INITIAL PROJECT PLAN - GESTURE CONTROLLED MUSICAL CONDUCTING

By Kieran Flay

Supervisor
Dave Marshall
Dave.Marshall@cs.cardiff.ac.uk

Module Information
One Semester Individual Project
1516-CM3203
40 Credits

Contents

Project Description

Technology has continuously helped enable the discovery of new platforms and ways to express music. An instance of this might be through the digitalisation of various instruments, such as an electric drum kit at a basic level or a musical videogame at a more advanced level. Musical artists are always looking to technology to find the next way of creating music in new and innovative ways to express themselves.

The market for gesture control and hands-free technology has been exploding over recent years, due to the fact that the technology offers the user an interesting and new way of interacting with devices. This opens up the doors to many possible use cases. For example, gesture control and hands-free technology has already been introduced into several areas, such as the automotive, gaming and electronics industries. As such, gesture control could allow for interesting new ways to create music that have not been previously explored.

My project is to explore the use of gestural control devices to capture human movements and gestures that will allow for the simulation of musical conducting. I will create the software that will act a virtual MIDI device, which will learn to recognise gestures and manipulate a piece of music accordingly based on user input. The gestures captured from the input device will send messages to a digital audio workstation, which will allow the user to manipulate several properties of the track being played, such as its tempo or volume. Eventually the gestures will allow for the manipulation of individual instruments and other advanced commands.

Example uses cases include (but are not limited to):

- Training for junior conductors who can experience conducting at a basic level
- Simulation environment for professional conductors who can experiment with the way they conduct a piece of music
- Entertainment purposes so users can have fun experimenting with music

Project Aims and Objectives

Primary Objectives

1. **Leap Motion sensor**
   The application should successfully connect to a Leap Motion sensor and process the input gesture data into something meaningful

2. **Machine learning**
   The application should recognise relevant hand gestures from the input and map them to actions through machine learning

3. **Gestures to MIDI actions**
   The application should successfully link hand gestures to MIDI actions by processing the input mapping it to a MIDI output

4. **MIDI bridge**
   The application should send MIDI commands to a soundboard, which will manipulate a track being played

5. **Soundboard output**
   The soundboard should output the track being played plus any effects being set in real-time by the user

Secondary Objectives

1. **Application aesthetics**
   The application should be easy-to-use by the user and will look visually appealing to any audience

2. **Advanced gestures**
   The application should be able to detect more advanced gestures for controlling individual instruments

3. **User tutorial**
   The application should teach the user how to perform interactions and inputs with the system with a helpful guide

Tools & Technology

MIDI (Musical Instrument Digital Interface)
A technical standard that describes a protocol, digital interface and connectors and a wide variety of electronic musical instruments, computers and other related devices to connect and communicate with each other.

MIDI API
Implements a bridge so that MIDI commands can be sent between an application and a digital workstation.

Leap motion sensor
A physical device that connects to a computer via USB. It captures hand gestures from a user and turns it into digital data.

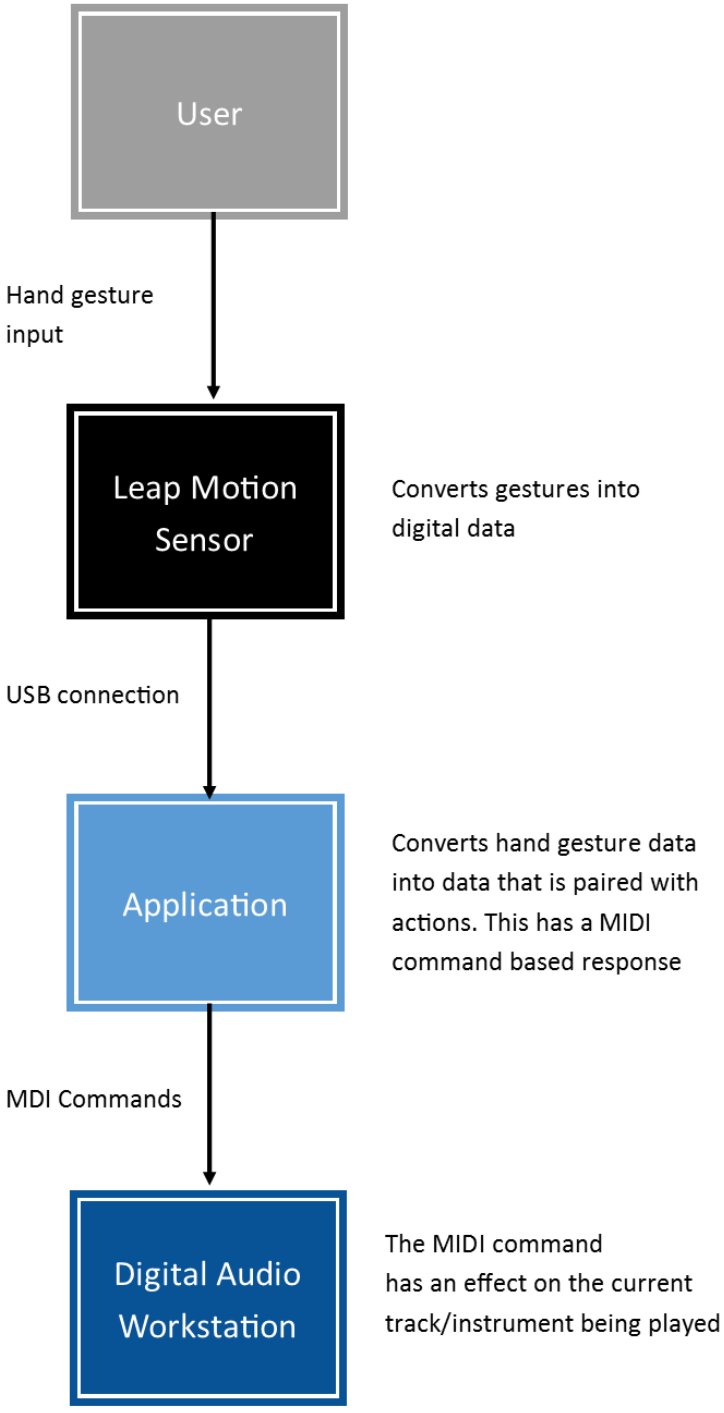
LeapSDK
The leap motion developer kit that allows for the creation of gesture-based applications.

Ethics

Due to the nature of the project, there are no ethical issues to consider. As my project will only require gesture movements from users and no personal data, it is not necessary for my work to be reviewed for ethical approval.

Project Architecture

As the project involves several components, each with multiple inputs and outputs, the architecture has been visualised below:



Work Plan

Below is a plan that outlines the deliverables, objectives and tasks for each week. It also contains expectations of the project state over the weeks, which will help to see if progress is being made at a sufficient rate for the project to achieve completion on time along with all relevant documentation that needs to be handed in.

Wednesday at 12pm have been scheduled as a weekly spot with my supervisor so that we can discuss project progress, review current work, discuss any issues that have arisen and plan the next steps for the project.

Week 1 (25^{th} Jan – 31^{st} Jan)

- Write the initial plan.
- Have the initial plan reviewed by the project supervisor and perform any required changes.
- Submit the initial plan.

Week 2 (1^{st} Feb – 7^{th} Feb)

- Install and setup any relevant programs, software and tools required for the project. This includes: LeapSDK, digital audio workstation package, Eclipse and Leap Motion drivers.
- Set up and configure Leap Motion sensor hardware.
- Research gesture control and motion capture theory.

Week 3 (8^{th} Feb – 14^{th} Feb)

- Research existing projects/samples that could be considered similar to the project to and document any possible ideas.
- Research into MIDI theory.
- Test that the input sensor successfully works as configured.

Milestone: The program should be completely planned and the software/hardware should be set-up and ready to begin development.

Week 4 (15^{th} Feb – 21^{st} Feb)

- Start coding the application foundations.
- Implement the feature that allows the user to input hand gestures and generate some meaningful data on the application.

Week 5 (22^{nd} Feb – 28^{th} Feb)

- Implement the mapping between user input and actions.
- Test the input data to check it is understandable by the user and the computer and it successfully maps to a gesture that is denotable from the input movement.

Milestone: The program should be able to process basic gesture data from the Leap Motion sensor and map it to a relevant pre-defined action.

Week 6 (29th Feb – 6th Mar)

- Research existing MIDI-based applications.
- Implement the generation of a sample MIDI command.

Week 7 (7th Mar – 13th Mar)

- Implement the feature that allows the application is set up to act a virtual MIDI device and create a link to the digital audio workstation.
- Test the link is working and a sample MIDI command can be sent to the workstation.

Week 8 (14th Mar – 20th Mar)

- Add the function to send multiple MIDI commands to a digital audio workstation.
- Test that the MIDI commands being sent are having an expected effect on the music.

Milestone: The application can successfully connect to a digital workstation as a virtual MIDI device and a MIDI command can be sent from the application to the workstation.

Week 9 (21st Mar – 27th Mar)

- Implement the connection between the two core parts of the application (input gesture to action and MIDI command to workstation).
- Implement any extra secondary objectives if there is time.

Week 10 (28th Mar – 3rd Apr)

- Test that the input gestures are being correctly mapped to MIDI actions.
- Implement any extra secondary objectives if there is time.

Milestone: The application has connected the MIDI and gesture parts of the application together so that input from the user generates a mapped MIDI action that is sent to the digital audio workstation

Week 11 (4th Apr – 10th Apr)

- Start the test plan write up.
- Start performing the tests on the main application.
- Fill out the test plan and correct any unexpected errors in the application.
- Meet with the project supervisor to review the current build of the project and perform any required changed.
- Start the documentation for the project.

Milestone: The application has been completed, any secondary features have been implemented and it is ready to begin the testing phase

Week 12 (11th Apr – 17th Apr)

- Continue testing the application.
- Application has been completely tested.
- Continue the documentation for the project.

Week 13 (18th – 24th Apr)

- Continue the documentation for the project.
- Correct any errors that have been discovered from the testing.

Milestone: The application has been completely tested and a final build should be ready

Week 14 (25th Apr – 1st May)

- Meet with the project supervisor to review the current state of the of the documentation and perform any required changed.
- Continue documentation for the project.

Week 15 (2nd May – 8th May)

- Finish documentation for the project.
- Submit the project and all relevant documentation by 6th May (final report deadline).

Milestone: Application will be completed along with all relevant documentation that is sufficient for marking of the project