Smart, Secure and Sustainable Robotic Hand

Dion Mariyanayagam
Department of Communications
Technology and Mathematics
London Metropolitan University
London, United Kingdom
dionysusmiroy@gmail.com

Dr Pancham Shukla
Department of Communications
Technology and Mathematics
London Metropolitan University
London, United Kingdom
p.shukla@londonmet.ac.uk

Abstract— **Robotic manipulators have a wide range of applications in many sectors from automation to medical surgery. Since decades, researchers and developers have been proposing creative designs in leveraging specific applications or to demonstrate proof of concepts in bespoke scenarios.**

This paper presents a novel smart, secure and sustainable framework for the design of a 3D printed robotic hand. The smart robotic hand, consisting of two elements: i) a user-wearable glove and ii) a mechanical hand, is a seamless entity in which the movement of the mechanical hand is controlled by the user-wearable glove. A real-time communication between the glove (server) and the mechanical hand (client) is facilitated through a secure wireless client-server model running on a commercially available microcontroller making it light-weight and cost-effective. In order to promote sustainable 3D printing, a conscious decision was taken to use recyclable and biodegradable Polylactic acid (PLA) based wood filament for the casing of the mechanical hand as well a cotton and linen based material for the user glove.

Keywords - 3D printing, Robotics, Security, Encryption, Client-server model, PLA, CAD, HMI, Smart system

I. INTRODUCTION

The origins of cybernetics started with Norbert Wiener's book - "Cybernetics, or the Control and Communication in the Animal and the Machine" published in 1948. It was stated that Cybernetics is "the scientific study of control and communication in the animal and the machine" [1]. The word itself comes from the Greek word Kubernetes (κυβερνήτης) which means to steer, pilot, or captain. It becomes 'governor' in Latin. The term first explored in Plato's Republic, Book I (Chapter 332e) as Plato describes the art of government through the thought experiment called Ship of Theseus.

James Watt and Matthew Boulton in 1788 invented one of the first cybernetics mechanisms to control the speed of the steam engine which was called the governor. "Cybernetics has the same root as government: the art of managing and directing highly complex systems" [2]. In general, control and communication between the self (animal) and the other (machine) is a crucial aspect in cybernetics. In fact, an amicable marriage of control and communication is at the heat of many robotic systems.

In proposing our robotic hand, we carried out research on several similar designs – mainly to evaluate their methodology and see the strengths and weaknesses of those designs. A study looking into the control and design of an under actuated artificial hand [3] had several useful features such as opposable thumb, 3D force sensors for the angular position using hall effect sensors and EMG controlled. However, it did not replicate a human hand as the design had two fingers and an opposable thumb. Additionally, the use of hall effect sensors would prove to be problematic as any electromagnetic interference would offset the results, causing

the hand to malfunction. Arguably, this could be filtered out using a Kalman or moving average filter. Nonetheless, the inclusion of an opposable thumb is a crucial feature of human biological hand, hence it has been adapted in our current project.

The organism inspired tactile sensory system approach [4] was a comprehensive study with a full substitution of the upper limb which was perceived quite natural. The strengths of this prosthetic hand were the ability to provide tactical sensation using on-off contact sensor arrays and triaxial force sensors and the ability to grasp objects. The key weakness was the system-feedback connected to a PC which rendered the system more suitable for a machine to machine interface (MMI) and not a human to machine interface (HMI). Even though researchers suggested a framework, it was never implemented an HMI.

A study in [5] explored an HMI based on neurophysiological peripheral nerve interfaces such as EMG-based control and other HMI based on optical sensors. This study did not make a practical impact in the engineering community due to the complexity, cost, and equipment needed for a viable option to explore muscle (e.g. EMG) controlled cybernetic hand. Similarly, [14] also investigates HMI using brainwaves but the key hurdle seems to be expensive sensors or a bespoke sensor design and development. To adapt to the current project, with a cost-effective methodology for hand capture, we envisaged flexible sensors instead of EMG.

Lastly, a study focusing on the Cyberhand to be a prosthetic for interfacing with humans [6] is mechanically similar to the study of [3] however, [6] focuses on the cybernetic hand to grip objects. A study of [7] investigates soft robots which may be a plausible alternative to the approach given in [6]. The placement of actuators in the forearm is a critical aspect that we could adapt in our design.

In the context of this study, the cybernetic hand is a 3D printed electro-mechanical robotic hand whose movements are controlled wirelessly through WiFi based connection by user's hand-glove through a client and server model. The human-machine interaction for the proposed robotic hand is investigated through an integrated approach involving sensors, servos, IoT devices, and 3D printed models. We would like to highlight that the missing feature, in all the studies we surveyed, was security. Our proposed robotic-hand-design harnesses the potential of IoT elements for encryption of the data sent and received for sucre wireless communication between the glove (server) and the mechanical hand (client). The HMI components of the robotic hand (i.e. a mechanical hand and a glove) are shown in Fig. 1.

II. RESEARCH METHOD

A. 3D Modelling and printing

The type of 3D printing used is FDM (Fused Deposition Modelling) that uses a thermoplastic filament, which is
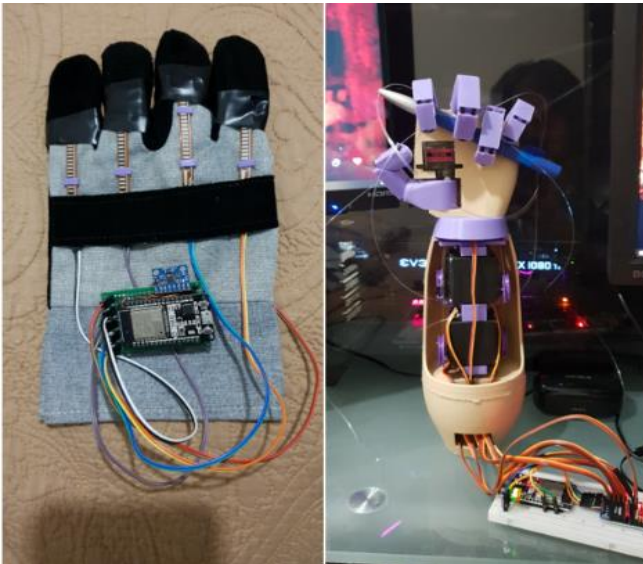
Fig. 1. Robotic hand elements: a) glove (server side), b) mechanical hand (client side).

heated to its melting point and then extruded layer by layer to create a three-dimensional object.

Designing of the 3D printed mechanical hand is modular and separated into several pieces: fingers, opposable thumb, palm, forearm servo holder, forearm, and forearm rotation. The use of tension to keep the finger in an upright position avoids having additional mechanisms to reset the fingers. The design of the opposable thumb included an indent for an M23-L micro servo horn and followed the same principles as the fingers.

The design of the palm has several tunnels that connect the top of the palm to the bottom of the palm (connected to the servos in the forearm). Additionally, the placement of the opposable thumb required an image of the micro servo placed on top of the CAD model then techniques called extrusion and intrusion [9] were used to achieve the indent of the servo.

The forearm servo-holder stacks two standard servos on top of each other to maximise the spacings of the servos with guiding holes for the strings (tendons) from the fingers to be attached to the servos, and the wrist securely places the palm to the forearm cover.

Lastly, the forearm rotation module allows the entire forearm to rotate around the y-axis. There are indents and holders of a standard servo and an indentation for easier management and access of wiring.

B. Electronic components

As outlined in Fig. 2, two ESP32 microcontrollers (MC) allowed the mechanical hand to communicate with the glove through a client and server model. MCs enabled reading of real-time data from the sensors placed on the glove to detect hand movements and then move the servos on the mechanical hand. They were also exploited for light-weight real-time encryption on both systems for wireless security.

As highlighted in the flow-chart of Fig. 3, server-side ESP32 microcontroller mounted on the glove collects the data from the flex sensors. A flex sensor or bend sensor is a sensor that measures the amount of deflection or bending as the sensor produces a resistance output correlated to the bend radius. The resistance produced follows a well-established voltage divider formula: $V_{out}=V_{in}(R1/(R1+R2))$ [10],
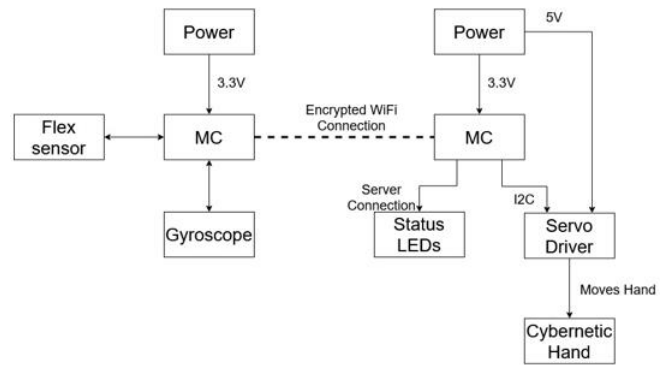

Fig. 2. Block diagram of the server-client model in which left side is server (glove) and right side is client (mechanical hand). Microcontroller (MC) is ESP32.

where, resistor one (R1) represents variable flex sensor resistor, and resistor two (R2) is a fixed resistor. The implementation of the flex sensor is to gather data of the user's hand when user bends a finger or fingers. This data received by the mechanical hand then aids to mimic the glove movements. The hand movement sensed through the flex sensor is used to calculate a rotational degree measure for driving a servo motor on the mechanical hand. A three-axis gyroscope sensor MUP 6050 with I2C connection is used to return the rotation information of user's hand around x-axis. The rotation around the x-axis is mapped to a degree value for the rotation/movement of the forearm servo of the mechanical hand.

As highlighted in the flow-chart of Fig. 4, the client-side ESP32 microcontroller connects to the PCA9685 servo driver via an I2C connection which controls five MG996R metal-geared servos controlling: thumb, index finger, middle finger, ring finger, and the forearm (wrist) rotation.
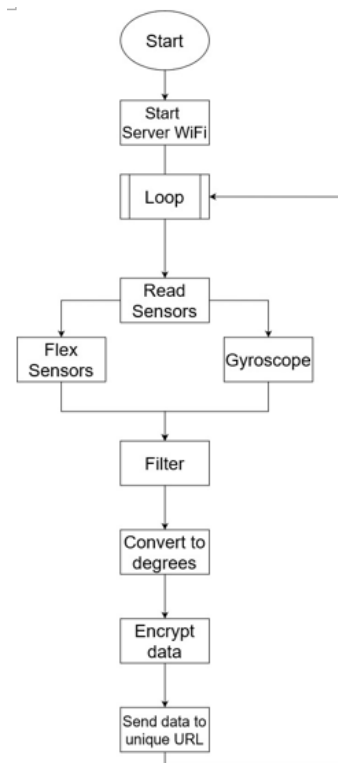

Fig. 3. Server-side flowchart demonstrating key operations of the microcontroller mounted on the glove.

In fact, Adafruit PCA9685 is a16-channel servo-driver that can control 16 servos using an I2C interface. To control a servo, an electrical pulse with modulation (PWM) signal (at 60Hz) with parameters such as the minimum pulse width, maximum pulse width, and repetition rate are sent through the control wiring. The microcontroller on the client-side (mechanical hand) accesses data stored on the server-side (glove) through the 'get' function of the HTTP.

A. Networking and security

The server placement is on the glove, which reads the real-time data of the hand movements. The data from the server is stored locally. The client (placed on mechanical hand) gets the data through HTTP 'get' requests, which allows the desired mobility and portability of the robotic hand. Due to HTTP being a text-based protocol, the encryption as discussed below could be light-weight and efficient in this model.

Encryption is the process of converting information or data into a code to prevent unauthorised access [11]. There are two types of encryptions: i) symmetric cyphers also referred to as secret key encryption which uses a single key, this key is known only to the sender who encrypted the data and whoever has the secret key to decrypt the data. ii) asymmetric cyphers also referred to as public-key encryption which uses two different but logically linked keys [12].
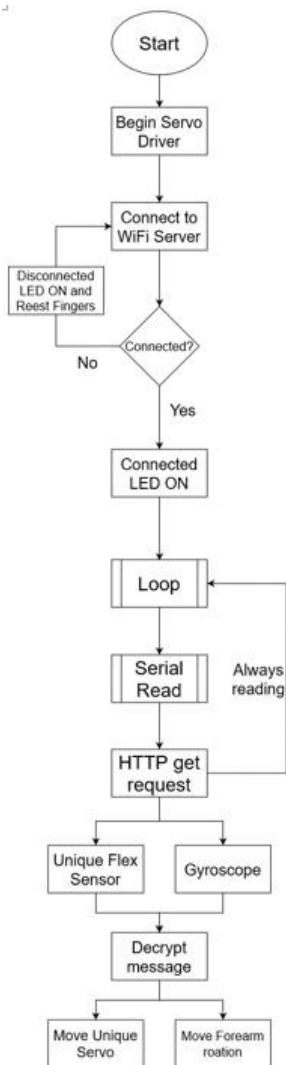

Fig. 4. Client-side flowchart demonstrating key operations of the microcontroller mounted in the mechanical hand.

Prime numbers are used to create keys as it is harder to reverse-engineer the encryption.

In our WiFi-based client-server model, there was no need to use an asymmetric cypher for the communication between two specific devices with the same secret key stored in their dedicated firmware. Such firmware driven arrangement makes our symmetric cyphers light-weight and friendly to real-time operation. Due to the limited processing power of the microcontroller, the encryption method used for this project is an XOR (Exclusive OR) cypher. The ASCII characters to be encrypted are converted to their binarised form and then XORed with the binarised form of the secret key. For ESP32 microcontrollers, this cypher offers a good balance between processing time and the effectiveness to hide the data from a 'man in the middle' attack.

III. TESTING, RESULTS AND DISCUSSION

A. 3D Modelling and printing

After number of experimentation, we found that the layer height of 0.2mm efficiently fused the layers while not becoming too coarse (e.g. 0.4mm scenario) and is not time-consuming compared to 0.1mm that would double the component print time.

Among various infill patters, Gyroid infill pattern is used as it is a naturally occurring pattern found in cells. In most cases, it provides robust backbone for any component keeping the infill density low without sacrificing strength and overusing material resource. Extruder temperature of 230 °C set to the maximum for Polylactic acid (PLA) filament allowed the flowrate to be constant and prevented any clogging.

Lastly, we used the build plant adhesion with a brim to keep the component in place with a margin of expansion from the component. This ensured increased surface area and decreased the likelihood of the component from displacing itself.

B. Electronic components

The placement of the flex sensors allows the wires to the microcontroller without disrupting the user. The flex sensor has a conductive paint, and its construction is similar to a voltage divider rule governed by $Vout = Vin(\frac{R1}{R1+R2})$ where R2 is a fixed and R1 is variable. Table I shows the measurements around flex sensor with respect to its bend in degrees assuming R2=10KOhms and Vin=3.3V.

The readings on the analogue pins correspond to the degrees of bend from 0 to 180. The rotation of the user's forearm around the x-axis is detected by a three-axis gyroscope. As expected, real-time raw data had a lot of random noise in form of spikes, corrupting the general trend. To get rid of any random anomalies, we used commonly used signal processing technique known as moving average filtering.

TABLE I. FLEX SENSOR READINGS

R1 (Ohms)	Vout (V)	Bend in Degrees	Pin Reading
30,000	2.475	0	2,800
45,000	2.700	30	2,893
60,000	2.829	60	2,964
75,000	2.912	90	3,035

The moving average filter takes one thousand real-time readings of movements around x-axis detected by the gyroscope and sums all the values dividing them by the number of readings taken. Mathematically, it follows:

$$\frac{1}{n}\sum_{i=1}^{n} Ai \qquad (1)$$

Where, n = length of an array, Ai = i_{th} value of the reading.

These readings are read by the client in form of encrypted message through HTTP 'get' request URL and are stored it in a buffer. The readings are then decrypted to get values in expected integer range to drive the servo accordingly. In a simple, test gyroscope was placed still on the desk, and the filtered and unfiltered data were printed out on the serial monitor as well as on Microsoft Excel to generate the stacked column bar chart.

Fig. 5 shows filtered and unfiltered x-axis data from the gyroscope. The light-grey bars represent unfiltered data which ranges from +50 to -540. The filtered version of the data shown in the dark-grey bars ranges from +10 to -108. We noticed that the filtering improves sporadic variations by 17%. It is evident that a bigger array provides better smoothing of the data. However, a bigger array requires more processing time for the microcontroller which deteriorates overall response of the system. Therefore, a balance was made between the processing time and amount of filtering.

C. Networking and security

Broadly, there were three stages to test the client and server model. Firstly, it was checked if the data stored on the server can be acquired by the client. To test this server kept keep counting, and the client read the count. In the setup, 'this' function returns the value and displays it on the webpage URL called 'count'. Secondly, the code stored on the client machine was executed with parameters from the server. This was completed by sending a command to turn on and off the inbuilt LED on the client ESP32 from the sever. The client listens to each URL of the buttons, and if the correct message appears, then the LED activates. Lastly, the code from the client was executed with real-time input from the server. By recording the real-time data from the flex sensor and saving it locally to the server allows the client to access the data through the 'get' function of the HTTP library.

For actuation, the client reads the URLs and moves the servos in accordance with the degree values communicated by the client-server model. Analytical testing of the XOR encryption was through a selected key "D". The key "D" in ASCII binary is 01000100. We tested this key to encrypt the message "hello".

Each character of the message is XORed with the key resulting in the encrypted message. Decrypting the message requires the same process on encrypted message. Table II shows encryption process for the given message "hello". As you can see that the encrypted messaged is ",!((+". The theoretical results were tested through our cypher code running in real-time on ESP32 microcontroller with results displayed on serial monitor.
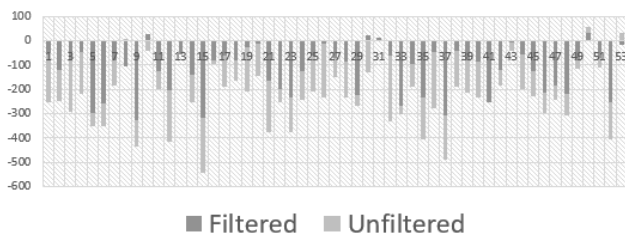

Fig. 5. Comparison of filtered and unfiltered x-axis data from the gyroscope

TABLE II. MANUAL DECRYPTION

D	01000100
h	01101000
Output = ,	00101100

D	01000100
e	01100101
Output = !	00100001

D	01000100
l	01101100
Output = (	00101000

D	01000100
o	01101111
Output = +	00101011

The deployment of the "strlen()" function returns the length of the desired message that allows a for loop iterator to go the message one character at a time to be XORed with the key. From server-side, all messages are put through the encryption function and then sent to the URLs. A pseudo-code for the server-side encryption is shown in Fig. 6.

To further enhance wireless security, we tested different encryption methods such as block cyphers that usually incorporate basic XOR methodology described earlier. Cypher Block Chaining (CBC) adds a feedback loop for cryptography. CBC is achieved by first encrypting the plaintext using an XOR cypher. However, a single bit error in the cyphertext will still cause the entire block to produce a wrong plaintext. As we have used microcontroller as a server and for signal processing and encryption, it was observed that the light-weight XOR cypher was more appropriate for the real-time response for the robotic hand. We believe that powerful microcontrollers and multi-processor operation may support advanced encryption methods.

```
1. String function called buttonOne()
2. {
3.   Read the button pin and store it in buttonState1
4.   if (buttonState1 is HIGH)
5.   {
6.     Char array called confirm which stores value ON1
7.     Char value called xorKey stores value D
8.     counts the length of the message and store it as an
   integer called len
9.     literates through the message character at a time
10.    {
11.        XOR each character with Key for the confirm array
12.    }
13.    Serial print confirm
14.    return the String confirm
15.  } else {
16.    Char array called decline which stores value OFF1
17.    Char value called xorKey stores value D
18.    counts the length of the message and store it as an
   integer called len
19.    literates through the message character at a time
20.    {
21.        XOR each character with Key for the decline array
22.    }
23.    Serial print decline
24.    return the String decline
25.  }
26. }
```

Fig. 6. Pseudo-code for server-side encryption

IV. Conclusion

To conclude, we presented a novel framework for the design and control of a 3D printed robotic hand that allows light-weight, real-time and secures wireless data communications between the server (glove) and the client (hand). Our cost-effective and sustainable design relies on commercially available microcontrollers and electro-mechanical components.

V. Further work

We envisage that such design efforts will contribute towards more advanced designs of the robotic hands with potential applications in areas such as psychotherapy and rehabilitation, controlled operations in hazardous environments, motion models for cinematography, and immersive virtual reality environments.

Acknowledgment

References

[1] N. Wiener, Cybernetics: or Control and Communication in the Animal and the Machine, Cambridge, Massachusetts: MIT Press, 1948.

[2] F. Heylighen, "History of Cybernetics and Systems Science," 24th October 2000. [Online].
Available: http://pespmc1.vub.ac.be/CYBSHIST.html. [Accessed 10th April 2020].

[3] H. v. Foerster, Cybernetics : circular causal and feedback mechanisms in biological and social systems , New York: Josiah Macy, Jr. Foundation, 1953.

[4] M.C. Carrozza, G. CappIello, E. Cavallaro, S. Micera, F. Vecchi and P. Dario, "Design and control of an underactuated cybernetic artificial hand," Proceedings World Automation Congress, 2004, Seville, Spain, 2004.

[5] B.B. Edin, L. Beccai, L. Ascari, S. Roccella, J.J. Cabibihan, and M.C. Carrozza, "Bio-inspired approach for the design and characterisation of a tactile sensory system for a cybernetic prosthetic hand," em Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006., Orlando, FL, USA, 2006.

[6] Silvestro Micera et al., "On the Use of Longitudinal Intrafascicular Peripheral Interfaces for the Control of Cybernetic Hand Prostheses in Amputees," IEEE Trans. Neural Syst. Rehabil Eng. 2008 Oct; 16(5): 453-72.

[7] Z. Jianshu, C. Xiaojiao, C. Ukyoung, L. Jui-Ting, C. Ching, C. Yonghua, H. Yong, W. Zheng, "A Soft-Robotic Approach to Anthropomorphic Robotic Hand Dexterity" Proceedings 2019 IEEE Access (Volume 7), Hong Kong, China, 2019.

[8] M.C. Carrozza, P. Dario, F. Vecchi, S. Roccella, M. Zecca, and F. Sebastiani, "The Cyberhand: on the design of a cybernetic prosthetic hand intended to be interfaced to the peripheral nervous system," Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453), Las Vegas, NV, USA, USA, 2003.

[9] Ultimaker, "Technical data sheet PLA," 3rd September 2018. [Online]. Available:
https://ultimaker.com/download/74599/UM180821%20TDS%20PLA%20RB%20V10.pdf. [Accessed 22nd June 2020].

[10] D. Norman, The design of everyday things, New York: Basic Books, 2013.

[11] T. Duncan, Electronics for today and tomorrow, 1st ed., London: John Murray, 1985.

[12] P. M. Monika Agrawal, "A Comparative Survey on Symmetric Key Encryption Techniques," International Journal on Computer Science and Engineering (IJCSE), vol. 4, no 1, pp. 870-885, 2012.

[13] G. C. Kessler, "An Overview of Cryptography," Embry-Riddle Aeronautical University, Daytona Beach, Florida, 2019.

[14] O. Rușanu, L. Cristea, M.C. Luculescu, S.C. Zamfira., "Experimental Model of a Robotic Hand Controlled by Using NeuroSky Mindwave Mobile Headset", Proceedings 2019 E-Health and Bioengineering Conference (EHB), Iasi, Romania, 2020.