Remote control of a robotic hand using a leap sensor

DAWES, Felix, PENDERS, Jacques <http://orcid.org/0000-0002-6049-508X> and CARBONE, Giuseppe <http://orcid.org/0000-0003-0831-8358>

Remote Control of a Robotic Hand Using a Leap Sensor

Felix Dawes[1], Jaques Penders[1], Giuseppe Carbone[2]

[1]*Dept. of Engineering and Mathematics, Sheffield Hallam University, Sheffield, United Kingdom, J.Penders@shu.ac.uk*

[2]*LARM: Laboratory of Robotics and Mechatronics, University of Cassino and South Latium, Cassino, Italy, giuseppe.carbone@unicas.it*

Abstract. This paper presents a low-cost gesture-based remote control of a robotic hand. The proposed control architecture is based on a commercial leap motion sensor and an Arduino board, which have been chosen due to their low-cost and user-friendly features. A specific Matlab code has been implemented to collect data from the leap motion sensor and to generate proper instructions to control a robotic hand, which has been 3D print at Sheffield Hallam University. Experimental tests have been carried out validate the effectiveness of the proposed remote control for performing various grasping tasks.

Keywords: Robotic hands, remote control, leap sensor,

1 Introduction

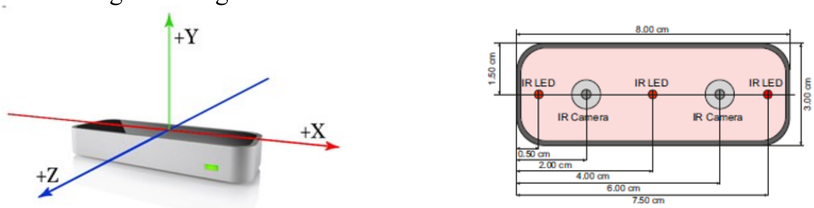
Human grasping has been investigated for centuries aiming at achieving solutions for assisting humans in difficult tasks or to develop functional prosthesis for amputees [1]. In the last decades several robotic hands have been developed such as, for example, Stanford/JPL Hand, DLR Hand, BUAA Hand, Colobi Hand, Barrett Hand, TUAT/Karlsruhe Hand [1-5]. Several of the above-mentioned prototype have very complex and expansive control architectures, which makes difficult and expansive the training and interaction with end-users. Many other robotic hands have been developed or currently under development with key open issues still being, among others, user friendliness in combination with cost oriented solutions, such as proposed with specific experiences at LARM in Cassino, as reported for example in [6-11].

In recent years, hand gesture recognition [12-13] has attracted a growing interest due to its applications in many different fields, such as human-computer interaction, robotics, computer gaming, automatic sign-language interpretation. Gesture-based remote operation is potentially one of the most effective means of communication with a robotic hand as it is one of the most effective and intuitive means of human communication. Nevertheless, cameras and vision control have been for long time expansive and challenging solutions to achieve gesture recognition. Recently the introduction of low cost solutions like Microsoft's Kinect [14], has suggested to exploit the depth information acquired by these devices for achieving proper gesture recognition in a low-cost and user-friendly manner. Recently, the Leap Motion has been introduced as an inexpensive choice with a proper accuracy for detecting human hand motions, [15].

In this paper, we aim at validating the appropriateness and effectiveness of using the Leap Motion for the direct mapping of human fingertips for achieving a low-cost and user-friendly remote control of a robotic hand. For the purpose, an InMoov robotic hand has been 3D print and assembled at Sheffield Hallam University. A specific Matlab code has been implemented to collect data from the leap motion sensor and to generate proper instructions to control the robotic hand via a wireless Arduino board. Several grasping tasks have been experimentally tested to validate the engineering feasibility and effectives of the proposed solutions.

2 Leap Motion sensor

The Leap Motion Controller, Fig.1 is considered a breakthrough device in the field of hand gesture controlled human-computer interface. This device introduces a novel gesture and position tracking system with sub-millimeter accuracy. Its operation is based on infrared optics and cameras instead of depth sensors. Its motion sensing precision is unmatched by any currently available depth camera. It can track simultaneously all the human fingers. As stated by the manufacturer, the accuracy in the detection of each fingertip position is approximately 0.01mm, with a frame rate of up to 300 fps [15]. The controller is considered to be an optical tracking system based on stereo vision. Within its surface area of 24 cm², the controller has three IR (Infrared Light) emitters and two IR cameras [16-17]. The field of view of the controller is very wide, up to 150°, which gives the user the opportunity to move his hand in 3D motions. The Software Development Kit (SDK) supplied by the manufacturer delivers information about Cartesian space of predefined objects such as the finger tips, pen tip, hand palm position. Also, information about the rotations of the hand (e.g. Roll, Pitch, and Yaw) are available as well. All delivered positions are relative to the Leap Motion Controller's center point, which lies between the two IR cameras, just above the second IR emitter. Differently from the Kinect and other similar devices, the Leap Motion does not return a complete depth map but only a set of relevant hand points and some hand pose features such as position of the fingertips, the position of the palm center, the hand orientation. In a leap motion the position of the fingertips Fi, with i=1,...,N, represent the 3D positions of the detected fingertips with N being the number of recognized fingers.



a) b)

Fig. 1 A schematic view of the Leap Motion Controller: a) a 3D view with axes; b) details of component locations.

The Palm center C roughly corresponds to the center of the palm region in the 3D space. The Hand orientation is based on two unit vectors; h is pointing from the palm center to the fingers, while n is perpendicular to the palm plane pointing downward from the palm center. However, the estimation of h and n is not very accurate and depends on the fingers arrangement. An important observation is that, while the computed 3D positions are quite accurate, the sensor is not always able to recognize all the fingers. For example, fingers touching each other, folded over the hand or hidden from the camera viewpoint are not captured. In many configurations some visible fingers could be lost, especially if the hand is not perpendicular to the camera. Furthermore, protruding objects near the hand, like bracelet or sleeves edges, are easily confused as fingers. This is quite critical since in different executions of the same gesture the number of captured fingers could vary. Therefore, approaches simply exploiting the number of captured fingers do not work very well.

It is important to note that the current version of the Leap Motion software does not return any information about the matching between the acquired points and the corresponding fingers with collected values just randomly ordered. Considering this characteristic, we sort the collected values on the basis of the fingertips angle with respect to the hand direction h. This corresponds to order them from the thumb to the pinky and it assigns each captured finger to a specific region according to the angle between the projection of the finger in the plane and the hand direction h. Note that there is not a one-to-one matching between sectors and fingers. Accordingly, some of the sectors Si could contain more than one finger and others could be empty. In case two fingers lie in the same angular region, one of the two is assigned to the nearest adjacent sector if not already occupied, otherwise the maximum between the two feature values is selected.

All the features values, except angles, are normalized in the interval (0,1) by dividing the values for the distance between the hand center and the middle fingertip in order to make the approach robust to people with different hands of different sizes. The scale factor S can be computed when the user starts to use the system. To simplify the control of each finger of the robotic hand the distance between the user's fingertip and palm positions are required, this will allow a single value to be mapped to each servo motor actuating each finger.

The actual number of captured fingers strongly affects the hand orientation h and so the fingertips angles. The obtained values Ai have been scaled to the interval (0.5,1) to better discriminate the valid interval from the missing values that have been set to 0. These values have also been used to assign each finger to the corresponding sector.

Calculated parameters are the fingertips distance Di and fingertips elevation Ei. Fingertips distance are computed as $Di = \|\mathbf{Fi} - \mathbf{C}\|/S$ (with i =1,...,5), as they are the 3D distances of the fingertips (defined by the vector **Fi)** from the hand center (defined by the vector C**)** divided by the scaling factor S. Note that there is at most one distance value for each sector and the missing values is set to 0. Fingertips elevation has been calculated as $Ei = \mathrm{sgn}(\mathbf{Fi} - \mathbf{F\pi i})(\|\mathbf{Fi} - \mathbf{F\pi i}\|/S)$ (with i =1,...,5) as it represent the distances of the fingertips (defined by the vector **Fi)** from the plane corresponding to the palm region, whose projection is defined by the vector **Fπi**. The sign takes into account the fact that the fingertips Fi can belong to any of the two semi-spaces defined by the palm plane π. Also in this case, the distances are made dimensionless by using the scaling factor S. Similarly to fingertips distances, each sector has only one fingertip elevation value Ei for each sector. Any missing value is set to be 0. Note that as for the fingertips angles, the values range has been scaled to the interval (0.5,1). A similar approach has been reported, for example, in [17]. It is worth noting the symbols Fi, C, h, n

Data collected by the leap motion are saved in matricial form in Matlab environment where the distance of each fingertip from the user's palm is calculated, making use of Matlab "distance" function. Figure 2 shows an example of the data collected from the leap motion output in terms of positions along x direction (in mm) for a finger moving from left to right six times.. Figure 3 shows the calculated positions relative to the palm of the five fingers versus time (1-thumb, 2-index, 3-medium, 4-anular, 5-pinky) when the hand is at hand fully open (Fig.3a) and at fully close configuration (Fig.3b). As the position of the fingers are quite accurately recorded enabling even some simple gesture recognition. Further analysis on the analysis of the motions could be performed by referring to the hand grasping taxonomy as proposed for example in [18].

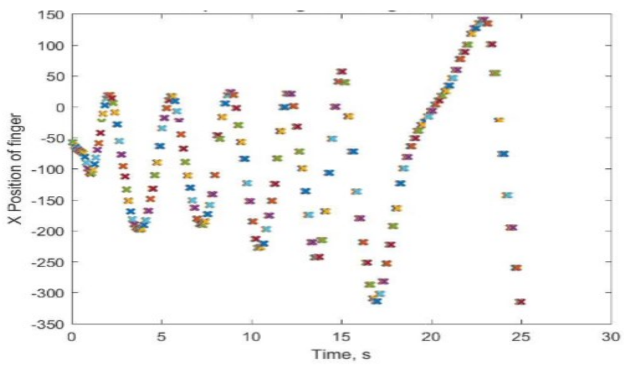
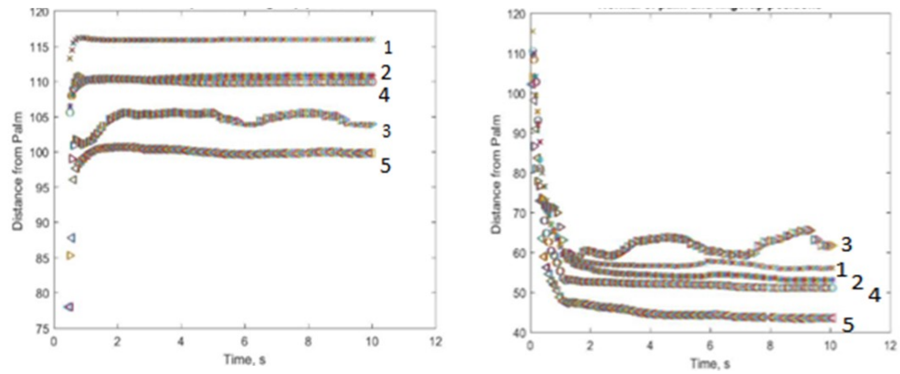


Fig.2 An example of leap motion output along x direction (in mm) for a finger moving from left to right six times.



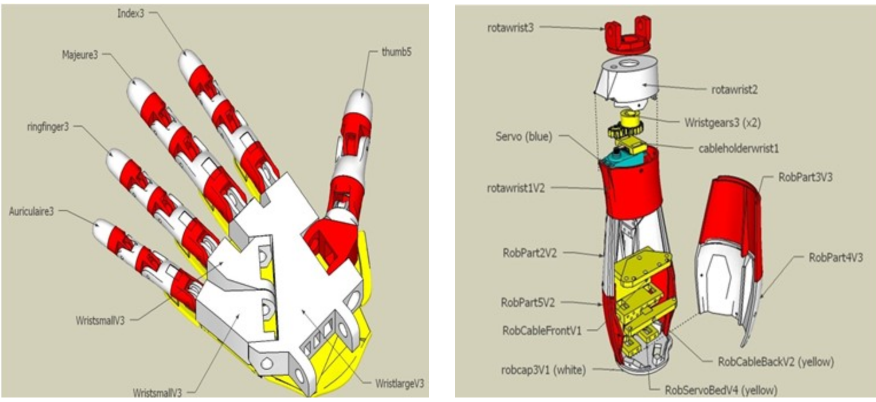
a) b)

Fig.3 Calculated positions relative to the palm of the fingers versus time (1-thumb, 2-index, 3-medium, 4-anular, 5-pinky): a) hand fully open; b) hand fully close.

3 The InMoov robotic hand

Many robotic hands are currently even commercially available as also investigated by the authors for example in [1-5]. Main requirement for this work has been to identify a robotic hand with low-cost and user-friendly features. It needs to be sufficiently human like to allow its remote control through a direct mapping of human fingertips using the Leap Motion controller.

The InMoov open source 3D printable robotic hand has been chosen to be used for this work as it has five human-like fingers, [17]. Each finger has one active degree of freedom and it is driven by tendons and pulleys as shown in the schemes of Fig.4. The 3D CAD drawings of the InMoov hand are openly available from the InMoov website [19]. The hand has been 3D printed at Sheffield Hallam University in blue and white ABS-P430 - 65c1 (920cc). S03N STD servo motors have been purchased and assembled to actuate the fingers via tendons rated at 30Kg. The hand and arm parts were printed, the support material removed and the pieces assembled at Sheffield Hallam University as detailed in Fig.5. A commercial braided fishing line has been chosen for the tendons as it does not stretch over time and allows the required tendon loading over 30 Kg. The tendons are kept taught by running them through 1cm long tension springs as shown in Fig.5. The hand control board and interfaces have been wired and placed in a small box. This was then connected to the servomotors to drive the tendons. The hand was mounted on a stand to allow trouble-free operation and to give a nice aesthetic for testing and showcasing. The servomotors require a 6V power supply so a LM2596 buck convertor was used with a 12V, 2A DC main power supply and an HC-12 bridge for a proper operation of the whole system, [20]. The arm stands 410mm tall and has a maximum width of 150mm.



a) b)

Fig.4 A 3D model of the InMoov robotic hand set-up: a) the hand; b) the a detail of the forearm and wrist, [19].

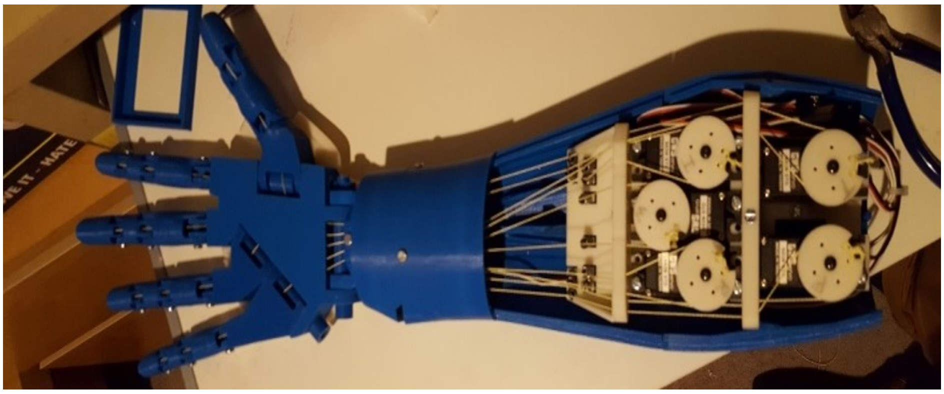


Fig.5 The 3D printed InMoov hand at Sheffied Hallam University during the assembly process.

4 Control architecture

The control architecture composed of Leap Motion controller, an Arduino board, and the hand control board consisting of a wireless communication board and proper cabling to connect all the servomotors and their power supply. The Leap Motion controller detects the human hand gesture input. The leap software with visual studio is used to collect X, Y, Z, Pitch, Yaw, Roll coordinates of the hand as referring to its center and the position of each fingertip versus time. A dedicated Matlab code elaborates the collected data and outputs the required angular positions of each servomotor versus time. This information is sent via serial communication to the Arduino board, which drives the servomotor for the desired hand control as summarized in the scheme of Fig.6. Additionally, Fig.7 summarizes the data transfer and communication logics between Matlab and Arduino.

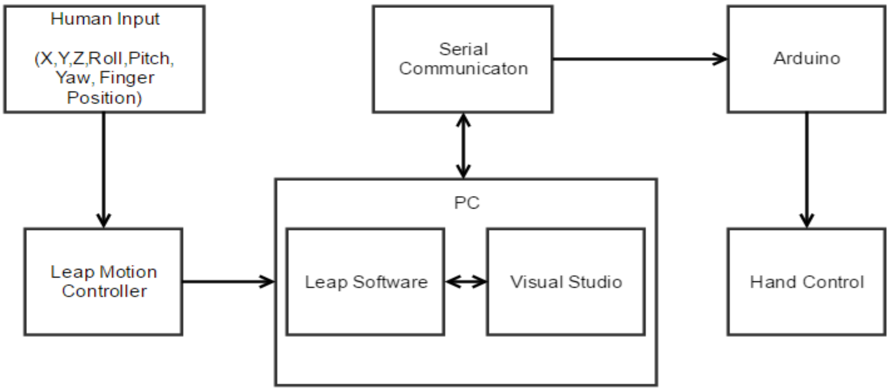


Fig.6 A scheme of the hardware set-up.

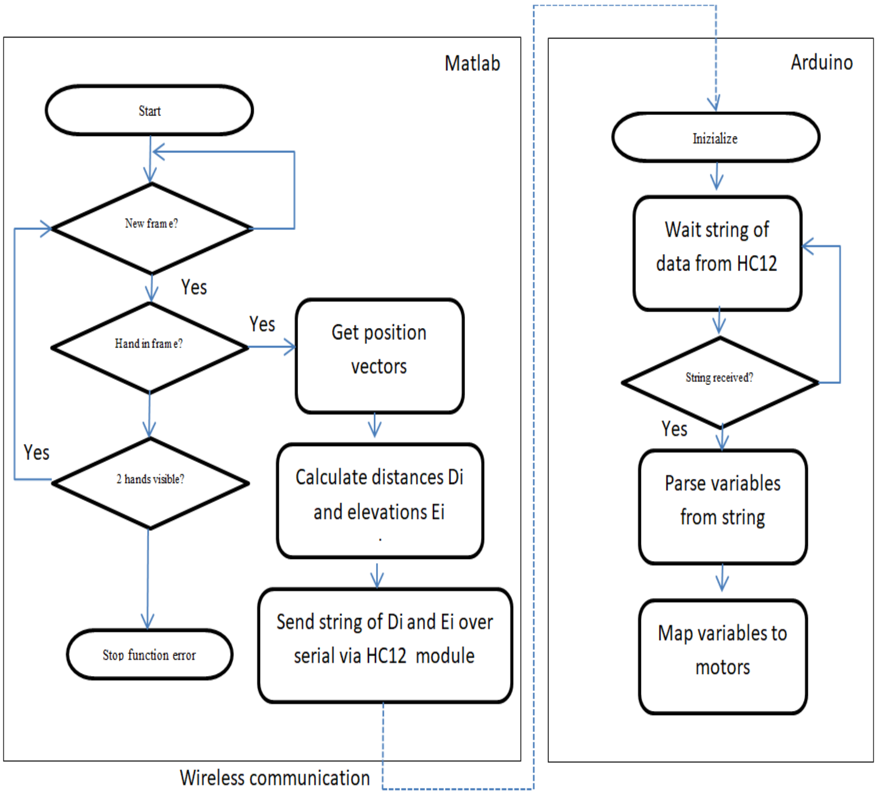


Fig.7 A flow-chart of the operation set-up of the proposed system.

5 Experimental tests

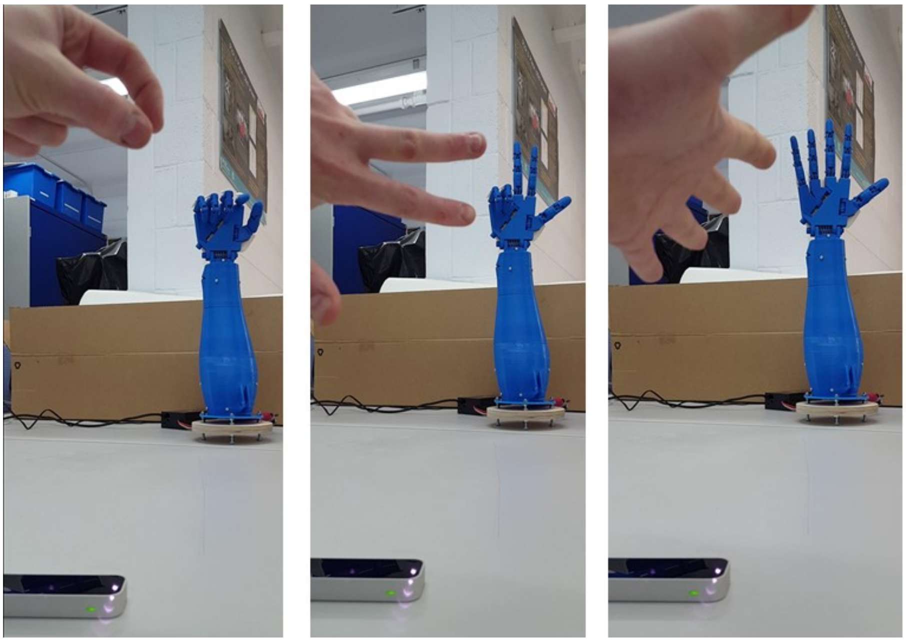
Several tests have been specifically carried out with the proposed set up. A first set of tests consisted of basic movements with individual fingers to verify the proper operation of each component and the related settings.

Following the success with basic movements, the hand was then used for more complex tasks. For example, tests have been made for validating the dexterity of the hand consisting of holding a selection of common tools. The hand has a span radius of 95mm, this makes handling large tools easier than handling small ones. It also means that the dexterity of the hand is reduced, making it harder to complete relatively intricate tasks. The hand can effectively operate scissors, pliers and other basic hand tools with ease when controlled with the Leap Motion and implemented software.

Further tests consisted of direct serial commands containing the necessary finger positions, which were sent via a wired connection and then via the HC12 module for validating accuracy and repeatability of the hands movements. In these tests the hand responded exactly as expected in both connection options reproducing accurately different finger positions and grips. The wireless test was conducted at around 5 meters without any noticeable time delay between the wired and wireless connections.

Remote control of finger counting from 1 to 5 was successfully achieved as shown in Fig.8. Moreover, several other basic grasping tasks of daily used have been tested. For example, the hand can operate buttons and simple switches such as shown, for example, in Fig.9a). Power grasp is also possible as shown, for example, in Fig.9b). Envelop grasping of a ball is shown in Fig.10. The hand has strength enough to hold its own weight.

Further tests were based on changing the operator hand size and skills to verify the operator influence in the control performance. Regrettably, people with different sized hands struggled to control the robotic hand using the Leap with the same accuracy implying the need of some preliminary training before use of the proposed control. Furthermore, there is a need of implementing a proper calibration when the operator is changed. As the hand is controlled by a user's movements the most prominent cause of accuracy error in repeating grips and positions, is caused by the user. The biggest issue when controlling the hand is visual occlusion with the Leap Motion. Bright daylight can cause sporadic values to be produced, which in turn make the hand act erratically. Best test conditions are in an office, or darker, environment.

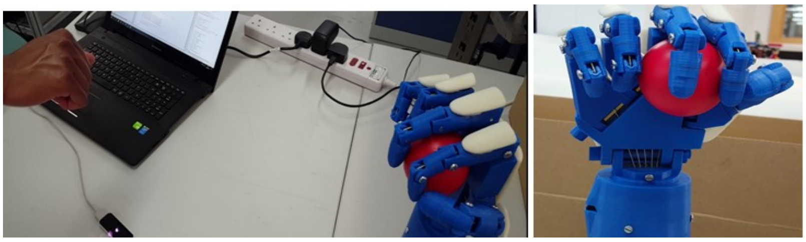


a) b) c)

Fig.8 Examples of the operation of the InMoov hand using the leap sensor: a) pinch grasp configuration with hand fully closed; b) three open fingers; c) hand fully open.

a) b)

Fig.9 Examples of successfull graspings: a) thumb opposition for key grasp; b) power grasping for holding a pole.



a) b)

Fig.10 An example of successfull grasping: a) holding a ball with envelope grasping; b) z zoom view of the grasping.

6 Conclusions

This paper has addressed the possibility of achieving a low-cost gesture-based remote control of a robotic hand. An InMoov robotic hand has been manufactured and the proposed low-cost control architecture has been implemented as based on a commercial leap motion sensor and an Arduino board. Experimental tests have been reported showing the effectiveness of the proposed remote control for performing various grasping tasks.

References

[1] Carbone G. (Ed.), “Grasping in Robotics”, Springer, Dordrecht, 2013.

[2] Mason MT, Salisbury JK, “Robots hands and the mechanics of manipulation”, MIT Press, Cambridge, 1985.

[3] M. V. Liarokapis, P. K. Artemiadis, and K. J. Kyriakopoulos, “Telemanipulation with the DLR/HIT II robot hand using a dataglove and a low cost force feedback device”, 21st Mediterranean Conference on Control and Automation, pp. 431–436, 2013.

[4] Arns M.; Laliberté T.; Gosselin C., “Design, control and experimental validation

of a haptic robotic hand performing human-robot handshake with human-like agility”, IEEE/RSJ International Conference on Intelligent Robots and Systems IROS, pp.4626 – 4633, 2017.

[5] Raparelli T, Mattiazzo G, Mauro S, Velardocchia M, “Design and development of a pneumatic anthropomorphic hand”. J Robot Syst 17(1):1–15, 2000.

[6] Carbone, G., Iannone, S., Ceccarelli, M., “Regulation and control of LARM Hand III”, Robotics and Computer-Integrated Manufacturing, Volume 26, Issue 2, pp. 202-211, 2010.

[7] Carbone, G., Ceccarelli, M., “Experimental tests on feasible operation of a finger mechanism in the LARM hand”, Mechanics Based Design of Structures and Machines, 36(1), pp. 1-13, 2008.

[8] Yao S., Ceccarelli M. , Carbone G., Zhan Q., Lu Z., “Analysis and Optimal Design of an Underactuated Finger Mechanism for LARM Hand”, Frontiers of Mechanical Engineering, vol.6, n.3, pp.332-343, 2011.

[9] Carbone G., Iannone S., Ceccarelli M., “Regulation and Control of LARM Hand III”, Rob. and Comp. Int. Manu., Vol.26, pp.202-211, 2010.

[10] Russo M., Ceccarelli M., Corves B., Hüsing M., Lorenz M., Cafolla D., Carbone G., “Design and test of a gripper prototype for horticulture products”, Robotics and Computer-Integrated Manufacturing, Vol.44, pp.266-275, 2017.

[11] Yao S., Ceccarelli M., Carbone G., Dong Z., “Grasp configuration planning for a low-cost and easy-operation underactuated three-fingered robot hand”, Mechanism and Machine Theory, Vol. 129, pp.51–69, 2018.

[12] H. Cheng, L. Yang, and Z. Liu, “A survey on 3D hand gesture recognitions,” IEEE Trans. Circuits Syst. Video Technol., 2015, doi: 10.1109/TCSVT.2015.2469551.

[13] Juan Pablo Wachs, Mathias Kolsch, Helman Stern, and Yael Edan, “Vision-based hand-gesture applications,” Commun. ACM, vol. 54, no. 2, pp. 60–71, Feb. 2011.

[14] Ren Z., Yuan J., Meng J., Zhang Z., “Robust Part-Based Hand Gesture Recognition Using Kinect Sensor”, IEEE Trasactions on Multimedia, 15(5), pp.1110-1120, 2013.

[15] Leap Motion, “Leap motion sensor”, available online at https://www.leapmotion.com/, Retrieved 1st June 2018.

[16] Weichert, F., Bachmann, D., Rudak, B., Fisseler, D., Analysis of the accuracy and robustness of the leap motion controller, Sensors, 13(5), pp.6380–6393, 2013.

[17] Bock, T., Linner, T., Guettler, J., Georgoulas, C., & Bassily, D., “Intuitive and Adaptive Robotic Arm Manipulation using the Leap Motion Controller”, IEEE 41st Int. Symposium on Robotics ISR/Robotik 2014, Munich, pp. 1-7, 2014.

[18] T. Feix, J. Romero, C. H. Ek, H.-B. Schmiedmayer, and D. Kragic. A Metric for Comparing the Anthropomorphic Motion Capability of Artificial Hands. IEEE Transactions on Robotics, 29 (1):82–93, 2013.

[19] InMoov Robot webpage, “hand-and-forarm”, available online at http://inmoov.fr/hand-and-forarm/, last retrieved 1st June 2018.

[20] Elecrow webpage, “HC12 User Manual HC-12 Wireless Serial Port Communication Module User Manual V1”, available online at https://www.elecrow.com/download/HC-12.pdf, last retrieved 1st June 2018.