Gripping Motion of Robotic Arm using EMG Sensor

Raj Shah, rshah11@sfsu.edu, 922991378; Sayali Samjiskar, ssamjiskar@sfsu.edu, 922968147; Shalman Chakravarthi Tatigiri, statigiri@sfsu.edu, 922965469

***Abstract*— The hand, being the most dependable part of the human body, serves as the inspiration for this project. By utilizing Electromyogram (EMG) signals, which are generated upon muscle contraction, the feasibility of controlling the gripper and rotation is demonstrated. The project utilizes an Arduino as the main controlling unit to analyze and control the commands to the robotic gripper. Based on the received signal from EMG, the servo motor or robotic gripper is controlled accordingly. To facilitate data transmission, the NRF24L01 module is utilized for radio communication. With this setup, the project enables the control of motion such as the opening/closing of the robotic gripper, with the integration of sensors and Arduino-based control.**

I. Introduction

The hand is the most dependable part of the human body. It is the first body part used to perform tangible tasks. It consists of multiple phalanges (fingers) for grasping purposes. These phalanges allow the hand to accommodate various shapes of objects for a firm grasp. Electromyogram (EMG) generates a small signal amplitude upon muscle contraction. It is difficult to distinguish the significance of its EMG response but yet its control feasibility has been proven by many researchers. The electrode placement for signal extraction plays a major role in the significance of EMG signal reading [1].

One crucial aspect of the human hand's functionality is its ability to grasp objects of different shapes and sizes. This grasping mechanism relies on muscle contractions, generating electrical activity. In this project, we aim to explore the concept of controlling a robotic gripper by using EMG and gyro sensor signals. These signals which are in analog form, are converted into digital format so it can be easily processed by the Arduino. The Arduino analyzes the incoming EMG signals and interprets the movement of the robotic gripper accordingly. For communication between the two, Arduino and robotic gripper, NRF24L01 is used to facilitate real-time control and feedback [2].

II. Design And Implementation

A. Hardware Architecture

The hardware architecture combines EMG sensors, Arduino microcontroller, a robotic gripper/servo motor, motion sensors (accelerometer and gyroscope), and a wireless communication module (NRF24L01) to create a comprehensive control system. The integration of these components enables the system to interpret the user's muscle contractions, analyze hand position and movement, and provide precise control over the gripper's motion and rotation.

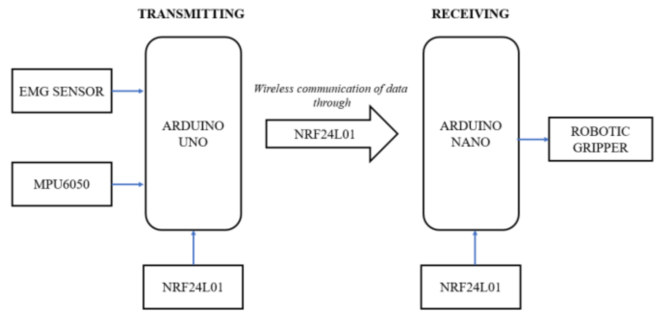


Figure 1 Block Diagram

B. Requirements

1. Electromyogram (EMG) Sensors: EMG sensors are essential for capturing the electrical activity generated by muscle contractions. These sensors are placed strategically on the user's hand to detect and amplify the EMG signals, which provide insights into the user's intentions for controlling the robotic system [3].

2. Arduino Microcontroller: The Arduino serves as the main control unit of the system. It receives the amplified EMG signals from the sensors, processes the data, and generates corresponding commands to control the robotic gripper and rotation. The Arduino's versatile I/O capabilities and programmability make it a suitable choice for interfacing with the various components of the system.

3. Robotic Gripper/Servo Motor: The robotic gripper or servo motor is responsible for physically manipulating objects based on the control commands received from the Arduino. The gripper may consist of mechanical fingers or an actuated mechanism designed to provide a firm grasp object. The servo motor enables controlled rotation of the gripper [3].

4. Accelerometer and Gyroscope: These motion sensors enhance the control capabilities of the system. The accelerometer measures the angle of inclination or orientation along the three axes, providing feedback on the hand's position or movement. The gyroscope measures angular velocity around the three axes, offering insights into the hand's rotational motion. These sensor inputs

enable more precise control of the servo motor's rotation and the gripper's motion.

5. NRF24L01 Wireless Module: The NRF24L01 module facilitates wireless communication between the Arduino and the robotic gripper/servo motor. It enables real-time transmission of control commands and feedback data, eliminating the need for physical wiring between the control unit and the robotic system.

C. *Detailed Implementation*

The detailed implementation involves several steps, including setting up the hardware components, programming the Arduino, and establishing the necessary connections.

1. Hardware Setup:
   - Connect the EMG sensors to the Arduino UNO board using appropriate analog or digital I/O pins. Follow the sensor manufacturer's instructions for wiring and connections.
   - Connect the accelerometer and gyroscope to the Arduino UNO using the required communication protocol (e.g., I2C or SPI). Make sure to connect the appropriate power and ground pins.
   - Connect the robotic gripper or servo motor to the Arduino Nano's digital I/O pins, ensuring that the control signals (such as PWM) are properly connected for controlling the gripper's motion and rotation.
   - Connect the NRF24L01 wireless module to the Arduino UNO and Nano, following the module's pinout and communication protocol requirements. Take care to connect the necessary power and data pins.

2. Software Development:
   - Install the Arduino IDE. Write a program to read and process the EMG sensor signals. Use the appropriate analog or digital input functions to read the sensor values and apply any necessary signal processing techniques.
   - Write code to interpret the EMG signals and map them to specific control commands for the gripper and rotation. Define the thresholds or patterns that correspond to different actions, such as grip strength or rotation speed.
   - Incorporate code to read data from the accelerometer and gyroscope, and implement algorithms to calculate the hand's position, orientation, or movement based on the sensor inputs.
   - Integrate the wireless module code to establish communication between the Arduino. Implement the necessary data transmission and reception protocols to send control commands and receive feedback data.

3. Calibration and Testing:
   - Calibrate the EMG sensors by performing controlled muscle contractions and mapping the sensor readings to desired actions. Adjust the sensitivity or thresholds as needed to ensure accurate and reliable control.
   - Test the system by executing various gripping and rotation actions using the EMG signals and sensor inputs. Verify that the gripper and rotation mechanisms respond appropriately to the user's muscle contractions and hand movements.
   - Fine-tune the control algorithms, thresholds, or mappings based on testing results to optimize the system's performance and responsiveness.

4. Enclosure and Finalization:
   - Design or acquire an enclosure to house the Arduino, sensors, and wireless module. Ensure proper spacing and protection for the components while allowing access to necessary ports and connectors.
   - Mount the components securely within the enclosure, taking care to organize the wiring and ensure proper connections.
   - Conduct thorough testing of the fully assembled system to verify its functionality, reliability, and safety.
   - Make any final adjustments or refinements based on testing results.

III. EXPERIMENTS

Firstly, the project started with the calibration of EMG sensor. Measured and recorded the EMG signals while the user performs gripping actions with different force levels. So, with this we calculated the threshold. Then, we analyzed the relationship between the EMG signal amplitude and the actual grip strength exerted by the robotic gripper. Plotted and compared the EMG signal amplitudes against known grip force values to establish a calibration curve.

Secondly, we integrated the sensors individually with the Microcontroller to check the accuracy and reliability. Then we combined the entire circuit. We measured and analyzed the response time of the system from the moment the user initiates a gripping action until the gripper mechanism responds accordingly. Evaluated the system's responsiveness by conducting tests that require quick and accurate grip adjustments. Quantify the delay between the user's muscle contractions (captured by the EMG signals) and the resulting gripper or movements.

Thirdly, subjected the system to various challenges which include hand positions at different angles, pickup and drop objects, moving objects from one place to another and also considering the object size and weight. Measured the system's ability to maintain accurate and consistent control. Identify any limitations or failure points and explore strategies to mitigate them.

Lastly, performed continuous transmission of EMG data to monitor and record the transmission errors (delay) during the data transmission process. Then calculated the error rate to evaluate the transmitted EMG data. Further, analyzed any patterns or trends in the errors and identified potential causes or areas for improvement.

IV. Results And Discussion

HandPosition(DOF)	Objects	Trial (pickup & drop)	Hit rate	Miss rate	Total Percent
0	Airpods	10	8	2	80%
30	Mobile	15	10	5	66.67%
45	Pen	22	21	1	95.45%
90	Screwdriver	13	9	4	69.23%

Table 1 Result Table

The information in this table includes trial results for picking up and dropping the relevant objects in various hand positions (referred to as DOFs). The hit rates, miss rates, and overall percentages are also shown in the table.

A particular hand position is represented by each row in the table. The columns are separated out as follows:
Hand Position (DOF): This term denotes the degree of freedom (0, 30, 45, or 90) that represents the hand's angle or position.
Objects: It describes the kinds of items that were used in the experiments.
Trial (pick up & drop): This indicates how many times the objects were picked up and dropped.
Hit rate: This statistic shows how many pickups and drops for each hand position were successful.
Miss rate: This gauges the quantity of dropped and unsuccessful pickups for each hand position.
Total Percent: This value represents the proportion of pickups and drops that were successful out of all the trials that were completed for each hand position.

Let's now analyze the information in the table:

1. Hand Position 0 (Airpods): This hand position was used for 10 trials of picking up and dropping Airpods.
Eight of those ten trials resulted in hits, while only two resulted in misses. The Airpods were successfully picked up and dropped in 80% of the trials, according to the 80% hit rate (8 out of 10 trials).

2. Mobile Hand Position 30: This hand position was used for 15 trials of taking up and dropping mobile phones.
Ten of those 15 trials resulted in hits, while only five resulted in misses. The mobile phones were successfully picked up and dropped in around 66.67% of the trials, according to the hit rate of about 66.67% (10 out of 15).

3. Pen Hand Position 45: 22 attempts were made to pick up and drop pens in this hand position. Of those 22 tries, 21 ended in success (hits), while 1 ended in failure (miss). The pen was successfully picked up and dropped in 95.45% of the trials, which is a very high hit rate (21 out of 22 trials).

4. Hand Position 90 (Screwdriver): In 13 trials, screwdrivers were picked up and dropped from this hand position.
Nine of those 13 trials resulted in hits, while four failed to do so (misses). The screwdrivers were successfully picked up and dropped in approximately 69.23% of the trials, according to the hit rate of 9 out of 13 trials.

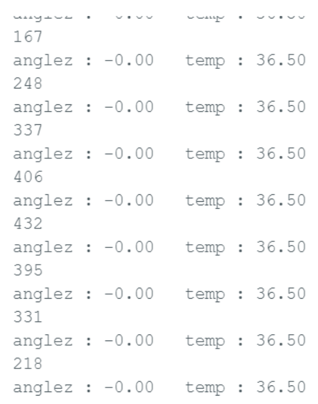


Figure 2 EMG data, hand closed

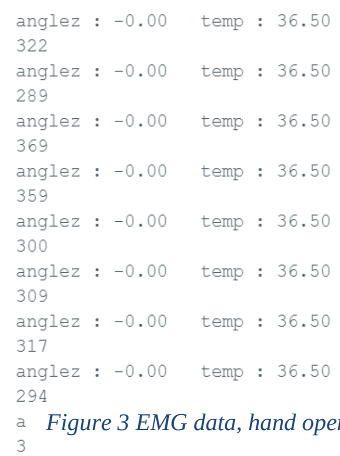


Figure 3 EMG data, hand open

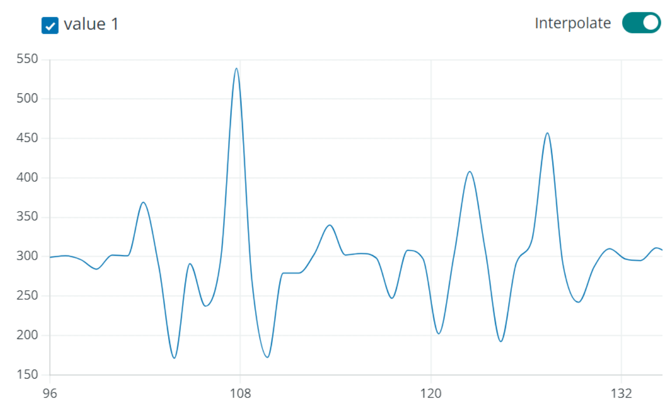


Figure 2 Waveform, hand closed

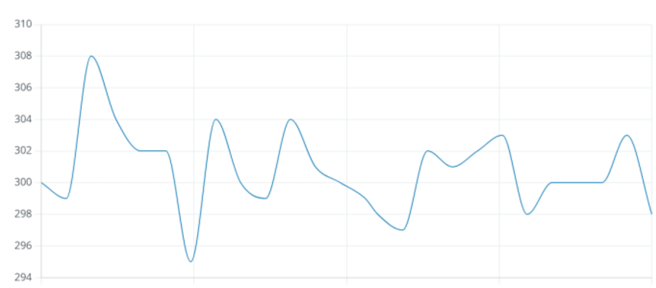


Figure 3 Waveform, hand open

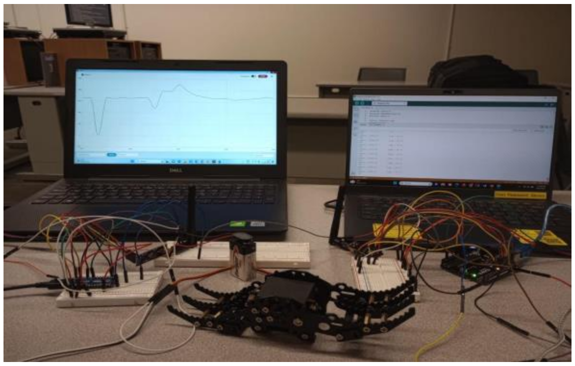
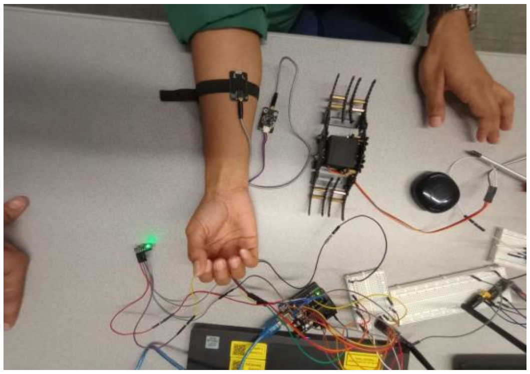


Figure 4 Experimental Setup



V. Roles of Team Members

Chakravarthi and Sayali started with the calibration, testing of the sensors, and system integration. Raj and Sayali worked on the coding part and debugged all the errors that occurred. Chakravarthi and Raj fixed the wireless communication delay problem and also created a video of the entire project. All three members of the group worked on the final report.

VI. Conclusion And Future Work

In conclusion, the table details the success rates of picking up and dumping different things while using various hand positions. It implies that while some hand positions may have higher failure rates than others, some hand positions—like Hand Position 45 with pens—yield higher hit rates, indicating superior success in controlling such things.

Acknowledgment

We would like to extend our heartfelt gratitude to everyone who supported us in completing our project. We are especially thankful to our project guide and instructor, Prof. Xiaorang Zhang for her constant guidance and motivation. Our team member's assistance and constructive criticism were invaluable throughout the project period.

References

[1] Muhammad Hadi Mat Rosly, Md Mozasser Rahman,, "Gripping Motion of Artificial Hand Using EMG Signal" , 2016 International Conference on Computer & Communication Engineering.

[2] T. O'Sullivan, D. E. O'Sullivan, and C. M. O'Sullivan, "Control of a Robotic Arm using EMG and Flex Sensors," International Journal of Advanced Robotic Systems, Vol. 11, No. 1, 2014.

[3] Parker, P. A. (2001). Myoelectric control: State of the art. Journal of Medical Engineering & Technology, 25(1), 10-16.

[4] Spong, M. W., Hutchinson, S., & Vidyasagar, M. (2005). Robot Modeling and Control. John Wiley & Sons.

- H. F. Machado, A. L. F. Loureiro, and G. A. Borges, "EMG signal normalization for controlling a robotic exoskeleton," IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 18, No. 4, pp. 347-355, 2010.

- NRF24L01 Datasheet: https://www.sparkfun.com/datasheets/Components/nRF24L01_prelim_prod_spec_1_2.pdf

- RF24 Library GitHub Repository: https://github.com/nRF24/RF24

- Arduino Official Website: https://www.arduino.cc/