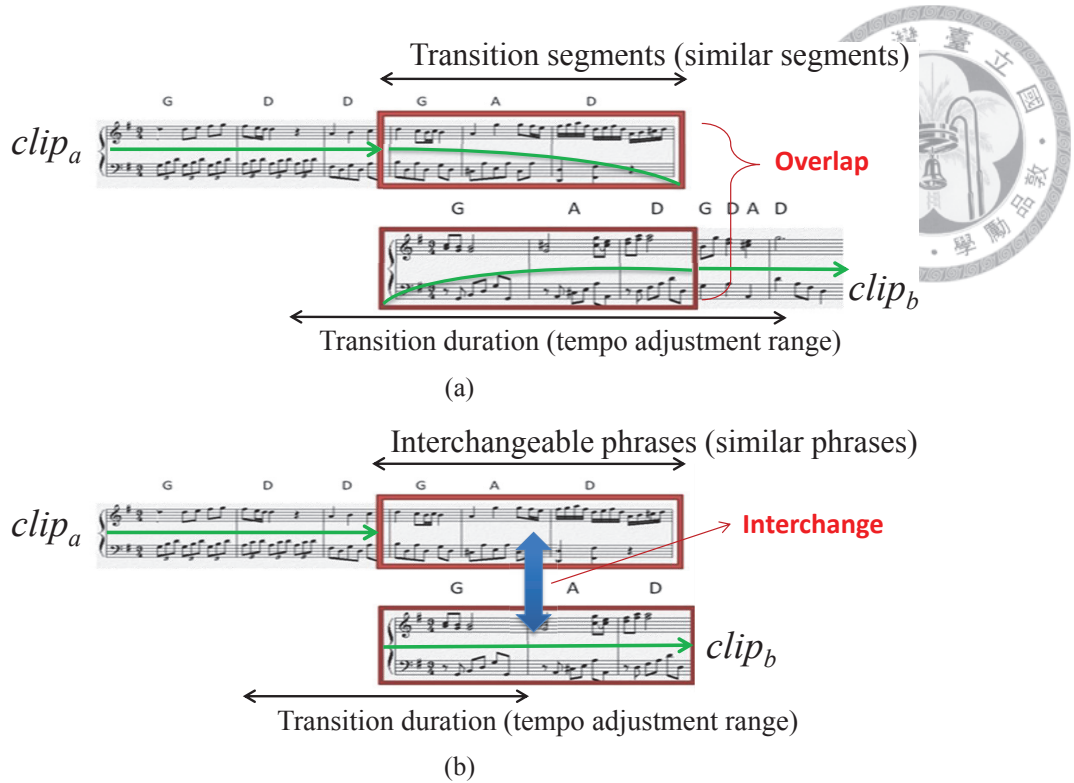


Figure 1.4: Two proposed schemes for concatenative audio music re-composition.

and the past music he or she has listened [15]. As a result, in our first project: Music Paste [4], we proposed to connect the music materials at the position where the former clips is most similar to the latter ones — the former clip and the latter clip will be overlapped at their most similar part. Then, that part becomes the transition segment. The resultant music from the beginning through the connecting-position to the end will all conform to the listener’ anticipation. To smooth the change in tempo between the consecutive clips, an adequate duration (transition duration) is a must for gradually adjusting the tempo from one clip to another based on the concept of just noticeable difference (JND) [16].

In our second project [5, 6], we focused more on developing a personalized framework for medley creation and improving the concatenation by phrase detection. That is, the music materials will be first pre-processed and turned into music phrases. And then, we will concatenate the music materials at their phrase boundaries and apply the same tempo adjustment process, which was developed in our first project. A volume normalization method was also proposed to further improve the concatenation quality.

Finally, in our third project [17], we further improved the concatenation method developed in the second project. We proposed to also cut the music materials at their phrase

boundaries but align them with the bar information, so as to improve the beat counting experience of listeners. Besides, we improved the tempo adjustment by considering the double/half or quadruple/quarter of the original tempi to dealing with tempo-distinct cases.

To sum up, we divide our concatenation methods into three steps: transition segments locating, tempo adjustment, and synthesis. At transition segments locating step, we provide three options for users: at the most similar position, at phrase boundaries, and with bar alignment. Then, psychoacoustics-based tempo adjustment methods are proposed to smooth the tempo of concatenated music. For cases of distinct tempo or volume, corresponding techniques for doing dual tempo adjustment and volume normalization schemes have also been studied, respectively.

1.3.2 Personalized Material Selection Scheme

In our first project [4], we just simply select and order the music materials by the similarity values of the transition segments between connecting material pairs, while in our second project [5, 6], we took another view for material selection. As shown in Figure 1.4(b), we assume that music materials are interchangeable if they are similar enough. As a result, clips will be chosen to be put after the former clip if they are similar enough to the phrase just after the former clip in the original music. We knew that the aesthetic appeal of music is highly subjective and is subject to personal tastes. Based on the aforementioned concept in Figure 1.4(b), we proposed a personalized material selection scheme in reflection to the rarely considered user preferences in previous related studies.

As shown in Figure 1.5, users can specify the structure of the target medley, and optionally select a few materials at certain positions in the medley, i.e. the darker parts in the figure. The system then completes the medley with materials selected from the music collection provided by the user. We built a flexible scheme to create medleys based on user preference, thus even users with no understanding of music theory can compose medley songs from their favourites tracks.

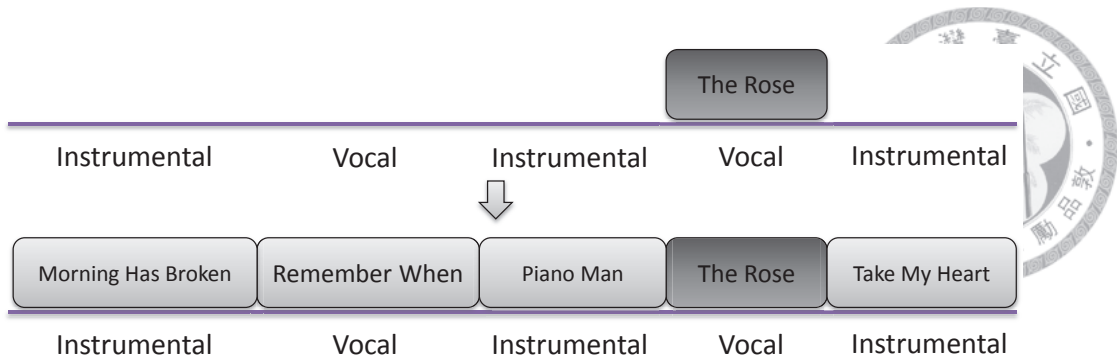


Figure 1.5: Operational scenario of the proposed system. Upper part: the user specified medley structure and must-used clips. Bottom part: the completed medley produced by our system. The result of this example is available at <http://www.cmlab.csie.ntu.edu.tw/~known/medley/results/scenario.wav>

1.4 Thesis Organization

The rest of this dissertation is organized as follows. In the next chapter, we will review literature related to concatenative audio re-composition. In [Chapter 3](#), we will introduce the domain knowledge and audio features have been used in the dissertation. Then, the concatenation techniques used in our three projects will be re-organized and presented in [Chapter 4](#). In [Chapter 5](#), the selection schemes and the overall system structure of our first two projects will be presented. Afterwards, the effectiveness of the investigated methods will be discussed in [Chapter 6](#). Finally, the conclusion and future study directions will be presented in [Chapter 7](#)



Chapter 2

Review of the Literature

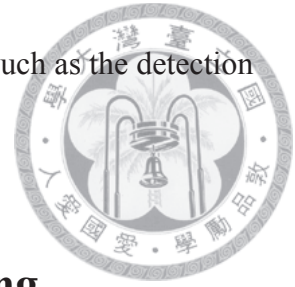
If I have seen further it is by standing on the shoulders of giants.

– Isaac Newton, 1676

2.1 Music Re-composition in Symbolic Domain

Re-composing music clips in the symbolic domain has been studied since the 1980s [18]. Cope [8] conducted various experiments and developed a music-generating system based on the concept of the dice game [15]. In the system, music clips from master composers are analyzed and recombined to generate a new master style music pieces. Cope also argued that the recombination of existing excerpts is a basic technique frequently used by composers. More recently, Shan and Chiu [19] used machine learning techniques to analyze existing music samples and proposed a top-down algorithmic composition system that generates music pieces similar to the given samples. Combining music clips in the symbolic domain causes no obvious audible artifacts in the auditory aspect. The music is usually synthesized and performed by the same artist so that the key, tempo, and the instruments used in the song clips of the generated music can be easily altered. Nonetheless, the approaches in the symbolic domain cannot be easily applied to the audio domain because current music transcription and separation techniques are still not accurate enough to extract all the music notes from polyphonic audio clips. Instead, in this dissertation, we

use more applicable (but coarser¹) audio music analysis paradigms, such as the detection of beats, chords, and phrases.



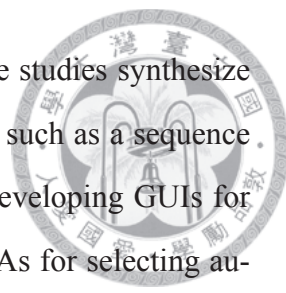
2.2 Self Re-composition – Audio Retargetting

Audio retargetting is a self re-composition (c.f. [Section 1.1.2](#)) approach that changes the duration of a given audio track to arbitrary lengths by inserting or deleting a portion of the same track. A common approach for audio retargetting includes two subtasks: (i) segment the song into short clips and group together near-identical clips, and (ii) find an appropriate method to concatenate these clips under the constraints given by users, usually the duration of the output audio. Wenger and Magnor [9] segmented a given song by calculating its self-similarity matrix of music signals. Liu et al. [10] employed time-stamped lyric information as well as the self-similarity matrix of chroma features to find appropriate cut points. Grouping segments with the self-similarity matrix only allows for the identification of near-identical segments, and thus can hardly be applied to our case, which involves multiple and potentially dissimilar songs. Besides, using the self-similarity matrix to segment songs does not necessarily guarantee accurate segmentation, and the corresponding computational complexity grows dramatically with the number of involved clips. To avoid selecting near-identical clips and to better capture the musical content of the clips, this study takes higher level features including chord sequence similarity into account.

2.3 Short Material Re-composition – Concatenative Synthesis

Concatenative synthesis [20] focuses on synthesizing speech, music or environmental sounds based on pre-collected “short” audio snippets. Schwarz et al. [21] provided a com-

¹Compared to “all the musical notes”, beat detection provides only time indexes for each beat, while chord detection reports only the chords at each time index, and phrase detection provides the phrase boundaries in the songs.



prehensive survey of the techniques of concatenative synthesis. Some studies synthesize music notes from a corpus according to a user-provided description, such as a sequence of pitches or midi files [22]. Some recent works have focused on developing GUIs for interactive performance artists to control synthesis results [23, 24]. As for selecting audio snippets, some studies [24, 25] adopted approaches similar to the proposed one in our second project [5] – they pre-clustered the snippets and chose them according to certain statistics of consecutive snippets in the clusters. However, the unit for concatenative synthesis can be as short as a musical note/onset (usually less than a second). Consequently, the music pieces produced by concatenative synthesis techniques will not keep the phrasing of the original songs, i.e., the resultant output is a totally new song rather than a combination of existing song excerpts, thus losing the spirit of a medley song. Here, we need to handle additional issues to compose songs with musically meaningful units, such as chord/note sequence similarities, tempo differences between clips, and the smoothness of the music as it transitions from one clip to the next.

2.4 Overlaid Material Re-composition – Mashup Creation

Mashup is another paradigm for music re-composition, where the clips are overlaid with each other, usually with the vocal track of one song and the instrumental track of another. That is, in a mashup, the clips from different songs are played concurrently with the original songs while in a medley, the clips are played successively. Automatic mashup creation is still new and few studies focused on it. Griffin et al. [26] used a phase vocoder to adjust the tempo of each one of the user-specified clips, and combine them after synchronizing their beats. The AutoMashupper was recently proposed by Davies et. al. [27] to automatically create mashup music from multiple song tracks. In their system, users first pick a song track as the basis song. Then, the system will segment the picked track into short phrases. For each phrase, clips with the highest mashability – by chromagram similarity – will be overlaid on the phrase to create the final mashup. A limitation of AutoMashupper is that the resultant mashups should follow the structure of the basis song while in the system of our second project [5], users can specified their own structures of the resultant

medley.

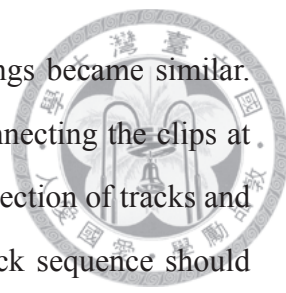


2.5 Material Selection – Playlist Generation

The study of playlist generation emphasizes on selecting suitable songs for playback successively. A general approach to these tasks is to select similar songs for being included in the playlists based on some specific criteria. Users may specify a seed song, then the song best matched with the specific criteria will be chosen as the next song by the system, and so on. Sometimes, slightly random factors may be used, to increase the novelty/interest of playlists. Commonly used criteria usually fall into one of the following types: (i) meta-data based (e.g. same artist, album, or genre), (ii) content-based (e.g. audio feature similarity [28, 29], key [30], tempo/rhythm [31]), and (iii) collaborative filtering based (e.g. occurrence of songs in user’s friends’ playlists [32]). The present case deals with song excerpts (~ 10 sec on average) rather than whole songs and, as such, the properties used to select adjacent clips are stricter than those used to select adjacent songs. In other words, the song selection approaches used for playlist generation cannot be directly applied to the selection of song excerpts for medley generation. In playlist generation, the next song only begins after the previous ends, and each song in the playlist is complete and played without interruption. In contrast, a medley consists of partial song clips sequenced to form a new song, with the next song excerpt starting before the previous song excerpt has finished. Therefore, while playlist generation merely considers the global similarities between songs, this is insufficient for medley generation. Local audio similarities between consecutive song excerpts should also be considered to meet the listeners’ expectations for seamless musical flow in the resultant medley.

2.6 Material Concatenation – Automatic DJ tools

The studies on Automatics DJ tools [33, 34, 35] often emphasize more on the issues of music concatenation. The songs or clips mentioned in these studies are specified by users, and the tools only have to deal with the concatenation issues. Basu [34] aligned two



clips through scaling and shifting so that the energy of the two songs became similar. Jehan [33] realized a DJ system by extracting auditory features, connecting the clips at rhythm-similar segments and aligning the beats of clips. Given a collection of tracks and a tempo trajectory of the tracks, Cliff [35] determined how the track sequence should be played in accordance with their tempi. In [36], a detailed discussion was presented on how two given clips should be concatenated without producing listener discomfort. In the clip selection phase, audio clips with similar Mel-frequency cepstral coefficients (MFCCs) were selected. Their tempi were then adjusted by computing their optimal tempo adjustment coefficients (OTAC), and the two clips were then aligned and concatenated by matching the strong beats of the clips. However, in the aforementioned work, “rhythm similarity” and “beat alignment” are emphasized most because the concatenated songs are often used for dancing. Given that medleys are also a kind of music composition, in this work we focus more on the chordal euphoniousness of the consecutive clips.

We summarized the related studies of playlist generation and automatic DJ tools in [Table 2.1](#).



Studies	Goal	Unit	Pre-processing	Selection criteria	Composition	User involved
Logan 2002 [28]	Playlist	Song		Timbre (MFCC)		Seed song
Baccigalupo et al. 2006 [32]	Playlist	Song		Co-occurrence of songs in past playlists		Seed song
Flexer et al. 2008 [29]	Playlist	Song		Timbre (MFCC)		start and end songs
Lin et al. 2010 [31]	Playlist for jogging	Song		Rhythm	Crossfade, Beat Sync	Seed song, tempo profile
Cliff 2000 [35]	DJ tool	Song		Tempo	Crossfade, Beat sync	Song set, tempo trajectory
Basu 2004 [34]	DJ tool	Song			Crossfade, Energy matching	Given two songs
Jehan 2005 [33]	DJ tool	Song			Crossfade, Beat sync	Song set
Ishizaki et al. 2009 [36]	DJ tool	Song		Timbre (MFCC)	Crossfade, Beat sync	Seed song
Chararandini et al. 2011 [30]	Playlist, DJ tool	Clip	Segmentation by novelty curve	Tempo, Key, Timbre, Mood	Crossfade, Beat sync	Seed song, choose next song from a list

Table 2.1 : Related studies in playlist generation and automatic DJ tools.



Chapter 3

Domain Knowledge and Audio Music

Features

If music be the food of love, play on.

– William Shakespeare (Twelfth Night, 1601-02)

In this chapter, we will briefly introduce the domain knowledge and audio music features we used. The used features reflected the basic properties of an audio music: temporal related factors, pitch related factors, dynamics factors, and timbre factors. We will detail these factors in the following sections.

3.1 Temporal Related Factors

Temporal related factors are music properties that related to music events along the time axis, such as beat, onset, tempo, measure, rhythm, etc.. The used temporal related factors are listed below.

- Beat

Beat is the basic unit of time in music [12]. It is often indicated by the conductor's moving hand or baton. People can also interpret the beats by listening to music, though different people may feel different beat timings. Fortunately, most of the time, we can find consensus in beats times. Beside, the interpreted beats are often

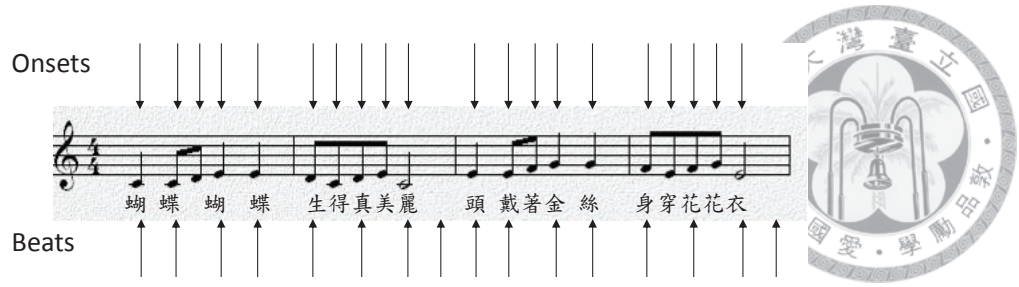


Figure 3.1: Example of beat and onset's position

integer multiples (often double or half) of each other. As a result, there were numerous studies dedicated to detect beats from audio recordings. In this dissertation, the beat information is extracted using BeatRoot [37], a state-of-the-art beat-detection tool that won the Music Information Retrieval Evaluation Exchange (MIREX) contest¹ in 2006 with a P-score of 0.575. Beats are also used as the unit with respect to other audio features when analyzing music signals.

- Onset

The term “onset” in the field of music information retrieval often means the note onset, i.e. the starting instant of a music note [38]. Figure 3.1 provides an example to show the locations of onsets and beats on time axis. The detected onset locations often used to further interpret beats or represent the rhythm features of the music. Bello et al. [38] provided a good overview of onset detection techniques. In this dissertation, we use the onsets extracted along with beats in BeatRoot system[37].

- Tempo

In musical theory, tempo is defined as the speed of a given piece [11], usually measured by the number of beats per minute (BPM). The tempo value (in BPM) at the i^{th} inter-beat-interval (IBI, in second), $T(i)$ can be calculated as,

$$T(i) = \frac{60}{IBI_i}. \quad (3.1)$$

- Measure (Bar)

According to [12], bar is a kind of musical notation, which is a line drawn vertically

¹http://www.music-ir.org/mirex/wiki/MIREX_HOME

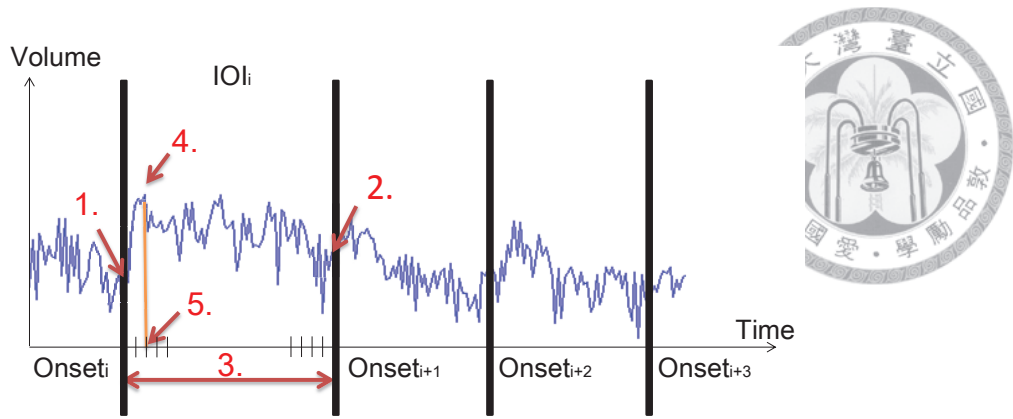


Figure 3.2: Illustration that indicates the 5 dimensions in the rhythm feature.

through a staff or staves, indicating the metrical unit (e.g., two, three, or four beats) of the music piece is divided into. “Bar” also indicates the metrical unit itself, and the line notation is called “bar-line”. In American usage, “bar” is the line itself, and the metrical unit is named “measure”. In some studies, the authors use “downbeat tracking” to describe the action of extracting bar information from audio signals because a downbeat is the first strong beat of a measure [12]. In this dissertation, we use the EchoNest API² to extract bar information from music signals.

- Rhythm

Rhythm is the pattern of movement in time [13]. The term covers “everything related to the time aspect of music,..., i.e. it includes the effects of beats, accents, measures, grouping of notes into beats, grouping of beats into measures, grouping of measures into phrases, etc.” [39]. As a result, it is not feasible to exactly describe the rhythmic property of a music piece. Many past studies have proposed ways to extract audio features that could capture the rhythmic property [40, 41]. Here we use the rhythmic feature proposed by Cicconet [42] because this method can be combined with other beat-sync features easier. To extract the rhythmic feature by [42], we first detect onset positions of an audio. Then, as illustrated in Figure 3.2, for each inter-onset-interval (IOI), we extract 5 values to represent current IOI: volume levels at the beginning and the ending of current IOI, the duration of current IOI, maximum volume in the current IOI, and the position of the maximum volume. Af-

²<http://echonest.github.io/remix/apidocs/echonest.remix.audio.AudioAnalysis-class.html>

terwards, we average the 5-dimension values of each IOI in each inter-beat-interval to have a beat-sync rhythmic feature.



3.2 Pitch Related Factors

Pitch is “the location of a sound in the tonal scale, depending on the speed of vibrations from the source of the sound, fast ones producing a high pitch and slow ones a low” [39]. A melody – pitched sounds arranged in musical time – and a chord – simultaneous sounding of two or more notes – are all examples of pitch related factors. In this dissertation, two pitch related factors are used, chroma feature and chords.

- Chroma

Chroma vector is a 12-dimensional feature, representing the energy of 12 pitch classes. Some researchers also called it pitch class profile. A common approach to calculate chroma feature is to first calculate the frequency response according to the frequency of each musical note for each audio frame. One may map the energy of each band in FFT to that of musical notes [2], or use the Constant Q transform [43] to directly compute the frequency response for each semitone. Then, as illustrated in Figure 3.3, we add up the frequency response of each pitch class over all the frequency ranges. For example, the frequency response values of the every E notes are summed up as the 5th element of the chroma vector. Finally, the 12-dimension vector represents the energy of 12 pitch classes for the current audio frame. This feature is often used in the studies of chord detection, music structure analysis, and cover song detection.

- Chord

A chord is defined as “the simultaneous sounding of two or more notes” [44]. The most frequent-used chords in Western music are the triads, which comprise of a root note with two superposed 3rds [40]. The most common triads are the major and minor triads. The terms *major* and *minor* are referred to as chordal quality³. Chords

³[http://en.wikipedia.org/wiki/Chord_\(music\)](http://en.wikipedia.org/wiki/Chord_(music))

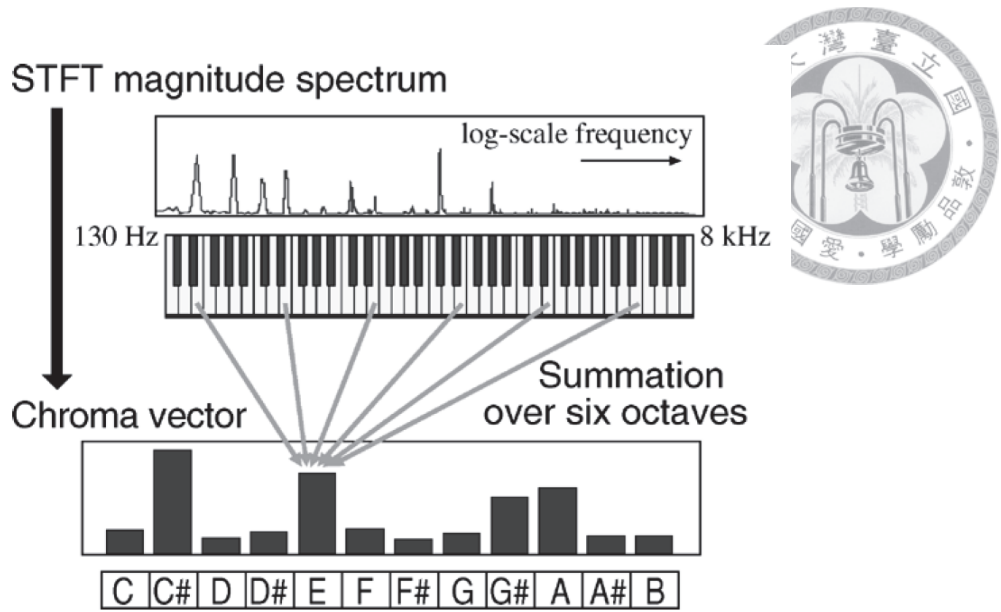


Figure 3.3: Schematic diagram for calculating the chroma feature. (Images are taken from [2])

can also be classed by their root notes, for example, a C major chord means the chord is made of a triad with major quality and its root is a C note. In this dissertation, we detect chords in songs with the Harmony Progression Analyzer (HPA) [45], a state-of-the-art chord estimation system. HPA ranked first in the MIREX Audio chord description contest in 2012 and achieved an superior accuracy of 75% to 85% for most musical genres. The number of possible chords estimated is limited to 25, including 12 major chords, 12 minor chords, and no-chord for silence. Beat-synchronized chord sequences are then extracted by aligning the chord estimation results with the detected beats.

3.3 Dynamics Factors

In music, dynamics normally indicates the relative intensity (loudness, volume) and degree of accentuation of sounds or notes [12]. The correspondence of dynamics of a music in its signal is the amplitude. To calculate the volume of the music signal, we first separate the signal into short frames of length n (often about 5 to 10 ms). For each frame, the volume

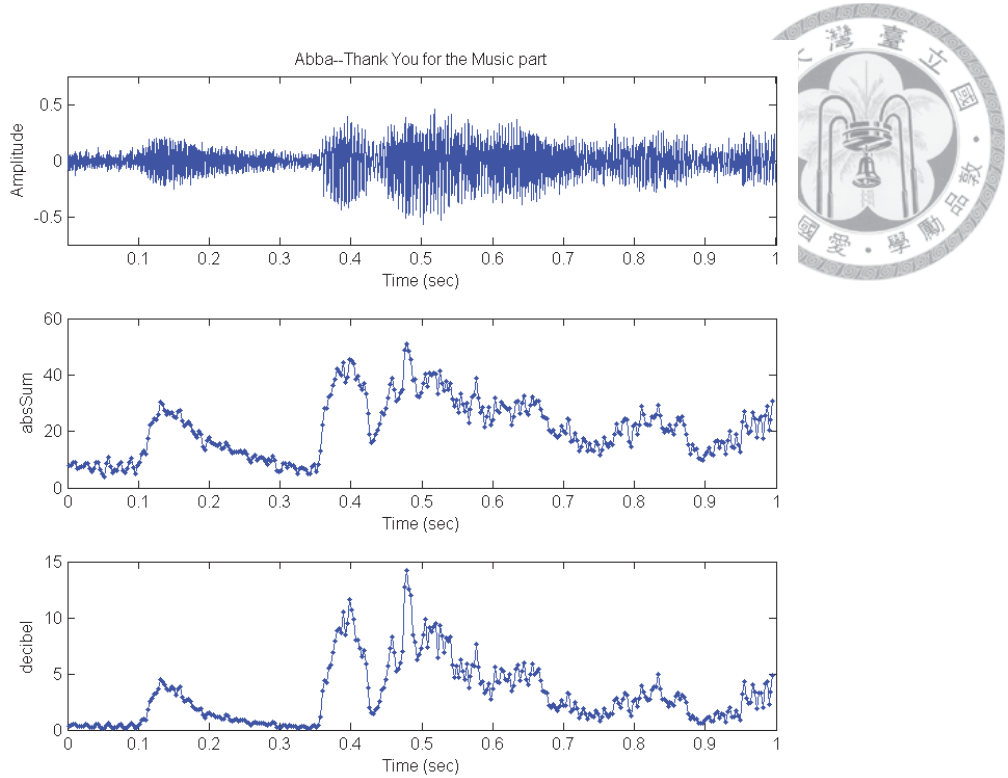


Figure 3.4: Example result of the volume levels of an excerpt from the song “Thank you for the Music (by Abba)”. Top: the origin waveform; middle: volume levels calculated by Equation (3.2); bottom: volume levels calculated by Equation (3.3).

is calculated as the sum of the amplitude of each samples in that frame, that is,

$$volume = \sum_{i=1}^n |s_i|, \quad (3.2)$$

where s_i is the i^{th} sample in the current frame. Another approach turns the amplitude into logarithmic scale (in db), which is closer to human auditory perception, as follows.

$$volume = 10 * \log_{10} \sum_{i=1}^n s_i^2 \quad (3.3)$$

Figure 3.4 illustrates an example of the calculated volume levels by using asp toolbox⁴.

⁴[http://mirlab.org/jang/books/audioSignalProcessing/basicFeatureVolume.asp?title=5-2%20Volume%20\(%AD%B5%B6q\)](http://mirlab.org/jang/books/audioSignalProcessing/basicFeatureVolume.asp?title=5-2%20Volume%20(%AD%B5%B6q))

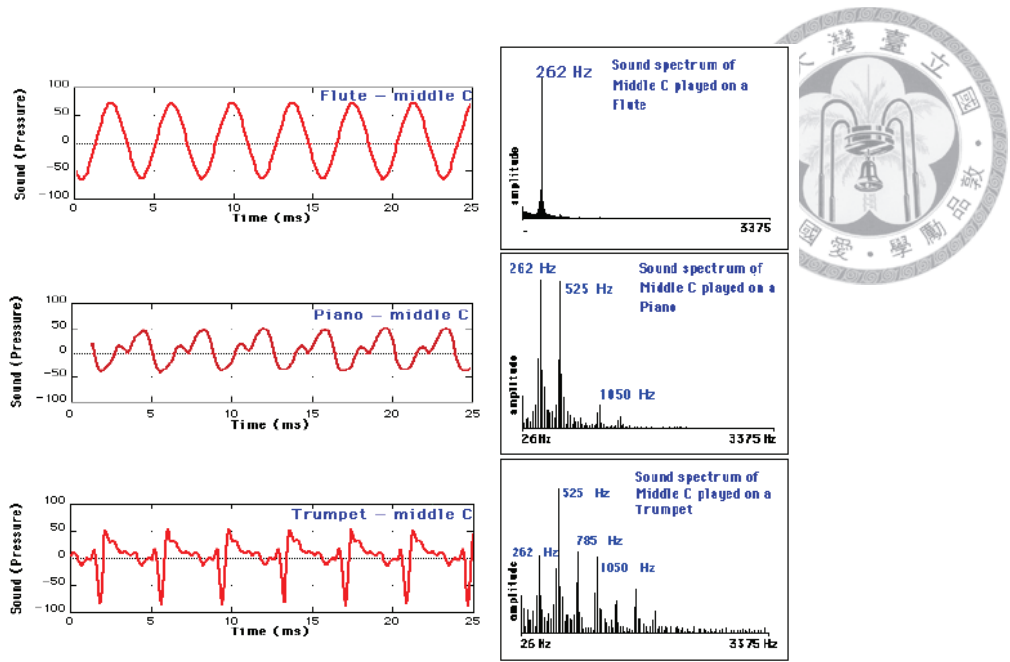


Figure 3.5: Waveform and spectrum of flute, piano, trumpet in middle C. (Images are taken from [3])

3.4 Timbre Factors

Timbre, also known as tone color, tone quality, texture, is the quality of a musical note (or sound or tone) that distinguishes the types of sound production⁵. For example, a piano and a guitar produce the same note at the same loudness are said to have different timbres. Timbre is a more complex property than other music factors like pitch and loudness [44]. Pitch and loudness can be represented in one-dimensional scale (pitch: high–low, loudness: loud–soft) [44]. However, timbre may be affected by multiple factors: the shape of frequency spectrum, the patterns in starting transients and the time envelope of the sound, etc. [44]. As far as we know, the shape of frequency spectrum is of great importance in determining the timbre. As shown in Figure 3.5, three different instruments are played with the same note–middle C. The fundamental frequencies are the same, but the energy of the corresponding harmonics varies, which makes the 3 instruments sound different. To model the timbre property, several features were proposed, e.g., spectral centroid⁶, spectral flatness⁷, etc..

In this dissertation, we use the mel-frequency cepstral coefficient (MFCC) [46], which

⁵<http://en.wikipedia.org/wiki/Timbre>

⁶http://en.wikipedia.org/wiki/Spectral_centroid

⁷http://en.wikipedia.org/wiki/Spectral_flatness

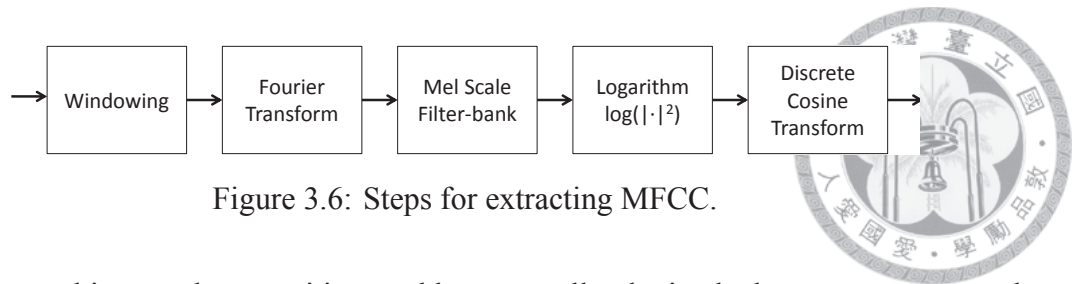


Figure 3.6: Steps for extracting MFCC.

is widely used in speech recognition, and has generally obtained a better accuracy at relatively low computational complexity. MFCC, in some sense, measures the frequency of mel-frequency, which is a “perceptual scale of pitches judged by listeners to be equal in distance from one another⁸”. As a result, in the studies of music signal analysis, MFCC is often used to represent the timbre. The general steps for extracting MFCCs are illustrated in Figure 3.6.

⁸http://en.wikipedia.org/wiki/Mel_scale



Chapter 4

Concatenation Methods

We cannot direct the wind, but we can adjust the sails.

– anonymous

In this chapter, we will describe the concatenation methods used in our 3 projects [4, 5, 6, 17]. To explain the methods clearer, we will disassemble and re-organize the components used in the projects. We will focus on the concatenation methods of given materials (clips), specifically for two given clips. Music concatenation can be divided into 3 steps: transition segments locating, tempo adjustment, and synthesis processes.

4.1 Transition Segments Locating Process

To concatenate clips, we first have to find out where to connect the clips – locating the transition segments. As we have shown in [Figure 1.4\(a\)](#), the transition segments are the parts in the clips that will be overlapped in concatenation. There are three schemes to locate the transition segments, (i) at the most similar position, (ii) at the phrase boundary, and (iii) with bar alignment.

4.1.1 At the Most Similar Position

The first proposed scheme [4] is to connect the clips at the most similar positions of the clips. That is, the transition segment in clip a is the most similar part to the one in clip b .

To locate this kind of segments between clips a and b , similarity/distance based method [47] is applied. We extract chroma (c.f. Section 3.2) and rhythm features (c.f. Section 3.1) per inter-beat-interval (IBI) and then calculate their Euclidean distances. The smaller the values are, the more similar the segments are. Let $D_C(a, b)[i, j]$ and $D_R(a, b)[i, j]$ represent the chroma and rhythm distance values between clip a 's i^{th} IBI and clip b 's j^{th} IBI, respectively. That is,

$$D_C(a, b)[i, j] = \|\vec{C}_{ai} - \vec{C}_{bj}\|_2, \quad (4.1)$$

$$D_R(a, b)[i, j] = \|\vec{R}_{ai} - \vec{R}_{bj}\|_2, \quad (4.2)$$

where \vec{C}_{ai} and \vec{C}_{bj} denote clip a 's i^{th} and clip b 's j^{th} chroma vectors, respectively. And similarly, \vec{R}_{ai} and \vec{R}_{bj} represent the rhythm feature vectors. The two matrices $D_C(a, b)[i, j]$ and $D_R(a, b)[i, j]$ are linearly combined into a new matrix $D_{CR}(a, b)$ (as shown in Equation (4.3)), which is the distance matrix we used for finding transition segments:

$$D_{CR}(a, b)[i, j] = \alpha D_C(a, b)[i, j] + (1 - \alpha) D_R(a, b)[i, j] \quad (4.3)$$

where $\alpha \in [0, 1]$. Figure 4.1(a) depicts the distance matrix ($D_{CR}(a, b)$) of 2 clips chosen from Chinese pop songs: “Real man (《大丈夫》)” (clip a) and “Let’s move it” (clip b), respectively. The darker the color is, the more similar the segments are.

Then, we want to find consecutive IBIs in both clips a and b which are similar. So we trace the values diagonally by applying an overlapping window with L_{min} to L_{max} IBIs long and compute the average distance value within each window. The window with the minimum average value is picked and the corresponding segments are the transition segments. Moreover, for the purpose of reducing the computational load and avoiding promptly switching clips, we consider only the last half of clip a and the first half of clip b . Figure 4.1(a) shows the most similar segment we found. Figure 4.1(b) shows the ignored areas marked with thick crosses. Mathematically, the process can be described as

$$[i^*, j^*, L^*] = \arg \min_{i, j, L} \frac{1}{L+1} \sum_{l=0}^L D_{CR}(a, b)[i+l, j+l] \quad (4.4)$$

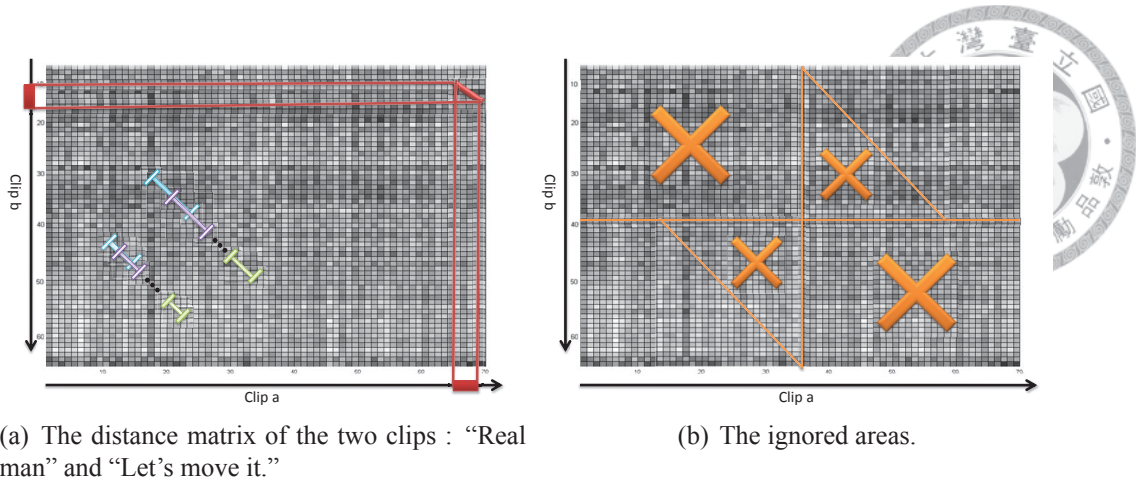


Figure 4.1: Distance matrix of two clips chosen from Chinese pop songs: “Real man (《大丈夫》)” (clip a) and “Let’s move it” (clip b), respectively.

where $L \in [L_{min}, L_{max}]$, $i \geq \frac{N}{2}$, $j \leq \frac{M}{2}$, N and M are the total beat number of clip a and clip b , respectively.

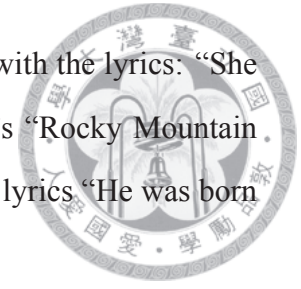
4.1.2 At the Phrase Boundary

The second scheme [5] is to connect the clips at phrase boundaries. According to Webber [48], phrasing is one of the most important factors to be considered when concatenating different tracks. Interruption occurring in the middle of a musical phrase is just as unpleasant and unexpected as the interruption of an oral sentence in a conversation. Therefore, the transition between clips should occur at the end of each musical phrase. As a result, we proposed to cut the music materials into phrases, and then use the phrase clips as a unit to re-compose music.

Musical Phrase

A musical phrase is usually subjective and not well-defined. We take the definition by [49], which described a musical phrase as “any group of measures (including a group of one, or possibly even a fraction of one) that has some degree of structural completeness.” According to [12], musical phrases may come in different lengths, but are most frequently of four bars. In [48], the author also mentioned many pop songs and dance records use musical phrases that are multiples of four bars long and is a half or a quarter of a verse or chorus section in popular songs [48]. For example, in Billy Joel’s “She’s Always a

Woman to Me”, the phrases are 4-bar long, the first vocal phrase is with the lyrics: “She can kill with a smile...with her casual lies”, while in John Denver’s “Rocky Mountain High”, most phrases are 8-bar long, the first vocal phrase is with the lyrics “He was born in the summer...place he’d never been before”.

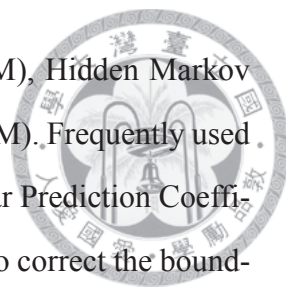


Phrase Detection via Singing Voice Detection

There are many studies investigating on audio music structure analysis and song segmentation of popular songs. However, the task is still challenging. As noted by Paulus et al. [50], common approaches in music structure analysis can be categorized into repetition-based, novelty-based or homogeneity-based methods. The first two methods often determine segments via the self-similarity/distance matrix [51] of audio signals or feature sequences. The last method often adopts HMM or HMM-like approach to cluster similar segments [52]. A more recent approach was proposed by Pauwels et al. [53], which combines the novelty-based method and their previously proposed harmony-based approach to jointly estimate keys, chords and structural boundaries in a probabilistic framework.

In our application, identifying correct segment boundary is more important than recognizing correct section labels. Besides, the musical phrases that we deal with here are shorter than the segmentations targeted by previous studies. So we turn to another view, via singing voice detection. The key idea is, at least, not to cut the songs in the middle of a vocal phrase since singing voice is usually the leading character in a popular song and medleys are composed of parts of popular songs. Therefore, we cut the songs into clips based on the boundaries of detected vocal segments, and use these as the basic unit for creating medleys.

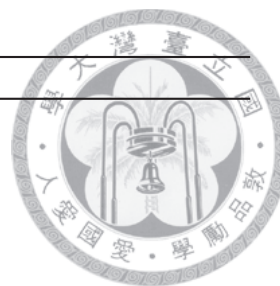
Singing voice detection aims to categorize which parts in a given track are vocal or instrumental segments. We define an instrumental segment as a segment consisting of purely instrumental sounds. A vocal segment, on the other hand, is defined as a singing voice with or without background music, as defined in [54]. Many studies have proposed solutions for singing voice detection, typically by extracting frame-based audio features and then training a two-class classifier to classify each audio frame as instrumental or vocal.



Commonly used classifiers include Gaussian Mixture Models (GMM), Hidden Markov Models (HMM) and their variants, and Support Vector Machines (SVM). Frequently used features include Mel Frequency Cepstral Coefficients (MFCC), Linear Prediction Coefficients (LPC), and Zero-Crossing Rate (ZCR) [55]. Techniques used to correct the boundaries of the segments have also been widely explored. Temporal smoothing techniques are often used to constrain the length of each vocal and instrumental segment afterwards to prevent over-segmentation [56, 55].

Here, we employ beat-synchronized MFCCs as audio features and HMM as the classifier. The use of beat-synchronized features is based on the fact that vocals are more likely to join the accompaniment at beat onsets [57], as noted in [58]. During the training phase, MFCCs and the beat information are first extracted (c.f. [Section 3.4](#) and [Section 3.1](#)). Then, MFCCs within an inter-beat interval (IBI) are regarded as the observed sequence of an HMM classifier. In the test phase, each IBI in a test song can then be classified as vocal or instrumental. Consecutive vocal/instrumental IBIs can then be connected as vocal/instrumental segments. To avoid over-segmentation, we then apply a moving median filter of 3-IBI long (i.e., about 1 seconds) to the singing voice detection result.

From our observations, there is often a short instrumental segment after a vocal segment, which is likely to appear when the singer transits from one phrase to another. The instrumental segment is too short and it should be regarded as a trailing part of the leading vocal segment. Similarly, a short vocal segment between two instrumental segments is likely to be short humming (such as interjection) or noise, which should be ignored by merging it with the neighboring instrumental segments. Based on these observations, we derive a “hybrid grouping” method to further refine the vocal/instrumental segments and convert them into musical phrases. The pseudo code of the grouping method is shown in [Algorithm 1](#). Note that the input candidate segments **seg** are alternate with vocal and instrumental because these segments are grouped from consecutive vocal/instrumental IBIs.



ALGORITHM 1: The pseudo code for hybrid grouping method

input/output: An array of candidate segments *seg*

```
for  $i \leftarrow 1$  to numSeg do
  if  $seg[i].segLen < G$  then
    if  $seg[i - 1].label$  is VOCAL then
      | combine  $seg[i - 1]$  and  $seg[i]$  into one segment;
    else //  $seg[i - 1].label$  is INSTRUMENTAL
      | combine  $seg[i - 1]$ ,  $seg[i]$ , and  $seg[i + 1]$  into one segment;
      | skip  $seg[i + 1]$ ;
    end
  end
end
```

Concatenation

When concatenation, for the alignment of phrase boundaries of the consecutive clips, we extend the clips by x IBIs if a $2x$ -IBIs crossfade is specified by the user. These $2x$ -IBIs long segments around the phrase boundaries are regarded as the transition segments (to be overlapped). With this scheme, the specified crossfade duration should not be too long. In our experience, crossfade duration ≤ 4 IBIs is preferred.

4.1.3 With Bar Alignment

As we mentioned in [Section 4.1.2](#), the transition between clips should occur at the end points of musical phrases. However, there are many songs contain pick up notes¹. If we directly connect the songs at phrase boundaries, even with the beats matched [5], the connected clips will be still temporally unsmooth (sounds like losing tempo). So in our third scheme [17], we suggest to cut the songs at phrase boundaries but connect the songs with bar alignment. Then, the transition segments will be determined according to the results of bar alignment, so that users cannot specify the crossfade duration with the aid

¹One or more notes preceding the first metrically strong beat of a phrase. Also called anacrusis or upbeat [13].

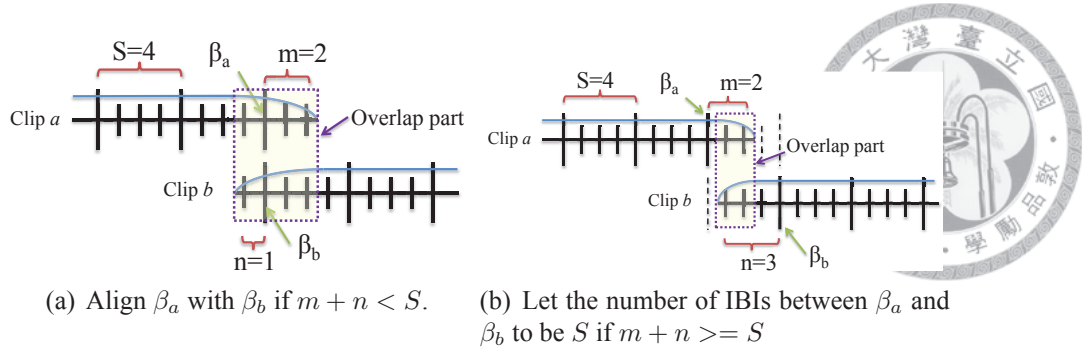


Figure 4.2: Proposed bar alignment method.

of this scheme. We first extract bar and beat information (c.f. Section 3.1) from music signals. Then, we proceed the following two processes according to the conditions at the boundaries of phrases, as shown in Figure 4.2. For each pair of consecutive phrase clips a and b , suppose that there are m remaining inter-beat intervals (IBIs) after the last bar of clip a , and n IBIs before the first bar of clip b . Let S be the average number of IBIs per bar of the former song, which can be treated as the time signature of clip a . Let β_a and β_b respectively denote the last bar of the former clip and the first bar of the latter clip. If $m + n < S$, we align β_a with β_b , as shown in Figure 4.2(a), otherwise, if $m + n \geq S$, we let the number of IBIs between β_a and β_b to be S (see Figure 4.2(b)).

4.2 Tempo Adjustment Process

After locating the transition segments, now we need to adjust the tempi of the clips so that the concatenated clips will sound smooth in beat counting.

4.2.1 Transition Duration Determination

For each pair of clips a and b , we adjust the tempi of the two clips to make them smooth. To gradually adjust the tempi from the tempo of clip a , T_a , to the tempo of clip b , T_b , a transition duration of K IBIs is determined to ensure that the tempo change ratio, r , at each beat, is small enough so that the change in the speed of the songs would not sound abrupt to the listener. That is, r should lie in the range of the Just Noticeable Difference

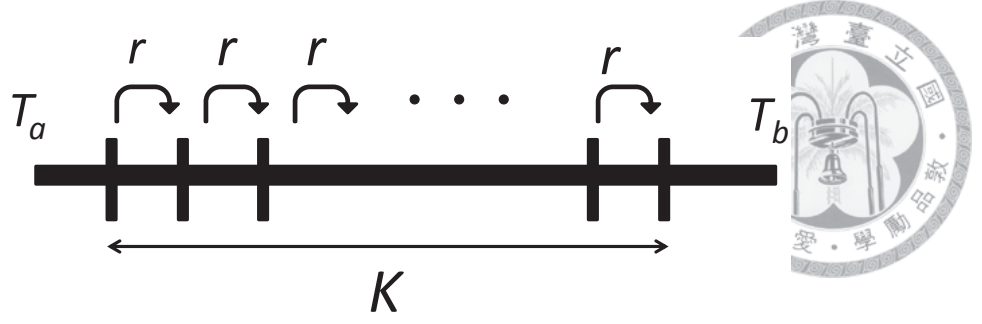


Figure 4.3: Diagram for changing tempi within a duration of K IBIs.

(JND) [16] in the domain of psychoacoustics, which can be calculated as

$$r = \sqrt[K]{\frac{T_b}{T_a}}. \quad (4.5)$$

Figure 4.3 shows a diagram describing the ratio r .

JND is defined as the minimum difference of stimuli that people can perceive. These stimuli include loudness, tempo and pitch. According to *Weber's law*, JND can be computed with the *Weber's Constant*. However, the *Weber's Constant* of tempo varies with changes in the environment. Thus, inspired by Thomas [59], we conduct experiments to find the JND of tempo on our music datasets (c.f. Section 6.1.5). For fast tempo clips (120 ~ 180 BPM), we found out that the ratio of the tempi from 0.96 to 1.03 will not be perceived. For slow tempo clips (40 ~ 90 BPM), the JND ranges from 0.97 to 1.04.

Since real world music clips may contain more than one tempo, (e.g., the pieces with *accelerando* or *ritardando*), we developed Algorithm 2 to find the transition duration and the ideal target tempi T_a^* and T_b^* . The procedure is also illustrated in Figure 4.4. We start from the boundaries of transition segments, each time extend the transition duration with 1 IBI, and check the value of r . The iteration stops when r lies within the range of JND. We then compute the corresponding ideal tempi for clips a and b . Afterwards, phase vocoder [60] is used to adjust the tempi from T_a and T_b to T_a^* and T_b^* , respectively. Figure 4.5 shows the example results of two song excerpts from: Jolin Tsai's "Let's move it" (蔡依林《Let's move it》) and "Real man" (《大丈夫》). The ratio of change appears like a linear decay because the ratios are usually very close to 1.



ALGORITHM 2:

Input: the tempi of clip a and clip b : $T_a(i), T_b(j)$, for $i = 1 \dots N, j = 1 \dots M$ IBIs, and the start indexes (in IBI), i^*, j^* , duration L^* of transition segments in clips a and b , respectively.

- 1: **for** $x = 0$ to $i^*, y = 0$ to $(M - L^* - j^*)$ **do**
- 2: $i_{tmp} \leftarrow (i^* - x)$
- 3: $j_{tmp} \leftarrow (j^* + L^* + y)$
- 4: $r \leftarrow x+y+L^* \sqrt{\frac{T_b(j_{tmp})}{T_a(i_{tmp})}}$
- 5: **if** r is within JND **then**
- 6: **break**
- 7: **end if**
- 8: **end for**

$$T_a^*(i) \leftarrow \begin{cases} T_a(i), & \text{for } i \leq i_{tmp} \\ T_a(i_{tmp}) \times r^{(i-i_{tmp})}, & \text{otherwise.} \end{cases}$$

$$T_b^*(j) \leftarrow \begin{cases} T_b(j), & \text{for } i \geq j_{tmp} \\ T_b(j_{tmp}) \times r_c^{-(j_{tmp}-j)}, & \text{otherwise.} \end{cases}$$

Output: the target tempi T_a^*, T_b^*

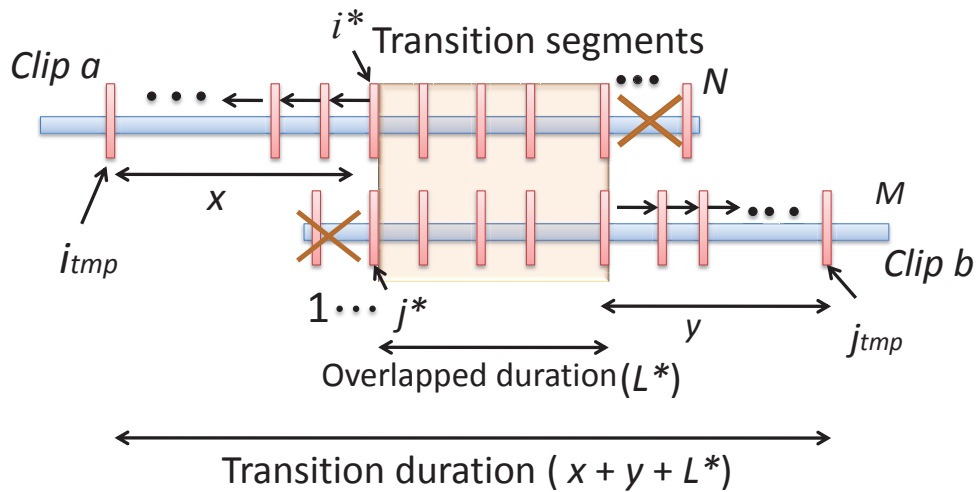


Figure 4.4: Schematic diagram of finding the transition duration.

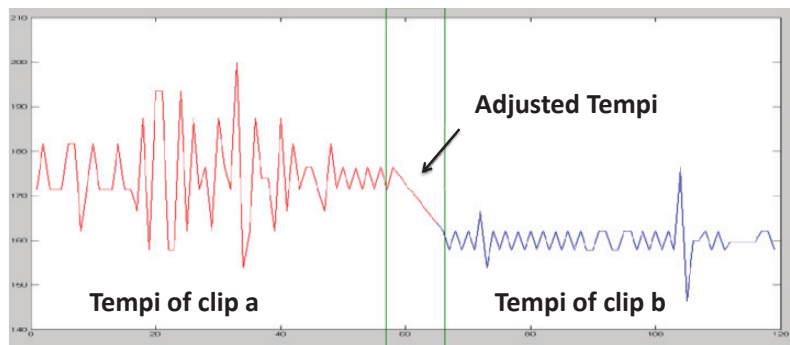


Figure 4.5: Tempi changes in the transition duration of the clips of “Let’s move it” and “Real man.”

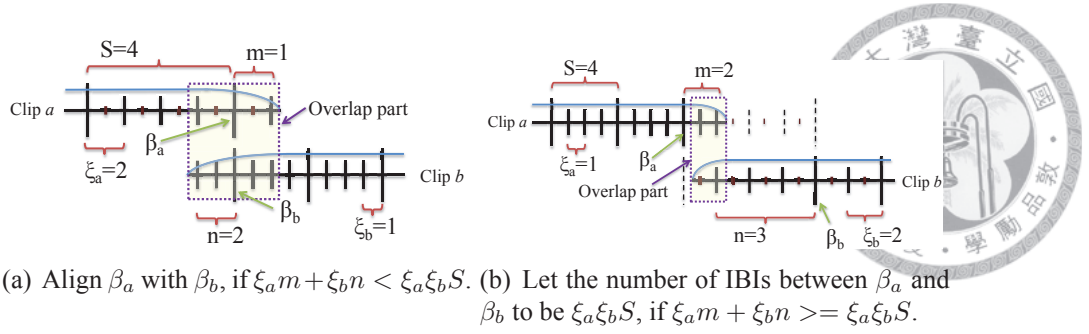


Figure 4.6: Proposed bar alignment method (with consideration of dual tempo adjustment).

4.2.2 Dual Tempo Adjustment

Sometimes the consecutive music clips a and b may be with large tempo differences. In such a case, we may not find a long enough transition duration to gradually change the tempi. Besides, the found ideal tempi will be far from the original ones and will lead to explicitly audible artifacts. Moreover, most beat detection algorithms have a common issue with double/half errors [36]. To handle this issue, we incorporate a concept similar to what Ishizaki et al. mentioned in [36] to deal with it: take the dual tempo into account. That is, match the IBIs of clip a to its double/half or to its quadruple/quarter of that of clip b . Let T_a and T_b be the average tempi of clips a and b . The weighting factors representing the multiples relations of clips a and b can be calculated as

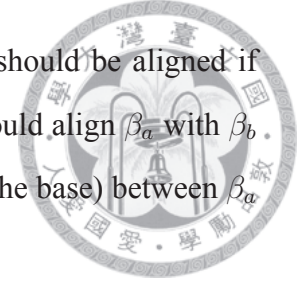
$$\xi_a = \arg \min_{i \in \{1, 2, 4\}} |i \cdot T_a - T_b|, \quad (4.6)$$

$$\xi_b = \arg \min_{i \in \{1, 2, 4\}} |i \cdot T_b - T_a|. \quad (4.7)$$

For example, in Figure 4.6(a), $\xi_a = 2$ and $\xi_b = 1$ while in Figure 4.6(b), $\xi_a = 1$ and $\xi_b = 2$. Then, to apply the tempo adjustment method presented in Algorithm 2, we should make the numbers of overlapped IBIs to be the same for both clips. So, we up-sample the tempo value sequence and weight the tempo values in the slower-tempo clip s by its factor ξ_s . After this pre-processing, we could apply Algorithm 2 without any change while still taking the dual tempo into account.

Note that the bar alignment method presented in Section 4.1.3 should be modified accordingly as follows. As shown in Figure 4.6, S , m and n are respectively weighted

by factors of ξ_a and ξ_b for clips a and b . Accordingly, β_a and β_b should be aligned if $\xi_a m + \xi_n < \xi_a \xi_b S$. On the other hand, if $\xi_a m + \xi_n \geq \xi_a \xi_b S$, we should align β_a with β_b in a way to make the number of IBIs (take IBIs of the faster clip as the base) between β_a and β_b to be $\xi_a \xi_b S$.



4.3 Synthesis Process

The final step is the synthesis process. In this step we handle the volume levels of the music materials.

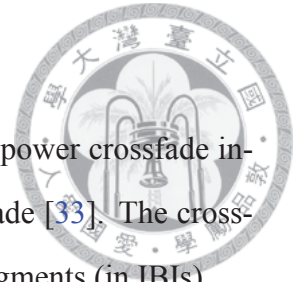
4.3.1 Volume Normalization

Sometimes the volume levels of music materials are quite different, causing the transition quite intrusive (e.g., the sound suddenly becomes loud.). To handle such a condition, the volume levels of the clips are then normalized so that the volume levels near the transition segments of the segued clips are consistent across the clips. After scaling every selected clip so that the amplitude of the signals falls into the range of -1 to 1, the logarithmic intensity² (c.f. [Section 3.3](#)) within a small window (approximately 3 seconds) at both the beginning and the ending of each clip is calculated. The beginning or the ending window whose volume level represents the median of all selected clips is selected as the reference. Starting from the clip containing this reference window, we adjust the volume of each neighboring clip one-by-one so that the beginning or the ending window alongside of the reference window has the same volume level as the reference. In other words, for a clip a , we denote its beginning and ending windows as w_s^a and w_e^a , respectively. Let $V(w)$ denote the volume level of a given window w . Given a set of clips $\{u_1, u_2, \dots, u_N\}$, if the selected reference window is at the end of clip u_i . We adjust the global volume levels of clip u_{i+1} and u_{i-1} to make $V(w_s^{u_{i+1}}) = V(w_e^{u_i})$ and $V(w_e^{u_{i-1}}) = V(w_s^{u_i})$, respectively. The volume levels of $w_e^{u_{i+1}}$ and $w_s^{u_{i-1}}$ should be changed accordingly. We then adjust the global volume levels of clip u_{i+2} , u_{i-2} , and so on.

²i.e., the volume level

4.3.2 Crossfading

The clips can then be concatenated with each other using a constant-power crossfade in-between to make the perceptual energy “constant” during the crossfade [33]. The crossfade duration is determined according to the duration of transition segments (in IBIs).





Chapter 5

Material Selection

There is no perfect pickle; there are only perfect pickles.

– Howard Moskowitz (in Malcolm Gladwell’s TED talk, 2004)

In this chapter, we will describe the selection schemes and the overall system structures in our first two projects: “Music Paste [4]” (with a straightforward selection scheme) and “Audio Musical Dice Game [5, 6]” (with a graph-assisted and personalized selection scheme).

5.1 Straightforward Scheme

The first selection scheme is quite straight forward, we just remove unfitted clips, and then order the remaining clips based on the similarity values of found transition segments. [Figure 5.1](#) illustrates the system flows of the first project [4]. First, all the music features we need for the input music are extracted, such as volume, chroma, rhythm, and tempo. Then, in the selection stage, we filter out distinct clips by pair-wise comparisons, and then order the clips by similarity values of transition segments between all pairs of clips. After that, we perform the steps we mentioned in [Chapter 4](#). The transition segments are located according to chroma and rhythm similarities (c.f. [Section 4.1.1](#)). The tempi are adjusted based on [Section 4.2.1](#) and the clips are synthesized without volume normalization because clips with distinct volume levels will be filter out in previous steps. The materials filtering

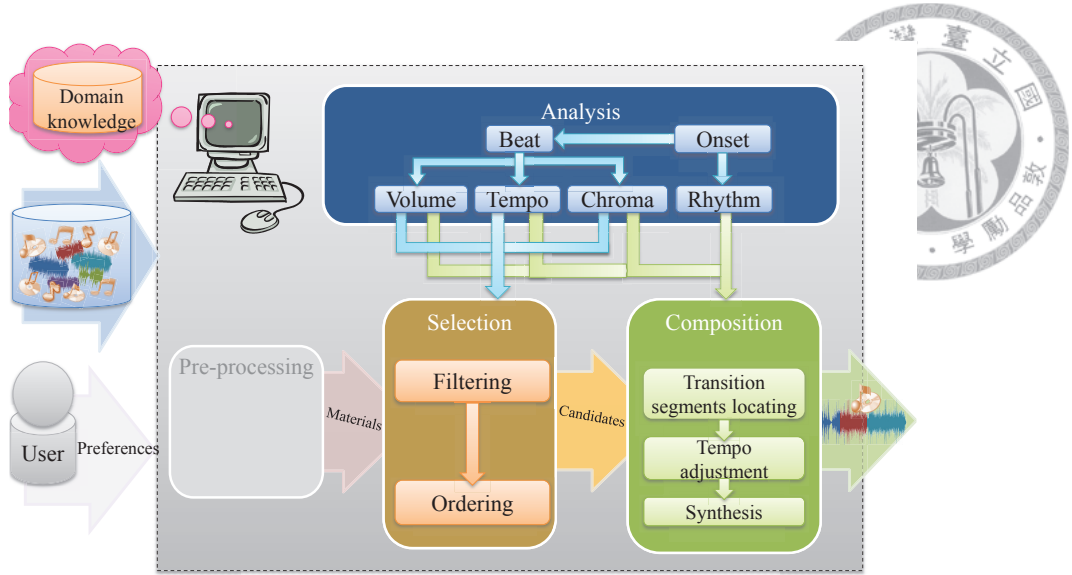


Figure 5.1: System Framework for our first project: “Music Paste [4]”’s

and ordering schemes will be detailed in the following sections.

5.1.1 Filtering

In order to reduce the probability of concatenating quite distinct clips and the computational load in the ordering process (c.f. Section 5.1.2), we remove clips with extreme values by pair-wise comparison. A clip a is said to be with extreme values and should be removed if there are more than half of the other clips (clip b) in the database dissimilar to clip a . The dissimilarity and similarity of any two clips are measured sequentially as follows.

Loudness Dissimilarity

The loudness dissimilarity is defined by the ratio $r_L(a, b)$ of the average volume levels of two clips clip a and clip b , as shown in Equation (5.1),

$$r_L(a, b) = \frac{|Ld_a - Ld_b|}{Ld_a}, b = 1 \dots W, b \neq a, \quad (5.1)$$

where Ld_a and Ld_b are the average volume values of the a^{th} and the b^{th} clips in the datasets and W is the total number of clips. The volume values are computed by accumulating log-energy (in db) in all the frequency bands (c.f. Section 3.2). Clips a and b are said to be

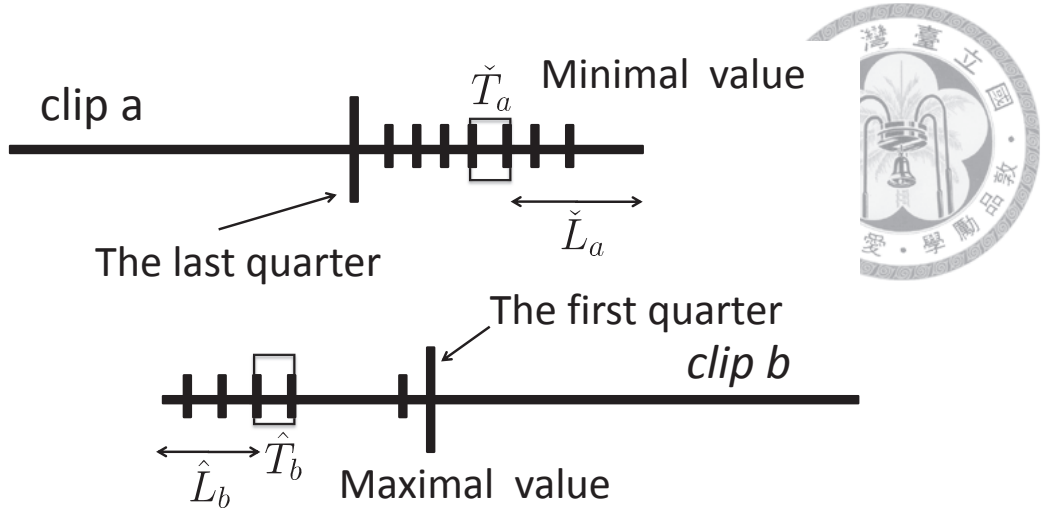


Figure 5.2: How to find the tempo dissimilarity.

loudness-dissimilar if $r_L(a, b)$ is greater than a certain threshold. By Weber's law [16], the JND of loudness in db is 0.1, i.e. we will perceive the loudness change between clip a and clip b when the changing ratio ($r_L(a, b)$) is greater than 0.1 db. Since we will apply crossfade in-between concatenated clips, we set the threshold value as 0.2 instead of the original strict standard.

Tempo Dissimilarity

Clips a and b are said to be tempo-dissimilar if there are not enough durations for them to gradually adjusting the tempi from one to the other. The tempo dissimilarity is defined as $r_T(a, b)$, that is

$$r_T(a, b) = \left(\frac{\hat{T}_b}{\check{T}_a} \right)^{\frac{1}{\check{L}_a + \hat{L}_b}}, a = 1 \dots W, b \neq a, \quad (5.2)$$

where \check{T}_a and \check{L}_a are the minimal tempo value of the last quarter in clip a and the corresponding length (in IBIs) taken from the position of \check{T}_a to the end of clip a . Similarly, \hat{T}_b and \hat{L}_b are the maximal tempo value of the first quarter in clip b and the corresponding length, as shown in Figure 5.2. If $r_T(a, b)$ does not lie in the range of JND mentioned in section 4.2.1, there will not be enough transition length for changing tempi from clip a to clip b and they should be regarded as tempo-dissimilar.



Chroma Histogram Similarity

In this module, we tend to avoid concatenating clips with different pitch distribution. The reason is as follows: the resultant medley will be unpleasant if we directly combine clips of different tonalities (e.g. C Major \rightarrow e \flat minor) without modulation. Generally speaking, music clips with the same tonality contain similar pitch distributions. Thus, we construct a chroma histogram for each clip to represent its dominant pitch distribution and compare the clips by the corresponding histograms. For the 12 dimensional chroma vector (\vec{C}_{ai}) of the i^{th} IBI in clip a , we choose the index of its maximal value to represent the chroma dominant pitch (\hat{C}_{ai}) of the IBI. That is,

$$\hat{C}_{ai} = \arg \max_u \vec{C}_{ai}(u), u = 1 \dots 12 \quad (5.3)$$

The chroma histogram of clip a (CH_a) is constructed from the statistics of dominant pitch represented by \hat{C}_{ai} . Inspired by the commonly used color histogram intersection method [61] in the computer vision field, we define the chroma histogram similarity between clip a and clip b by

$$S_H(a, b) = \frac{\sum_{u=1}^{12} \min(CH_a(u), CH_b(u))}{\sum_{u=1}^{12} CH_a(u)} \quad (5.4)$$

where $b = 1 \dots W$, $a \neq b$. Analogous to the two previous subsections, clip a and clip b are viewed as dissimilar if $S_H(a, b)$ is less than 0.5.

5.1.2 Ordering

In the music ordering process, we tend to find an appropriate order to minimize the average distance values between each clip pair. For example, if the transition segments between $clip_a$ and $clip_b$ is not similar enough, maybe $clip_p$ can be the bridge for them. Besides, the transition segments from $clip_a$ to $clip_b$ may be less similar as compared with the transition segments from $clip_b$ to $clip_a$. Therefore, the ordering problem can be formulated as finding a path which goes through all clips in the datasets with minimum cost in the ordering matrix

	<i>clip</i> ₁	<i>clip</i> ₂	<i>clip</i> ₃	<i>clip</i> ₄
<i>clip</i> ₁	0	0.3486	0.329	0.342
<i>clip</i> ₂	0.3936	0	0.4704	0.4577
<i>clip</i> ₃	0.2609	0.537	0	0.4806
<i>clip</i> ₄	0.2898	0.4826	0.3732	0



Figure 5.3: An example of the ordering matrix for 4 clips.

(D_o) defined as follows:

$$D_o[a, b] = \min_{i,j,L} \frac{1}{L+1} \sum_{l=0}^L D_{CR}(a, b)[i+l, j+l] \quad (5.5)$$

where $L \in [L_{min}, L_{max}]$. To reduce the computation loads, we use a method analogous to the greedy algorithm but the path found cannot be guaranteed to reach the global optimum. The procedure is as follows:

1. Find the minimum value in the ordering matrix and set the corresponding two clips as the initial clips.
2. Find the minimum value in the row that corresponding to the last clip in the order found previously (each clip can only be visited once) and then add the corresponding clips to the order.
3. Repeat step 2 until all the values in the target row are larger than a predefined threshold or all clips have been visited.

Figure 5.3 shows an example of an ordering matrix constructed by four clips. First, we look for the minimum value in the matrix: 0.2609. We set the order as 3 \rightarrow 1. Then, we check the values of first row: {0, 0.3486, 0.3290, 0.3420}. Since the first entry (0) represents *clip*₁ going to *clip*₁ itself and the third entry (0.3290) means *clip*₁ going to *clip*₃ again, we would not consider these two values. We find the minimum value of the rest: {0.3486, 0.3420} is 0.3420. Thus, the order becomes 3 \rightarrow 1 \rightarrow 4. Next, we check the

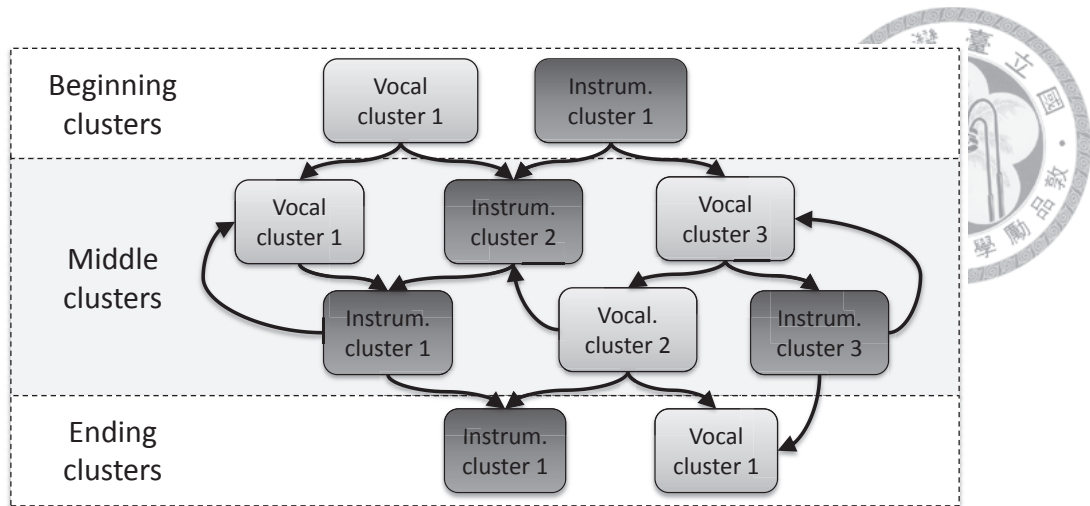


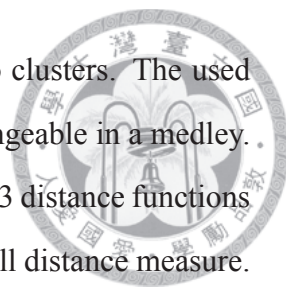
Figure 5.4: Example of a musical dice graph.

fourth row and find 0.4826 is the only left value, so we compare it with the predefined threshold. If it is smaller than the threshold, the order would become $3 \rightarrow 1 \rightarrow 4 \rightarrow 2$. Otherwise, we would not concatenate $clip_2$ and the order would be just $3 \rightarrow 1 \rightarrow 4$. Currently, the threshold is 0.5.

5.2 Graph-assisted and Personalized Scheme

In the second selection scheme, we took another view of material selection. The medley creation is turned into an audio version musical dice game. The musical dice game, also known as *Musikalische Würfelspiele* [15], is a kind of music composition which originated from the European classical era. In a musical dice game, players throw dice to randomly choose short pieces of melodies from a pool of pre-composed interchangeable musical figures¹ for each bar. Aside from providing entertainment value, this kind of composition enables people without music knowledge to compose music on their own, i.e. they can “generate” multiple new pieces of music simply by throwing dice. Similarly, our system generates medleys by choosing the clip at a given position from a set of interchangeable clips. To create sets of interchangeable clips, we first analyze the songs in the user-provided collection and cut the songs into clips (based on musical phrase detection we have mentioned in Section 4.1.2) and determine the type of the clips, i.e. vocal

¹A short musical phrase [13].



or instrumental. For each clip type, we then group similar clips into clusters. The used distance measures should make the clips in the same cluster interchangeable in a medley. Many distance measures can be used in this regard. Here we combine 3 distance functions for chord sequence, timbre and tempo, respectively, to form the overall distance measure. We then connect clusters according to the transition probability calculated from clip connectivity in the songs from which they were originally extracted. The result is referred to as a “musical dice graph” in which the vertices are the clusters and the edges are weighted by the calculated transition probability. Each path on the graph is a version of a medley. [Figure 5.4](#) shows an example of a musical dice graph. With this graph, we can generate various medleys based on user preferences and the transition probability. This allows us to transform the steps of concatenative music re-composition, “material selection” and “material composition” into “musical dice graph construction” and “medley generation from the walk on the graph” respectively. [Figure 5.5](#) illustrates the proposed framework. The selection step has been divide into musical dice graph construction and a part of medley generation, path finding. After path finding, the selected clips are concatenated via methods we have mentioned in [Chapter 4](#). The transition segments are decided at phrase boundaries. Tempo adjustment scheme is the same as the one mentioned in [Section 4.2.1](#), and in the synthesis process, the clips are concatenated after volume normalization. The methods used in the selection scheme will be detailed in the following section.

5.2.1 Musical Dice Graph Construction

We divide the construction of a musical dice graph into two steps: clip clustering, and cluster connecting.

Clustering

After dividing songs into phrase clips, we group these clips according to their degree of similarity of chord sequences, timbre, and tempo. In a musical dice game, the interchangeable musical figures are often chord-similar or dominant-pitch-similar [15]. Besides, since the task deals with audio music, similarity of timbre and tempo must be accounted for, an

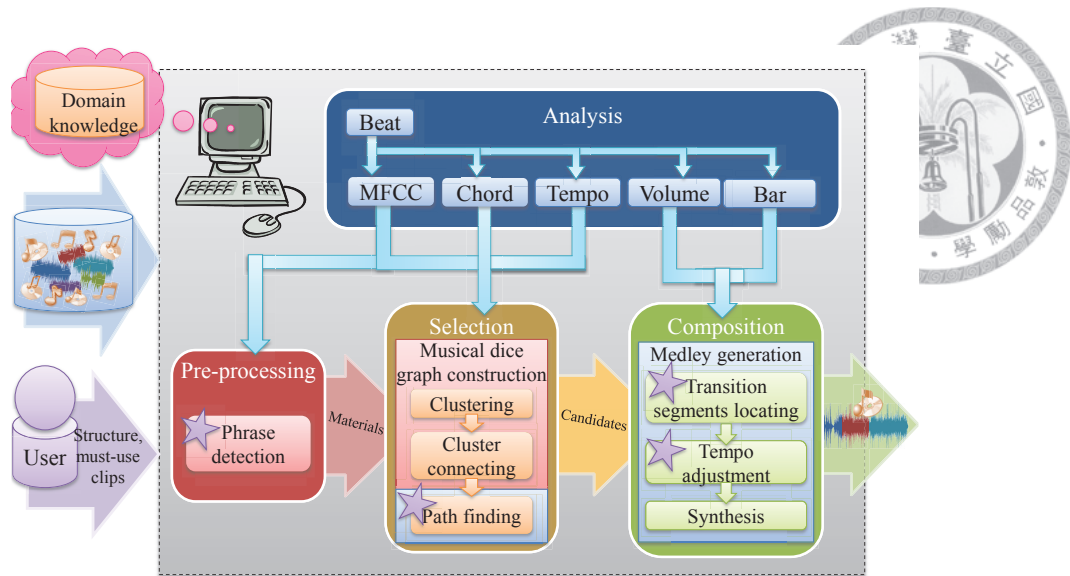


Figure 5.5: System Framework for our second project: “Audio Musical Dice Game [5, 6].” Purple stars marks the function blocks that can take user preference into account.

issue that does not arise in the musical dice game in the symbolic domain. Two clips of different timbres and tempi can sound quite different even if their score notations are the same, e.g., the same song played with different instruments with different tempi. The clip clustering process can be divided into two steps: distance computation, and clustering.

Distance of Chord Sequence To compute the chord sequence distance of two clips, we first detect the beat-synced chord sequence appearing in the given clips (c.f. Section 3.2). We then measure the chord sequence distance between each clip pair. First, our system measures the similarity between two given chord sequences by using an edit-based approach proposed by [62] via local alignment. To better capture the harmonic relationship between two chord sequences, the substitution score used to calculate similarity varies with the consonance of the interval between two given chord roots, as proposed by [7]. Consonant intervals are the intervals that sound stable [13], and chords whose roots are more consonant will be given higher substitution scores. For example, substituting a C chord with a G chord (the fifth chord of the C chord) may affect the chord sequence less than substituting it with an Am chord. As a result, the sequence pair: “C F C G” and “C F G G” has a higher score (8.55) than that of “C F C G” and “Am F C G” (6.55). The used substitution scores are listed in Table 5.1. The pseudo code for calculating chord sequence similarity is stated in Algorithm 3. Finally, the similarity score of two chord sequences



Pitch Differences in Semitones	Associated Score
0	+2.850
1	-2.850
2	-2.475
3	-0.825
4	-0.825
5	+0.000
6	-1.800

Table 5.1: Different substitution scores of the edit-algorithm according to the intervals between chord roots.[7]

will be transformed to a distance score after normalization (c.f. [Section 5.2.1](#)), in order to be combined with the other two distance functions: timbre and tempo.

ALGORITHM 3: The pseudo code for computing chord sequence similarity

input : two chord root sequences C_1 and C_2 of lengths M and N , respectively

output: the chord sequence similarity score simScore

$\text{delScore} \leftarrow -1$; $\text{insertScore} \leftarrow -1$;

$\text{subScore} \leftarrow [2.85, -2.85, -2.475, -0.825, -0.825, 0, -1.8, -0.5]$;

for $i \leftarrow 1$ **to** M **do** $\text{localScore}[i, 0] \leftarrow \text{localScore}[i - 1, 0] + \text{delScore}$;

;

for $j \leftarrow 1$ **to** N **do** $\text{localScore}[0, j] \leftarrow \text{localScore}[0, j - 1] + \text{insertScore}$;

;

for $i \leftarrow 1$ **to** M **do**

for $j \leftarrow 1$ **to** N **do**

$\text{del} \leftarrow \text{localScore}[i - 1, j] + \text{delScore}$;

$\text{ins} \leftarrow \text{localScore}[i, j - 1] + \text{insertScore}$;

$d \leftarrow |C_1[i - 1] - C_2[j - 1]|$;

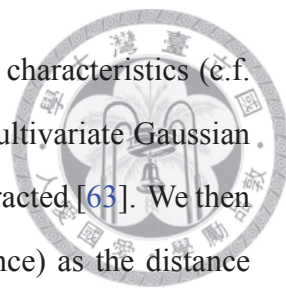
$\text{sub} \leftarrow \text{localScore}[i - 1, j - 1] + \text{subScore}(\min(d, 12 - d))$;

$\text{localScore}[i, j] \leftarrow \max(\text{del}, \text{ins}, \text{sub})$;

end

end

$\text{simScore} \leftarrow \text{localScore}[M, N]$;



Distance of Timbre MFCCs [46] are used to represent the timbre characteristics (c.f. Section 3.4). To calculate the distance of MFCC between clips, a multivariate Gaussian model is used to describe the distribution of MFCCs after they are extracted [63]. We then employ the symmetrized Kullback-Leibler divergence (KL divergence) as the distance measure between the Gaussian models of different clips, as suggested in [63].

Distance of Tempo The average tempo (measured in BPM) of a clip a , T_a , can be calculated as:

$$T_a = \frac{1}{N-1} \sum_{i=1}^{N-1} T_a(i), \quad (5.6)$$

where $T_a(i)$ is the tempo of the i^{th} IBI of clip a (c.f. Section 3.1), and N is the number of IBIs in clip a . The tempo distance between clips a and b is the absolute difference between T_a and T_b .

Clustering We then normalize the chord sequence distance, the timbre distance, and the tempo distance of all clip pairs by subtracting the corresponding minimum distance scores from the distances and dividing them by their ranges so that the distance scores lie between 0 and 1. A mixed distance score of all pairs is then calculated by performing a weighted average over the three distance measures mentioned above. Given the distance between each pair of the clips, we could then cluster the clips by average-linkage hierarchical clustering. Each clip is first categorized into 6 types based on its properties (vocal or instrumental) and its positions (beginning, middle, or end). The clips of different types are then clustered separately. In other words, we would have 6 types of clusters in total: beginning, ending and middle clusters, each of which can be either vocal or instrumental. For a 100-song collection, each type consists of an average of around 38 clusters, with an average of about 6 clips per cluster.

Cluster Connecting

Finally, we connect the clusters according to the transition probability, as defined in Equation (5.7). For two arbitrary clusters A and B , the transition probability $P(B|A)$ is defined

as the proportion of clips in cluster A that is originally concatenated with clips in cluster B , that is

$$P(B|A) = \frac{|S|}{|A|}, S = \{(a, b) | a \in A, b \in [N(a) \cap B]\}, \quad (5.7)$$

where a and b stand for two arbitrary clips, and $N(a)$ is the set of clips that appearing just after clip a in the original song.



5.2.2 Medley Generation

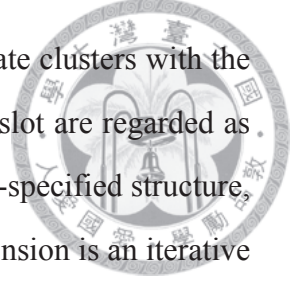
Once the musical dice graph is constructed, we can then compose medleys by finding a path on the graph and concatenating² them according to the clips selected from the clusters on the path.

Path Finding

Here we describe how to find a path on the musical dice graph with the maximum transition probability. First, we pick candidate clusters according to the user-specified medley structure. For example, the user may designate the structure as $I \rightarrow V \rightarrow I \rightarrow V \rightarrow I$, where “I” and “V” respectively stands for instrumental and vocal clips. For the previous example, we then choose clusters conforming to the types the user specified in the structure as candidate clusters, that is, instrumental-beginning clusters for the first clip slot, and vocal-middle clusters for the second, and so forth.

Then, for those slots where the user has specified must-use clips or songs, we assign the corresponding cluster(s) which conform to the user specified condition as candidates. For instance, if the user assigns clip a to slot n , then the cluster to which clip a belongs is directly assigned to slot n . If the user specifies that slot n should be filled with a clip from song A , then the clusters contain clips of song A are chosen as candidate clusters for slot n . The user can also specify a desired range of duration for each chosen clip. The clusters which do not have any clips within the desired duration range are eliminated from the set of candidate clusters.

²using the methods we mentioned in [Chapter 4](#)



We then use Viterbi algorithm [64] to find a path through candidate clusters with the maximal transition probability, where the candidate clusters at each slot are regarded as the states used in the algorithm. If the path does not exist for the user-specified structure, the system will automatically extend the structure. The structure extension is an iterative process. At each iteration, a new slot will be inserted after the first found slot where the transition probabilities from all the candidates at that slot to the candidates at its following slot are zero. For example, if the transition probabilities from all the candidates at slot n to the candidates at slot $n + 1$ are zero, a new slot will be inserted between n and $n + 1$. After slot insertion, the system will check whether the path can be found with the new structure. The process will be iterated several times until a path has been found or a specified maximum number of iterations is reached.

After path finding, we randomly select one clip per cluster along the path³, since clips in the same cluster are assumed to be interchangeable. Here, we also design another option for the system to have less probability of selecting consecutive clips from the same song. The idea is to reduce the probability of a clip to be chosen if any clip from the same song has been selected in the previous slots of the medley. The degree of probability reduction is based on the distance between the previously selected same-song clip to the current slot η , as follows:

$$p_\eta(a) = p_\eta(a) - \sum_{n=1}^{\eta-1} p_\eta(a) \cdot \gamma^{\eta-n} \cdot \Gamma(\text{clip } a \text{ and the chosen clip at slot } n \text{ belong to the same song}), \quad (5.8)$$

where $p_\eta(a)$ is the probability of clip a in the current cluster at slot η , γ is a parameter between 0 and 1, and $\Gamma(\cdot)$ is an indicator function. After this modification, all of the selected clips are then used to compose the final medley by methods mentioned in [Chapter 4](#).

5.3 User Interface

The quality of the generated medley is highly subjective and depends greatly on users' preferences. For example, it is hard to determine the ending position of a singing voice

³except for slots that the user has specified certain clips.



Input Parameters

Duration per clip (sec): 0 ~ Inf

Reduce the prob of choosing same song use GT phrase

Crossfade Duration: 1 beat

Go!!!

**It takes about 30 seconds to generate a medley.
(The running time depends on the number of segments)

Instrumental Vocal Instrumental Vocal Instrumental

4. All 4 One-I Swear

2. Choose Phrase

All 4 One-I Swear
** double click on the phrase to assign that phrase.
** If you cannot see the waveform, please change your browser to the newestest Chrome.

Time (sec) Label

0.0~8.9	vocal
8.9~16.5	vocal
16.5~22.7	instrumental
22.7~40.8	vocal
40.8~63.6	vocal

1. Choose Song

1. Abba--Thank You For The Music
2. Air Supply--Only One Forever
3. Alan Jackson--Remember When
4. All 4 One-I Swear
5. Andrea Bocelli and Sarah Brightman--Time to Say Goodbye
6. Audrey Hepburn--Moon River
7. Barbara Dickson--Another Suitcase in Another Heart
8. Bee Gees--I Started a Joke
9. Bette Midler--The Rose
10. Billy Joel--Piano Man
11. Billy Joel--She's Always a Woman to Me
12. Bob Dylan--Don't Stop Believin'
13. Bob Dylan--Forever Young
14. Bob Dylan--The Times They Are a-Changin'
15. Bob Dylan--The Times They Are a-Changin'
16. Brothers Four--Yellow Bird
17. Carol Kidd--When I Dream
18. Cat Stevens--Morning Glory
19. Cat Stevens--Morning Glory
20. Celine Dion--My Heart Will Go On
21. Charlie Rich--Dejeu
22. Chicago--If You Leave Me This Way
23. Cliff Edwards--When You Wish Upon a Star
24. Debby Boone--You Light Up My Life
25. Don McLean--American Pie
26. Doris Day--Fly Me to the Moon

Figure 5.6: Screenshot of our graphical user interface for medley creation, where users can specify the medley structure, must-use clips, and other parameters

Result

Waveform

** double click on the phrases region to adjust their boundaries.
** If you cannot see the waveform, please change your browser to the newestest version of Chrome.

Wayton Jennings--Amanda, phrase 4

Adjust Boundaries

Used Clips

** click on the phrase label to hear each phrase.

Time (sec)	Label
0.0~7.1	Wayton Jennings--Amanda, phrase:1
6.1~22.0	Alan Jackson--Remember When, phrase:11
21.0~29.1	Wayton Jennings--Amanda, phrase:14
28.3~47.1	All 4_One-I Swear, phrase:4
46.5~66.4	Randy Vanwarmer--JustWhen_I_Needed_You_Most, phrase:22

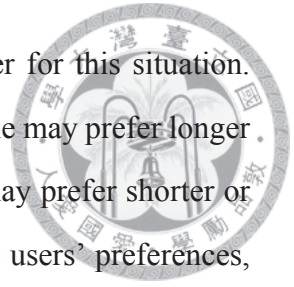
Notice! Phrase boundaries have been changed, click "Refine!!" to generate a new medley!

Adjust the crossfade duration: 2 beat

Create New Refine!!

Figure 5.7: Screenshot of our graphical user interface for medley generation, where the user can adjust the phrase boundaries manually.

that fades out gradually, and there is no absolutely “correct” answer for this situation. The crossfade duration between two clips is also subjective since some may prefer longer overlaps to increase the smoothness of the transition, while others may prefer shorter or even no crossfade to avoid blurring sounds. To better satisfy various users’ preferences, we have developed a GUI (as shown in [Figure 5.6](#) and [Figure 5.7](#)) where users are allowed to specify parameters, modify the segmentation result, and so on. The demo site can be found at <http://www.cmlab.csie.ntu.edu.tw/~known/medley/demo/>.





Chapter 6

Experiments

*The best and most beautiful things in the world cannot be seen nor even touched,
but just felt in the heart.*

– Helen Keller, 1891

In this chapter, we will describe experiments about the performance of the proposed concatenative music re-composition system. Most of the experiments are conducted by subjective evaluations because the aesthetic appeal of music is highly subjective and is subject to personal tastes. As a result, it is not feasible to compare all the combinations of methods we proposed. In the rest of this chapter, parameter explorations and effectiveness of the components were investigated individually. And the comparisons among different combinations of components were done according to the versions in our projects[4, 5, 17, 6]. For those comparisons or settings we did not provide, interested readers may judge for themselves by listening the used samples at <http://www.cmlab.csie.ntu.edu.tw/~known/medley/results/>. and exploring our demo site at: <http://www.cmlab.csie.ntu.edu.tw/~known/medley/demo/>

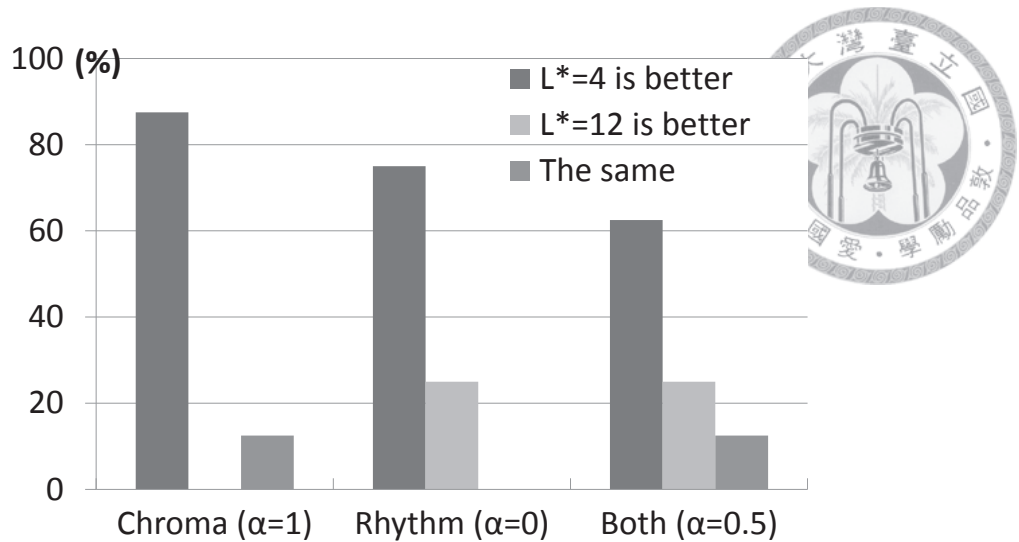


Figure 6.1: Comparisons among various overlap durations.

6.1 Evaluations on Concatenation Methods

6.1.1 Overlap Duration of Similarity-based Transition Segments

In this experiment, we discussed the effect of overlap durations in similarity-based transition segments (c.f. [Section 4.1.1](#)). Fifteen evaluators were invited to report their satisfaction. 8 sets of clips (≈ 40 secs/clip) taken from different types of Chinese pop songs are used. We generate medleys with 2 overlap durations (force $L^* = 4, 12$ IBIs in [Equation \(4.4\)](#)), and each of them are with three different α values (we set $\alpha = 0, 0.5, 1$ in [Equation \(4.3\)](#)) when locating transition segments. [Figure 6.1](#) presents the overall results. The vertical axis denotes the percentages of how many people prefer each method. We found that results with longer overlap duration are not necessarily more acceptable than the shorter ones. The reason is probably that the similarity of transition segments decreases as the overlap duration grows. Another observation is that the evaluator's acceptance varies with the types of the music clips. For instance, the accepted overlap duration between two rap clips may be shorter than those of two lyric clips. Over 60% of the evaluators preferred 4 IBIs as the overlapping duration. Hence, we set the default overlap duration to 4 IBIs long in the next section to compare the influence of different similarity measurements.

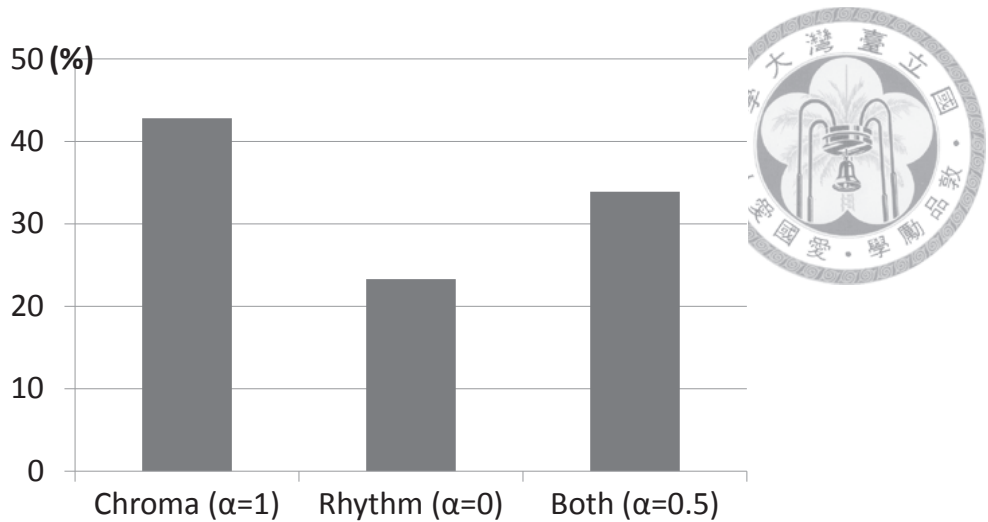


Figure 6.2: User preference comparisons of similarity measurements.

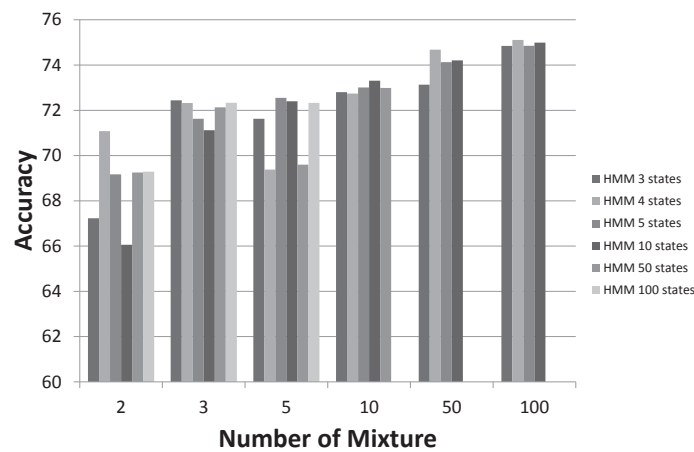


Figure 6.3: The results of singing voice detection with different HMM parameters¹.

6.1.2 Similarity Measurements in Similarity-based Transition Segments

This experiment discussed the similarity measurement for locating similarity-based transition segments (c.f. [Section 4.1.1](#)). The compared similarity measurements are chroma only, rhythm only and both chroma and rhythm, i.e. $\alpha = 0, 1, 0.5$. The overlap duration is set to 4 IBIs. We utilized 8 sets of clips from songs in different languages. Fifteen evaluators gave scores from 1 to 10 to represent their satisfactions (higher score means better satisfaction) with respect to the feeling of intrusion. [Figure 6.2](#) shows the percentages of how many people prefer each method. The results show that chroma only may be the most preferred measurement. Thus, we choose the chroma measurements to conduct other experiments of concatenation with similarity-based transition segments.

6.1.3 Effectiveness of Phrase Detection



This experiment investigates the effectiveness of the proposed approach for phrase detection, which is based on the result of singing voice detection (c.f. Section 4.1.2).

The used music dataset consists of 100 English hit songs from the 1950s to the 1990s, collected from Youtube³. These songs correspond to various genres, including folk, pop, jazz, Broadway musical and movie soundtrack, with track length ranging from 1.5 to 5.5 minutes. Two annotation sets were manually built to create the ground truth of vocal/instrumental segments and musical phrases, respectively. In total, there are 1409 musical phrases and 1716/1813 vocal/instrument segments in the dataset. For each track, both annotations were performed by the same person to avoid inconsistencies. All songs in this dataset have both singing and instrumental parts, i.e., none is purely instrumental nor a cappella⁴.

We used HMM with tied Gaussian mixtures for singing voice detection, which basically classifies an IBI into two categories of “vocal” and “instrumental”. All audio files are 22050 Hz-sampled, and 26 MFCCs are extracted from each frame of 256 samples, with 50% overlap. We changed the numbers of states and mixtures to obtain the performance based on 5-fold cross validation, as shown in Figure 6.3. From the figure, we can see that the accuracy of singing voice detection approximately increases with the number of mixtures in each HMM state. On the other hand, the number of HMM states does not seem to affect the accuracy in an obvious manner. Since HMM with 4 states and 100 mixtures achieves the highest accuracy, we adopt the settings for phrase detection.

To evaluate the performance of phrase detection, we compare our method with three publicly available systems for song segmentation, including the Dynamic Texture Model based approach (DTM) [65], the sparse Shift-Invariant Probabilistic Latent Component analysis based approach (SI-PLCA) [66], and the EchoNest audio analysis tool⁵. The

¹The main reason that some settings we did not test is that it takes too much memory to proceed experiments with them.

²This method only segments tracks, did not predict the segments are vocal or instrumental.

³Song names and URLs are listed at: <http://www.cmlab.csie.ntu.edu.tw/~known/medley/EnglishSongsDataset.html>.

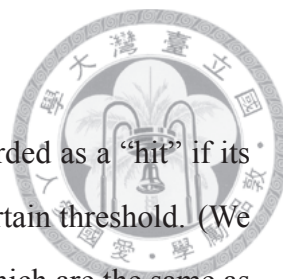
⁴“A cappella”: choral music without instrumental accompaniment [13]

⁵<http://echonest.github.io/remix/apidocs/echonest.remix.audio.AudioAnalysis-class.html>

Method	Phrase Detection Results										
	Median (sec)		0.5s hit (%)			3s hit (%)			IBI accuracy (%)		
	G-to-T	T-to-G	Pre.	Rec.	F-meas.	Pre.	Rec.	F-meas.	Pre.	Rec.	F-meas.
GT	1.14	0.00	42.30	87.28	56.98	67.81	94.52	78.97	67.81	94.52	78.97
GT-beatSync	1.17	0.13	41.15	84.08	55.26	67.38	92.89	78.10	67.38	92.89	78.10
GT-beatSync-hybrid	0.17	0.15	67.20	74.18	70.52	76.49	83.92	80.04	76.49	83.92	80.04
HMM-raw	3.44	0.26	13.81	73.01	23.22	45.63	92.05	61.01	45.63	92.05	61.01
HMM-median [5]	2.56	2.74	24.73	30.22	27.20	56.51	53.00	54.69	56.51	53.00	54.69
HMM-hybrid (Proposed)	1.91	2.07	37.20	37.13	37.16	58.95	57.45	58.19	58.95	57.45	58.19
DTM [65]	2.40	4.96	22.40	17.71	19.78	59.76	43.00	50.02	59.76	43.00	50.02
SI-PLCA [66]	2.36	5.98	27.29	14.65	19.07	57.72	33.66	42.52	57.72	33.66	42.52
EchoNest	2.26	4.50	22.87	15.68	18.60	57.75	39.80	47.12	57.75	39.80	47.12

Table 6.1: The phrase detection results.





evaluation is based on several different metrics:

- *Precision, recall and F-measure*: A detected boundary is regarded as a “hit” if its time difference from the nearest true boundary lies within a certain threshold. (We used two thresholds of 0.5 and 3 seconds in our experiment, which are the same as in [65].)
- *Medians of Guess-to-true (G-to-T) and true-to-guess (T-to-G)*: The median of the time differences between detected boundaries and the closest true ones, and the median of the time differences between true boundaries and the closest detected ones, respectively, as defined in [65]. (The median is computed over all songs.)
- *IBI accuracy*: The percentage of IBIs that are correctly labeled as vocal/instrumental phrases.

The experimental results are presented in [Table 6.1](#), where the rows can be grouped into 3 parts:

- The upper part of the table presents the results of phrase detection that used ground-truth singing voice annotations as reference, including GT, GT-BeatSync, and GT-BeatSync-hybrid.
 - *GT*: The boundaries of singing voice annotations were used directly as the predicted phrase boundaries.
 - *GT-BeatSync*: The GT boundaries are aligned to the beat locations detected by BeatRoot [37].
 - *GT-BeatSync-hybrid*: The “hybrid grouping”(Algorithm 1) is applied to the boundaries of GT-BeatSync.

As shown in the table, the performance difference between GT and GT-beatSync is quite small, indicating the alignment operation is not a critical factor. In contrast, the performance difference between GT-beatSync and GT-beatSync-hybrid is significant, indicating that the “hybrid grouping” is an effective operation for improving the performance. Notice that the performance of GT-beatSync-hybrid should be

viewed as the upper bound of the proposed phrase detection since the method uses the GT for singing voice detection.



- The middle part of the table shows the phrase detection results based on HMM-based singing voice detection: HMM-raw, HMM-median, and HMM-hybrid.
 - *HMM-raw*: consecutive vocal/instrumental IBIs are regarded as vocal/instrumental phrases.
 - *HMM-median* [5]: a median filter is applied to the results of HMM-raw.
 - *HMM-hybrid* [6]: the proposed “hybrid grouping” is applied to the results of HMM-median.

As can be seen from the table, the HMM-hybrid method generally outperforms HMM-raw and HMM-median, demonstrating the effectiveness of “hybrid grouping”.

- The bottom 3 rows of the table show the results of the three publicly available segmentation systems. As shown in the table, the proposed method outperforms these three systems in most of the metrics. Since DTM, SI-PLCA, and EchoNest tend to identify less boundaries than ours, their precision are generally higher than the recall. Different values of the parameters (such as the length of the median filter used to prevent over-segmentation) may also influence the trade-off between precision and recall. In this study, our primary goal of segmentation is to avoid interruption in the middle of a phrase when concatenating songs. Hence a higher precision is preferable. However, if too few boundaries are detected (low recall), then the clip will become too long which makes the medley boring. Thus, we still need to strike a balance between precision and recall.

6.1.4 Comparison Between Similarity-based and Phrase-based Transition Segments Locating Methods



This experiment aims at comparing the similarity-based (Section 4.1.1) and phrase-based (Section 4.1.2) transition segments locating methods. Six pairs of medleys with two different settings for crossfading are used in this experiment. Each pair contains two medleys, and each medley is composed of two song clips. Three of them are in the format of “vocal+vocal”, two are “vocal+instrumental”, and one is “instrumental+vocal”. The two song clips used in the medleys of a test pair are the same. For one medley in each pair, the transition, based on the phrase-based method, happens immediately at the end of a phrase of the first clip. (We extended the clips by x IBIs if a $2x$ -IBIs crossfade is desired, for the alignment of phrase boundaries of the successive clips.) In order to avoid the potential bias due to inaccurate segmentation, human-labeled annotations were used to indicate musical phrase boundaries. For the other medley, the transition, based on the similarity-based method, could be anywhere in the middle of the clips such that the short-term chroma features of two clips are best matched. (We extended the clips by 4 IBIs at both ends first.) In Section 6.1.1 we have found that crossfading with 4 IBIs outperforms longer crossfading durations. Thus, we adopted crossfading durations of 1 and 4 IBIs to explore their perceptive differences in this experiment. For counterbalance, half of the participants evaluated the 1 IBI setting first and the other half evaluated the 4 IBIs first. 25 participants were invited to listen to the 6 medley pairs and assess them in terms of the transition smoothness between two adjacent clips. The questions were designed using a 7-point Likert scale [67]. The higher the score, the smoother the evaluators perceived the transition to be.

The average scores of the six test sets concatenated with 1-IBI crossfades and 4-IBI crossfades are shown in Figure 6.4. At the 1-IBI crossfade duration setting, the average score of the medleys generated by the phrase-based method is significantly higher than that created by the similarity-based method in every set with a confidence level of 95%. As in 4-IBI setting, the mean scores of the medleys concatenated by the phrase-based approach are significantly higher than that of the medleys concatenated by the similarity

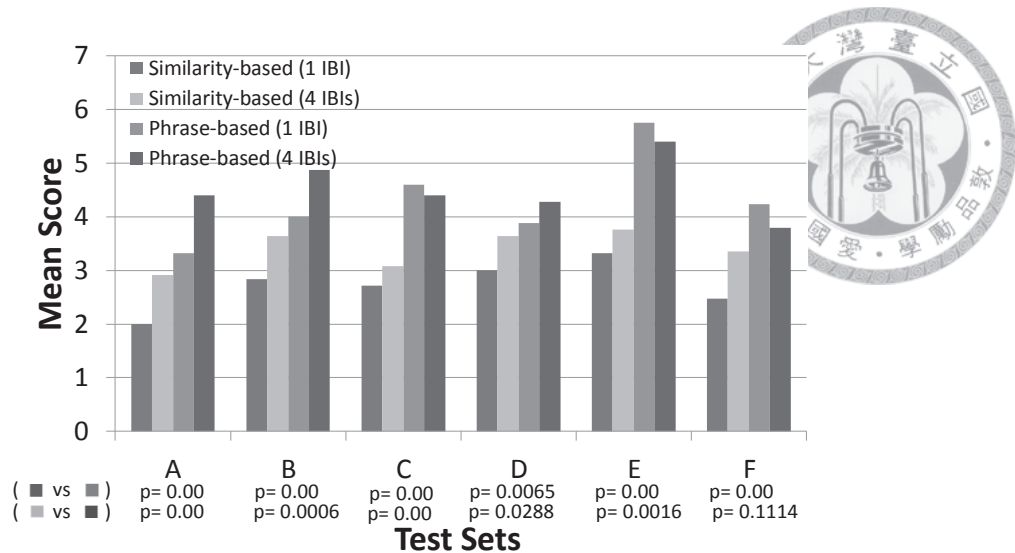


Figure 6.4: Results of user evaluation for clip concatenation with 1-IBI and 4-IBI crossfades, in which the relevant p-values of pairwise t-tests between phrase-based and similarity-based methods are displayed under the corresponding bars of each experiment.

approach in 5 out of 6 test sets.

In conclusion, no matter how long the duration of the crossfade applied to concatenate two clips, participants found the medleys generated with the phrase-based approach, i.e., concatenating clips at phrase boundaries, will be more pleasant. In addition, the comparison between medleys concatenated with crossfades of different durations also shows that users prefer longer crossfades over short ones when similarity-based method is used, while no clear user preference was found for crossfade durations when phrase-based method is used to concatenate the clips. Finally, segmenting songs into clips according to their musical phrases helps retain the characteristics of the concatenated clips, and thus results in smoother transitions between clips.

6.1.5 The Just Noticeable Difference of Tempo

In this section, we will describe the details about finding the just noticeable difference of tempo for the tempo adjustment component in Section 4.2.1. Inspired from [59], two groups of songs are used: fast tempi (120 ~ 180 BPM) and slow tempi (40 ~ 90 BPM). Each group contains 6 Chinese pop songs of different types. Each song has been adjusted to have two different tempi. 25 evaluators are invited to judge if they could tell apart the tempi in the two samples of the same song. The corresponding results are illustrated in

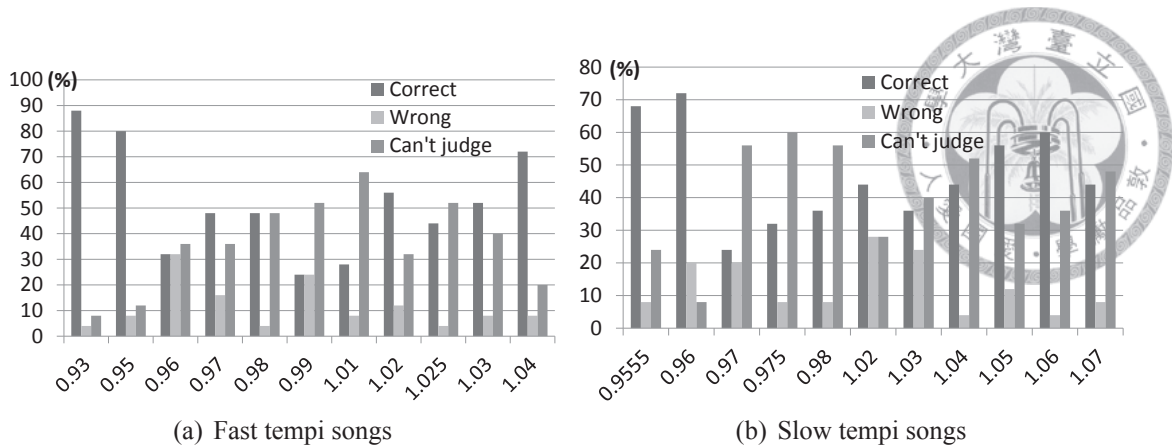


Figure 6.5: Percentages of evaluators who can recognized the tempi difference of the samples

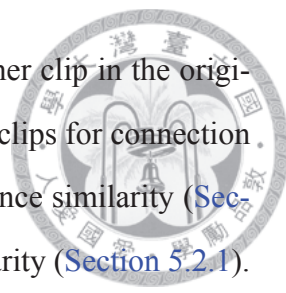
Figure 6.5. The bars of each set represent the percentages of the users who recognize the changes of tempi and judge them correctly, wrongly, or cannot tell apart, respectively. The value on the horizontal axis denotes the tempo ratio between the two samples of the same song. From Figure 6.5, we can infer that users may not notice the change of tempo when the tempo ratio is close to unity since the third bar (evaluators cannot judge) reaches high values. For quick (i.e. fast tempo) music clips, we found out that the ratio of the tempo from 0.95 to 1.03 will not be perceived. For slow music clips, the non-perceivable JND range is from 0.96 to 1.04.

6.1.6 Effectiveness Bar Alignment and Dual Tempo Adjustment

This experiment aims to verify the effectiveness of connecting clips with bar alignment (c.f. Section 4.1.3) and dual tempo adjustment (c.f. Section 4.2.2). As a result, we compared the concatenated clips using bar alignment and dual tempo adjustment (denoted as BD method) with those did not (denoted as PT method), that is, just connected at phrase boundary (c.f. Section 4.1.2) and use normal tempo adjustment (c.f. Section 4.2.1). Beside, another Echonest⁶ method was also compared as a reference. Echonest connects clips at timbre similar positions, matches beats, and adjusts tempi linearly from clip to clip, and does take dual tempo adjustment into consideration.

To systematically analyze the effectiveness of connecting methods, we proposed to use

⁶<https://github.com/echonest/remix/tree/master/examples/capsule>



the similarity between “the latter clip” and “the phrase after the former clip in the original song” as the metric to represent the suitability of the consecutive clips for connection. (c.f. [Figure 6.6](#)). Three kinds of similarity are used, the chord sequence similarity ([Section 5.2.1](#)), the timbre similarity ([Section 5.2.1](#)), and the tempo similarity ([Section 5.2.1](#)). Based on these metrics, we can measure the performance of different connecting methods on the clip pairs with various connecting-suitability. To choose the clip pairs that cover various suitability, we first compute the three types of similarity on each pair of phrases in the dataset ⁷. Then, we divide all the computed similarity values into 3 groups: the highest 30 % similar, the middle, and the lowest 30 % similar. After that, total 20 different types of music clip pairs are chosen, according to the relation between the two clips. The 20 types are composed of 5 similarity types multiplied by 4 different timbre types. The 5 similarity types consist of the clip pairs that their similarity characteristics between “the latter clip” and “the phrase after the former clip in the original song” belong to one of the following situations:

- LLL: low similarity⁸ in all dimensions (chord, timbre, tempo).
- LLH: low chord and timbre similarity, but high tempo similarity.
- LHL: low chord and tempo similarity, but high timbre similarity.
- HLL: low timbre and tempo similarity, but high chord similarity.
- HHH: high similarity in all dimensions.

The 4 timbre types are:

- V-V: both of the clips are vocal.
- V-I: the former clip is vocal, but the latter clip is instrumental.
- I-V: the former clip is instrumental, but the latter clip is vocal.
- I-I: both of the clips are instrumental.

⁷the same as the set we use in [Section 6.1.3](#)

⁸in the the lowest 30 % of all the similarity values

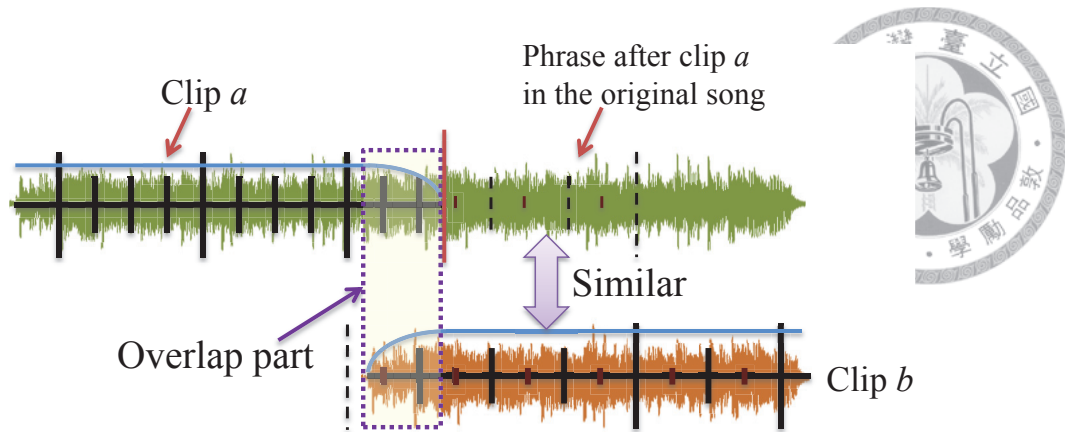


Figure 6.6: We use the similarity between “the latter clip” and “the phrase after the former clip in the original song” as the metric to measure the suitability of the consecutive clips for connection.

The former clips will be fixed for the five similarity types to reduce the influence of variation. We then randomly choose 3 clip pairs from each one of the clip pair types, that is, there are 60 clip pairs in total.

For each one of the 60 chosen clip pairs, we connect them based on the above-mentioned 3 methods, and there are 168 resulting connected clips⁹ in total. We divide the connected clips into 12 groups, each group contains 9 ~ 15 connected clips of 3 ~ 5 different similarity types and 1 timbre type. The former clips in the connected samples are the same for each group. Our user evaluations are performed through the aid of a web interface, and the tested clips are presented in random order. Users are invited to listen to one group of clips per time, taking about 10 minutes to finish each test. The questions are designed using a 7-point Likert scale [67]—users are asked to report their opinions of the connected clips from the following options: very pleasing, pleasing, somewhat pleasing, neutral, not so pleasing, not pleasing, and very displeasing.

46 males and 11 females, aged around 20 to 40, participated in this experiment. Each user involved 1 to 2 groups, and each test sample was listened by 5 different people. Figure 6.7 shows the mean scores of each one of the connecting methods with all tested clips and with the test clips of each similarity type only. The paired Wilcoxon signed rank test is applied to analyze the results. The corresponding p-values are reported in Figure 6.7, each line reported the p-values of “BD vs. PT”, “BD vs. Echonest”, and

⁹Some clip pairs contains too short phrases so that the Echonest method cannot produce the result. So we remove them.

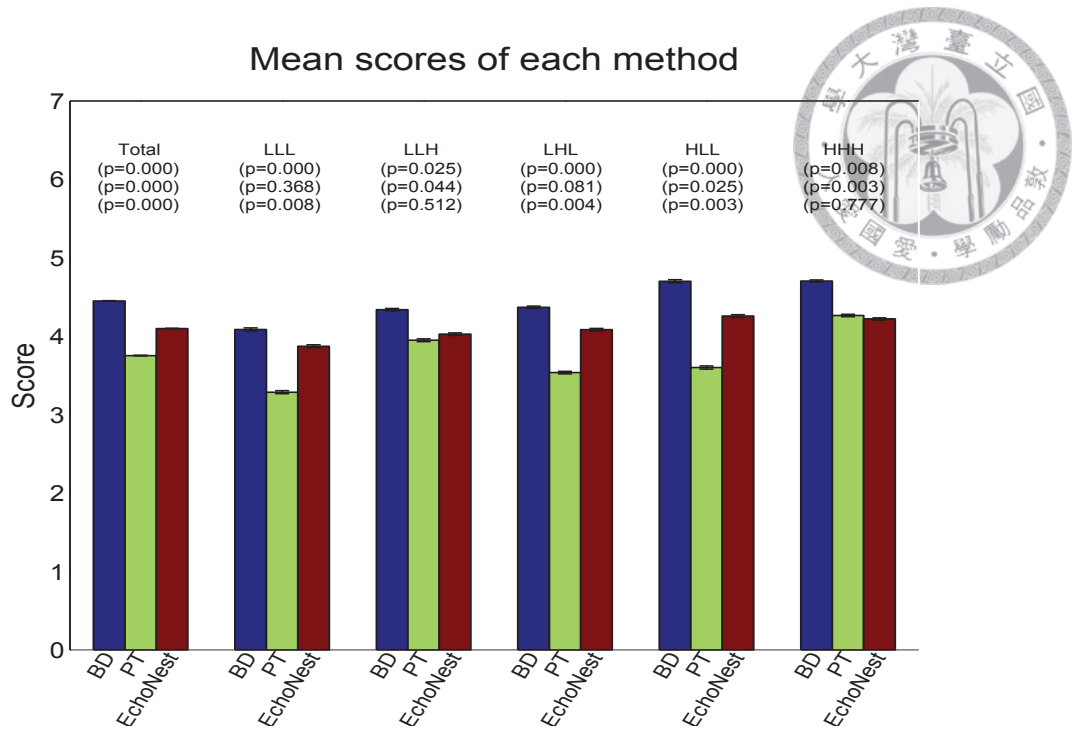


Figure 6.7: Mean scores of each one of the temporal adjustment methods for total data and for each similarity type of the test samples, in which the relevant p-values of paired Wilcoxon signed rank test on “BD vs. PT”, “BD vs. Echonest”, and “PT vs. Echonest” are displayed above the corresponding bars of each experiment, respectively.

“PT vs. Echonest”, respectively. The overall result shows that BD method did improve the smoothness of clip concatenation under a confidence level of 95%. For those low-tempo-similar (LLL, LHL, HLL) clips, BD method and Echonest’s approach out perform the PT method, which shows that dual tempo adjustment is relatively more important in tempo-dissimilar cases. In those high-tempo-similar (LLH, HHH) cases, the mean score of the BD method is higher than those of both PT and Echonest, which indicates that bar alignment did improve the temporal adjustment since, now, it is no need to apply dual tempo adjustment to high-tempo-similar clips.

6.2 Evaluations on Selection Schemes

6.2.1 Effectiveness of Clustering Criteria

This experiment verifies if the proposed clustering criteria (Section 5.2.1) can put similar clips into the same cluster, and if clips in the same cluster are mutually interchangeable. We used 5 sets of medleys of the form “vocal+vocal” to perform subjective test, where

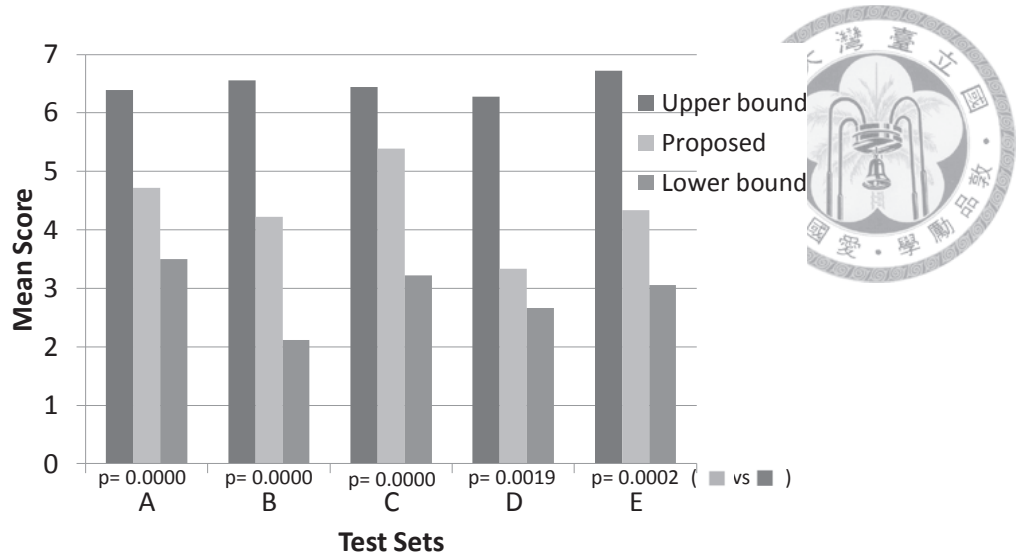


Figure 6.8: Results of user evaluation on clip selection with the proposed clustering criteria, in which the relevant p-value of pairwise t-test of the proposed and the lower bound methods is displayed under the corresponding bars of each experiment.

2 of them are male artists while 3 are female. Each set contains 3 medleys, and each medley is composed of 2 clips only. Within each set, the first clips of the 3 medleys are the same, while the second clips were obtained via three different methods. More specifically, suppose that each medley $M_i, i = 1, 2, 3$ in a set can be expressed as the concatenation of two song clips $[a_i \text{ and } b_i]$, then the selection of b_i is based on the following methods:

- *The upper bound:* b_1 is the original clip that follows the first clip in the original song. This selection serves as the “upper bound”, or the best selection.
- *The proposed method:* b_2 is randomly selected from the cluster containing b_1 (but not b_1 itself).
- *The lower bound:* b_3 is randomly selected from the cluster that is the least similar to the cluster containing b_1 . This selection serves as the “lower bound” to check if the similarity score computed by the proposed method conforms to the interchangeability perceived by humans.

These three methods can be put into mathematical notations as follows:

$$\begin{cases} b_1 = N(a), \\ b_2 \in C(N(a)), \\ b_3 \in C(d), \quad d = \arg \max_{c, c \in \{\text{all clips}\}} D(c, N(a)), \end{cases} \quad (6.1)$$



where $N(a)$ is the clip that immediately comes after clip a in the original song, $C(a)$ is the cluster containing clip a , and $D(a, b)$ is the distance between clips a and b , as we proposed in [Section 5.2.1](#).

We invited 36 participants to listen to the medleys (multiple times if they preferred) and score each medley based on its subjective appeal. The three medleys in the test sets were randomly ordered, and all questions were designed using a 7-point Likert scale [67]. The higher the score, the more cohesive and pleasant the medley is (based on the participant’s perception). Participants were also asked to indicate whether they were familiar with the song from which the first clips were taken.

[Figure 6.8](#) shows the mean score of each set for this subjective test. We performed a pairwise t-test to analyze the results. Overall, the medleys composed of clips selected with the proposed method significantly outscored the medleys composed of “lower-bound” clips under a confidence level of 95%. In addition, the medleys composed of clips selected with the proposed method achieved an average score of 4.97 out of 7 points, while the medleys composed of “lower-bound” clips achieved an average of only 3.05. When 5 test sets were evaluated individually, we found that participants were satisfied with the medleys composed by the proposed method in test sets A, B, C, and E, with a score above 4. The result demonstrates that the proposed method is capable of putting similar and interchangeable clips in the same cluster.

6.2.2 Effectiveness of Path Finding

This experiment aims to verify the proposed path finding-scheme (based on the Viterbi algorithm, c.f. [Section 5.2.2](#)) which finds an optimum path with given constraints in a musical dice graph. We used 5 pairs of medleys for subjective test. The number of transi-

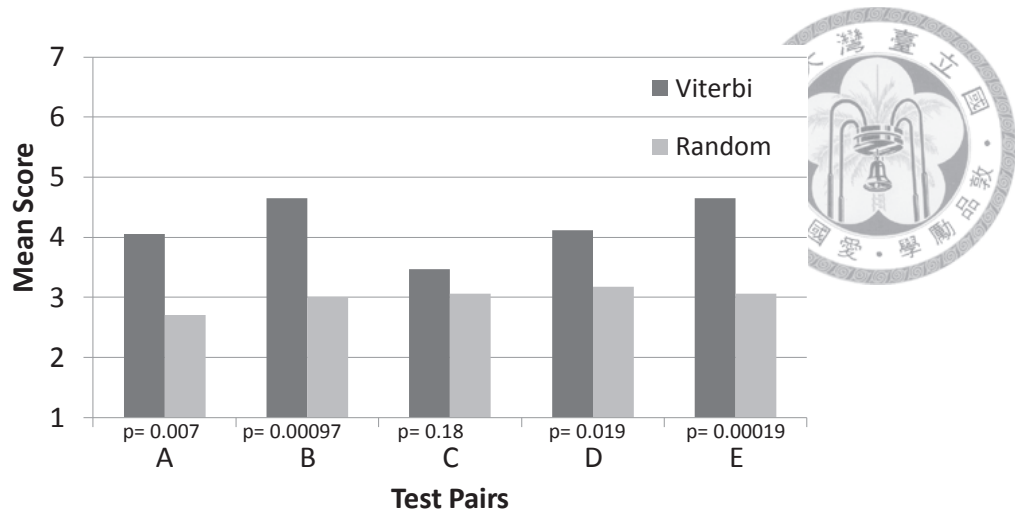


Figure 6.9: Results of user evaluation on the proposed path-finding scheme based on the Viterbi algorithm, in which the relevant p-value of pairwise t-test is displayed under the corresponding bars of each experiment.

tions and the structures of the two medleys in a pair are the same, and the beginning and the ending clips of the medleys are the same as well. For one medley in each pair, the chosen clusters (from which the clips are randomly selected) and their order are decided by the proposed Viterbi-like path finding. For the other medley, clips are selected randomly. We invited 17 participants to evaluate the generated medleys. Each pair of medleys was played twice.

Figure 6.9 illustrates the score of each test pair. Overall, the mean score of medleys constructed by the proposed path-finding scheme is significantly higher than that of the one based on random selection, with a confidence level of 95%. If the 5 test sets are evaluated separately, the mean score of medleys constructed by the proposed path-finding scheme is significantly higher than that of the randomly generated ones in 4 out of 5 test pairs (i.e., pairs A, B, D, and E, with $p < 0.05$). The result shows that the proposed path-finding scheme based on the Viterbi algorithm is effective in selecting clips that sound pleasant when they are concatenated and played in sequence.

6.3 Overall Performance

In this section, we compare the overall performance of our two projects: “music paste” [4] and “audio musical dice game” [6]. In music paste [4], two clips are concatenated at the

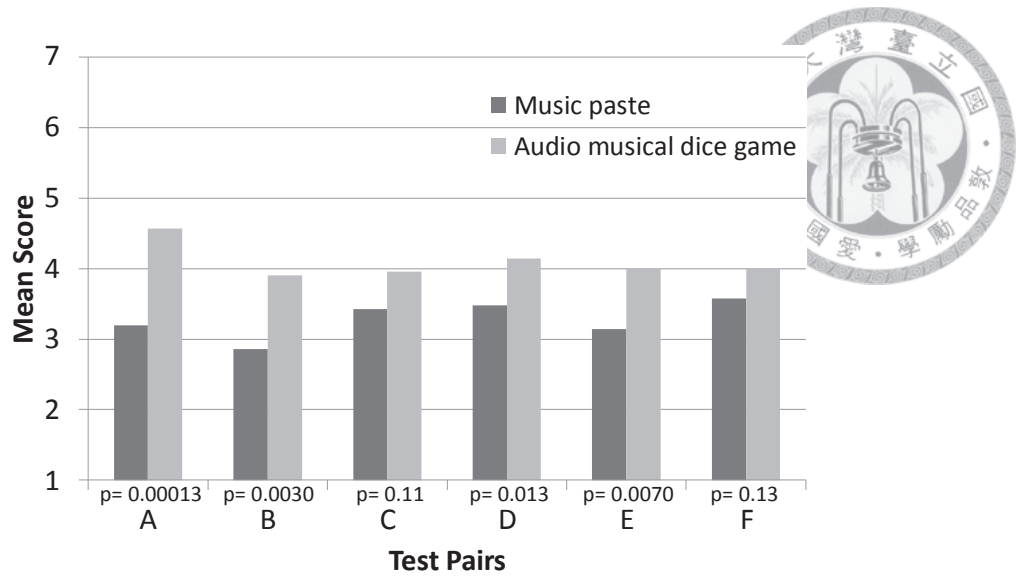
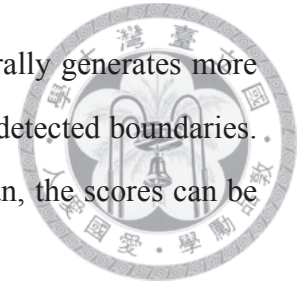


Figure 6.10: Results of user evaluation on the overall performance of “audio musical dice game” [6] when compared with “music paste” [4], in which the relevant p-value of pairwise t-test is displayed under the corresponding bars of each experiment.

position where the chroma vectors are the most similar (c.f. Section 4.1.1). For each music clip, its following clip is selected by picking the one with the highest similarity value at the connecting position (Section 5.1.2). In audio musical dice game [6], clips are concatenate at phrase boundaries (c.f. Section 4.1.2), with volume normalized (c.f. Section 4.3.1), the order of the clips are determine by path finding (c.f. Section 5.2.2). In this experiment, we follow similar settings of the experiment in Section 6.2.2, that is, the structures of the two medleys in a test pair are the same, but here only the first slots are specified. Besides, we use detected phrases as the unit for medley creation. We also set the range of duration of each chosen clips to be 10 ~ 30 seconds to avoid bad segmentation. 21 people are invited to evaluate the overall pleasantness of the medleys. The participants are allowed to listen to the medleys multiple times if they preferred.

Figure 6.10 illustrates the mean score of each test pair generated by “music paste” and “audio musical dice game”. In general, the mean score (4.17) of medleys constructed by “audio musical dice game” is significantly higher than that (3.33) of “music paste”, with a confidence level of 95% tested via pairwise t-test. In the 6 test pairs, the mean scores of “audio musical dice game” are significantly higher than “music paste” in 4 out of 6 test pairs (i.e., pairs A, B, D, and E, with $p < 0.05$). Due to audible phrase boundary errors, the mean scores of test pairs C and E are not significantly higher than that of “music paste”.

These results indicate that “audio musical dice game” system generally generates more satisfactory medleys than the approach in “music paste”, even with detected boundaries. If the detected phrase boundaries can be slightly corrected by human, the scores can be improved further.



6.4 Discussion

This section discusses the observations in our experiments.

6.4.1 The Influence of Accompanied with Visual Content

To investigate the influence of accompanied visual contents on the quality of medleys, we conduct two subjective tests.

In the first test, users listened to each test sample twice. At the first time, they just listened, but at the second time, they were informed the position of transition (by viewing labels) while listening. 12 evaluators are invited to grade their satisfactions from 1 to 10 points, and answered the number of transitions they recognized. Three test samples composed of Chinese pop songs are used. [Figure 6.11](#) illustrates of the percentages of how many evaluators scored lower, the same, and higher when they listen to the samples at the 2nd time, respectively. Over 60% of the evaluators rated the same score after knowing the position of the transition points.

In the second test, a medley is played with and without photo slideshows generated by [1], respectively. Twelve evaluators attended the test. Over 90% of evaluators think that the medleys are more euphonious after playing with tiling slideshow.

To sum up, we may infer that knowing the position of the transition points did not affect the satisfactory of the medleys but visual contents may still distract the users from the intrusions in the medleys.

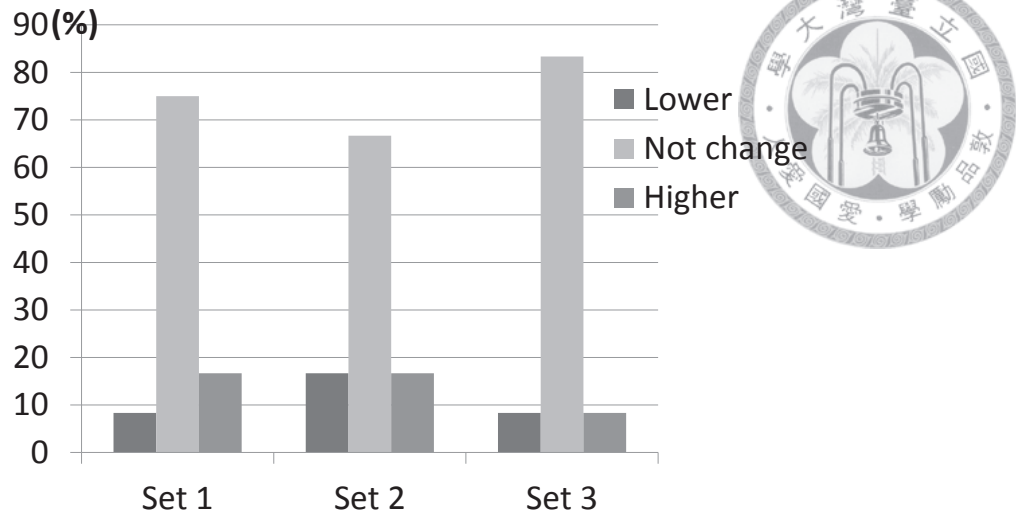


Figure 6.11: Percentages of the how many evaluators scored lower, the same, and higher when they listen to the sample at the 2nd time, respectively

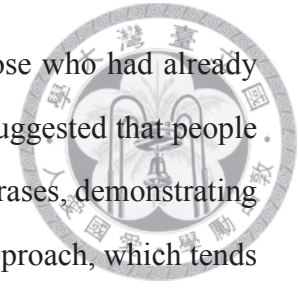
6.4.2 The Influence of User Familiarity with the Songs

In the experiments conducted in [Section 6.1.4](#) and [Section 6.2.1](#), users were asked to reveal whether they were familiar with the song clips used to compose the medleys in each test set.

From the results of [Section 6.2.1](#), we found that participants who had heard of the first of the two clips in a medley gave significantly higher scores to the “upper bound” test medleys, compared with those who had not. This was probably due to the well-known “mere-exposure effect” [68] from cognitive psychology – people tend to show a positive inclination for the familiarity. Therefore, users who are familiar with the songs may prefer the “upper bound” group, which consists of song clips selected as they appeared in the original song. On the other hand, “lower bound” and “proposed” test medleys received lower scores from those who had heard the songs before. This can also be explained by the mere-exposure effect since participants who knew the first clip often held higher expectations of the original following clip in the song, and took stricter criteria in evaluating the replaced second clip.

We also noticed that in the experiment of [Section 6.1.4](#), participants who had not heard either of the clips used in the medleys gave higher scores to the medleys concatenated with similarity-based transition segment approach (c.f. [Section 4.1.1](#)) than those who were already familiar with the clips. On the other hand, medleys concatenated with the phrased-

based approach (c.f. [Section 4.1.2](#)) received higher scores from those who had already heard the original songs than those who had not. This observation suggested that people who knew the songs may be relatively more sensitive to the music phrases, demonstrating a stronger preference for medleys concatenated with the proposed approach, which tends to retain complete musical phrases.



6.4.3 Other Criteria that Might Contribute to Better Clip Selection

In the experiment of [Section 6.2.2](#), we asked participants to suggest some possible factors that might influence the perceived smoothness of the transition between clips in the composed medleys. The most frequently given factor was the volume of the clips (mentioned by 11 out of 17 participants). Other frequently factors include timbre, tempo/rhythm, key and genre, along with the singer's gender. Some also suggested that the lyrics of the clips influence how well two clips are concatenated. However, it should be noted that the importance of the role these factors play when composing medleys is highly subjective and may vary from person to person. While some people are sensitive to key changes in a song (e.g., people with absolute pitch), others may not consider it as a serious concern. Similarly, most people prefer medleys composed with clips characterized by similar emotions, whereas some like the thrill brought by unexpected emotional transitions between clips. Therefore, a more personalized graph construction scheme based on user preferences as clustering criteria should be introduced in the future to better satisfy different user's needs. For instance, the users should be able to specify the weighting of chords, tempi/rhythms, and timbre.

6.4.4 Comparison with Human Created Medley

As compared to human made medleys, the proposed system cannot handle several skills that are commonly used in expert generated medleys, including:

- *More elaborative excerpts* : The expert may use a unit that is shorter than a musical phrase to generate medleys that sounds more natural. For example, the last phrase

in the Beatles Movie Medley (3'48.37 ~ end) is a combination of the first half of a phrase (0'50.70 ~ 0'52.44) and the last half of another phrase (2'43.32 ~ 2'54.38) in the song "Get Back".



- *Newly composed tracks* : The expert may compose new music tracks to mix with the concatenated music clips according to the clips' characteristics.

Amateur users may still use these skills, but perhaps not as proficiently as expert users can do.

To make a pleasant medley, the most time consuming step for human is to find appropriate song excerpts to be put together. In the proposed system, we used a statistical way to find suitable excerpts based on the assumption that clip *a* is suitable to connect with clip *b* if clip *b* is similar to the next phrase of clip *a* in the original song. The proposed system may not generate medleys that are as pleasant as those generated by human, but it can surely be a computer-assisted tool for human, which is useful for and effective in suggesting clips for concatenation.





Chapter 7

Conclusions and Future Work

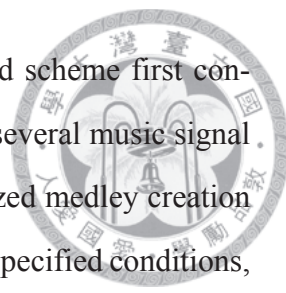
Learn from yesterday, live for today, hope for tomorrow.

The important thing is not to stop questioning.

– Albert Einstein

7.1 Conclusions

In this dissertation, systematic techniques for concatenative audio music re-composition has been developed. The re-composition process is divided into four steps: content analysis, pre-processing, selection, and composition. We have briefly reviewed the content analysis techniques that are useful in audio music re-composition. Based on the content analysis techniques, music theory, and psychoacoustics, various composition and selection schemes have been proposed and investigated. We divide the composition step into three parts, transition segments locating, tempo adjustment, and synthesis. In locating transition segments, three options are proposed: at the most similar positions, at the phrase boundaries, and with bar alignment. In order to find phrase boundary for pre-processing, an approach based on singing voice detection is adopted. Then, psychoacoustics-based tempo adjustment methods are proposed. To handle the cases of distinct tempo and volume, we also examined the corresponding dual tempo adjustment and volume normalization schemes, respectively. For the selection, two schemes are discussed. The straightforward scheme filtered out unfitting clips by pair wise comparison and ordered the rest



clips by similarity values of transition segments. The graph-assisted scheme first constructed a musical dice graph from the pre-processed clips based on several music signal analysis techniques. Then, with the graph, we can provide personalized medley creation service, which generates various pleasing medleys conforming to the specified conditions, such as medley structures or must-use clips. Besides, we also provide a GUI for the users to choose clips, specify parameter and adjust concatenation boundaries. The experiment results have shown the effectiveness of individual components, comparisons among methods, and provided guidelines for users to choose parameters.

7.2 Future Work

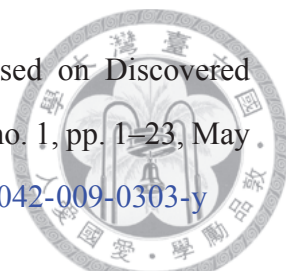
Many aspects of our system can be improved in the future. First, the current clip clustering method can be modelled as an optimization problem, where the parameters of the “mixed distance score” could be set according to user preferences, resulting in customizable clip clustering and selection criterion. More clip similarity measures (e.g., meter, genre, mood, etc.) can also be introduced during the clip clustering phase. Second, cluster types can be extended into sections of different roles in a pop song, such as “first half verse”, “second half chorus”, “bridge”, etc.. The length of each phrase can also be taken into account during clip selection, enabling users to specify the desired lengths of each clip and the overall medley. Third, song segmentation with singing voice detection restricted this work to vocal songs. In the future, we will improve the current phrase detection method and explore other music segmentation methods that are able to recognize phrases in instrumental music. Other learning-based boundary detection methods [69, 70] are also worth exploring. In addition, the current research ignored potential impacts of lyrics and the language in which the songs are performed. Our future studies will explore how these two factors could be used to help with composing lyrically-meaningful medleys. Finally, automatic separation of background music from foreground singing voices may enable us to create and add intermediate bridges, allowing for greater flexibility in medley generation.




Bibliography

- [1] J.-C. Chen, W.-T. Chu, J.-H. Kuo, C.-Y. Weng, and J.-L. Wu, "Tiling slideshow," in *Proc. ACM MM*, 2006, pp. 25–34.
- [2] M. Goto, "A Chorus Section Detection Method for Musical Audio Signals and Its Application to a Music Listening Station," *IEEE Trans. ASLP*, vol. 14, no. 5, pp. 1783–1794, Sep. 2006. [Online]. Available: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=1677997>
- [3] D. Davis, "Principles of Physics I - Chapt 16. Characteristics of Sound," 2002. [Online]. Available: <http://www.ux1.eiu.edu/~cfadd/1150/16Waves/char.html>
- [4] H.-Y. Lin, Y.-T. Lin, M.-C. Tien, and J.-L. Wu, "Music Paste: Concatenating Music Clips Based on Chroma and Rhythm Features," in *Proc. ISMIR*, Kobe, 2009. [Online]. Available: <http://ismir2009.ismir.net/proceedings/PS2-4.pdf>
- [5] I.-T. Liu, Y.-T. Lin, and J.-L. Wu, "Music Cut and Paste: A Personalized Musical Medley Generating System," in *Proc. ISMIR*, Curitiba, PR, Brazil, 2013.
- [6] Y.-T. Lin, I.-T. Liu, J.-S. R. Jang, and J.-L. Wu, "Audio Musical Dice Game: A User-preference-aware Medley Generating System," *ACM TOMM*, 2014.
- [7] P. Hanna, P. Ferraro, and M. Robine, "On Optimizing the Editing Algorithms for Evaluating Similarity Between Monophonic Musical Sequences," *J. New Music Res.*, vol. 36, no. 4, pp. 267–279, 2007. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/09298210801927861>


- 
- [8] D. Cope, *Experiments in Musical Intelligence*. Madison, Wisconsin, USA: A-R Editions, 1996.
- [9] S. Wenger and M. Magnor, “Constrained Example-based Audio Synthesis,” in *Proc. ICME*, Barcelona, Spain, 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6011902
- [10] Z. Liu, C. Wang, J. Wang, H. Wang, and Y. Bai, “Adaptive Music Resizing with Stretching, Cropping and Insertion,” *Multimedia Syst.*, vol. 19, no. 4, pp. 359–380, Jul. 2012. [Online]. Available: <http://www.springerlink.com/index/10.1007/s00530-012-0289-6>
- [11] R. Cole and E. Schwartz. (2012) Virginia Tech Multimedia Music Dictionary. [Online]. Available: <http://www.music.vt.edu/musicdictionary/>
- [12] A. Latham, “The Oxford Companion to Music,” 2011. [Online]. Available: http://www.oxfordmusiconline.com/subscriber/book/omo_t114
- [13] D. M. Randel, *The Harvard Dictionary of Music*. Belknap Press, 2003. [Online]. Available: <http://www.credoreference.com/book/harvdictmusic>
- [14] “iTunes Store Sets New Record with 25 Billion Songs Sold,” *Apple Press Info*. [Online]. Available: <https://www.apple.com/pr/library/2013/02/06iTunes-Store-Sets-New-Record-with-25-Billion-Songs-Sold.html>
- [15] G. Loy, *Musimathics*. USA: The MIT Press, 2006, vol. 1, pp. 295–296, 347–350.
- [16] G. T. Fechner, *Elements of Psychophysics I*. New York: Holt, Rinehart & Winston, 1860.
- [17] Y.-T. Lin, C.-L. Lee, J.-S. Jang, and J.-L. Wu, “Bridging Music via Sound Effects,” in *Proc. IEEE ISM*. (Best Student Paper Award), 2014.
- [18] D. Cope, “Experiments in Music Intelligence,” in *Proc. ICMC*, San Francisco, USA, 1987.


- 
- [19] M.-K. Shan and S.-C. Chiu, “Algorithmic Compositions Based on Discovered Musical Patterns,” *Multimedia Tools and Applications*, vol. 46, no. 1, pp. 1–23, May 2010. [Online]. Available: <http://link.springer.com/10.1007/s11042-009-0303-y>
- [20] D. Schwarz, “Corpus-based Concatenative Synthesis,” *Signal Processing Magazine, IEEE*, vol. 24, no. 2, pp. 92–104, 2007. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4117932
- [21] —, “Current Research in Concatenative Sound Synthesis,” in *Proc. ICMC*, Barcelona, Spain, 2005. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.60.8530&rep=rep1&type=pdf>
- [22] R. B. Dannenberg, “Concatenative Synthesis Using Score-Aligned Transcriptions Music Analysis and Segmentation,” in *Proc. ICMC*, New Orleans, USA, 2006, pp. 352–355.
- [23] D. Schwarz, R. Cahen, and S. Britton, “Principles and Applications of Interactive Corpus Based Concatenative Synthesis,” *Journées d’Informatique Musicale (JIM)*, 2008.
- [24] G. Bernardes, C. Guedes, and B. Pennycook, “EarGram : an Application for Interactive Exploration of Large Databases of Audio Snippets for Creative Purposes,” in *Proc. CMMR*, London, UK, 2012, pp. 19–22.
- [25] R. Kobayashi, “Sound Clustering Synthesis Using Spectral Data,” in *Proc. ICMC*, Singapore, 2003. [Online]. Available: <http://nagasm.org/ASL/icmc2003/closed/CR1052.PDF>
- [26] G. Griffin, Y. Kim, and D. Turnbull, “Beat-Sync-Mash-Coder: a Web Application for Real-Time Creation of Beat-Synchronous Music Mashups,” in *Proc. ICASSP*, Dallas, Texas, USA, 2010, pp. 2–5. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5495743
- [27] M. E. P. Davies, P. Hamel, K. Yoshii, and M. Goto, “AutoMashUpper : An Automatic Multi-Song Mashup System,” in *Proc. ISMIR*, Curitiba, PR, Brazil, 2013.


- 
- [28] B. Logan, “Content-based Playlist Generation: Exploratory Experiments,” in *Proc. ISMIR*, Paris, France, 2002, pp. 2–3. [Online]. Available: http://pdf.aminer.org/000/439/408/content_based_playlist_generation_exploratory_experiments.pdf
- [29] A. Flexer, D. Schnitzer, M. Gasser, and G. Widmer, “Playlist Generation Using Start and End Songs,” in *Proc. ISMIR*, Philadelphia, USA, Sep. 2008, pp. 173–178. [Online]. Available: http://ismir2008.ismir.net/papers/ISMIR2008_143.pdf
- [30] L. Chiarandini, M. Zanoni, and A. Sarti, “A System for Dynamic Playlist Generation Driven by Multimodal Control Signals and Descriptors,” in *Proc. MMSP*, Hangzhou, China, 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6093850
- [31] Q. Lin, L. Lu, C. Weare, and F. Seide, “Music Rhythm Characterization with Application to Workout-Mix Generation,” in *Proc. ICASSP*, Dallas, Texas, USA, 2010, pp. 69–72. [Online]. Available: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5496203>
- [32] C. Baccigalupo and E. Plaza, “Case-based Sequential Ordering of Songs for Playlist Recommendation,” *LNCS*, vol. 4106, pp. 286–300, 2006. [Online]. Available: http://link.springer.com/chapter/10.1007/11805816_22
- [33] T. Jehan, “Creating Music by Listening,” PhD Dissertation, Massachusetts Institute of Technology, 2005.
- [34] S. Basu, “Mixing with Mozart,” in *Proc. ICMC*, Miami, USA, 2004. [Online]. Available: http://research.microsoft.com/en-us/um/people/sumitb/papers/MixingWithMozart_icmc2004.pdf
- [35] D. Cliff, “Hang the DJ : Automatic Sequencing and Seamless Mixing of Dance-Music Tracks,” HP Labs Technical Report, Tech. Rep., 2000.
- [36] H. Ishizaki, K. Hoashi, and Y. Takishima, “Full-Automatic DJ Mixing System with Optimal Tempo Adjustment based on Measurement Function of User Discomfort,”

- in *Proc. of ISMIR*, Kobe, Japan, 2009, pp. 135–140. [Online]. Available: <http://ismir2009.ismir.net/proceedings/PS1-14.pdf>
- [37] S. Dixon, “Evaluation of the Audio Beat Tracking System BeatRoot,” *J. New Music Res.*, vol. 36, no. 1, pp. 39–50, 2007. [Online]. Available: <http://www.elec.qmul.ac.uk/people/simond/beatroot/>
- [38] J. P. Bello, L. Daudet, S. A. Abdallah, C. Duxbury, M. Davies, and M. B. Sandler, “A Tutorial on Onset Detection in Music Signals,” *IEEE Trans. ASLP*, vol. 13, no. 5, pp. 1035–1047, 2005.
- [39] M. Kennedy and J. Bourne, “The Oxford Dictionary of Music.” [Online]. Available: http://www.oxfordmusiconline.com/subscriber/book/omo_t237
- [40] E. Pampalk, “Computational Models of Music Similarity and Their Application in Music Information Retrieval,” Ph.D., Vienna University of Technology, 2006. [Online]. Available: <http://www.pampalk.at/publications/presentations/sigmus06similarity.pdf>
- [41] T. Pohle, D. Schnitzer, M. Schedl, P. Knees, and G. Widmer, “On Rhythm and General Music Similarity,” in *Proc. ISMIR*, no. Ismir, Kobe, Japan, 2009, pp. 525–530. [Online]. Available: <http://ismir2009.ismir.net/proceedings/OS6-1.pdf>
- [42] M. Cicconet. Rhythm Features. [Online]. Available: <http://w3.impa.br/~cicconet/cursos/ae/spmirPresentation.html>
- [43] J. C. Brown and M. S. Puckette, “An Efficient Algorithm for the Calculation of a Constant Q Transform,” *Journal of the Acoustical Society of America*, vol. 92, no. 5, pp. 2698–2701, 1992.
- [44] D. Root, P. V. Bohlman, J. Cross, H. Meconi, and J. H. Roberts, “Grove Music Online,” 2014. [Online]. Available: http://www.oxfordmusiconline.com/subscriber/book/omo_gmo



- 
- [45] Y. Ni, M. McVicar, P. Santos-Rodriguez, and T. D. Bie, “An End-to-End Machine Learning System for Harmonic Analysis of Music,” *IEEE Trans. ASLP*, vol. 20, no. 6, pp. 1771–1783, 2012. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6155600
- [46] S. B. Davis and P. Mermelstein, “Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences,” *IEEE Trans. ASLP*, vol. 28, no. 4, pp. 357–366, 1980.
- [47] M. Cooper and J. Foote, “Automatic Music Summarization via Similarity Analysis,” in *Proc. ISMIR*, Paris, France, 2002, pp. 81–85.
- [48] S. Webber, *DJ Skills: The Essential Guide to Mixing and Scratching*. Focal Press, 2007.
- [49] C. Burkhart, “The Phrase Rhythm of Chopin’s A-Flat Major Mazurka, Op. 59, No. 2,” in *Engaging Music: Essays in Music Analysis*, D. J. Stein, Ed. Oxford University Press, 2005, pp. 3–12.
- [50] J. Paulus, M. Müller, and A. Klapuri, “Audio-based Music Structure Analysis,” in *Proc. ISMIR*, Utrecht, Netherlands, 2010, pp. 625–636.
- [51] J. Foote, “Visualizing Music and Audio Using Self-Similarity,” in *Proc. ACM MM*, 1999, pp. 77–80. [Online]. Available: <http://portal.acm.org/citation.cfm?doid=319463.319472>
- [52] B. Logan and S. Chu, “Music Summarization Using Key Phrases,” in *Proc. ICASSP*, vol. 2, Istanbul, Turkey, 2000, pp. 749–752. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=859068
- [53] J. Pauwels, F. Kaiser, and G. Peeters, “Combining Harmony-based and Novelty-based Approaches for Structural Segmentation,” in *Proc. ISMIR*, Curitiba, PR, Brazil, 2013.

- 
- [54] T. L. Nwe, A. Shenoy, and Y. Wang, “Singing Voice Detection in Popular Music,” in *Proc. ACM MM*, 2004, pp. 324–327. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1027602>
- [55] N. C. Maddage, C. Xu, M. S. Kankanhalli, and X. Shao, “Content-based Music Structure Analysis with Applications to Music Semantics Understanding,” in *Proc. ACM MM*, 2004, pp. 112–119. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1027549>
- [56] L. Regnier and G. Peeters, “Singing Voice Detection in Music Tracks Using Direct Voice Vibrato Detection,” in *Proc. ICASSP*, Taipei, Taiwan, 2009, pp. 1685–1688. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4959926
- [57] Y. Li and D. Wang, “Separation of Singing Voice From Music Accompaniment for Monaural Recordings,” *IEEE Trans. ASLP*, vol. 15, no. 4, pp. 1475–1487, 2007.
- [58] I. Leonidas and J.-L. Rouas, “Exploiting Semantic Content for Singing Voice Detection,” in *Proc. IEEE ICSC*, ser. ICSC ’12, Washington, DC, USA, 2012, pp. 134–137. [Online]. Available: <http://dx.doi.org/10.1109/ICSC.2012.18>
- [59] K. Thomas, “Just Noticeable Difference and Tempo Change,” *Journal of Scientific Psychology*, 2007.
- [60] M. Dolson, “The Phase Vocoder: a Tutorial,” *Computer Music Journal*, vol. 10, no. 4, pp. 14–27, 1986.
- [61] M. J. Swain and D. H. Ballard, “Color Indexing,” *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.
- [62] P. Hanna, M. Robine, and T. Rocher, “An Alignment Based System for Chord Sequence Retrieval,” in *Proc. JCDL*, Austin, Texas, USA, 2009, p. 101. [Online]. Available: <http://portal.acm.org/citation.cfm?doid=1555400.1555417>
- [63] J. H. j. Jensen, M. G. s. l. Christensen, D. P. W. Ellis, and S. r. H. Jensen, “Quantitative Analysis of a Common Audio Similarity Measure,”

- 
- IEEE Trans. ASLP*, vol. 17, no. 4, pp. 693–703, 2009. [Online]. Available: <http://dx.doi.org/10.1109/TASL.2008.2012314>
- [64] J. G. D. Forney, “The Viterbi Algorithm,” *Proc. of the IEEE*, vol. 61, no. 3, pp. 302–309, 1973. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1450960
- [65] L. Barrington, A. B. Chan, and G. Lanckriet, “Modeling Music as a Dynamic Texture,” *IEEE Trans. ASLP*, vol. 18, no. 3, pp. 602–612, 2010. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5337999
- [66] R. J. Weiss and J. P. Bello, “Identifying Repeated Patterns in Music Using Sparse Convolutional Non-Negative Matrix Factorization.” in *Proc. ISMIR*, Utrecht, Netherlands, 2010.
- [67] R. Likert, “A Technique for the Measurement of Attitudes,” *Archives of Psychology*, vol. 22, no. 140, pp. 1–55, 1932. [Online]. Available: <http://psycnet.apa.org/psycinfo/1933-01885-001>
- [68] R. B. Zajonc, “Attitudinal Effects of Mere Exposure,” *Journal of Personality and Social Psychology*, vol. 9, no. 2, Part 2, pp. 1–27, 1968. [Online]. Available: <http://www.sciencedirect.com/science/article/B6X01-4NPKJ6B-1/2/3ceefd618facdf822d5a8478ea0b0e78>
- [69] D. Turnbull and G. Lanckriet, “A Supervised Approach for Detecting Boundaries in Music Using Difference Features and Boosting,” in *Proc. ISMIR*, Vienna, Austria, 2007, pp. 42–49.
- [70] M.-Y. Su, Y.-H. Yang, Y.-C. Lin, and H. H. Chen, “An Integrated Approach to Music Boundary Detection,” in *Proc. ISMIR*, Kobe, Japan, 2009, pp. 705–710.