Muscle signals can pilot a robot

By Brandon Truong and Zetao Wu

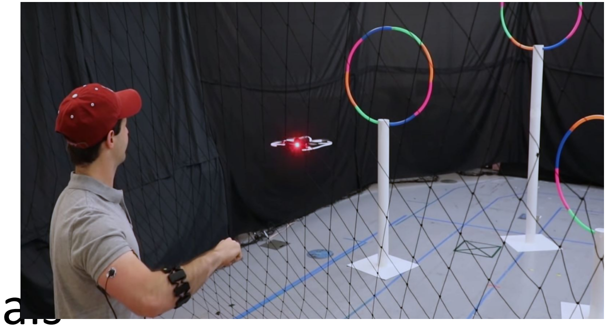
June 1, 2020

Who made it?

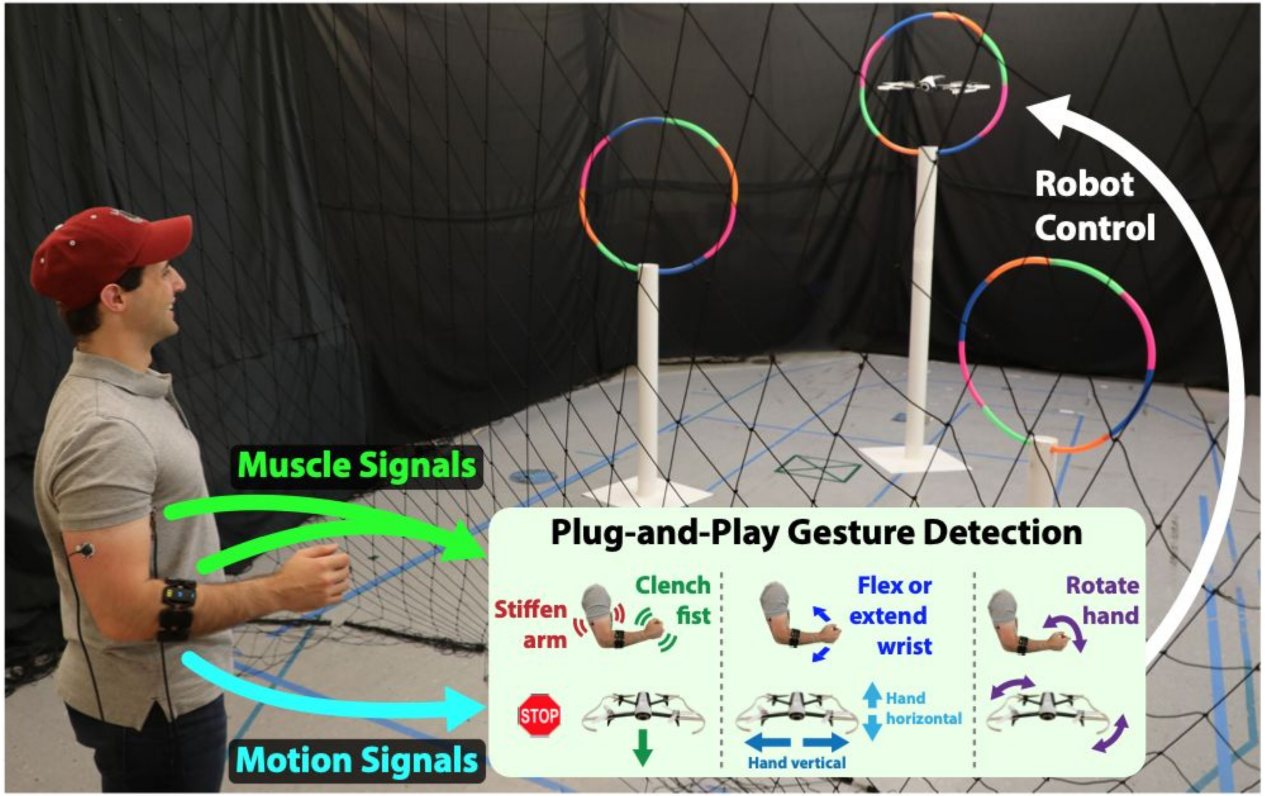
- MIT’s Computer Science and Artificial Intelligence Laboratory (CSAIL) came up with it
  - called “Conduct-A-Bot” which uses human muscle signals from wearable sensors to pilot a robot’s movement.
- Finished experiment during March 2020
  - Purpose to largely reduce the barrier to casual users interacting with robots
  - Experiment involved controlling a drone with muscle sensors

How does it work?

- Uses wearable surface electromyography (EMG) sensors and an inertial measurement unit (IMU) to detect muscle signals
  - Includes detecting arm stiffening, fist clenching, rotation of arm and left/right/up/down motion.



- Online clustering algorithm helps interpret these signals
  - Allows processing of gesture in real time without calibration and user data
  - Neural network of previous data supplements algorithm
    - Predicts wrist flexion and rotation from forearm muscle signals


Robot
Control
Muscle Signals
Motion Signals
Plug-and-Play Gesture Detection
Stiffen
arm
Clench
fist
Flex or
extend
wrist
Rotate
hand
STOP
Hand
horizontal
Hand vertical

Gesture Interpretation

Pros:

- Increased robustness by separating movement into easily interpretable gestures, thus improving accuracy.
- Increased deployability through reducing need of external equipment
- Requires only 2-3 wearable sensors rather than cameras and motion capture
- Less susceptible to environmental noise

Cons:

- Predefined gesture library sacrifices full control of movement for precision
- Wearing muscle sensors can be irritating during extended use

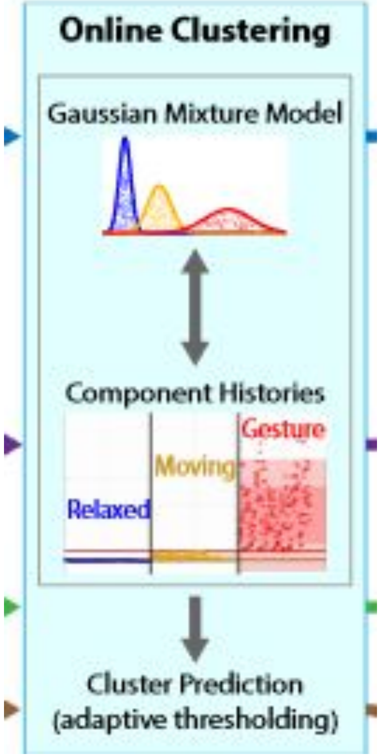
Intuitive and Force Recognition

<u>Advantages of Using Muscle Control</u>

- Easier to learn
- Easier to control for beginners
- You can feel how much force you are putting into an action

Online clustering algorithm

- Builds on Gaussian Mixture Models (GMM) to create a good model of gestures, movement, relaxation
- Streaming data online allows for no predefined GMM, dynamically creates a new GMM to fit the user from scratch



Software Overview

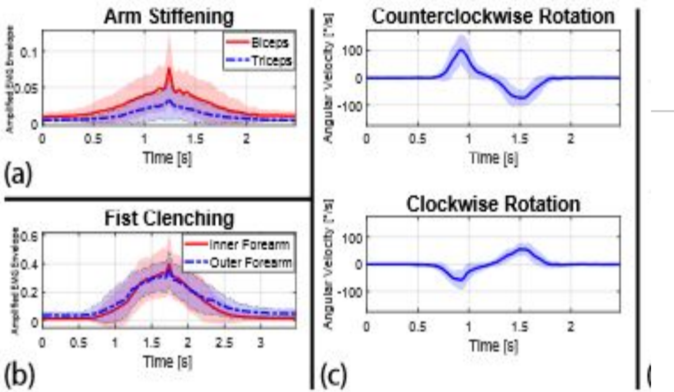
Algorithm 1: Online Clustering of Streaming Data

Input: n-dimensional signal sample $sample^{1\times n}$
Params: k, history length H, T_{refit}, $T_{initialize}$, $zscore\_bounds$
Output: cluster prediction $K_{predicted}$

Initialize $k+1$ rolling buffers: $histories^{(k+1)\times H\times n}$
while current time $< T_{initialize}$ **do**
  $histories[k+1] \leftarrow sample$ `// store as unknown cluster`
if T_{refit} seconds since last GMM fit **then**
  $allSamples^{H(k+1)\times n}$ = concatenate all $(k+1)$ $histories$
  gmm = `FitModel`($allSamples, k$)
  $K_{all}^{H(k+1)\times 1}$ = `PredictCluster`($gmm, allSamples[i]$)$\forall i$
  $histories[c] = allSamples[K_{all} == c] \forall c = 1 \ldots k$
  $histories[k+1]=[]$
$K_{predicted}^{1\times 1}$ = `PredictCluster`($gmm, sample$)
$histories[K_{predicted}] \leftarrow sample$
Function `PredictCluster`($gmm, sample$):
  $[posteriors^{1\times k}, zscores^{1\times k}]$ = `EvalGMM`($gmm, sample$)
  $K_{predicted}$ = cluster with $max(posteriors)$
  if any of $zscores$ violate $zscore\_bounds$ for $K_{predicted}$ **then**
    $K_{predicted} = k+1$ `// mark as unknown cluster`

- Buffer to store info before system is initialized
- Fit all data and predicts a good model based on it
  - Tests for z-score, statistic value for checking deviation

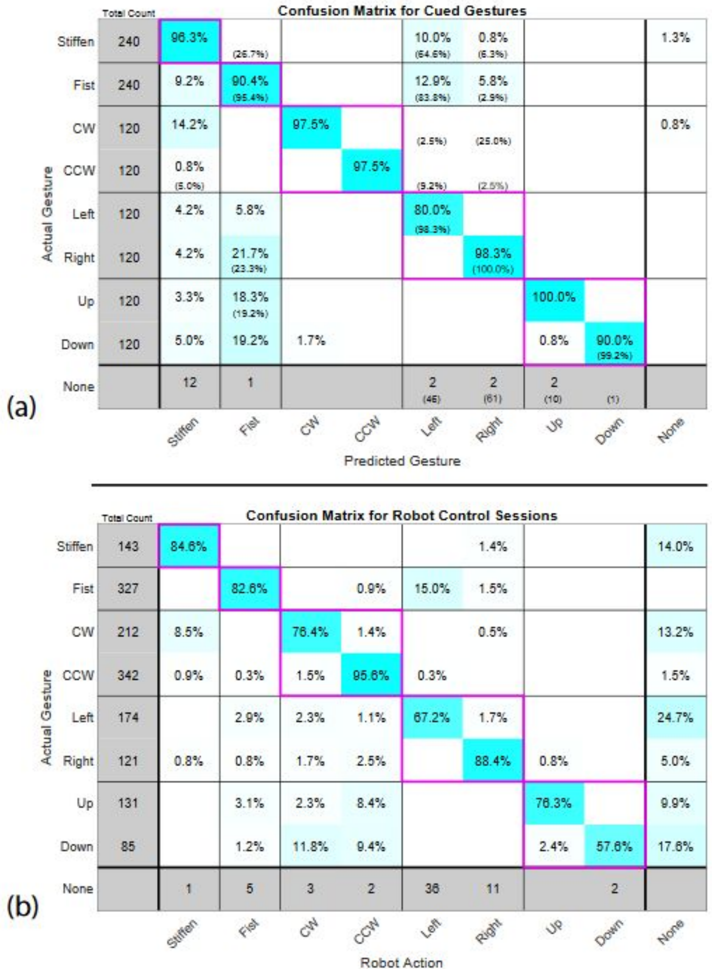
Hardware Overview



- Uses two EMG electrodes attached to upper arm
- Attached to triceps brachii and biceps brachii to detect muscle signals at upper arm
- 8 dry EMG electrodes on upper forearm samples muscle activity below the elbow
- IMU containing accelerometer and gyroscope detects rotation and acceleration of forearm.
- Upper arm sends data to data acquisition (DAQ) device and through USB while forearm sends data through Bluetooth to Simulink R2018b

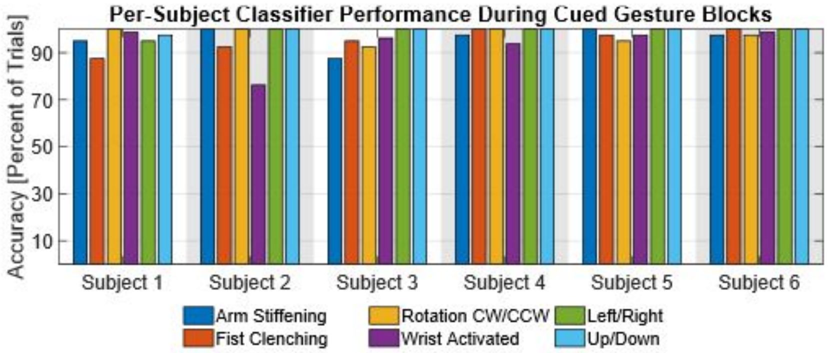
Results

- Classifiers identified 97.6% of 1200 cued open-loop gestures
  - Repeated patterns and gestures
- 81.6% of 1535 unstructured gestures was responded to by the robot (closed-loop)
  - Real life application of piloting a drone through hoops
  - Errors may have resulted from multiple gestures in a given time or change in handling (accelerating/rotating quicker than before)



Its Significance

- Reduces need for user to calibrate
- Increases range of possible motions by using muscle signals
    - Can tell how much force and torque to put into action



Possible Applications

- Remote handling
    - Can benefit those in hazardous jobs/ those that handle hazardous chemicals by improving safety through remote robotic movement without sacrificing accuracy
        - Ex. chemical waste management, industrial machinery operation, decontamination of rooms, etc.

Works Cited

DelPreto, Joseph, “Plug-and-Play Gesture Control Using Muscle and Motion Sensors”, ACM, March 23, 2020, https://dl.acm.org/doi/pdf/10.1145/3319502.3374823

Gordon, Rachel, “Muscle signals can pilot a robot”, MIT NEWS, April 27, 2020, https://news.mit.edu/2020/conduct-a-bot-muscle-signals-can-pilot-robot-mit-csail-0427