Sheet Vision

EE/SE/CPRE 491 - Spring 2019

Student Suggested Project

Faculty Advisor

Alexander Stoytchev

Team Members

Faure, Ricardo Fung, Bryan

Greenfield, Garrett Nance, Trevin

Svenddal, Walter

Table of Contents

List of Figures

List of Definitions

- Sheet Music - Music in its written or printed form. [1]
- Musical Notes - A sign or character used to represent a tone, its position and form indicating the pitch and duration of the tone. [2]
- Tabs - A form of written music, but instead of being represented in the traditional sense (what tone it makes), notes are represented by the specific position they are supposed to be played in.

Problem Statement

Reading sheet music is not easy task. With the creation of alternate ways to learn how to play music, such as tabs and youtube tutorials, there has been a decline in the amount of people who can properly read sheet music. The problem with tabs and other kinds of methods of reading music is that they lack the complexity to be able to convey all of the specific nuances that a specific piece may have. The best option that captures all of the nuances that most musicians wish to convey when writing music, is sheet music. The problem with sheet music is that it can be very difficult at first, and since there is a decline in the amount of people that can read it, it can be difficult to find a proper way to read it.

Our solution for this is Sheet Vision, an application that can read and show a user how the sheet music is played, and how it is supposed to sound. This will lead to the user being able to draw parallels between what is on the sheet, and the music being played, supplementing the learning process of reading sheet music. Not only will our application play the music on the sheet, but it will also listen to the user playing it, and will give proper feedback to the user, to fix mistakes they may be making.

Operating Environment

Our product is expected to be used quiet indoor environment, with appropriate lighting. This is for our application to more accurately pick up sound from the user when they're playing along with our on-screen piano prompt, showing which keys should be played. The lighting is important to allow the camera to pick up an image clear enough for the computer vision algorithm to properly pick up the symbols needed to find which notes should be played, and when they should be played.

Intended Users

This product is intended for beginners and veteran music readers alike. This application should provide instructions simple and clear enough for even first-time musicians can keep up with using our product, yet powerful enough to ease some of the struggle of reading more complex songs for veteran musicians.

Intended Uses

Sheet Vision is intended to be used as a supplemental product for learning how to play sheet music. Sheet Vision will show musicians how the song is supposed to

sound based on the music written on sheet, with the use of audio and visual cues, as well as detect if the user is playing the music correctly. The application is mainly produced for educational purposes.

Assumptions and Limitations

Assumptions

- Our product will be able to read most written sheet music documents.
- Algorithm will be run on the client, so no internet connection is needed.

Limitations

- Some features will be limited based on the users possession of sheet music and a musical instrument.
    - The user needs to provide their own sheet music to scan, since our application will not provide it for them.
    - The user will need an instrument to make use of the play-along feature.
- The quality of our output will rely on the quality of both a users camera and microphone, depending on which feature they are making use of.

Expected End Product and Other Deliverables

- Sheet Vision Desktop Application - Expected Delivery Date : December 2019
    - User-friendly and responsive user interface.
    - System to read in images of sheet music.
    - Computer vision system used to decode sheet music into information useful for the application.
    - System that uses information provided by the computer vision system to select what notes should be played and when.
    - Audio processing system that can detect and decode audio into information useful for application.
    - System that uses information provided by audio processing system to notify the user if they played the song correctly or not.

Related Works / Market Survey

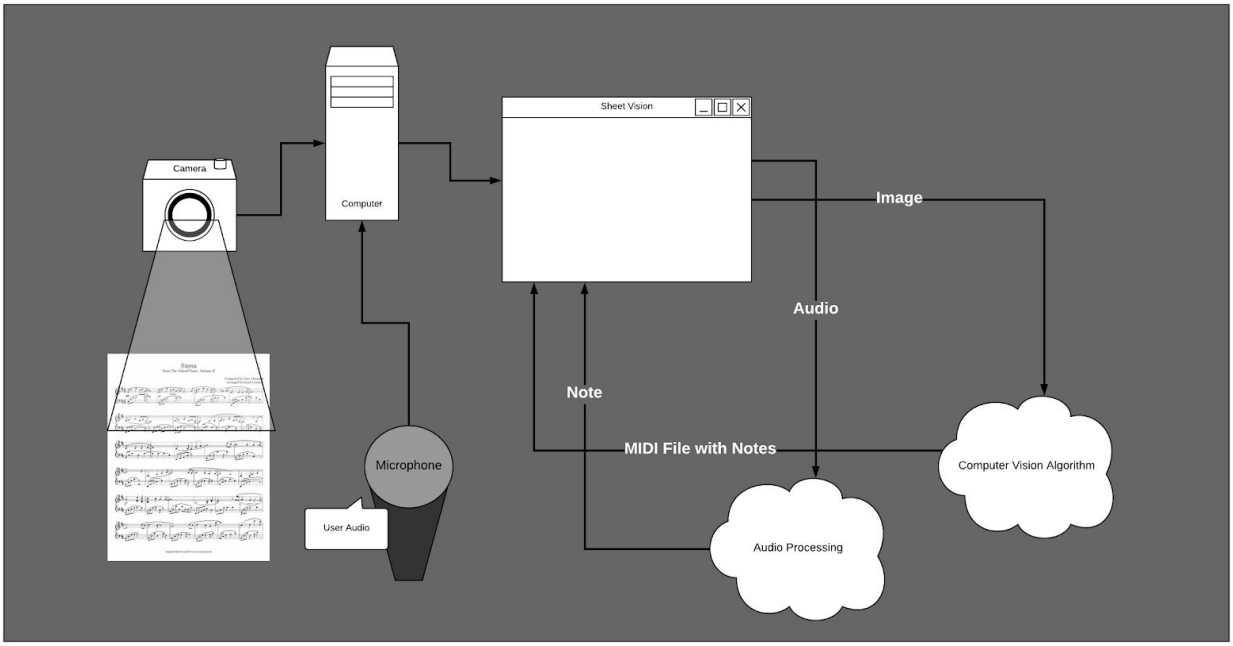
- During our market research, we found no products which could directly compete with our projected finished product.
- Products with similar goals in mind (Not providing the same service):

- Songsterr - Scroll through and play tabs for musicians who want a better understanding on what they are reading.
- Ultimate-Guitar - Provides the largest database for tabs in the world. Also supports scrolling through tabs and playing music, yet it does this for a fee.
    - None of the mentioned products provide this service for sheet music, they do this exclusively for guitar/bass/drum tabs.

Relevant Literature

- http://www.simplifyingtheory.com/sheet-music-guide/
- https://www.musicnotes.com/blog/2014/04/11/how-to-read-sheet-music/

High-level Block Diagram of System



Functional Requirements

- Access images from the device
- Access images from the camera
- Access audio from microphone
- Process images and provide correct output.
- Process audio and provide feedback on audio correctness (check against data provided from processed image).
- Play audio based on data provided from processed image.
- Functional and easy-to-use user interface.

Constraints Considerations

Our team has little experience with machine vision and react native so we will have to do a great amount of research and learn quickly to complete the project. We would like to have a feasible model by May 2019, so this particular deliverable will need to be completed by that time.

We will also need to work closely with our team members and faculty to ensure that our system is developed in a way where it could be eventually incorporated into IOS and Android apps. We also need to take into account that our team is composed of undergraduate students who are working part-time, so some of the timelines might need to be readjusted.

Technology Considerations

Due to the low processing power on mobile devices we will need to be as efficient as possible with our on board image processing. If image processing is too computationally expensive on most mobile devices we plan to move image processing to be a server side operation.

Technical Approach Considerations

Since our end goal is to have an application which can be used on a computer as well as a mobile device we are limiting ourselves to using a portable front end framework for our application. As well as implementing our machine vision in a language which will be easily interfaced with through our front end application, and is portable to different devices.

Testing Requirements Considerations

Testing will be done by feeding a plethora of different songs written in the format of sheet music to the application and seeing if the output of the algorithm is coherent with what is on the sheet. For testing the audio we will play an instrument and make sure the audio processing algorithm detects the correct note that was played. Finally after all individual components have been tested we will test use-cases, in which we have the application read sheet music, play it correctly and then have a user play the song themself, the application then should be able to determine if the user is playing it correctly or not, based on the information outputted from the individual components.

Security Considerations

Our product will not send nor store sensitive/confidential information. Our application will have access to the camera, so if this section of code is exploitable then by exploiting this section of code an attacker could gain access to the devices camera.

Safety Considerations

Our project/product currently has no safety concerns. Our application is only intended for safe activities. Any harm which comes to developers during the creation of our product or to the user during use is purely coincidental, and not caused by our product or its development.

Previous Work/Literature Review

None.

Possible Risks and Risk Management

There is a risk that we may not be able to implement all of our planned features. To avoid this we need to make sure that we have a realistic project plan and schedule that we follow. We need to ensure we give ourselves enough time to research and execute our tasks. We also need to make sure that if we are not getting a good accuracy with what we want to do, we should reach out to the faculty advisor for help.

Project Proposed Milestones and Evaluation Criteria

- ReactJS desktop application for windows.
- Application can read images from camera/file directory.
- Algorithm can recognize music notes in sheet music.
- Application can play the correct notes that come from the processed sheet music.
- Application can listen to user audio using the microphone.
- Algorithm can recognize what music notes are being played by the user based on audio input.
- Application can compare output from audio processing and image processing to determine if user is playing the correct notes.

Project Tracking Procedures

We will use Trello to assign tasks into two week sprints. These tasks will be assigned according to team roles, what has been accomplished in the past weeks, and our goals moving ahead. We will try to split up tasks such that they can be completed within a two week time period. Our team will meet weekly or once every two weeks, both as a team and with our faculty advisor to make sure we are on track with our sprints and deliverables.

Personnel Effort Requirements

- Expected 6 hours a week per team member
- Constant availability for communication
- Meeting participation and attendance

Resource Requirements

- Access to a computer with a Windows OS for React Native testing
- Access to an android cell phone device for testing on Android

Financial Requirements

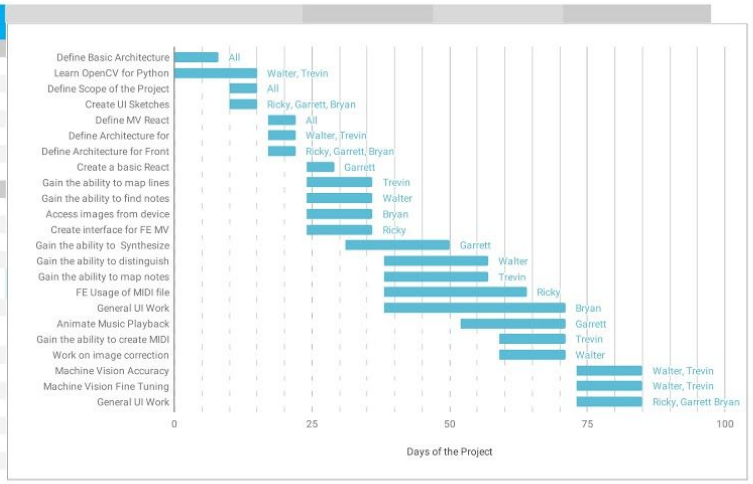
- Book of sheet music see here
    - Price: ~$20
- High-Resolution Camera see here
    - Price: ~$130
- High-Precision Microphone see here
    - Price: ~$100
- Incentive for student feedback
    - Price: ~$100

Schedule

Sheet Vision Spring 2019 Plan

* = an automatically calculated cell

TASK NAME	START DATE	END DATE	START ON DAY*	DURATION* (WORK DAYS)	TEAM MEMBER
Planning					
Define Basic Architecture	2/15	2/22	0	8	All
Learn OpenCV for Python	2/15	3/1	0	15	Walter, Trevin
Define Scope of the Project	2/25	3/1	10	5	All
Create UI Sketches	2/25	3/1	10	5	Ricky, Garrett, Bryan
Define MV React Communicaton	3/4	3/8	17	5	All
Define Architecture for Machine Vision	3/4	3/8	17	5	Walter, Trevin
Define Architecture for Front End	3/4	3/8	17	5	Ricky, Garrett, Bryan
Development					
Create a basic React Application	3/11	3/15	24	5	Garrett
Gain the ability to map lines from SM	3/11	3/22	24	12	Trevin
Gain the ability to find notes from SM	3/11	3/22	24	12	Walter
Access images from device	3/11	3/22	24	12	Bryan
Create interface for FE MV communication	3/11	3/22	24	12	Ricky
Gain the ability to Synthesize Midi FIles	3/18	4/5	31	19	Garrett
Gain the ability to distinguish different note types	3/25	4/12	38	19	Walter
Gain the ability to map notes	3/25	4/12	38	19	Trevin
FE Usage of MIDI file	3/25	4/19	38	26	Ricky
General UI Work	3/25	4/26	38	33	Bryan
Animate Music Playback Progress	4/8	4/26	52	19	Garrett
Gain the ability to create MIDI files	4/15	4/26	59	12	Trevin
Work on image correction	4/15	4/26	59	12	Walter
Machine Vision Accuracy Testing	4/29	5/10	73	12	Walter, Trevin
Machine Vision Fine Tuning	4/29	5/10	73	12	Walter, Trevin
General UI Work	4/29	5/10	73	12	Ricky, Garrett Bryan

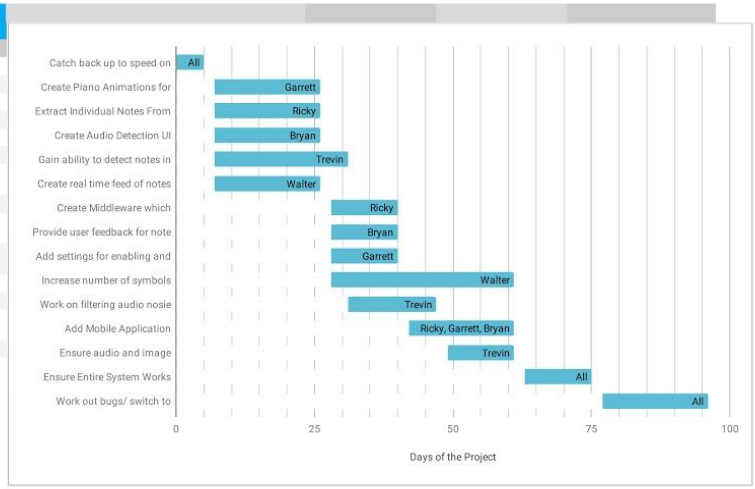


Sheet Vision Fall 2019 Plan

* = an automatically calculated cell

TASK NAME	START DATE	END DATE	START ON DAY*	DURATION* (WORK DAYS)	TEAM MEMBER
Development					
Catch back up to speed on project	8/26	8/30	0	5	All
Create Piano Animations for playback	9/2	9/20	7	19	Garrett
Extract Individual Notes From MIDI	9/2	9/20	7	19	Ricky
Create Audio Detection UI	9/2	9/20	7	19	Bryan
Gain ability to detect notes in audio	9/2	9/25	7	24	Trevin
Create real time feed of notes detected	9/2	9/20	7	19	Walter
Create Middleware which compares real time feed to expected value	9/23	10/4	28	12	Ricky
Provide user feedback for note values	9/23	10/4	28	12	Bryan
Add settings for enabling and disabling animation	9/23	10/4	28	12	Garrett
Increase number of symbols recognized by MV	9/23	10/25	28	33	Walter
Work on filtering audio nosie	9/26	10/11	31	16	Trevin
Add Mobile Application Functionality	10/7	10/25	42	19	Ricky, Garrett, Bryan
Ensure audio and image processing works with mobile	10/14	10/25	49	12	Trevin
Ensure Entire System Works together	10/28	11/8	63	12	All
Work out bugs/ switch to production build/ wrapping up	11/11	11/29	77	19	All

Glossary
SM - Sheet Music
MV - Machine Vision
FE - Front End



Closing Summary

Throughout these two semesters we will work on creating a application that can scan through a sheet of music and create a simple animation of which keys on a piano you should press that correlate to the beats that will be playing. With this you can change features and after hearing what it should sound like and seeing what you should play; you can turn the application into practice mode and it can tell you if you are playing the correct notes.

References

- Use React Native
  http://www.reactnative.com/

- OpenCV
  https://opencv.org/

- Full Gantt Chart
  https://docs.google.com/spreadsheets/d/1bJtgowXIafPeUwFRCIq-trlTMmdOy6Jp7IHifeHZWHk/edit?usp=sharing

- Definitions [1] [2]
  https://www.dictionary.com/browse/sheet-music
  https://www.dictionary.com/browse/notes?s=t

Appendices

TBD