Evaluation of 6 DOF Robotic Arm Using Leap Motion Sensor

Mohanad Mezher
The University of Mashreq
Baghdad,10011, Iraq

Amjed Razzaq AlAbbas
AlSalam University College
Baghdad,10011, Iraq

Received: August 2, 2022. Revised: January 6, 2023. Accepted: February 9, 2023. Published: March 6, 2023.

Abstract— **Due to their increased functionality, robotic arms provide a well-organized method for developing assistive devices. By 2035, statistics indicate that half of Germany's population will be over the age of fifty, and every third person will be over the age of sixty. These aging societies face numerous obstacles when it comes to performing basic activities of daily living, or "ADLs." A growing body of research is focusing on Ambient Assisted Living, or "AAL," as a novel approach to addressing the needs of elderly people. A critical objective of AAL is to improve the quality of life for the elderly and disabled and to assist them in maintaining an independent lifestyle. Robotics and technology-enabled environments will be critical in enabling elderly and physically disabled people to maintain a self-determined, independent lifestyle in their familiar surroundings. The purpose of this article is to propose the implementation of a novel intuitive and adaptive manipulation scheme by creating a human-machine communication interface between the Leap Motion controller and the 6-DOF Jaco robotic arm. An algorithm is developed to optimize the mapping between the user's hand movement and the Jaco arm, as tracked by the Leap Motion controller. By constantly adapting to the user's hand tremor or shake, the system should enable a more natural human-computer interaction and smooth manipulation of the robotic arm. The implementation would significantly improve people's quality of life, particularly those with upper limb problems, by assisting them in performing several essential Activities of Daily Living "ADLs." The applications of this human-robot interaction will be discussed in relation to Ambient Assisted Living, with the introduction of several use case scenarios.**

Keywords—6 DOF, Leap Motion, Robotic Arms, Sensors.

I. INTRODUCTION

Meanwhile, there is interaction between humans and computers, which is largely derived from the use of pointing devices or typewriter-style devices. Additionally, as demonstrated in [1,] robots and implantable devices equipped with WBANs are critical for human health monitoring. This type of relationship, in which accepted methods of treatment involving the hands are limited, may result in the complication of simple tasks. One of the overcomplicated control examples is rotating a three-dimensional object [2]. A client must capture the object and navigate around it using a mouse on a personal computer, which will be derived only in a two-dimensional space. The alternation process represented by the mouse movement is imperceptible to humans, and clients require some time to understand how it will work in the real world. However, the rotation function is natural, and thus it is straightforward to rotate the objects in the desired direction using the hands as shown in [3]. When we discuss robots, societies tend to believe that robots are only useful for custom in manufacturing or for inventors to investigate new machineries. On the other hand, the primary purpose of robots is to assist humans in assuming responsibility for their jobs, whether in businesses or simply performing routine domestic tasks. To overcome the conventional wisdom that "robots are only for activities," the internet will be used. Our task in this work is to increase the organization of a robotic arm, as discussed in [4]. The advancement of robot arms that can be controlled remotely via the internet by a computer. This robot can be used to demonstrate that a robot can be used in a home environment to perform routine human tasks. The robotic arm is powered by an arduino Uno, which is connected to the internet via an arduino ethernet shield. This project required two types of examinations: servo motor examination and exactness check [5]. The correctness examination reveals that the true productivity of the servo motor when connected to the arduino Uno via the internet is between 97 and 99 percent. The robot demonstrated that the prepared remained positive. This client-friendly robot is expected to bridge the divide between robots and domestic errands [6]. As a result, there must be a demand for more conventional human-computer organizations. One of the proposed approaches involves creating a sign by hand and having it interpreted by a computer. The use of hands across the human-computer interface is justified by the fact that they are still used for non-verbal communication, such as body language or cipher. Additionally, scheming tasks accomplished with hands, in fact the entire world, can be understood as a sequence of signs and used as computer input. Utilization of the table upper article is one of the primary routine responsibilities when utilizing intelligent robots. It combines the robot's capabilities for visualization, object gratitude, image processing, and hand-arm exploitation. Nonetheless, the actual covered situation is significantly more complicated than the trial situation. Occasionally, the concept of the robot is unable to provide us with sufficient data for successfully performing a few difficult tasks, such as picking, placing, or assembling nearly minor items. In those instances, it will be difficult to accurately slice two objects when they are so close to one another; additionally, a few obstruction belongings frequently occur in a real indoor environment. Thus, the tele-operative manifestation technique is a well-organized method for surmounting these obstacles [7]. The latest sensors provide data that can be used to effectively recognize and control a computer. At the moment, there are a plethora of devices that provide useful data for signal identification. We can use Microsoft Kinect as an example of a manager of this type. It displays a three-dimensional spot cloud of the user's experiential vision, but was designed for applications that interpret the user's entire body movement. As a result, it falls short of the required accuracy for hand sign detection [8]. Another device that is used to track the movements of a hand and fingers is the Leap Motion Controller, which was released to the market in July 2013 by Leap Motion, Inc. The Leap Motion is a small device that fits in front of a computer. It denotes incredible three-finger detection accuracy of up to 0.01 mm. The controller provides location information for each finger and hand detected in the observed space. The SDK pre-installed on the device enables the device to recognize three pre-defined signals: sphere movement with one finger, swipe exploit, and tapping on the virtual keys. The Leap Motion Controller provides us with information about each sensed hand. This device transmits data at a rate greater than 100 Hz. The Leap Motion Controller is a sensing device that has the potential to disrupt human-computer interactions. This is why the leap motion sensor is the ideal choice for controlling the project; data from the sensor is sent to the arduino microprocessor, which controls the motors that move the arm in the desired direction [9]. For several years, these expression methods were used on developed robots. For example, a controller equipped with switches or a six-dimensional mouse is used to control the robot and communicate key positions and orientations to the robot, allowing it to plan its path and accurately reach each key position with the desired orientations and perform a smooth movement. However, this type of demonstration method's interface is inefficient for a smart robotic system. And because the majority of schemes simply record the robot's position and location without elaborating on them, these schemes are unsuitable for additional complex tabletop object manipulation tasks. For substation, an additional accepted technique based on a kinesthetic edge is used. One is aware of how to slog the robotic arm in order to track his activities, as demonstrated by [10]'s research on humanoid robots. However, this procedure also applies to route following rather than signal gratitude. Additionally, this is a unique interaction regulation technique in which humans work in a similar environment to the robot. As a result, it is rarely used in unfavorable humanoid environments. As a result, non-contact tele-control methods are more suited to these circumstances. For robotic systems, for example, some mechanical, optical tracking, or vision-based master-slave devices and tele-operation systems are being developed. In comparison to mechanical devices, optical and vision tracking systems are less expensive and easier to mount in a variety of environments. arduino is an open-source prototyping platform that is based on (simple-to-use) hardware and software. arduino boards are capable of reading inputs from sensors, buttons, or Twitter messages and converting them to outputs - starting a motor, spinning an LED, or 4 displaying a few things online. As mentioned previously, a robotic arm has been enclosed a variety of meadows for manufacturing production, medical treatment, security control, and other applications [20]. It performs the functions of an assistant, an operative, or even a work associate, similar to "Jarvis." DFLG 6DOF is a bionic robotic arm that is equipped with six servo motors that correspond to the arm, elbow, and wrist (2 degrees of freedom), five joints, and a rotating base. Each joint has a certain range of motion, and the base can be rotated about 180 degrees. All actions can be controlled via PC software with the 24 channel Veyron servo controller, which supports online clearing and wireless control. In a nutshell, this is a high-cost routine with an easy-to-use robot arm. It is an excellent robot expression educational stage and do-it-yourself robotic arm. The servo can be connected directly to the arduino IO expansion shield or the Romeo robot microcontroller, and the arduino servo library simplifies the process significantly more than previous works in the same area.

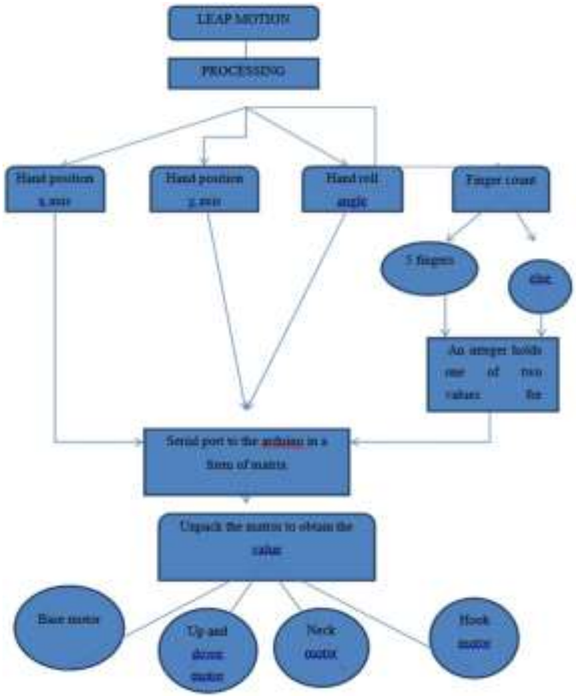

Fig.1. Block diagram of leap motion.

II. LEAP MOTION

As you can see from the hardware perspective, the leap motion controller is quite simple. The device consists of two cameras and three electromagnetic LEDs. The Path Ultraviolet bright range has a wavelength of 850 nanometers, and those nanometers are visible outdoors. The Leap Motion controller is a small USB peripheral device that is intended to be placed on a physical desktop, facing upward. Additionally, it can be mounted on top of a virtual reality headset. The device examines an approximately semi-circular region to a distance of approximately 1 meter using two monochromatic infrared cameras and three infrared LEDs. The prototype LEDs emit significantly less infrared light, while the cameras reproduce nearly 200 frames per second. This is followed by transmission via USB cable to the host PC, where it is analyzed by the Leap Motion software, which uses "complex maths" in an unspecified manner to create 3D point data by evaluating the 2D frames produced by the two cameras. In a 2013 study, the controller's overall regular accuracy was determined to be 0.7 millimeters [11]. The simplified examination portion and advanced declaration of the expedient distinguish the creation from the Kinect, which is more suitable for entire figure tracking in an interplanetary the size of a bodily area. According to a statement to CNET, the controller is capable of performing tasks such as determining the path of a web, increasing the visibility of signs on plans, sketching with extraordinary accuracy, and utilizing compound 3D information imaginings. Leap Motion initially distributed thousands of components to developers tasked with the responsibility of developing presentations for the expedient. Originally transported in July 2013, the Leap Motion controller remained. Leap Motion released a significant beta update to its core software in February 2016. The software, dubbed Orion, is designed for pointer pursuing in effective realism [12].

Fig.2. Typical leap motion

A. Processing

The Leap motion detects and routes information through hands, fingers, and finger-like tools. The device activates with high accuracy and a consistent frame rate when it detects a familiar environment. The leap motion software verifies the items practically within the device's field of view. It distinguishes between tools, fingers, and human hands, indicating both fixed locations and motion. The Leap field of view is shaped like an upturned pyramid and is centered on the device. The useful range of the Leap is approximately 25 to 600 millimeters above the device (1 inch to 2 feet). Axis of Hand Positions (x and y) as showed in Fig. 3.

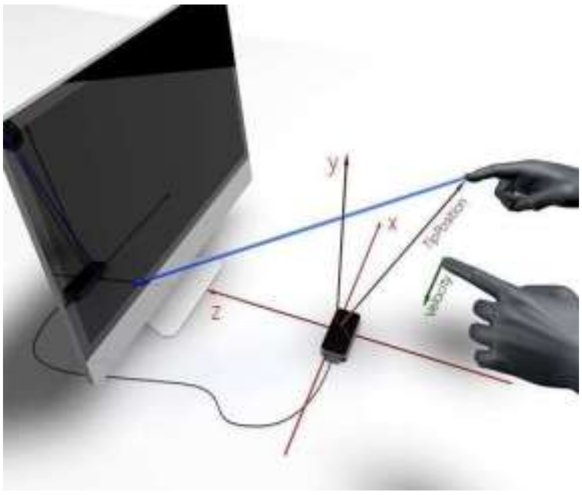

Fig.3. Hand positioning

B. History of Robots and Robotic Arms

The primary programmable robotic device is invented through George Devol. He coined the period in 1954 of the Universal Automation. Two men one is called Devol and the other is Joseph and he was an engineer Engel Berger procedure the biosphere's first robot establishment, Unimation in 1956. Unimation is obtained by Condec companies and improvement of Unimate Robot organizations starts. American engine and Foundry, in a while identified as AMF Establishment, markets a robot, known as the Versatran, intended by Harry Johnson and Veljko Milenkovic in1960.The initial manufactured robot was online in a General Motors sports car factory in New Jersey. It was Devol and Engelberger's UNIMATE [13]. It achieved spot welding and removed die castings in 1962 German corporation for robots, KUKA, produced the earliest industrial robot with six electromechanically determined axes known the Famulus in 1973.A robotic arm that completed tiny pieces' assembly with criticism from tap and force sensors was planned. Professor Scheinman, the one who invented the Stanford Arm, forms Vicarm Inc. to promote an edition of the arm for engineering applications. The new arm is controlled by a minicomputer in 1974.In 1495, Leonardo da Vinci made the extended and multi-colored antiquity of robots. He conceived a multifaceted social robot that it shows in his draughts possibly will execute humanoid similar motions such as meeting up, moving its hands and falsification its skull and neckline. The robot will be shown to be a soldier or fighter, wearing clothes in German-Italian medieval shell. This was merely one of the quite a few hundreds of designs and practical drawings experienced again within the pages of two establish documents in 1950.There are quite a few specimens of robotic machines and motorized progress in mutually very ancient and extra contemporary past, but the initial really up-to-date robot to stand extra to the antiquity of mechanical device was designed by George C. Devol in 1954 as I mentioned before. This machine existed called Unimate. In 1961, Unimate became the earliest industrial robot to labor at the congregation line of a New Jersey General Motors plant. Programmed to take die castings from machines and execute fusing on auto bodies, Unimate was clearly very far from the idealistic notion of androids, or human-like robots [14].

Fig.4. Complex humanoid robot

C. Leap Motion History

To begin, allow me to define leap motion and what I mean by the term "MOTION," and then I will discuss the entire history of leap motion. The Leap Motion is a technique that utilizes the user's fingers to plot a route around the desktop without requiring the user to touch any physical switches; it enables the client to control the scheme solely by monitoring the user's finger waves [15]. This concept of a movement regulator is unique in that it has persisted in production for an extended period of time, and this workout of a person as the manager dates all the way back to the Industrial Revolution. Though, during that time period, there were no computers or other modern devices, we will be able to subordinate ourselves to the modern wave and assume responsibility for using the devices. The earliest motion control devices utilized a foot holder that was measured by the user to terminate a procedure mechanically via pulleys and engines. Employers' inability to develop programmed is what inspired societies to figure out and develop the wave control skills that are now in use. Engineers in the late 1800's used the push to develop these methods in order to provide customers with the first electronic technologies. Edison's DC Producer in the 1870's, communal control and Tesla's AC motor in the 1880's, and the most primitive electrical tool in the 1890's are all examples of these mechanisms. By 1915, these projects demonstrated the feasibility of electronic laundry machines and refrigerators. In 1927, Harold Black made a significant contribution to the performance of computerized developments by presenting the theory of undesirable response in loudspeakers. He was not the first human to propose the impression of response, but he did conclude that an insufficient amount of additional feedback from the amplifier could be used in the input formula. In the 1940's and 1950's, it became obvious to engineers that mathematical considerations could be used to plan and educate wave regulator procedures. (Motion Control's History) The subsequent epochs provided these engineers with the opportunity to learn, beginning with interplanetary transportables and combat. These proceedings enabled engineers to create several difficult procedures capable of controlling procedures in an infinitely enhanced manner. Additionally, this era established the foundation for microcontrollers, which enabled all of the functions of a typical manager to be performed in a significantly smaller component. In the 1990s, DSP-based wave governor crops enabled complex signal summarization and numerical communication within sequential networks. As a result of the design of fiber optic communication appearances, much faster communication progressions became possible [16]. Now I'm returning to my primary subject; the leap motion is the central device for this project; the technological aspects of Leap Motion began in 2008, concurrently with co-founder David Holz's pursuit of a Ph.D. in mathematics. Following an early angle improvement, Holz co-founded the company with the assistance of his friend Michael Buckwald in 2010. To be more precise, in June 2011 Leap Motion raised a $1.3 million seed funding round led by Highland Capital Partners in May 2012, with participation from venture capital firms Andreessen Horowitz, Founders Fund, and SOSV, as well as a number of angel investors. In January 2013, Leap Motion announced an additional $30 million series B round of financial support. as of 2010 Later, while working in silence, widely Leap Motion, formerly known as The Leap, was declared its initial invention to the entire world on May 21, 2012. In October 2012, the group launched a software developer program and distributed approximately 12,000 units to developers interested in submitting submissions for the expedient [17]. Although it was recommended to launch the device in May 2013, full-scale shipping was delayed until July. In March 2014, TechCrunch reported that Leap Motion had sold nearly 500,000 units on markets, falling far short of the initial opportunity; as a result, Leap Motion announced layoffs for 10% of its workforce, primarily in sales and marketing. In May 2014, Leap Motion released a public beta version of its version 2 software to developers. This is the same version that was used in this research.

III. General Theory and Main Components

A. Methodology

This article discusses the fundamentals of controlling an uncomplicated yet useful robotic arm. The robotic arm is a six-degree-of-freedom arm that is controlled by an arduino device with a well-matched board. We have connected the Leap Motion to the robotic arm as an input. It can visualize the input in real time as motion in the metal arm. Due to the processing library, it was possible to integrate the Leap Motion controller with the processing IDE (Integrated Development Environment) (mentioned in the code). The servo data was sent to the arduino well-matched board via the processing app since we used servo motors in this work in a very simple format: servo number,servo angle>. To illustrate, send the command "0, 90" to cause servo 0 (the base) to rotate to the heart location.

B. Servo Motors

A servomotor is a rotating actuator or linear actuator that enables precise control of the location, speed, and precipitation of a pointed or linear spot. It consists of a fitting motor coupled to a position sensor. Additionally, it requires a relatively advanced controller, which is frequently a dedicated component designed primarily for use with servomotors. Servomotors are not a recognized class of motors, despite the fact that the term is frequently used to refer to a motor suitable for use in a closed loop control system. There are numerous applications for servomotors in fields such as robotics, technology, and automated manufacturing. There are numerous types of servos based on their rotation angle, but the one used in this project is a 180-degree servo [18].

Fig. 5. Servo motor

C. 6 DOF Robotic Arm

The robotic arm is a piece of mechanical equipment that is frequently and widely used in robot knowledge fields nowadays; you can see it in engineering manufacturing, health check dealings, entertainment services, education, armies, and space examination fields. Over the preceding decades, designing and managing a robotic arm has not been an easy task [19]. Numerous thoughts must be tended to concurrently with the scheming and calculating robotic arm. Additionally, unlike robotic arm design, it is possible that a modified controller explanation will result. Additionally, it is difficult for the robotic arm to follow the assigned geometry trail with high precision and accuracy. The research that we are currently conducting details the development and progress of a 6-DOF (degrees of freedom) PC-Based Robotic Arm (PC-ROBOARM). The primary context for this learning is a six-degree-of-freedom robotic arm, which is demonstrated as three relationships, each connected to an appropriate servomotor. The robotic arm design and control solution is implemented using an urbanized PC program called SMART ARM. It is a workstation-assisted project, and you will be responsible for devising a solution for a 6-DOF robotic arm equipped with a graphical user interface accessible to the customer (GCI). It enables the client to create a mock-up or invent a functional robotic arm prior to connecting the real thing. As a result, the user can estimate the optimal size of a specific robotic arm at the outset, thereby lowering the cost of construction and ensuring compatibility with the surrounding environment. Additionally, once the actual robotic arm is constructed, the user will be able to reuse the software to control the actual robotic arm in a natural manner without wasting time developing new control solutions [12]. Additionally, the program exhibits imitation characteristics. During imitation in the graphical user interface, the program assists significantly in visualizing the robotic arm route preparation. The PC ROBOARM is a genuine robotic arm that has been urbanized to simulate fallout. The 6-DOF robotic arm design is based on the PUMA jointed arm model. Both point-to-point and continuous track motion are verified using both simulated and real arm panels [20].

Fig. 6. 6 DOF Robotic arm

D. Arduino Board

There are two specific meanings that are an element of each arduino plan: system () and Loop (). The setup () is called once, as soon as the plan begins. It's a high-quality lay to do setup responsibilities like setting join forms or initializing libraries. The loop () meaning is called over and over and is heart of main plans. It's important to contain both functions in your plan, even if you don't need them for something.

Void setup ()
{
Statements;
}
Void loop ()
{
Statements;
}

Setup ()

The setup () function is called once the program has started. Used to initialize the pin modes or begin serial communications.

Void setup ()
{
pinmode(pin number , output);
}

Loop ()

Later than calling the setup () purpose the circle () do precisely what its name involves and loops in a row letting the program to alter reply and control the 18rduino board.

Void loop ()
{
digitalWrite (pin , high);
delay(1000);   //delay is in milliseconds
}

Pin Mode (), digital Write (), and delay () he pin mode () utility arranges a pin as moreover an input or an output. To employ it, you go by it the digit of the pin to organize and the steady INPUT or OUTPUT. When configured as an input, a pin be able to become aware of the situation of a sensor similar to a push button. The digital Write () functions outputs a value on a pin. For example, the line: digital Write (l edpin, HIGH); The delay () makes a reason for the Arduino to hang around for the specified number of milliseconds earlier than continuing on to the subsequently line. There are 1000 milliseconds in a second, so the line: Delay (1000).

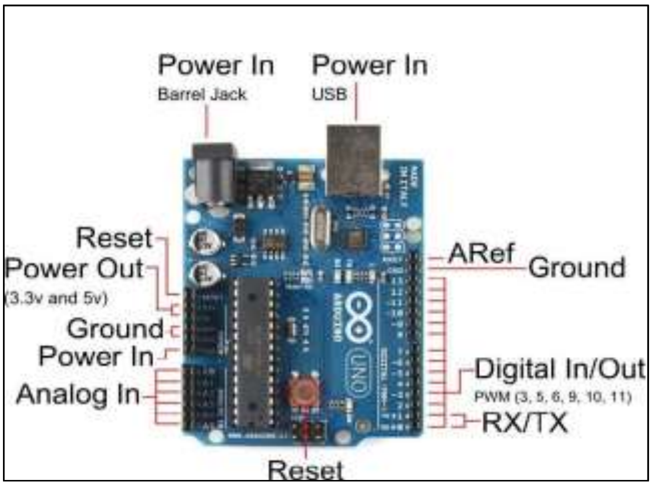

Fig. 7. Arduino UNO

The arduino Uno R3 is a microcontroller board based on the ATmega328 AVR microcontroller in a removable dual-inline package (DIP). It features a total of twenty digital input/output pins (of which 6 can be used as PWM Outputs and 6 can be used as analog inputs). It can be programmed using the simple-to-use arduino computer program. The arduino community is large, which makes it a very simple way to get started with embedded electronics. The arduino Uno R3 is the third and most recent revision [21].

IV. LEAP MOTION SENSORS

Starting with a hardware view point, the Leap Motion Controller is really pretty easy. The spirit of the leap motion made of 2 cameras and three infrared LEDs. These paths infrared light with a wavelength of 850 nanometers, which is outside the visible light spectrum.

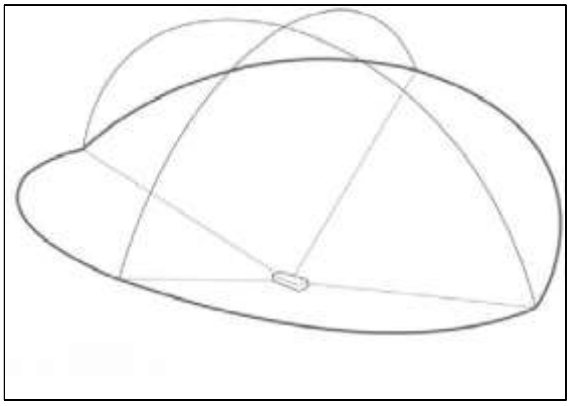

Fig. 8. Leap motion interaction area [20]

Due to the wide-angle lenses, the machine has a massive communication freedom of two cubic feet, which takes the shape of a reversed pyramid the connection of the microscope cameras' fields of view (fov) [22]. The Leap Motion Controller's display range is limited to barely (60 cm) above the device. This variety is limited by the spread of LED light across a space, from the point at which it becomes more difficult to imagine your hand's position in three dimensions away from a convinced distance. Finally, the intensity of the LED light is limited by the maximum current that can be drawn through the USB connection. Recent market introductions include novel 3D achievement devices such as strength cameras and the Leap Motion. While depth cameras enable a complete three-dimensional explanation of the framed prospect, the Leap Motion feeler is clearly targeted at hand sign identification and provides only a limited set of pertinent points [14]. Our work demonstrates how to collaborate on the development of two distinct types of sensors for precise gesture detection. Initially, an ad hoc result is obtained for the combined calibration of the two devices [23]. Following that, a set of novel characteristic descriptors for both the Leap Motion and depth data is introduced. Numerous systems based on the distances between the hand sections from the centroid, the curvature of the hand line, and the rounded hull of the hand shape are under development, as is the use of Leap Motion data to support characteristic mining [12]. Two distinct classifiers are used to process the planned characteristic sets: one based on multi-class SVMs and another developed using Random Forests. Additionally, singular characteristic assortment algorithms have been tested in order to reduce the difficulty of the move toward. Unpublished results demonstrate that the proposed technique is capable of achieving an extremely high degree of precision. Additionally, the current completion is capable of running in real-time [17]. Once the figure data is streamed to your computer, it's time for some serious mathematical elating. Contrary to popular belief, the Leap Motion Controller does not generate a depth chart; rather, it applies higher algorithms to the raw sensor data [22]. The Leap Motion examination is the software that was used to practice the scenarios on your PC. After adjusting for nearby objects (such as heads) and ambient ecological illumination, the descriptions are analyzed to reconstruct a three-dimensional representation of what the device sees [21]. Following that, the subsequent level equals the data required to extract the following data for Example members and gears. Our subsequent procedures comprehend three-dimensional facts and make assumptions about the locations of blocked substances. Sorting methods are realistic in that they ensure the information's flat chronological rationality. Previously, the Leap Motion Facility fed the consequences – articulated as a sequence of edges or photographs containing all of the following data attached to a convey procedure [12]. Once this procedure is complete, the facility will be connected to the Leap Motion Control Panel, as well as natural and web customer databases, via a secured socket link (TCP for native, Web Socket for web). The customer archive organizes data that is bound to an object-oriented API, ensures edge antiquity, and provides assistant purposes and sessions. The application logic then incorporates the Leap Motion input, allowing for motion-controlled communication [17].

Fig. 9. Leap motion arm position

The ability to control your PC with a swipe of your finger ensures that the sign-determined borders from sci-fi films such as alternative statement survive in perpetuity. Unfortunately, there is a significant gap between vision and reality, particularly when designing a tool that was not originally designed with the handle and/or motions in mind. For many citizens, the Leap Motion will offer little more than the novelty of playing touch-friendly PC games like Fruit Ninja and Censored the Line. Except if you discover a mostly decent routine for the Leap Motion and it behaves like this, it may be the most advanced piece of technology you've ever imagined [5]. To configure the Leap Motion, simply connect it via a sub input to your laptop and download the leap program from the Leap Motion website. Following connection, you are taken through a limited sample that enables you to interact with objects on the monitor and apply a small signal. Extending your hand to use the Leap Motion feels incredibly natural, similar to the first time you used a tap screen device. The initial demonstrations do not require a high degree of accuracy; they simply involve flapping your fingers nearby to interact with the models on the monitor. It's as if you're alarming a mist of atoms that are perfectly balanced and positioned directly in front of you [4]. While this method of flapping your fingers is quite tempting, it's a shame that the installer does not go the extra mile at this point to ensure the Leap Motion is happy with the illumination conditions. A small additional support upfront may prevent a slew of future annoyances [15]. Once you've established the connection, you'll be directed to the main monitor for Leap Motion's Territory. Here you'll find a few pre-installed applications, such as the Liquid Breakers screensaver, which allows you to simulate waves in a pool.

Additionally, you can download apps from the online Airspace market. Nowadays, you can download approximately 120 applications for Windows and Mac. While nearly half of all apps are games, there is also a trickle of learning, efficiency, and originality apps to discover. While the Leap Motion's extended expression feasibility is clearly based on a strong third party hold, in the short term, it faces larger usability issues [20]. The Leap Motion can be challenging to practice if the expedient is not properly located and suited to the lighting conditions, despite the fact that you are given limited direction in these areas. In less than ideal circumstances, you can still disorder through with games like Cut the Rope, but the sign border does not feel precise enough to handle much more at the moment. Finding your way around Google Earth or revolving a DNA thread in the Molecules app can be an aggravating experience, made worse by the fact that the apps provide so little guidance on how to use the sign controls [25]. For the time being, the Hand WAVE app provides necessary gesture assistance for navigating the browser and media player, but it's so unreliable that it's not difficult to simply reach for the keyboard or mouse. As is the case with the signal controls integrated into Samsung's Smart TVs and Sony's Vaio Pro Notebooks, the Leap Motion generally appears to be more trouble than it is once you get past the innovation of touch approachable competitions. Although the Leap Touchless apps for Windows and Mac suggest a higher level of control, you can't get past the fact that desktop boundaries were frequently not designed with motion and touch in mind [13]. One of the difficulties is that the Leap system allows multiple apps to run simultaneously, regardless of how they interact with one another. As a result, you're probably going to be consuming PhotoScape to operate images, and your belongings will become confused as Hand WAVE continues to work in the background. While you may be able to disable background apps' ability to obtain signal instructions in certain circumstances, an improved method would be to announce some form of intelligent app organization [11]. You can eventually categorize those application compatibility issues, but this ignores the fact that Windows and Mac desktops are designed to be used with a keyboard and pointer. While signals are likely to be useful for a few applications, they are frequently a source of concern rather than a source of information. Especially if they are unreliable [16]. This brings us to the central issue concerning the Leap Motion. It is extremely susceptible to intervention by visible and electromagnetic light. Inappropriately, it is not identical in terms of indicating when something is wrong or assisting you in resolving the issue once you become aware of it. As a result, approximately several people will disregard the Leap Motion for the sake of prevention [2]. I installed the Leap Motion primarily using my MacBook Pro seated at my dining room table, with the feeler resting on the table in front of the trackpad. Six downlights were directly above, with additional illumination coming in through the windows behind me. In this environment, apps that required any level of precision were unsatisfactory to use, even though the software made no indication that there was a problem with the lighting [23]. We discovered the option to recalibrate the device while poring over the Leap Motion's list of options, which appeared to be a respectable impression considering there was no declaration of standardization throughout the initial set-up. Until now, the standardization instrument has been refused permission to operate. As the Leap Motion appears to have been in "Robust mode" for an extended period of time, this is why the "existing lighting conditions may be inappropriate." That is clearly a significant issue, but for various reasons, the software did not consider it significant enough to bring it to my attention [14]. A google search revealed a slew of people curious about how elaborate the leap motion is in terms of illumination and how disappointing the fallout is when it silently switches to Robust mode – something that is most emphatically not stated during the setup procedure. Flowing the leap motion to a dissimilar area with a single down light and no nearby window was insufficient to elude robust mode for the time being. We couldn't start the standardization process until we turned off the overhead light and sat exclusively in the radiance of the MacBook Pro's screen. This is supported by correctness, but only as long as the light remained off [14]. Additionally, we discovered that repositioning the MacBook Pro slightly behind me and then pulling my chairperson further away from the table aided me as well. Here and now, the Leap Motion has been placed within a member's reach. The setup seminar clearly demonstrates that positioning the Leap Motion between you and the keyboard is appropriate, but getting past its visual realm to use the keyboard frequently activates it by accident. The Leap Motion appears to be well-suited to a desktop environment in which the feeler can be positioned between the console and the display, allowing it to be used independently of the console until you gain access to it over the console. We've mounted the Leap Motion on my Windows 10 examination outfit and concealed it with the feeler between the console and the screen to keep it out of the mode. Directly absent, a pop-up message informed me that "exterior ultraviolet light was detected, recompensing." At that point, we recognized immediately that this was destined to appeal in a strong manner while diminishing the exactness of the sensor and so we executed the single down light and sat in the shady once more. If you are required to practice the Leap Motion each day, it may have an effect on you, even though you may as well continue to be required to exist in the shade to keep the feeler functioning properly. We discovered that when we switched on old-style bright corms rather than down lights, the exactness and compassion of the feeler were released [7]. Although the leap motion is an exciting concept and its creators have created something wonderful, retrofitting signs to a fixed desktop border is a risky practice. Especially when the sensor is less than reliable. If the Leap Motion does not clarify a tedious problem or make an expert program easier to use, it will not advance your computing knowledge at this point. It's one to keep an eye on, but there's work to be done before signal takes control of sense on traditional desktops [14].

V. THE DEVELOPED ROBOTIC ARM

A. Hardware

This section of the article will discuss the hardware. It was accomplished by combining four servo motors: the hook, neck, up and down, and the base motor. Each motor is responsible for a single arm movement. Hook motors are responsible for opening and closing the hook. The neck motor is responsible for rotating the hook's entire body 180 degrees. The hook and up and down motors are responsible for lifting and lowering both neck motors in accordance with the corresponding code. The base motor is responsible for rotating the entire metal arm 180 degrees to the left and right.

B. Connecting the Circuit

This section discusses how the servo motors are connected to the Arduino and the breadboard. The laptop is connected to the arduino and the leap motion sensor. Each servo motor is equipped with three input wires, which are colored red, brown, and orange. The red wire is the servo motor's input voltage line. The brown wire is the servo motor's ground line. Orange wire: This is the data line from the Arduino to the servo motor. The voltage source is created by connecting the four servo motors described in the previous section in parallel via a breadboard. All motors and the arduino share the same ground. With regards to the data pins, each data wire from the servos is directly connected to a digital pin of type PWM on the Arduino, from which they obtain data in the form of integer values. These integer values represent the angles at which the servo responds in order to change its rotation angle. The leap motion sensor is directly connected to one of the USB ports, where it receives data and sends it to the arduino, which then sends it to the servos. The data obtained from the leap motion sensor is essentially as follows:

1. The position of the hand.
2. The angle of the hand.
3. The number of fingers …. And etc.

C. Software Works

Processing PDE and the Arduino IDE were used to write the code; both are open source programs that enable developers to program any electronic circuit. The Arduino IDE was built on top of the Processing IDE. The JAVA programming language was used in the code, which made it easier to utilize the serial ports more efficiently. Predefined functions in the JAVA language were extremely useful when writing the codes, such as converting a float to an integer. The Processing Development Environment (PDE) simplifies the process of inputting Distribution plans. Plans can be printed and tracked in the manuscript publishing supervisor by clicking the Track key. In Treating, a personal computer database is referred to as a draught. Draughts are saved in the Sketchbook, which is a computer file. Drafts are capable of creating both two- and three-dimensional graphics. The original renderer was designed for two-dimensional graphics sketching. The P3D renderer makes it possible to sketch three-dimensional graphics, including camera, illumination, and resource management. P2D is a fast renderer, but there are few precise renderers for two-dimensional graphics. Both the P2D and P3D renderers are accelerated if your PC is equipped with an OpenGL-compatible graphics card [11]. With Collections and Outfits, the capacity for indulgence is expanded. Libraries make it possible for plans to acquire possessions outside of the primary Treating cipher. The Processing society has added hundreds of libraries that can be added to your sketches to enable new effects such as playing sounds, performing computer vision, and working with advanced 3D geometry. Tools extend the PDE to make sketching easier for craftspeople by providing extended interfaces for tasks such as color selection [10]. Treating utilizes unique software design techniques that enable it to be possible to arrange draughts on various stages and database in a variety of ways. By default, the Java mode is used. Additional programming modes can be downloaded by selecting "Add Mode..." from the menu bar in the PDE's upper-right corner [15].

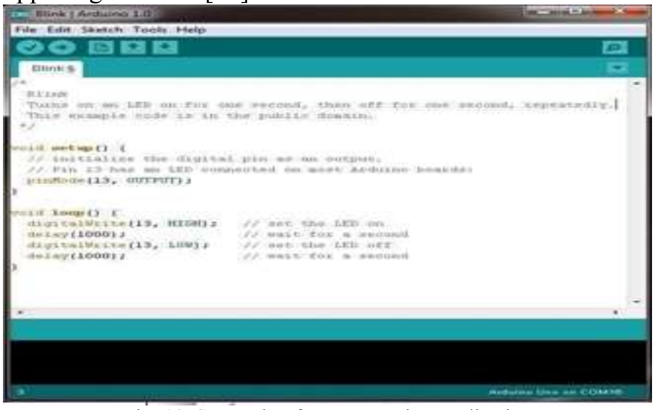

Fig. 10. Screenshot from processing application

D. Libraries and Functions

The leap motion library and the Processing. Serial library were included in the Processing IDE. There are numerous functions in the leap-motion library that make use of the hand tracking. Palmposition() returns the hand's position in the XYZ plane. The count() method of Fingers().extended() returns the number of open fingers at any point in time. palmNormal().roll() returns the hand's angle roll. The processing.serial library enables communication between the processing integrated development environment (IDE) and any USB port on a laptop. Using this line as an illustration Port = new Serial (this, "com3",9600); Com3 refers to the device's communication port. 9600 is the data transfer rate between Processing and the Arduino IDE. The Arduino IDE's Servo.h library was used. It enables programs to communicate with each other servo by sending the numbers indicating the servos' rotation. For instance, hook.write(120); indicates that the servo hook will rotate 120 degrees away from the X axis.

Fig. 11. Screenshot from arduino

VI. Practical Results and Schematic

The project's initial build was unable to move properly. The two servos responsible for moving the metal arm forward and backward could handle the arm's weight because the servos' torque was negligible in comparison to the arm's weight. It was necessary to remove the two failed servos. The metal arm was rebuilt with four servo motors, reducing the degree of freedom to four DOF; otherwise, the arm would be unable to move properly, as any servo rotating against its will would automatically open the circuit, turning everything off. After removing the two motors, the arm began to function normally. Now for the readings from the leap motion sensor: The leap motion returns data for the desired parts based on the code, such as the hand position in an XYZ plane. You can take any data you want, for example the X axis data used to move the base, so that if you move your hand to the left or right, the motor will rotate accordingly. Now, when you raise or lower your hand, the servo rotates and moves the hook and neck motors up and down. Finally, for the hook and neck motors, the neck motor responds to the angle of the hand, specifically the roll angle, and the vector for that angle is the palm normal vector as defined by leap motion developers.

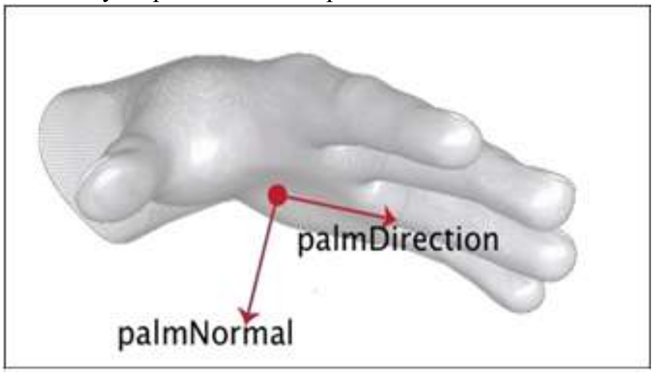

Fig. 12. Palm vector

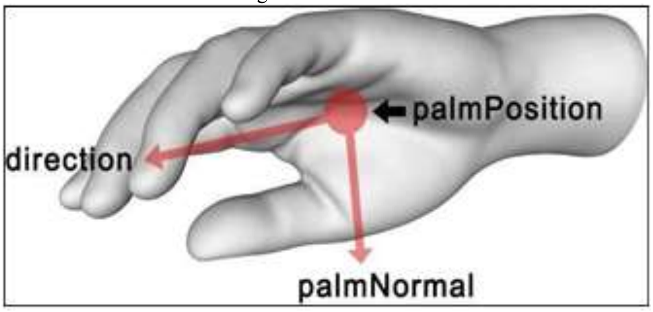

Fig. 12. Palm positions

VII. Conclusion

This work appears to be having a lot of issues with the torque of the servo motors; therefore, two possible solutions are to change the material of the arm, such as making it of plastic or something else lighter, or to replace the motors with more powerful ones. And another issue was with the arm's power supply; we've powered it with only 5 volts, which means that occasionally, particularly when we move our hand to the left side above the sensor, the robotic arm may shut down. The performance (The Actions) and the command are delayed by 0.0052 milliseconds. These solutions were not feasible due to the time constraint. Thus, when we compare our robotic arm controlled by leap motion to other robotic arms, we can see that our robotic arm is easier and smoother to use than the others and that we can perform additional movements in addition to the six we demonstrated in our project by programming the additional movements in our programming language, java. As previously discussed, the only way to resolve these issues was to dispose of two servo motors. In Table I, we compare the six-degree-of-freedom robotic arm to previous robotic arms.

TABLE I
COMPARISON BETWEEN ROBOTIC ARMS

Robotic arm type	Specs	Delay	Success	Accuracy
6 DOF controlled with leap motion	4 servo motors with Arduino and UNO sensor	500 Ms	6 movements succeed	Very accurate
The 2nd generation of OWI robot	Wired control of gripper 100 g lifting capacity	1000 Ms	4 Movements succeed	Poor accuracy
The haydraulic robotic arm	Generator, electric motors and combustion engine	900 Ms	4 Movements succeed	Good accuracy
The Cartesian robotic arm	Linear joints and linear motors	750 Ms	2-3 Movements succeed	Very Accurate

References

[1] M. A. Mezher, S. Din, M. Ilyas, O. Bayat, Q. H. Abbasi, and I. Ashraf, " Data Transmission Enhancement Using Optimal Coding Technique Over In Vivo Channel for Interbody Communication ," Big Data, vol. 00, no. 00, pp. 1–14, 2022, doi: 10.1089/big.2021.0224

[2] R. A. Khan, R. Imtiaz, A. Arain, A. S. SOOMRO, and D. KHAN, "Robotic Arm Controlled By Hand Gesture Using Leap Motion," Univ. Sindh J. Inf. Commun. Technol., vol. 3, no. 3, pp. 145–148, 2019.

[3] M. Alimanova et al., "Gamification of hand rehabilitation process using virtual reality tools: Using leap motion for hand rehabilitation," Proc. - 2017 1st IEEE Int. Conf. Robot. Comput. IRC 2017, pp. 336–339, 2017, doi: 10.1109/IRC.2017.76.

[4] et al., "Gesture Controlled Robotic Arm Using Leap Motion," Indian J. Sci. Technol., vol. 10, no. 45, pp. 1–7, 2017, doi: 10.17485/ijst/2017/v10i45/120630.

[5] R. D. Savatekar and A. A. Dum, "Design Of Control System For Articulated Robot Using Leap Motion Sensor," Int. Res. J. Eng. Technol., vol. 3, no. 3, pp. 1407–1417, 2016.

[6] Y. Fan, C. Yang, and X. Wu, "Improved teleoperation of an industrial robot arm system using leap motion and MYO armband," IEEE Int. Conf. Robot. Biomimetics, ROBIO 2019, no. December, pp. 1670–1675, 2019, doi: 10.1109/ROBIO49542.2019.8961758.

[7] C. Chen, L. Chen, X. Zhou, and W. Yan, "Controlling a robot using leap motion," 2017 2nd Int. Conf. Robot. Autom. Eng. ICRAE 2017, vol. 2017-December, pp. 48–51, 2018, doi: 10.1109/ICRAE.2017.8291351.

[8] S. Nicola, L. Stoicu-Tivadar, I. Virag, and M. Crisan-Vida, "Leap Motion supporting medical education," 2016 12th Int. Symp. Electron. Telecommun. ISETC 2016 - Conf. Proc., pp. 153–156, 2016, doi: 10.1109/ISETC.2016.7781080.

[9] J. Islam, A. Ghosh, M. I. Iqbal, S. Meem, and N. Ahmad, "Integration of Home Assistance with a Gesture Controlled Robotic Arm," 2020 IEEE Reg. 10 Symp. TENSYMP 2020, no. June, pp. 266–270, 2020, doi: 10.1109/TENSYMP50017.2020.9230893.

[10] M. H. Korayem, M. A. Madihi, and V. Vahidifar, "Controlling surgical robot arm using leap motion controller with Kalman filter," Meas. J. Int. Meas. Confed., vol. 178, no. April, p. 109372, 2021, doi: 10.1016/j.measurement.2021.109372.

[11] L. J. Wei, L. W. Sen, and Z. M. Sani, "Leap motion underwater robotic arm control," J. Teknol., vol. 74, no. 9, pp. 153–159, 2015.

[12] Z. Kaleem and K. Chang, "Public Safety Priority-Based User Association for Load Balancing and Interference Reduction in PS-LTE Systems," IEEE Access, vol. 4, pp. 9775–9785, 2016, doi: 10.1109/ACCESS.2016.2598198.

[13] M. R. S. B. de Souza, R. S. Gonçalves, and G. Carbone, "Feasibility and performance validation of a leap motion controller for upper limb rehabilitation," Robotics, vol. 10, no. 4, pp. 1–19, 2021, doi: 10.3390/robotics10040130.

[14] J. Guna, G. Jakus, M. Pogačnik, S. Tomažič, and J. Sodnik, "An analysis of the precision and reliability of the leap motion sensor and

its suitability for static and dynamic tracking," Sensors (Switzerland), vol. 14, no. 2, pp. 3702–3720, 2014, doi: 10.3390/s140203702.

[15] D. Bachmann, F. Weichert, and G. Rinkenauer, "Review of three-dimensional human-computer interaction with focus on the leap motion controller," Sensors (Switzerland), vol. 18, no. 7, pp. 1–39, 2018, doi: 10.3390/s18072194.

[16] J. S. Artal-Sevil and J. L. Montañés, "Development of a robotic arm and implementation of a control strategy for gesture recognition through Leap Motion device," Proc. 2016 Technol. Appl. to Electron. Teaching, TAEE 2016, 2016, doi: 10.1109/TAEE.2016.7528373.

[17] T. Venkata, S. Naidu, and S. Patel, "Real-Time Robot Control Using Leap Motion," Am. Soc. Eng. Educ., no. 1, pp. 1–5, 2015, [Online]. Available: https://scholarworks.bridgeport.edu/xmlui/bitstream/handle/123456789/1221/Real-Time Robot Control Using Leap Motion.pdf?sequence=1&isAllowed=y.

[18] M. Sakr, W. Uddin, and H. F. M. H. Van Der Loos, "Orthographic vision-based interface with motion-tracking system for robot arm teleoperation: A comparative study," ACM/IEEE Int. Conf. Human-Robot Interact., pp. 424–426, 2020, doi: 10.1145/3371382.3378311.

[19] I. A. Grout, "Hand Motion and Gesture Control of Laboratory Test Equipment Using the Leap Motion Controller," Proc. 19th Int. Conf. Image Anal. Recognit. (ICIAR 2017), Bangkok, Thail., vol. 11, no. 11, pp. 29–30, 2017.

[20] J. Chen, C. Liu, R. Cui, and C. Yang, "Hand tracking accuracy enhancement by data fusion using leap motion and myo armband," 2019 IEEE Int. Conf. Unmanned Syst. Artif. Intell. ICUSAI 2019, pp. 256–261, 2019, doi: 10.1109/ICUSAI47366.2019.9124812.

[21] K. Keerthana Rathan, A. Ajith, S. Aswathi, and V. M. Silpa, "Survey of robotic arm controlling techniques," Proc. Int. Conf. Intell. Sustain. Syst. ICISS 2017, no. Iciss, pp. 759–763, 2018, doi: 10.1109/ISS1.2017.8389277.

[22] G. Ponraj and H. Ren, "Sensor Fusion of Leap Motion Controller and Flex Sensors Using Kalman Filter for Human Finger Tracking," IEEE Sens. J., vol. 18, no. 5, pp. 2042–2049, 2018, doi: 10.1109/JSEN.2018.2790801.

[23] S. A. Jadhav, R. R. Yashod, A. P. Sabne, Y. K. Rokde, and C. R. Ghuge, "Leap Motion Sensor Technology Based Robo-Control System," Int. J. Comput. Sci. Mob. Comput., vol. 9, no. 12, pp. 50–58, 2020, doi: 10.47760/ijcsmc.2020.v09i12.007.