

Wrist Movement Detector for ROS Based Control of the Robotic Hand

Marcin Krawczyk¹, Zhijun Yang¹, Vaibhav Gandhi^{1*}, Mehmet Karamanoglu¹, Felipe MG, França², Priscila MV Lima³, Xiaochen Wang¹ and Tao Geng¹

¹Department of Design Engineering and Mathematics, Middlesex University London, UK

²Department of System and Computer Engineering, Federal University of Rio de Janeiro Rio de Janeiro, Brazil

³Tercio Pacitti Institute, Federal University of Rio de Janeiro, Brazil

Abstract

Robotic hands are used in a wide range of applications. They have many different shapes, constructions and capabilities. This work presents a new design of a robotic hand using tailor-made as well as widely available sensors and actuators. The information transferred between the sensor and actuators is processed using the Robot Operating System (ROS) topic mechanism. The robotic hand movement is remotely controlled by a movement detector mounted on the wrist of a human hand controller. Based on this simple hardware setup we demonstrate that the robotic hand can be remotely opened and closed thereby allowing to grasp objects flexibly.

Keywords: Robotic hand; ROS; Solid works; Control; Grasp

Introduction

There are many different methods and tools available for detecting wrist movements. Choosing the right technology depends on many factors such as the required precision, cost, the range of movements, etc. For precise movement detection of the wrist, the vision-based recognition system for motion and gesture detection can be used [1,2]. Devices using this type of technology processes images captured by the camera. The software implements advanced artificial intelligence algorithms to detect the wrist shape in the image and to analyze the video. This method allows to precisely detecting wrist movements without the user having to wear the device. However, such devices require good lighting condition, high level of computing performance and high-power consumption. Still, errors do creep in and the software does not provide 100% detection accuracy.

In recent years, several Micro-Electro-Mechanical Systems (MEMS) versions of accelerometer and gyroscope have become popular for movement detection in consumer electronic devices for e.g., smartphones and tablets. These sensors are usually very small as well as cheap to manufacture. They provide information about the position, speed and tilt of the moving object. Attaching these devices on the human wrist can effectively detect its movements [3]. However, there are disadvantages associated with these types of sensors. These sensors are sensitive to changes in magnetic field, temperature fluctuations as well as input voltage. They also do not have good performance characteristics and are prone to drift error in positioning which grows over time [4]. On the other hand, high-performance and accurate gyroscopes and accelerometers are large, consume more power and are expensive.

Another common way to detect movements of various human body parts is to use electronic sensors such as force and band sensors, Hall Effect sensors or rotary and linear potentiometers [5-7]. These sensors change their resistance when mechanical force is applied on them. This approach is simple, reliable and is the cheapest method amongst all. However, the disadvantage of this method is that the devices are usually bulky, heavy, have limited number of movements detected, and must be worn by the users with calibration needed for users.

Based on exploring existing devices, technologies and available tools for detecting human wrist movements, the aim of this work is to present two devices: a robotic hand and a wrist movement detector.

Both devices are connected via Bluetooth and the wrist movement detector can control the open-close movement of the robotic hand. The robotic hand is composed of a 3D printed, biomimetic hand with five fingers and controlled by two single-board microcomputers under the Robot Operating System (ROS) framework for signal processing [8-10]. The wrist movement detector has an ability to read horizontal and vertical human wrist movements and display these movements on the LED matrix. The vertical movements of the human wrist are also used to control remotely the robotic hand to open and close its fingers and the thumb.

The paper is organized into six sections. Section II introduces the materials and methods used in this experiment. Section III presents details about the construction and working of the robotic hand, while section IV presents the details of the wrist movement detector. Section V is dedicated to testing of the whole system. Lastly, section VI gives concluding remarks.

Materials and Methods

Robotic hand

The robotic hand is designed to consist of an Odroid-XU4 single-board computer, Arduino Uno microcontroller, 5 servo motors Tower Pro SG90 Micro Servo 9g, 5 force sensors made from Velostat and aluminum kitchen foil, plastic and rubber 3D printed parts, and electronic consumables.

The robotic hand is designed using Solidworks Computer Aided Design (CAD) package. The force sensors available in the market did not offer required shape and were expensive. The custom-built force sensors were made from Velostat (polymeric foil) material, which changes its resistance when pressure is applied. To improve the sensitivity and

***Corresponding author:** Vaibhav Gandhi, Department of Design Engineering and Mathematics, Middlesex University, London, UK, Tel: +44 (0)20 8411 5511; E-mail: V.Gandhi@mdx.ac.uk

Received June 26, 2018; **Accepted** July 23, 2018; **Published** July 30, 2018

Citation: Krawczyk M, Yang Z, Gandhi V, Karamanoglu M, Felipe MG, et al. (2018) Wrist Movement Detector for ROS Based Control of the Robotic Hand. Adv Robot Autom 7: 182. doi: [10.4172/2168-9695.1000182](https://doi.org/10.4172/2168-9695.1000182)

Copyright: © 2018 Krawczyk M, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

active area of the sensors, two layers of aluminium kitchen foil were placed on both sides of the Velostat. All materials were cut to match the shape of each finger and were placed under the rubber phalanges. This allowed to detect minute moments when the finger encountered an object. The Odroid XU4 is used as an ROS server to communicate with the Arduino for processing sensors' data and driving the motors.

Wrist movement detector

The main task of the wrist movement detector is to control the robotic hand remotely. Therefore, the detector is required to read the radial/ulnar deviation and the flexion/extension movements of the wrist in relation to the forearm [11] as shown in Figure 1. The hand movements are then used to decide whether to close or open the robotic hand allowing it to grasp the objects, via the Bluetooth 4.0 HM-10 modules installed on both the hand and detector sides. The materials used to build the wrist movement detector are: Arduino Mega 2560, plastic 3D printed parts, two 100 kOhm single-turn adjustable potentiometers, two rubber bands, one 8*8 dot LED display matrix, 3xAAA batteries, fabric hook and loop fastener, and electronic consumables.

The LED display matrix is used to display wrist movements in two directions simultaneously. It allows the user to see his/her own movements for calibration and providing instantaneous feedback. Although the matrix has 8x8 LEDs, only 7x7 LEDs are used. This allows using the LEDs at the middle row and column as the guide for the user to indicate if the wrist is in central horizontal or vertical position.

Designing the Robotic Hand

The robotic hand requires five servo motors and five force sensors. All of these sensors and actuators are controlled directly by an Arduino Uno. An extra Odroid XU4 works as a ROS server to control the information flow at the hand, and also serves as the receiver of commands from the wrist movement detector.

Hardware

The Barrett hand [12] is one of the few robotic hands which are used for industrial applications. Each finger has two degrees of freedom (DOF) but is controlled with one actuator only. These rotational joints are mechanically coupled, and moving one joint makes the other joint move as well. This structure is more durable and precise than tendons pulling fingers, therefore, this approach has been chosen as a method to control the fingers. This also allows to use fewer actuators than DOF (under-actuated system), which is more beneficial from an economic perspective and reduces the robotic hand's weight.

Another widely cited hand, namely the DLR hand [13], mounts all the servomotors in the wrist. The main advantage of this integrated

structure is that the hand does not have the arm and therefore can be attached to any type of a robotic arm. It is a self-contained device which occupies less space and also reduces robotic hand's weight.

Inspired by the above designs, the robotic hand presented in this paper is designed with the CAD tool (Figure 2), with which it is tested against parts collisions and contact points with the object were determined accordingly. The CAD model allows adjusting right proportions of parts to achieve full range of finger movements during a 180-degree servomotor rotation.

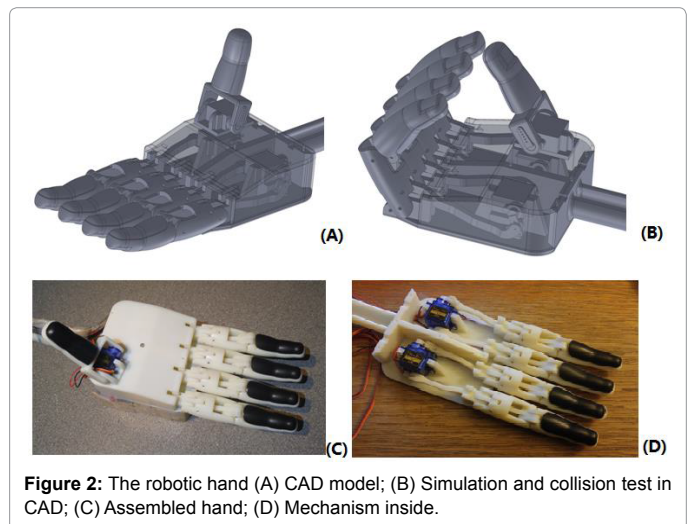
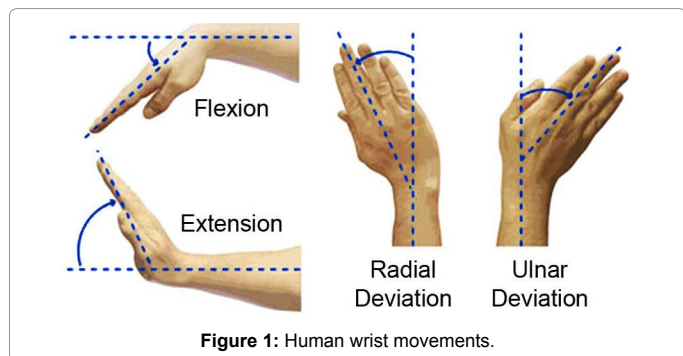
In addition to the mechanic part of the hand, the custom-made force sensors play a key role in the finger movement. The range of changing the resistance on each sensor is measured during tests, and the two resistors with 10 kOhm + 5.1 kOhm were chosen for voltage divider circuit. Figure 3 shows the connections between all components. It is worth noting that the servomotors and the sensors need to use separate 5 V power supplies to prevent the interference of changing electrical load with signals from sensors.

Software

The Odroid XU4 works as the ROS server. The Ubuntu 14.04 LTS and ROS Indigo Igloo frameworks are used. The Firmata protocol is employed to provide communication between the Odroid and Arduino boards. The robotic hand controller can operate in two modes; one is standalone operation, and another is the control by position information received from the wrist movement detector. The ROS framework is used for both these operation modes.

In the standalone operation mode, once the software on the Odroid and Arduino boards is ready, inside a catkin workspace, two nodes are created using Python to read values from the force sensors and to control the servomotor movements. There are two groups of 5 topics representing 5 fingers, and 5 finger sensors, respectively. The 2 nodes communicate with each other through publishing and subscribing these topics. The ROS rqt plot is shown in Figure 4 (cf. section V for more details).

In the wrist movement controlled mode, the difference is that, instead of the direct control loop between the sensors and the actuators, the control information comes from the remote wrist movement detector. The wrist position values received via Bluetooth from the wrist movement detector are converted into positions of each finger of the robotic hand. This process repeats every 50 ms, i.e., the scan cycle



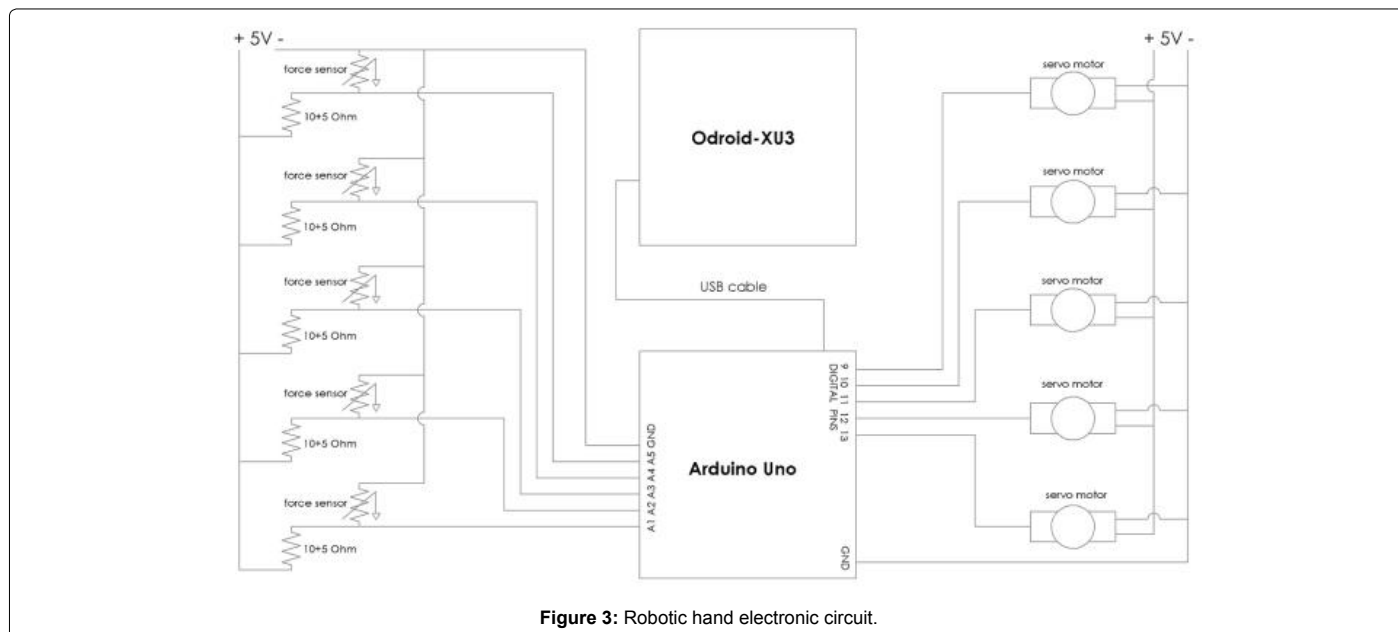


Figure 3: Robotic hand electronic circuit.

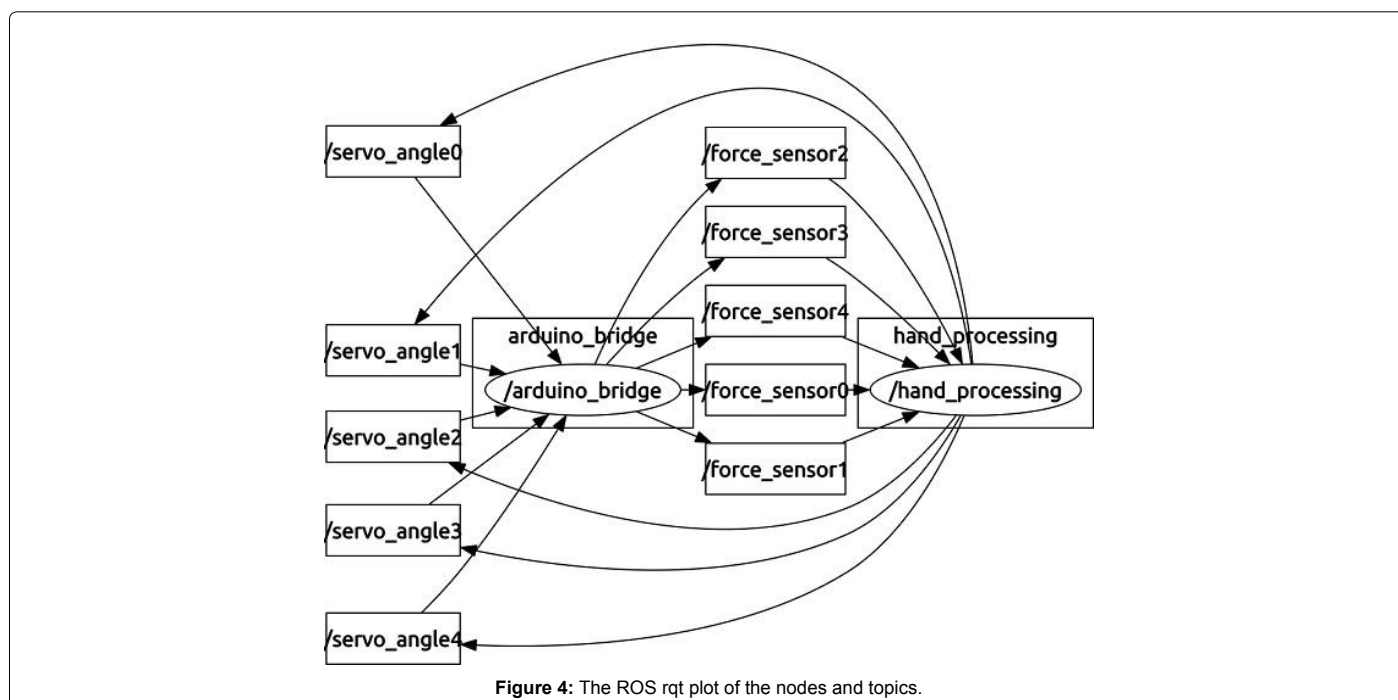


Figure 4: The ROS rqt plot of the nodes and topics.

is 50 ms and the new position of each finger is calculated based on the received position of the corresponding wrist information.

Designing the Wrist Movement Detector

The prototype wrist movement detector presented in this paper is designed using off the shelf components. The detector can detect movement in one DOF, i.e., flexion and extension. Detection of more DOFs is straightforward using similar strategy.

Hardware

The process of building this prototype detector begins from creating an electronic circuit. Two rotary potentiometers and one

LED matrix display are connected to the Arduino Mega 2560 board. The potentiometers are connected to analogue input pins in Arduino and the matrix display is connected to the digital pins. A battery bank provides 4.5 V power to the Arduino board. The LED display flashes upon the changes of two potentiometers' resistance values, representing the movements in two DOFs. One potentiometer changes the active row of the display while other the active column. All the components in the working circuit are glued together and two fabric hooks and loop fasteners are attached to the bottom of the prototype board. This allows to mount the whole device on the forearm of the user for testing and calibrating. Two rubber bands are placed between the potentiometers and two fingers allow turning the potentiometers when moving the wrist. A little hook is made using a hot glue gun tool at the end of each

potentiometer's shaft to attach the rubber band.

After binding the device with a forearm, the position of two potentiometers in relation to the fingers need to be adjusted so that the wrist movement can be detected proportionally. This adjustment is an empirical process which may be time-consuming to obtain the right angle, distance and correlation between each potentiometer and the finger. During this process, different fingers and device configurations are tested for the right hand. It is found that for radial/ulnar deviation the best result is achieved when the rubber band is placed on the middle finger. For the flexion/extension movements, the best result is achieved with the ring finger. The final installation of the device on the forearm is shown in Figure 5.

Software

The programming of the wrist movement detector system can be separated into three portions. First, configuring the Bluetooth modules with AT commands. Second, programming the wrist movement detector. And lastly, communicating with the robotic hand.

At the wrist movement detector side, the Bluetooth 4.0 HM-10 module is configured as a master while the module for the robotic hand side is configured as a slave. To pair the two modules, each one is connected to the PC through a FTDI USB-to-TTL adapter and is configured separately for the master and the slave mode, with AT commands through Serial Port Monitor in Arduino IDE.

The second step, once the Bluetooth configuration process establishes the communication, is to programme the wrist movement detector by using the Arduino. At the beginning, the code lights up all the LEDs on the matrix display for 1 second, to indicate that the device has turned on. This is followed by a calibration process. The user must move the wrist left/right/up/down as directed by the flashing LED. During this process, the wrist movements are detected by the rotary potentiometers and saved as a benchmark movement. This step is necessary, as every user may have a different range of movements, and the device must be recalibrated for every new user. When the device finishes the calibration process, the normal working process begins.

The device can now read signals from the rotary potentiometers whose positions depend on the user's wrist movements. When the

extension or flexion of the user's wrist is detected, the flashing LED moves up or down accordingly.

When the radial or ulnar deviation is detected, the LED moves left or right. The signals from the potentiometers are in the range from 0 to 255, but they are limited during calibration mode to smaller ranges depending on the user's wrist movements. These ranges are then divided by 7 because there are 7 LEDs in each row and column on the matrix display. Therefore, when the movement of the wrist is detected it is assigned to one of the 7 LEDs vertically or horizontally. The detected vertical wrist movements are then mapped to a range of values from 0 to 255 and sent through the serial port to the Bluetooth device. These values are used to control the movements of the robotic hand accordingly.

The Experiments

Testing the robotic hand

In our experiments with the robotic hand in the standalone mode, the hand can grasp objects of different shapes. Each finger runs independently and can stop at a different angle, depending on the signal information from the force sensor. The whole procedure of grasping and controlling fingers is programmed in two nodes and ten topics as shown in Figure 4.

The first node, namely `arduino_bridge`, communicates with the Arduino board and reads signals from the five force sensors. The sensor values are normalized in a range between 0 and 1. The collected data is published on five topics from `/force_sensor0` till `/force_sensor4`. Each topic stores the value from one sensor only. The second node, namely `hand_processing`, subscribes to these five topics and stores the last two values from each sensor. When the difference between these two values is lower than a pre-defined value (here we used 0.08), the node sends a command to increase the angle value of the relevant servomotor. If the difference is greater than 0.08, it does not increase the servo angle. The calculated angles for all five servomotors are published on five topics from `/servo_angle0` till `/servo_angle4`. Each topic stores the value for one servo only. The `arduino_bridge` node subscribes to these topics and sends the servo motors' angles back to the Arduino board. Based on these values, the Arduino board sets the angle for each servo motor.

In case of the wrist movement detector control mode, the angle information for the fingers is received from the remote controller, and not from the local Odroid board. The force sensors mounted on each finger is used to set the limit of movements for respective fingers.

Testing the wrist movement detector

The control effects of the wrist movement detector over the robotic hand have been tested on a group of 4 human subjects. Some modifications are made on the first prototype of the system, the limitations of that version will be discussed with the improved version presented here.

Whilst building the first prototype of the wrist movement detector several tests were performed. It is found that each user has a different range of movements and the wrist does not move along the horizontal and vertical planes. Wrist movements are complex and therefore changes were applied to the improved version of the device. The upgraded device allows adjusting position of each potentiometer by using a tailor-designed holder. Also, the software calibrates the device every time it starts. These changes allow reading the wrist movements precisely, and the device is appropriately setup individually for each end user.

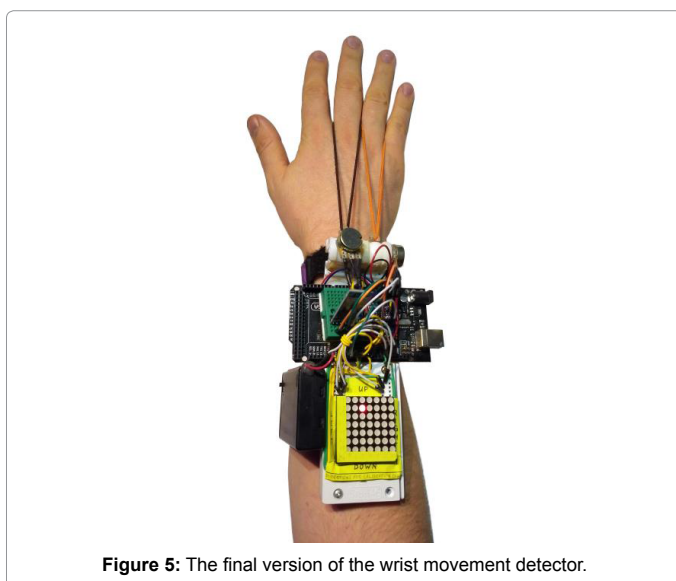


Figure 5: The final version of the wrist movement detector.

The tests of the improved device on four participants generated a similar result. It is found that each participant has a slightly different range of wrist movements. In addition, the ulnar deviation angle is always larger than the radial deviation. The flexion movement angle, however, is observed smaller than extension movement in all participants. Also, it was noted that these wrist movements are not perpendicular to the human body vertical axis and these movements slightly vary in each participant. This is presented in Figure 6.

The above study proves that middle and ring fingers provide the best correlation between wrist movements and the signals read by the potentiometers. It also shows that the positions of the potentiometers need to be adjustable because of different wrist movements for different people. A dedicated potentiometer holder (c.f. Figure 7) is therefore designed and printed using the same StratasysConnex Objet 500 3D printer as the robotic hand. The holder provides flexibility for accommodating different participants' wrist movements as it provides 5 DOFs to a potentiometer's position allowing precise adjustment of the wrist.

When the user moves the wrist, the device provides instant feedback on the matrix display. When the flexion or extension movements are detected, the device sends this data to the robotic hand. The fingers of the robotic hand close when the flexion is detected, and they open when extension of the user's wrist is detected.

The robotic hand allows performing object grasp [14], while the grasp is remotely controlled by the user. The speed of the grasp can be

controlled because the fingers of the robotic hand follow the movements of the user's wrist. When the user moves his wrist quickly the robotic hand also closes and opens quickly. For slow user movements, the robotic hand responds accordingly. A video link to show the control effects is available on [15,16]. The full design scheme and programs are open source, and will be available upon request.

Concluding Remarks

It is a challenging work to build a device to read and control wrist movements. Interacting with the human body is not a trivial task as each part of the human body is very complex. The wrist, which has many bones, joints, tendons and muscles, has several movement overlaps. Despite of the difficulties, the wrist movement detector has met the set requirements, i.e., detecting the wrist movements and control the robotic hand with acceptable precision. As always, improvements on this device is also possible.

One improvement is to include more rotary potentiometers. If there is a separate potentiometer for each of a user's fingers then the fingers of the robotic hand can potentially be controlled individually. However, in this case bend sensors could provide better feedback and could be easier to implement.

On the robotic hand side, although it performs well when grasping most of the objects, there are spaces to improve its capability. The most significant improvement can be achieved by applying more powerful and more precise servomotors. Another improvement could be an additional servo motor on the thumb. This would allow to rotate the thumb in different directions and perform more complex and dedicated grasps. The palm of the hand could have more anthropomorphic shape which would make the whole hand looking like the human hand. The tailor-built force sensors have worked reasonably well, and allows to detect contacts of different objects. They are easy to make and cheaper compared to commercial products. More work is necessary to feed the force sensor information back to the wrist movement detector so that the user wearing the detector can receive this feedback and be aware of the forces applied by the robotic hand on the grasped object.

Acknowledgment

This work was supported, in part, by the department of design engineering and the EPSRC grant EP/P00542X/1.

References

1. Pasarica A, Miron C, Arotaritei D (2017) Remote control of a robotic platform based on hand gesture recognition. *E-Health and Bioengineering Conference (EHB)*.
2. Pasarica A, Miron C, Arotaritei D (2017) Hand gesture detection using a stereo camera system and simulation of movement. *10th International Symposium on (ATEE)*.
3. Tong KY, Mak AFT, Ip WY (2003) Command control for functional electrical stimulation hand grasp systems using miniature accelero meters and gyroscopes. *Medical and Biological Engineering and Computing* 41: 710-717.
4. Woodman OJ (2007) An introduction to inertial navigation. *Computer Laboratory, University of Cambridge*.
5. Dipietro L, Sabatini AM, Dario P (2008) A survey of glove-based systems and their applications. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications* 38: 461-482.
6. Simone LK, Sundararajan N, Luo X, Jia Y, Kamper DG (2007) A low cost instrumented glove for extended monitoring and functional hand assessment. *J Neurosci Methods* 160: 335-348.
7. Dipietro L, Sabatini AM, Dario P (2003) Evaluation of an instrumented glove for hand-movement acquisition. *Journal of Rehabilitation Research and Development* 40: 179-189.

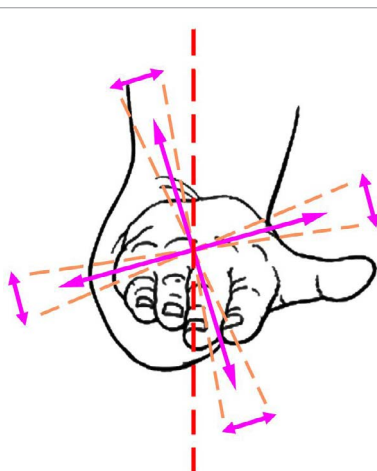


Figure 6: Human wrist movements in relation to vertical body axis.

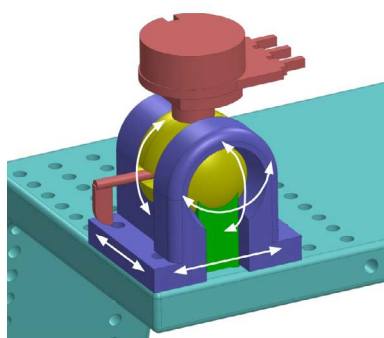


Figure 7: Potentiometer holder with 5 DOFs.

8. Quigley M, Conley K, Gerkey B, Faust J, Foote T, et al. (2009) ROS: an open-source Robot Operating System. ICRA workshop on open source software.
9. Goebel RP (2012) A Do-it-yourself Guide to the Robot Operating System: a Pi Robot Production.
10. Kalyani GK, Yang Z, Gandhi V, Geng T (2017) Using robot operating system (ROS) and single board computer to control bio-robot motion. *Towards Autonomous Robotic Systems*, pp: 41-50.
11. Jiang J, Englehart KB, Parker PA (2009) Extracting simultaneous and proportional neural control information for multiple-DOF prostheses from the surface electro myographic signal. *IEEE Transactions on Bio medical Engineering* 56: 1070-1080.
12. Townsend W (2000) The Barrett Hand grasper-programmably flexible part handling and assembly. *Industrial Robot: An International Journal* 27: 181-188.
13. Chen Z, Lii NY, Wimböck T, Fan S, Liu H, et al. (2014) Experimental analysis on spatial and cartesian impedance control for the dexterous DLR/HIT II hand. *Int J Robot Autom* 29: 1-13.
14. Singh N, Huyck C, Gandhi V, Jones A (2017) Neuron Based Control Mechanisms for a Robotic Arm and Hand. *World Academy of Science, Engineering and Technology* 11: 221-225.
15. Krawczyk M, Gandhi V, Yang Z (2018) Wrist movement detector video.
16. Krawczyk M, Gandhi V, Yang Z (2018) 3D printed robotic hand video.