Digital Docent

- Runs on Raspberry Pi with Satellite CCRMA
- Program written in both chuck and puredata
- Receives OSC signals
- Future direction: receive web requests
Canvas

Motivations:

- Touchscreen-native interface (non-skeuomorphic)
- Full user control of synthesized audio
Canvas

Interface:

- Loosely based on drawing
- Notes contained in circles
- Area between circles plays frequencies near those notes
Canvas

Layout:

- Chords are easy to visualize
- Scales are easy to play
- Intervals larger than a major third are not difficult to play, but are currently less controllable
- Trills are also easy to play, but less controllable when reverb is enabled
Canvas

Controlling the Voice:

- Sound produced on initial touch is a simple sine wave an octave above the circle's frequency.
- Sound produced when touch moves is a sine wave at the circle's frequency, plus a sine wave an octave higher with an amplitude modulation defined by the circle's fifth.
- Intervals larger than a major third are not difficult to play, but are currently less controllable.
Canvas

Mapped Attributes:

- Pitch to location, defined by circle
- Volume to touch velocity
- Reverb to location, defined by horizontal position
Canvas

Drawing Functionality:

- Has been implemented, but is currently too computationally expensive
- Each time a new sample is received, the view will draw the previously saved image, draw the last two points of all live touches, and save the current screen
Screen Protector:

- Provides topographical pitch differentiation
- Allows users to synchronize timing when using multitouch, and offers the ability to feel where the next note begins
Canvas

Future Directions:

- Optimize the drawing functionality
- Map additional audio attributes to the interface
- Enable the use of multiple devices to control the same instrument