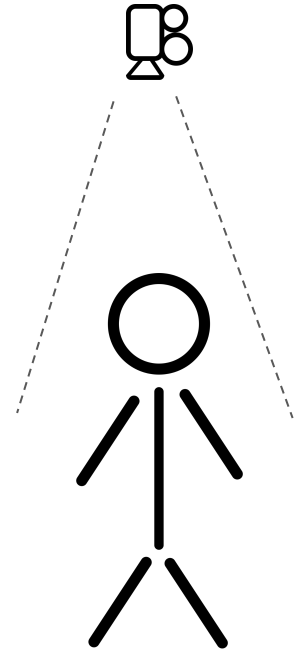


The Device

- Mechanically actuated keyboard that is managed by a microcontroller
- Provides a way to play the piano without needing the physical ability to press keys
- Limited to simple notes and chords
- Playing range = two octaves on piano keyboard, only white keys
- Currently, there are no widespread solutions for playing the piano without pressing keys or generating sound directly from a computer



Requirement #1: Hardware/Mechanism

- Design an external electrical system to play the keyboard
 - Designing an electrical system that can support multiple solenoids
 - Designing a program that could support turning on/ off these electrical components (i.e. actuators)

- Build a successfully, functioning external actuator system for the white keys
 - Using 14 solenoids
 - Similar to self-playing pianos in the market
 - Easy to use + install on the existing keyboard/ piano

Requirement #2: Gesture Detection

- Detect hand position > 90% of the time
 - Design a system that uses hand location on the screen to determine notes
 - Recognize hand positions when they are > 4 ft away and < 7 ft away

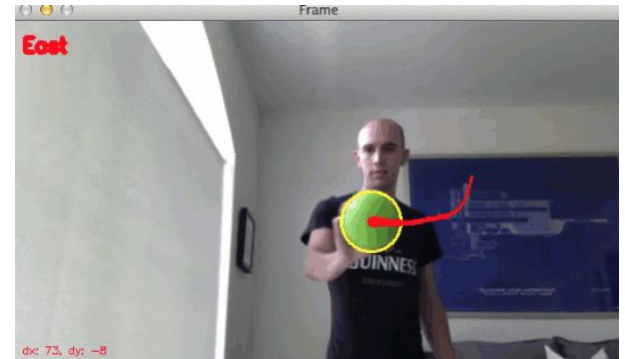
- **Goal:** Reduce latency as much as possible
 - Minimum Requirement: Response time (from time user makes motion to when the key is pressed) < 1 second
 - Long response time would take away from the feeling of actually playing an instrument

Requirement #3: Music Composition Generation

- Generate musical phrases of diatonic notes in the key of C major
 - This is a software only constraint, specifically to interface with the keyboard hardware and allow for future extensions of functionality
 - Can, in theory, also support different scale modes
- On the downbeat, notes should always fall on a chord tone
 - Although not necessarily musically interesting (like waiting for a resolution over multiple beats), the generated phrase will be “correct”
 - Still allows for interesting subdivided melodies
- Allow the user to switch between “player” and “generative” modes
 - Player mode will allow users to directly play piano via gesture software interfacing
 - Generative mode allows users to creatively express themselves while the software creates a fitting phrase to accompany them

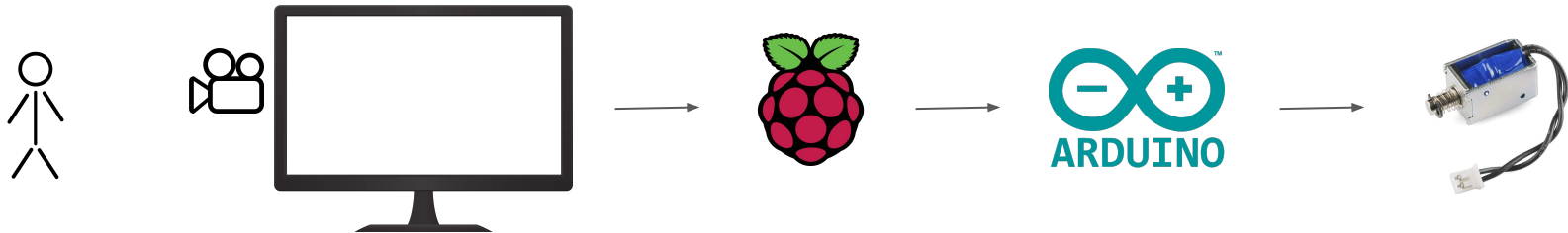
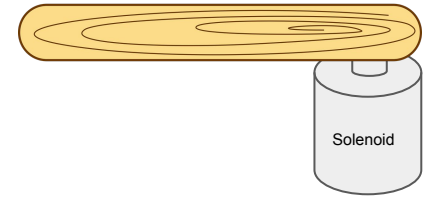
Technical Challenges

- External, actuating system to play the keyboard
 - Sourcing electrical components, designing a pcb board, and etc
- Learning curve for learning to use OpenCV
- Choosing the existing dataset to work with
 - Or even training the dataset ourselves
- Designing a comprehensive computer graphics system to detect a movement/position
 - Each position will correspond to a key or a chord
 - Using colored gloves on each hand for easy color detection



Solution Approach

- Power only the needed number of actuators
 - Avoids the use of unnecessary power
- Use OpenCV to track hand position in screen space
 - Initially, we can divide the screen into grids, then detect what grid the hand is a part of
 - Then, we can track the hand for some period, draw a trail, fill the shape between the start and end points, and use Shape Matching to perform additional actions
 - Lastly, we could use neural network to detect gestures that control the form of music

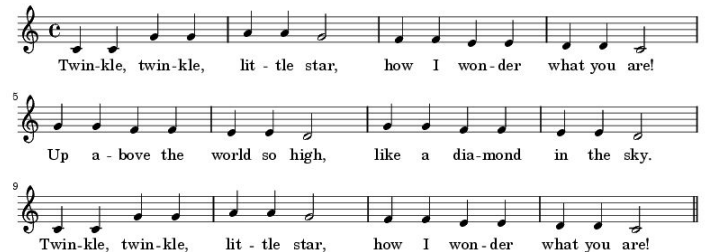


Solution Approach (contd.)

- Music Generation
 - Multiple Possible Solutions
 - One-to-one: Hand position directly correlates to notes
 - Accurate, but requires fast data processing and signalling
 - $< 125\text{ms}$ for 16th notes at 120 bpm
 - Most intuitive
 - Note-to-note: Generate subdivided melody between two detected notes
 - Plays well with slow pipelines
 - Pick one note, pick another, wait, repeat
 - Less intuitive, may create a sense of disconnect (“what I play isn’t what I hear”)
 - Gestural Generation: Read in gestures to create a continuous, mood-fitting piece
 - Plays well with any detection speed
 - Read gesture, play tune for a few measures, repeat
 - Not necessarily intuitive, but not clunky
 - Difficult/requires far more work for all software systems

Testing, Verification and Metrics

- Hardware: playing Twinkle, Twinkle Little Star correctly, with chords
- Computer Vision/ SW: measure accuracy of recognizing hands and their positions on the screen
- Measure response time from when user makes motion to when the key is played (goal: < 1 sec)
- Music Generation: Correctly pass notes to hardware at correct tempo (at least 60 bpm)



Division of Labor

- **Hardware:**
 - External piano keyboard system - Sun A
- **Computer Vision:**
 - Recognize both hands and position in the screen - Katherine
 - Translate position data into key notes - Katherine
- **Music Generation:**
 - Algorithmic scoring - Lance
 - Tempo and subdivision regulation - Lance
- **Integration:**
 - Everyone will contribute to integrating the music generation, computer vision data and hardware system.

Motional Keyboard

The Keyboarders

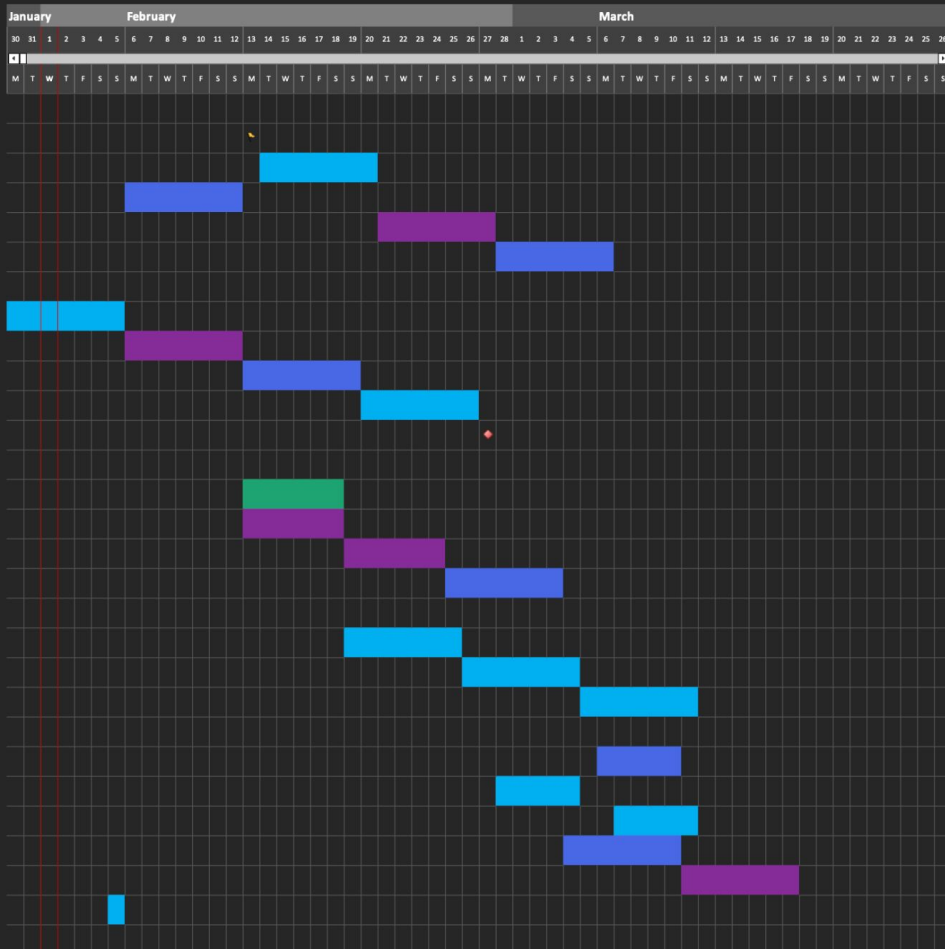
Tom Sullivan

Project Start Date: 1/16/23

Scrolling Increment: 0

Legend: On track Low risk Med risk High risk Unassigned

Milestone description	Category	Assigned to	Progress	Start	Days
Keyboard Hardware					
MIDI Keyboard Procurement	Milestone	Name	15%	2/13/23	1
Frame Creation	Low Risk	Suna	0%	2/14/23	7
Actuator Selection	Med Risk	Suna	5%	2/6/23	7
Actuator Integration	High Risk	Suna	0%	2/21/23	7
Software Integration	Med Risk	Suna	0%	2/28/23	7
CV					
Detection Method Selection	Low Risk	Katherine	5%	1/30/23	7
Hand Detection	High Risk	Katherine	0%	2/6/23	7
Gesture Mapping	Med Risk	Katherine	0%	2/13/23	7
Multi-feature Detection	Low Risk	Katherine	0%	2/20/23	7
Music Software Integration	Goal	Katherine	0%	2/27/23	1
Music Synthesis					
Sound Generation Algorithm	On Track	Lance	5%	2/13/23	6
Note Sequencing (In-time Quarter Notes)	High Risk	Lance	0%	2/13/23	6
Pitch Selection	High Risk	Lance	0%	2/19/23	6
Chord Playing	Med Risk	Lance	0%	2/25/23	7
Advanced Music Synthesis					
Subdivisions	Low Risk	Lance	0%	2/19/23	7
Syncopation	Low Risk	Lance	0%	2/26/23	7
Chords (Non-Triads)	Low Risk	Lance	0%	3/5/23	7
Verification and Validation					
Unit Testing	Med Risk	Lance	0%	3/6/23	5
Hardware Testing	Low Risk	Suna	0%	2/28/23	5
Software Testing	Low Risk	Katherine	0%	3/7/23	5
Music Synthesis Testing	Med Risk	Lance	0%	3/4/23	7
Integration Testing	High Risk	All	0%	3/11/23	7
Slack Creation	Low Risk	All	0%	2/5/23	1



To add more data, insert new rows ABOVE this one