

Automatic Rendering of Augmented Events in Immersive Concerts

Nick Creel, Marlboro College

Advisors: Zhiyao Duan, University of Rochester (ECE) & Matthew Brown, University of Rochester (Music)

Introduction

Using music feature extraction and audio-to-score alignment tools, we have developed a software which is capable of triggering audiovisual events based on a musician's position in a piece. Information is provided to the program in MusicXML format and matched to real-time audio performance, which triggers augmented events.

Motivation

Video Jockey (VJ) software is used by musicians to provide their audiences with an immersive audiovisual experience. Many of these software allow musicians to trigger events based on the musical features of their performance, using OSC or MIDI messages. Musicians who play traditional analogue instruments are interested in utilizing VJ tools during their performances, but without a way to connect their instruments to these software, performances must be facilitated with the help of a video operator. Operators are limited to triggering every event manually, following along with the score, which can be tedious and limits the range of events that can be triggered.

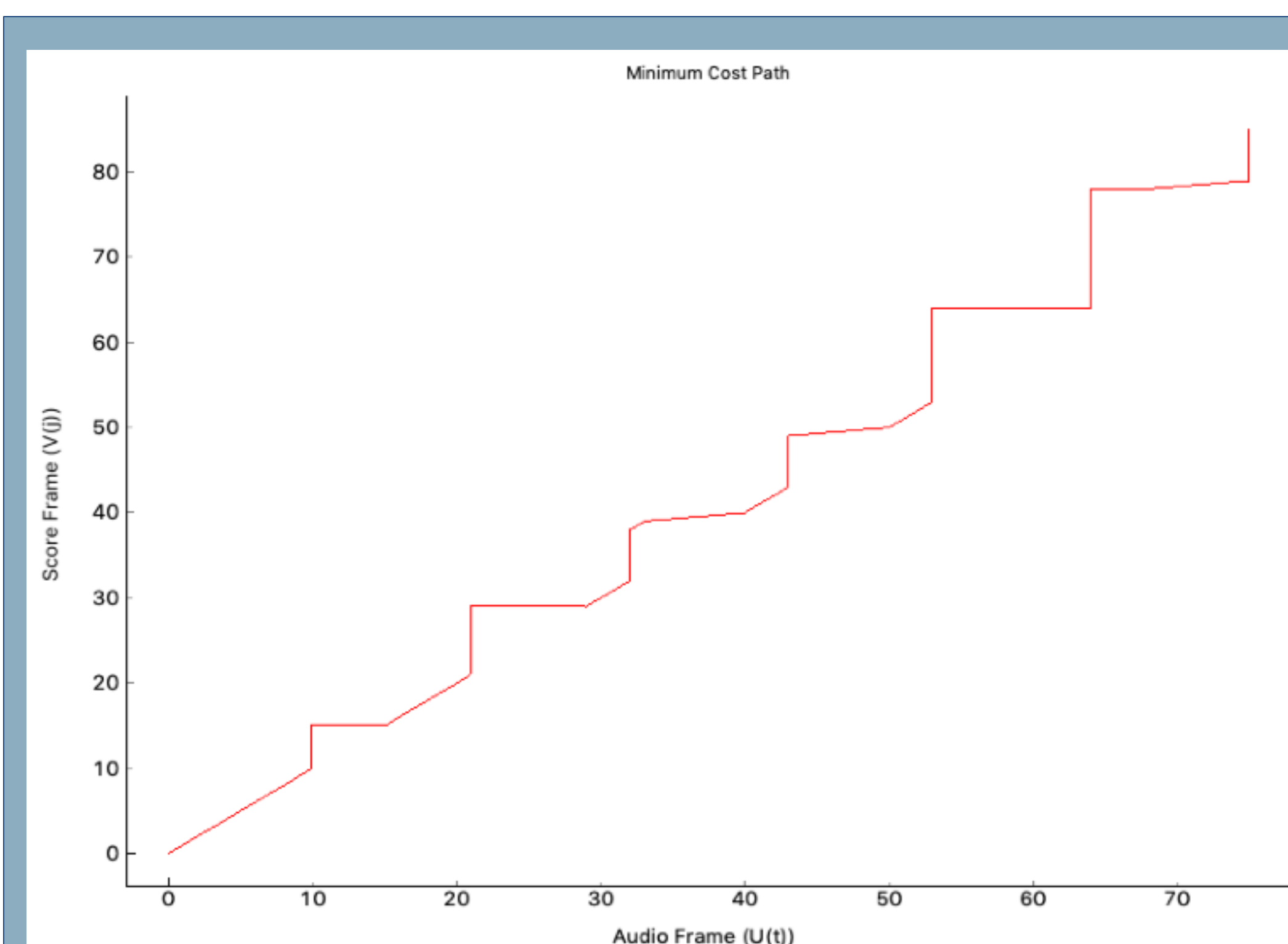


Figure 2: Alignment path for the first four measures of Twinkle Twinkle Little Star.

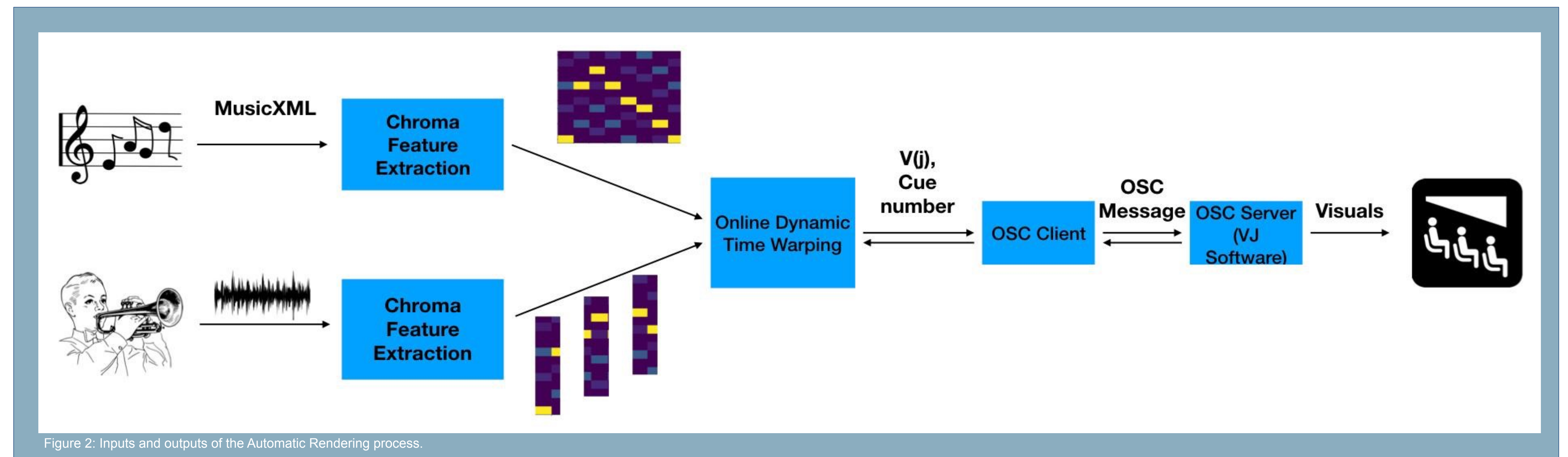


Figure 2: Inputs and outputs of the Automatic Rendering process.

Method

- Developed procedure for extracting chroma features from MusicXML (format often used to store and share musical scores, accessible to musicians)
- Extracted chroma features from audio using the `chroma_cqt` function contained in the audio analysis library `librosa`, discussed in McFee 2015.
- Frame size for audio and score analysis is limited to 1ms.
- Implemented the Online-Time Warping algorithm as presented in Dixon, 2005.
- UDP client sends OSC messages based on most recent score index matched. Client also listens for messages from server (in our case, a QLab workstation) and may change position in analysis based on feedback. (Users may still trigger cues manually, this must be communicated to alignment function if there is a delay)

Conclusion

Our software successfully aligns a sequence to itself, as well as a score-based sequence to an audio-based sequence. The software is capable of sending OSC messages based on the position in analysis, but does not yet react to OSC feedback. Further testing is needed to compare the efficacy of score-to-audio alignment compared to audio-to-audio alignment methods. GUI software for end-user still needs to be implemented, as well as the ability to change the position of the analysis dynamically to arbitrary positions in the score. One potential test for accuracy could be the ratio of user interventions to successful automatic matches.

Sources

B. McFee. "librosa: Audio and Music Signal Analysis in Python" Proc. Of The 14th Python In Science Conf., 2015, pp. 18-24.

S. Dixon. "Live-Tracking of Musical Performances Using On-Line Time Warping" Proc. of the 8th Int. Conference on Digital Audio Effects (DAFx'05), Madrid, Spain, September 2005, pp.20-22

Acknowledgements

This work was completed in the Audio Information Research lab at the University of Rochester, dept. of Electrical and Computer Engineering during the 2019 NSF REU in Computational Methods for Music, Media, and Minds. I would like to thank Christodoulos Benetatos and Bochen Li of the AIR lab for their wisdom during the debugging process and for their help implementing the OTW algorithm. I would also like to thank Christopher Winders of TableTopOpera for his support developing and testing the QLab communication feature of the software.

