

COMOVO

Design Review

Neeti Ganjur, Gauri Laxman, Shrutika Ruhela

Application Area

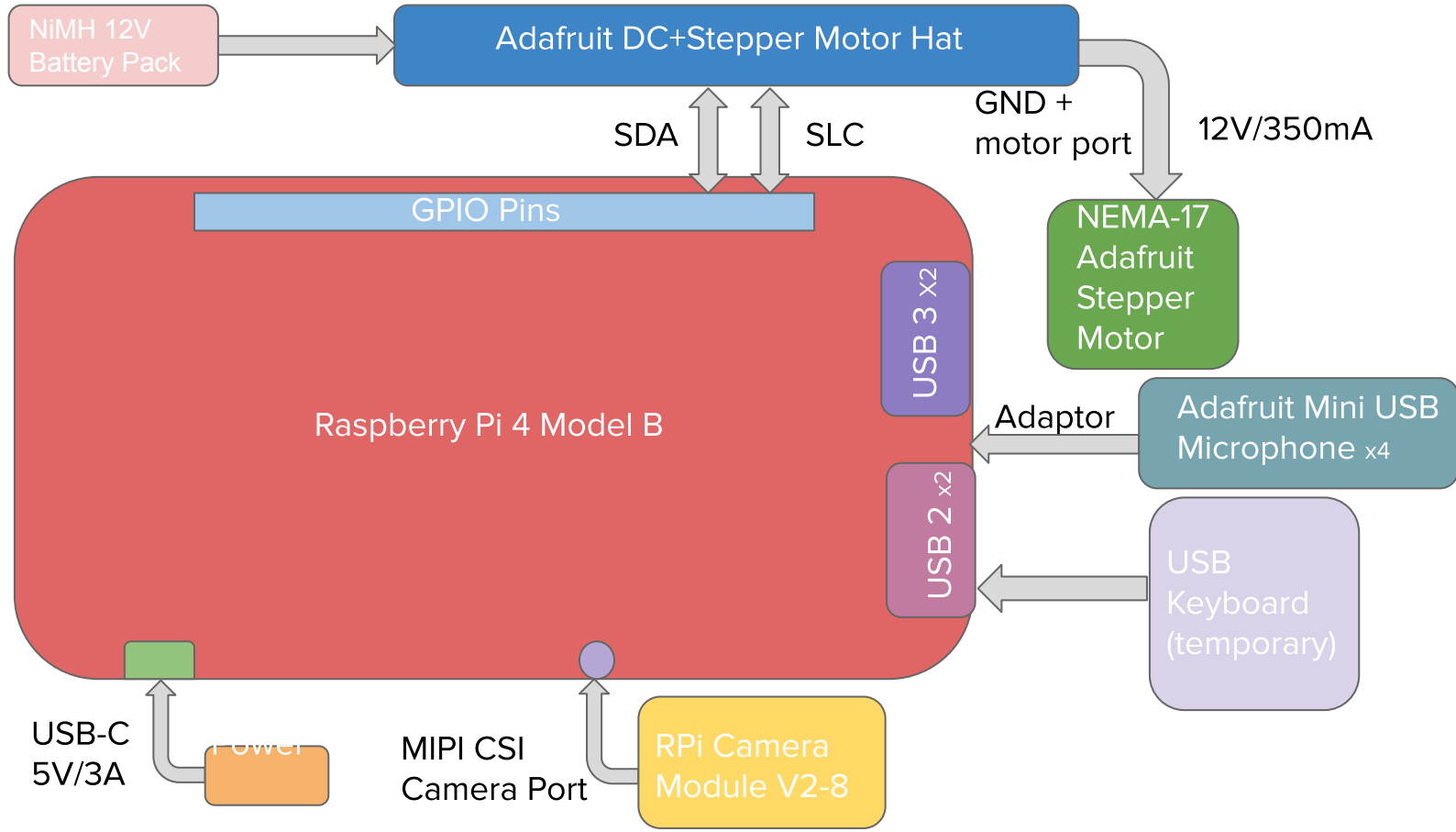
PROBLEM SPACE:

- Multiple-person video calling is a pain
- Have to keep moving the phone around to see everyone
- End up looking at the ceiling
- Currently no platform independent, inexpensive solutions for personal use

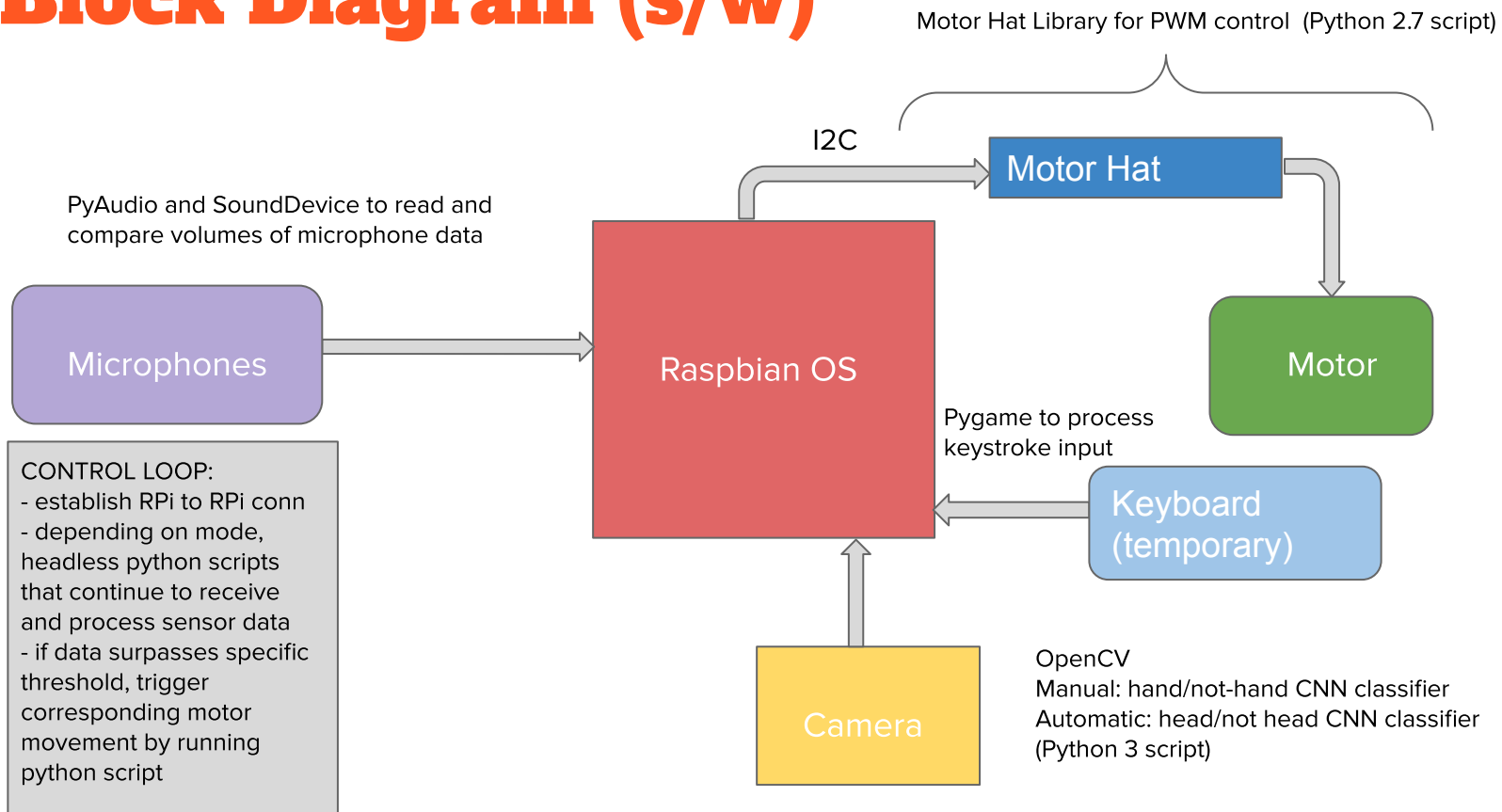
OUR SOLUTION:

- Device that sits on dinner table and holds your phone
- Can communicate with another device in your dorm room across the world
- Automatic and manual modes to rotate the phone when on video chat

Block Diagram (h/w)



Block Diagram (s/w)



Motor - RPi Communication

- Enable I2C on RPi
- RPi will be connected to motor hat through SDA and SCL GPIO pins
- Motor hat power terminal block will be connected to NiMH 12V Battery Pack
- Stepper motor will be connected to one of four motor ports on the hat
- Use the Python Raspi_MotorHAT library motorHAT PWM controller class
- Create an object for the stepper motor with port and steps per revolution arguments
- Use methods to set and move steps
- Use methods to set direction and speed

Camera - RPi Communication

- Camera will be connected to RPi through the MIPI CSI Camera Port
- Enable camera access on RPi using `sudo raspi-config`
- Directly use the Python OpenCV interface to read multiple frames from a video stream
- Manual: Build a CNN trained to distinguish “hand” and “not hand”. Feed frame into trained net to classify. Decide if “left” or “right” based on position of hand in the frame (left half or right half).
- Automatic: Build a CNN trained to distinguish “head” and “not head”. Use this in conjunction with sound localization position to narrow down on where to rotate to.

Microphone - RPi Communication

- 4 microphones positioned around the platform 90° apart
- Volume comparison to sense range of direction from where person is speaking
 - Plan: rotate camera to that range direction and look for a head. If multiple heads, pick one and stay there until sound volume information changes.
 - If volume difference detection is accurate enough (to the correct quarter), we can consider using more microphones to get more specific range information.
- Use PyAudio and SoundDevice libraries to receive, process and compare the sound signals from each microphone.
- Also considered using Time Difference of Arrival (TDOA)
 - Not a good idea for us because distance between microphones is too less
 - So difference between arrival times is not large enough to help.

RPi - RPi Communication

- Both RPis run a headless script which is the main control loop
- Establish a client-server model between the two through a TCP connection
 - Since we are first trying to implement a proof of concept, a direct TCP connection is enough.
 - In the future, if we extend this project and build a network of COMOVOs we could use cloud.
- Use socket programming to send and recv messages between the RPis.
- Both RPis boot up in automatic mode by default
- Change modes on both RPis by sending a message from one to the other.
- As sensor data gets processed on the controller's side in manual mode, send messages to other RPi directing motor rotation

Justification for Parts - I

- RPi 4 Model B:
 - Smaller RPi model
 - WiFi capability
 - More sensor input ports
 - Documentation and support readily available because newest model
- Adafruit DC & Stepper Motor Hat:
 - Contains motor controllers which are necessary to provide enough power to drive the motor
 - Sits on top of RPi
 - Motor Hat library is easy to use and provides high level abstraction
 - power terminal block to power stepper motor
- Adafruit Stepper Motor NEMA-17 size 200 steps/revolution:
 - High torque at low speeds
 - Precision (1.8 degrees per step)
 - Low cost
 - Availability of documentation, user guides, and examples

Justification for Parts - II

- Raspberry Pi Camera Module v2-8 Megapixel, 1080p:
 - Image quality (8mp)
 - Custom RPi add-on with RPi Camera module
 - Availability of user guides and documentation
- Adafruit Mini USB Microphone:
 - Low cost
 - Small
 - Compatibility with RPi
- 8x AA NiMH Battery Pack:
 - Flexible and simple
 - 12V

Metrics & Verification

Feature	Metric	Success Values
<i>(M)</i> Gesture Detection	% accuracy of classifier on testing data	> 85%
<i>(M)</i> Distance of person motioning from COMOVO	Distance in feet	< 1ft
<i>(A)</i> Accuracy of 'loudest speaker' detection	% times COMOVO rotates to correct quarter	> 95%
<i>(A)</i> Distance of people speaking from COMOVO	Distance in feet	~ 3ft
<i>(Both)</i> Latency	Time taken to receive, process, and execute command	Begin rotating in < 2.3s
<i>(Both)</i> RPi to RPi communication Latency	Time taken to transmit data over TCP	~ 300 ms

