

Lab 2 - CrossFade Specifications

Arnaud Blay, Dylan Hopper, Hyacinth Abad,
Joseph Wassell, Mohamed Abdullahi, Selai Naim, Virginia Vano Ruano

Old Dominion University

CS411W

CrossFade

12/01/2023

Version 2

Table of Contents

1. Introduction	3
1.1 Purpose.....	3
1.2 Scope.....	3
1.3 Definitions, Acronyms, and Abbreviations	5
1.5 Overview	7
2 Overall Description.....	7
2.1 Product Perspective.....	7
2.2 Product Functions	9
2.3 User Characteristics	10
2.4 Constraints	11
2.5 Assumptions and Dependencies	11

List of Figures

Figure 1: Menu Screen	7
Figure 2: Editor Screen.....	8
Figure 3: MFCD (Major Functional Components Diagram).....	8

List of Tables

Table 1: RWP vs Prototype (Transcription)	9
Table 2: RWP vs Prototype (Note Recognition)	9
Table 3: RWPvs Prototype (Transcription Error Correction)	10

1. Introduction

The musical field can be difficult to access, due to the amount of knowledge and the expensive equipment. Musical notation can prove to be very difficult, with many rules and symbols, which require a lot of time to learn and can dissuade many starting musicians from learning them.

1.1 Purpose

CrossFade is an application developed to help practicing musicians, since equipment to write and share musical pieces is expensive, and the knowledge required is high. It is meant for starting musicians as well as more experienced ones that want to share their music with the world but who do not have the knowledge to write it themselves. It can also be used by music teachers, as a help for their students.

CrossFade can listen to different kinds of input: live audio, a MIDI file, or an XML file. Then, it can transpose it to MIDI or XML, this last one also serving for sheet music format. Additionally, since transposing the input can have some errors, CrossFade can also correct them and allow the user for further changes if required.

CrossFade cannot record input, so it will not produce any audio files. It will not allow to compose a song from scratch, since some kind of base will be required to start working on it.

1.2 Scope

Starting practicing music can be daunting at the beginning. It not only requires obtaining an instrument, that is quite expensive by itself, but also other equipment to be able to record and share one's music with others. Music notation is also complicated, leaving the starting musician with no way to write down their music. With CrossFade, a practicing musician can play their music and CrossFade will write it for them, even correcting errors to make it more natural.

There are already applications that can transpose audio into music notation, but there is none that can transpose them without committing errors that make it difficult to interpret or unnatural in some contexts. CrossFade is made to correct those errors as

their main feature, since there are not that many applications that can do this. It also tries to make it easier to correct the errors, as no machine is perfect, saving time in the process.

That is why the prototype for CrossFade will focus on the error correction feature, detecting the errors and allowing the user to make further changes if necessary.

1.3 Definitions, Acronyms, and Abbreviations

Convolutional Neural Network (CNN) - Deep learning algorithm which can differentiate one image from another by assigning weights and biases to different aspects of the images. It is used in audio to differentiate different frequencies in a visual format.

Deep Learning - Subfield of machine learning which uses neural networks to solve complex problems. Learning comes directly from the data, instead of being hand-engineered by humans.

Keyboard - An electronic piano used to produce sound and MIDI information.

Monophony - A phrase of music in which only a single voice is played at a time.

Musical Instrument Digital Interface (MIDI) - A communications protocol used to connect physical and virtual music devices and instruments. MIDI files store note information which can be used to trigger instruments and devices.

MusicXML - A markup language format used to interchange and distribute digital sheet music.

Polyphony - A phrase of music in which more than a single voice is played at a time.

1.4 References

- Automatic Music Transcription and Ethnomusicology: A user study. (n.d.). Retrieved from <https://diva-portal.org/smash/get/diva2:1474663/FULLTEXT01.pdf>
- H. Takeda, T. Nishimoto and S. Sagayama, "Rhythm and Tempo Analysis Toward Automatic Music Transcription," 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, Honolulu, HI, USA, 2007, pp. IV-1317-IV-1320, doi: 10.1109/ICASSP.2007.367320.
- Huang, Z., Jia, X., & Guo, Y. (2019, June 29). State-of-the-art model for music object recognition with Deep Learning. MDPI. Retrieved February 8, 2023, from <https://www.mdpi.com/2076-3417/9/13/2645>
- Jovanovic, J. (2015, February 2). How does Shazam work? music recognition algorithms, fingerprinting, and processing: Toptal®. Toptal Engineering Blog. Retrieved February 8, 2023, from <https://www.toptal.com/algorithms/shazam-it-music-processing-fingerprinting-and-recognition>
- Scarlato, L. L. (n.d.). Continuous media. Audio. Retrieved March 1, 2023, from <https://www3.cs.stonybrook.edu/~lori/classes/GUI/sound.htm>
- Solanki, A., & Pandey, S. (2019, January 30). Music instrument recognition using deep convolutional neural networks - International Journal of Information Technology. SpringerLink. Retrieved February 8, 2023, from <https://link.springer.com/article/10.1007/s41870-019-00285-y>
- Team Crystal. (2023, September 18). Lab 1 - CrossFade Product Description. Retrieved November 13, 2023 from <https://arnaud-odu.github.io/CS411-Crystal-CrossFade>
- Zhang, X. (2022, March 11). Aided recognition and training of music features based on the internet of things and Artificial Intelligence. Computational intelligence and neuroscience. Retrieved February 8, 2023, from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8933112/>

1.5 Overview

The remaining sections provide an overview of how the prototype works and what can it do. It shows the interfaces that the user will interact with, and how the functionalities of the prototype differ with the ones from the complete product. It also shows the components that are used by CrossFade and how they interact with each other.

2 Overall Description

CrossFade allows to import a music file. The program then goes through it and corrects the transcription errors there might be, and then it shows the result a musical sheet, which can in turn be used to edit different parts of the song to correct errors that the program did not detect or revert those that were inaccurate. The program uses the new information to refine its error detection algorithm.

2.1 Product Perspective

CrossFade has 2 main screens:

- A menu screen (see Figure 1), where the user can select what kind of file they want to import.



Figure 1: Menu Screen

- The editor (see Figure 2), where the user can make changes to the imported file.

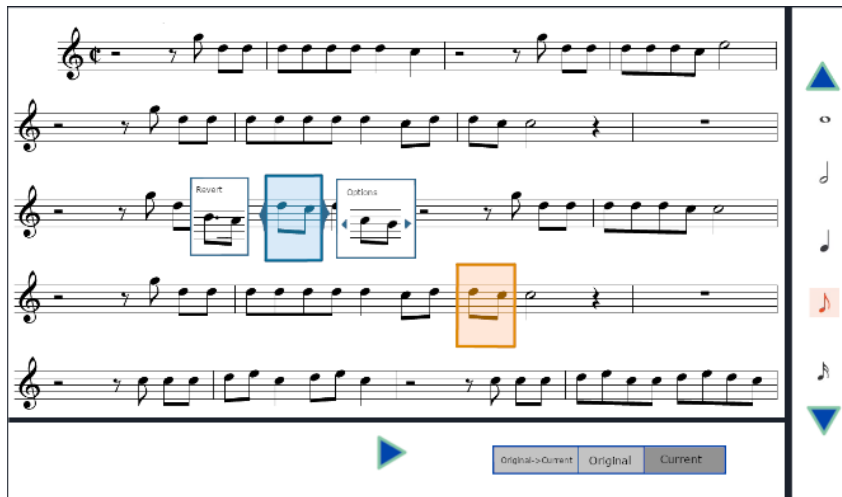


Figure 2: Editor Screen

On Figure 3, the components used by CrossFade are represented:

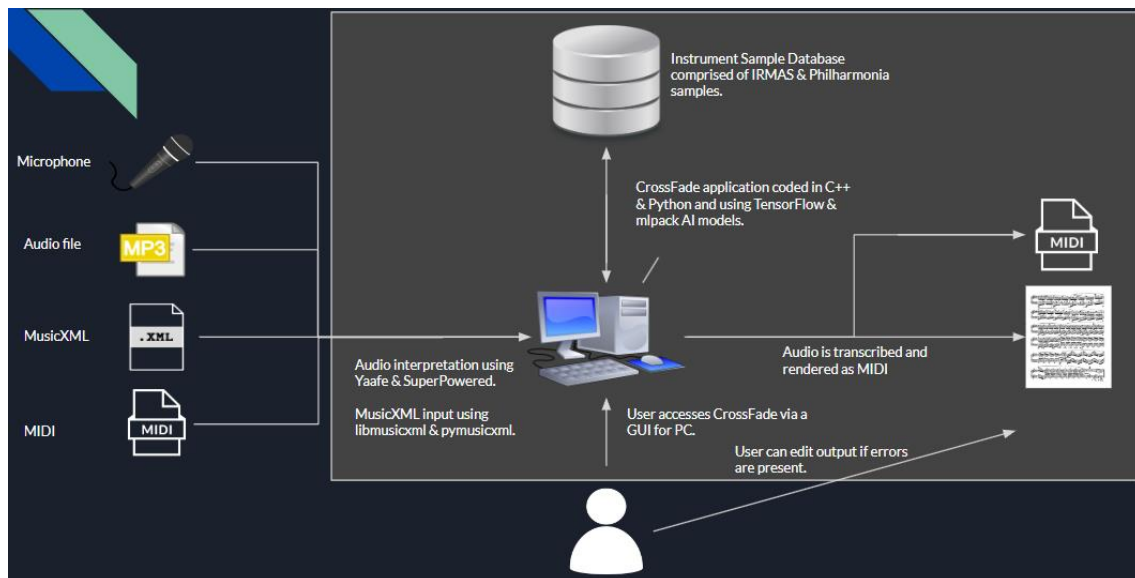


Figure 3: MFCD (Major Functional Components Diagram)

The computer in which CrossFade is opened is in the middle, with the user of the application located below it.

The possible inputs that CrossFade accepts are located on the left side of the figure. The prototype will not include the microphone input, as it will not be able to transpose audio yet. However, the audio input will be supported, as it will be used to be able to play the original audio if available for the sake of comparing it with the current version of the song after the automatic error corrections and further editing from the user.

The database of instruments is located at the top of the figure, which will be used for the transposition audio. This will not be present in the prototype.

The output from the program is located on the right side, which can be edited by the user.

2.2 Product Functions

CrossFade features are divided in three sections.

Table 1 refers to the transcription features. The prototype will not be able to transcribe audio in any way.

Table 2 refers to the note recognition capabilities. Since the prototype will not provide the transcription feature, none of these will be featured there.

Table 3 is the main focus for the prototype, so all of the features present in this table will also be featured in the prototype, being this the error correction features.

Table 1: RWP vs Prototype (Transcription)

Transcription		
Feature	Real World Product	Prototype
Live Audio Transcription	✓	
MIDI Input/File Transcription	✓	
MIDI Transcription Correction	✓	✓
Multitrack Transcription	✓	
MusicXML Compatibility	✓	✓
MIDI File Export	✓	✓

Table 2: RWP vs Prototype (Note Recognition)

Note Recognition		
Feature	Real World Product	Prototype
Monophonic Note Recognition	✓	
Polyphonic Note Recognition	✓	
Instrument Recognition	✓	
Instrument Distinction	✓	

Table 3: RWPvs Prototype (Transcription Error Correction)

Transcription Error Correction		
Feature	Real World Product	Prototype
Transcription Error Correction	✓	✓
Highlight Possible Errors	✓	✓
Take User Feedback	✓	✓
Offer Possible Solutions	✓	✓
Compare Single Note from Original to Written	✓	✓
Compare a Segment from Original to Written	✓	✓

Following the order in which they would be found while using CrossFade, the functional components would be the following:

- Importing a MIDI or XML file, accompanied by an audio file, if available, with the original audio used to create the MIDI or XML files.
- Correcting the errors found in the file, and marking them as changed, so that the user can revise them and changed them back, or to something new, if desired.
- Editing of the piece of music imported, allowing the user to change notes, clef, key, rhythm, and dynamics of the song.
- Listening of the original audio (if available) and current audio of the piece being edited, so that the user can compare between them, both to check if they sound the same, or if they like it better with the changes made, and being aware that they are different.
- Saving the current file to be able to resume work later.
- Exporting the file with the changes to MIDI or XML, to be able to share them with the world.

2.3 User Characteristics

Practicing musician: A person who practices playing music and might not be very knowledgeable of music.

Composer: A person who wants to create its own music and be able to make changes to it easily. They also have more knowledge about music.

Music teacher: A person who teaches music and wants their students to practice creating music.

2.4 Constraints

N/A.

2.5 Assumptions and Dependencies

CrossFade will be dependent on using a library to read XML files, to retrieve the information.

The prototype of CrossFade will assume that there is only one instrument in the file.