## Cubr: Cube Puzzle Solver

18500 S19 Team D6
Project Proposal
JT Aceron, Lily Chen, Sam Fazel-Sarjui

## Background

- 3x3x3 Rubik's Cube
- 43,252,003,274,489,856,000 unique cube states
- World record time set by Feliks Zemdegs: 4.22 s
- 3 pieces
- Center, Edge, Corner



## Use Case

- Uses CV to map the faces of the cube (signals \& systems)
- Uses software to find the sequence of moves to solve the cube (software)
- Uses an Arduino to interface our solution with stepper motors (hardware)


## Hardware and Software Overview

- Webcam
- OpenCV
- Arduino
- 6 stepper motors
- 6 stepper motor drivers
- Power Supply
- 3D printed coupling arms
- Basic breadboard
- 3D printed housing
- $3 \times 3 \times 3$ Speedcube



## Processing and Execution

- Mac:
- Cube state detection
- Logitech - C920 Pro Webcam
- Webcam compatible with OpenCV
- Written in C++ or Python
- Cube solving algorithm
- Written in Python

- Arduino Uno Rev3:
- Arduino receives solution string
- Master-slaves setup to communicate to the 6 stepper drivers
- Written in C


## Stepper Motors

- NEMA-17 Stepper motor
- 200 steps/rev
- 12V 350mA
- \$14
- A4988 Stepper Driver
- Operates from 8V-35V
- \$5
- 12 Volt DC wall adapter



## Physical Infrastructure \& Misc.

- 3D printed housing
- 3D printed cube coupling arms
- Breadboard
- Wires



## Solution: Cube State Detection

- Use webcam to scan all sides and make 2D map
- Process this 2D cube state to create a final solution string
- Stretch goal: incorporate RGB color sensors
- Color sensors can ensure our CV color detection is correct



## Solution: Solving Algorithms

- Intuitive (Beginner's method)

O Cube is broken down into states and solved with a set of basic algorithms

- Stretch goal: Non-intuitive (highly efficient)
- The most optimal solution path
- Two-phase Algorithm
- God's Number: 20 Algorithm
- Written in Python



## Solution: Physical Execution

- Arduino receives solution string using cube notation (R R' U U' etc)
- Arduino will communicate to each stepper driver individually to drive it's motor
- 18 possible moves
- Clockwise or counterclockwise
- 90 degree or 180 degree turn
- Motors must be timed and tuned: only one move/turn at a time

$u$

$\mathbf{U}^{\prime}$

R

F

$F^{\prime}$

D

D'


L

L'

в

B'


## Testing, Verification \& Metrics

- Cube state detection
- Verify how accurately we scan the cube using OpenCV
- Algorithm efficiency
- Number of cube movements to solve the cube
- Was a solution found?
- Stepper motor precision \& timing
- Verify correct movements are made for any given input
- Use metrics to record how fast a movement takes
- Is the cube solved?

Timeline

| Week | Feb 3-9 | Feb 10-16 (2/16: First status report due) | Feb 17-23 | Feb 24 - March 2 | March 3-9 (3/4: Design document due; $3 / 6$ : Ethics | March 10-16 <br> (Spring Break) | March 17-23 | March 24-30 | March 31 - Apr 6 (4/1-3: in-lab demos) | $\begin{aligned} & \text { App 7-13 (4/11-14: } \\ & \text { Carnival) } \end{aligned}$ | Apr 14-20 | Apr 21-27 (4/24: <br> in-lab demos) |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Purchase parts |  |  |  |  |  |  |  |  |  |  |  |  | JT Aceron |
| Cube state detection |  |  |  |  |  |  |  |  |  |  |  |  | Lily Chen |
| Cube state matching |  |  |  |  |  |  |  |  |  |  |  |  | Sam Fazel-Sarjui |
| Beginner's method implementation |  |  |  |  |  |  |  |  |  |  |  |  | Multiple/All |
| Learn how to control motors individually |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design master/slave configuration |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Stepper motor interface with Arduino |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design power supply hookup |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Construct basic housing and test all motors in sync |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design robot housing |  |  |  |  |  |  |  |  |  |  |  |  |  |
| $\begin{aligned} & \text { Design coupling } \\ & \text { arms } \end{aligned}$ |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3D print housing and arms |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing software accuracy |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tuning stepper motors |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Final testing and tuning |  |  |  |  |  |  |  |  |  |  |  |  |  |
| STRETCH: two-phase algorithm implementation |  |  |  |  |  |  |  |  |  |  |  |  |  |
| STRETCH: install more webcams for cube state detection |  |  |  |  |  |  |  |  |  |  |  |  |  |
| STRETCH: RGB color sensing for redundancy |  |  |  |  |  |  |  |  |  |  |  |  |  |

