

EE / CPRE / SE 491

Sheet Vision

Iteration 3 Report

2/15/19 - 3/01/19

Student suggested

Faculty advisor: Alexander Stoytchev

Team Members:

Bryan Fung — Frontend/Backend, Meeting Facilitator

Garrett Greenfield — Front end, Team Scribe

Ricardo Faure — Frontend/Backend, Meeting Facilitator

Trevin Nance — Machine vision, Chief Engineer Power System

Walter Svenddal — Machine vision, Report Manager

Past Week Accomplishments:

- Set up react native environment (Ricardo, Bryan, Garrett)
 - Set up and explored possibilities other than react native for our desktop application
 - Learned Python is compatible through the use of AWS and can be used for communication between the front and back end of the system.
- Created a Gantt chart (Trevin)
 - Created a chart that charted out the work schedules for the group
 - Learned how to do machine vision with discovering measure lines in sheet music
- Resource collection and Opencv (Walter)
 - Obtained sheet music to be used for the machine vision process
 - Learned how to do machine vision with discovering measure lines in sheet music

Pending issues:

- React-native for desktop applications requires windows
- React-native requires many dependencies.
- Very little online resources for OpenCV for Javascript.
- Too little resources for react-native-windows

Individual Contributions

<u>Team Member</u>	<u>Individual Contributions</u>	<u>Hours this week</u>	<u>Total Hours</u>
Bryan Fung	Tried to implement react-native-windows, also setted up android development using react-native	7	15
Garrett Greenfield	Worked on the mobile development of React Native because of a Windows Boot Crash that hindered React Native Windows Development	4	10
Ricardo Faure	Tried out different ways to implement desktop application aside from react-native-windows, extended research on web frameworks	4	10
Trevin Nance	Made a rough Gantt chart for the project. Worked on learning OpenCV for python.	5	15
Walter Svenddal	R&D of OpenCV on sheet music	4	8

Plans for Coming Week:

- Whole Team:
 - Create a Communications diagram for all of the frameworks
 - Research and confirm how the audio processing will work
 - Fully Define the Architecture of the project
 - Create a Dummy Application that has full communicative properties throughout the architecture
- Bryan Fung:
 - Create a small prototype application that can communicate with an Amazon AWS server
 - Research possible frameworks
- Garrett Greenfield:
 - Create a small prototype application that can communicate with an Amazon AWS server
 - Research possible frameworks

- Ricardo Faure:
 - Create a small prototype application that can communicate with an Amazon AWS server. Finalize an architecture for our product.
- Trevin Nance:
 - Implement a Dummy Application that will turn the music into grey vision and can receive a picture and send a midi file
 - Furthered development and understanding of how OpenCv.Js will see and process the sheet work
- Walter Svenddal:
 - Furthered development and understanding of how OpenCv.Js will see and process the sheet work
 - Make the Machine Vision see the lines and notes of the project