Full-Movement Simulation of Humanoid Robot Hand Controlled by Sensory Glove

Vu Le Huy[1,2*], Le Manh Trung[1], Pham Ngoc Viet[1], Tran Trong Tu[1], Le Van Tuan[1], Nguyen Duy Hung[1], Nguyen Chi Hung[3]

[1]Faculty of Mechanical Engineering and Mechatronics, PHENIKAA University, Yen Nghia, Ha Dong, 12116 Hanoi, Vietnam; huy.vule@phenikaa-uni.edu.vn; 21013314@st.phenikaa-uni.edu.vn; 21012363@st.phenikaa-uni.edu.vn; 21011453@st.phenikaa-uni.edu.vn; 21013321@st.phenikaa-uni.edu.vn; 21010382@st.phenikaa-uni.edu.vn

[2]PHENIKAA Research and Technology Institute (PRATI), A&A Green Phoenix Group JSC, No.167 Hoang Ngan, Trung Hoa, Cau Giay, 11313 Hanoi, Vietnam

[3]Faculty of Mechatronics, School of Mechanical Engineering, Hanoi University of Science and Technology, Dai Co Viet, Hai Ba Trung, Hanoi, Vietnam; hung.nguyenchi@hust.edu.vn

Abstract: The humanoid robots as well as the 5-finger robot hand have gradually appeared in life with more and more functions. This paper presents a design of system with a sensory glove and software to simulate fully movements of humanoid robot hand with the spread and flexion movements of the fingers. The kinematic problem and 3D model of the robot hand with 22 rotational degrees of freedom is built in this study according to the structure of the human hand. The glove uses 17 sensors of GY-521 6DOF IMU MPU6050 with the main board of Arduino mega 2560 to collect the movement data of phalanges and carpal in the real time. The measured data filtered by Kalmann filter is transmitted to the simulation software on the computer through the serial port to update the full movement of the 3D robot hand model. The first version of the sensorized glove and the software was successfully created and tested with the result showing that the 3D robot hand model followed well the human hand movement although there is still a large delay.

Keywords: humanoid robot hand; sensory glove; simulation

1 Introduction

The appearance of humanoid robot has changed a lot of things, where 5-finger robot hand is one of the most interests of researchers to develop the humanoid robot. Humanoid robot hand has been applied in the medical field or operations that need to simulate the movement of the user's hand from a distance. It is also expected to

replace human hands in the dexterous manipulation of objects, where some kinds of humanoid robot hand were reviewed [1, 2]. Some studies have developed humanoid robot hand equipped sensor to recognize objects or perform manipulations [3-5]. Recently, robotic hands have been used to assist doctors in therapeutic or rehabilitation of hands for inpatient stroke, combined assistance, and at-home rehabilitation ... [6-8], or used in sports training [5], operation in virtual space by combining with virtual reality glasses [10], applications of remote device control for robot arms, automation guided vehicle [11, 12], etc.

In order to simplify the study of structure, operating principle, solve kinematic problems of controlling the robot hand or serve for testing and pre-production testing, the use of simulation software is very necessary. For this goal, gloves with sensors are used to determine the movement of the fingers [13]. Some gloves use resistive sensors mounted along the finger length [11, 14]. This method gives the glove with a neat design, but it could not determine the exact rotation angle of each phalange. Recently, several studies have focused on developing gloves using the MPU6050 accelerometer sensor [9, 11, 15-18]. The MPU6050 sensors are attached to separate phalanges, thereby calculating the movement between the phalanges to obtain the rotation angle. Using MPU6050 sensor makes the glove bulkier in size, but it determines the movement of each phalange with high accuracy, and thus gives more useful applications. The previous studies have focused mainly on the development of glove design and data acquisition control program, but there has not been any research on the overall development of the system from glove design to real-time simulation software.

Nowadays, the manufacturing and production of 5-finger humanoid robot hands for disabilities as well as in civil applications has great potential for development. However, a big concern is the problems of simulation and control of the robot hand according to the gestures that the user is performing or wants to perform, along with the goal of building a software that can simulate the operation and proceed to control the robot hand. Simulating humanoid robot hand movements can help understand the principles of structure and operation, create appropriate control commands that can be applied to different tasks. Then, it can be easily developed and expanded into many useful applications in life. This paper presents a new designed system with a MPU6050 sensorized glove connected to a software in computer through the serial port to simulate fully movements of humanoid robot hand. Next, a 3D robot hand model is designed and its kinematics problem is established. Then the simulation software is introduced. Finally, the sensory glove as well as the simulation result based on the data obtained from the glove is presented.

2 Design and Kinematics of the Humanoid Robot Hand

Basically, the human hand consists of 3 parts which are the palm, the back of the hand, and the fingers. There are five fingers on a human hand named as thumb, index finger, middle finger, ring finger, little finger. Anatomically, the hand bone structure consists of 27 bones (8 wrist bones, 5 bones of the hand, 14 bones of the fingers), where the bones are joined together by 29 joints and at least 123 ligaments [19]. It is difficult to simulate exactly the very complex movements of the human hand. However, the design of humanoid robot hand should simulate the most necessary gestures as close as possible. Many studies suggested different designs of humanoid robot hand with reduction of degrees of freedom (DOF) [1, 2-5, 20-22]. In this study, the design of the robot hand is required to perform the basic movements of a human hand with 22 rotational DOF as shown in Fig. 1. The hand is connected to the arm by the wrist, where the arm is assumed to be a fixed link. Each finger of the robotic hand consists of four phalanges that simulate the spread and flexion of a finger. The remaining joints have the same function as the human hand. This model is then used to illustrate the movement of the robot hand on our simulation software.

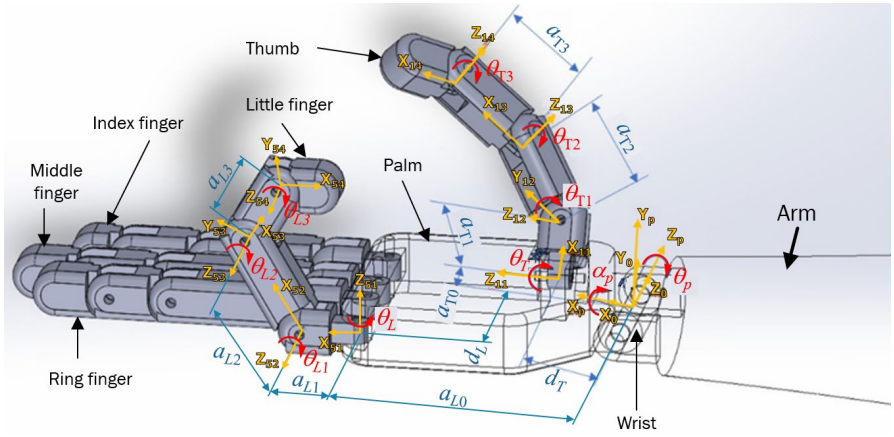


Figure 1
Design and coordinate systems of the humanoid robot hand

In order to implementation the simulation software with 3D model of the designed robot hand, kinematic problem is performed with the local coordinate systems attached to the parts by using the Denavit-Hartenberg (D-H) method [14]. The coordinate systems of the robot hand according to the D-H method are shown in Fig. 1 for the palm, thumb, and little finger. The coordinate systems for the index, middle and ring fingers are similar to those of the little finger. Table 1 shows the D-H parameters of the local coordinate systems used for the parts of the designed robot

hand. Each finger consists of 4 links numbered from 0 to 3, where the link 0 and the links from 1 to 3 help the fingers performing the spread and flexion motions, respectively. The coordinate transformation matrix [24] for the i^{th} link is formulated in D-H matrix as

$$ {}^{i-1}A_i = \begin{bmatrix} cos\,\theta_i & -sin\,\theta_i\,cos\,\alpha_i & sin\,\theta_i\,sin\,\alpha_i & a_i\,cos\,\theta_i \\ sin\,\theta_i & cos\,\theta_i\,cos\,\alpha_i & -cos\,\theta_i\,sin\,\alpha_i & a_i\,sin\,\theta_i \\ 0 & sin\,\alpha_i & cos\,\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) $$

Table 1
D-H parameter table of the designed robot hand

Part	**Link**	a_i	θ_i	d_i	α_i
Wrist	1	0	0	0	α_p
Palm	0	0	θ_p	0	α_p
Thumb	0	a_{T0}	θ_T	d_T	0
	1	a_{T1}	θ_{T1}	0	0
	2	a_{T2}	θ_{T2}	0	$\pi/2$
	3	a_{T3}	θ_{T3}	0	0
Index finger	0	a_{I0}	θ_I	d_I	0
	1	a_{I1}	θ_{I1}	0	0
	2	a_{I2}	θ_{I2}	0	0
	3	a_{I3}	θ_{I3}	0	0
Middle finger	0	a_{M0}	θ_M	d_M	0
	1	a_{M1}	θ_{M1}	0	0
	2	a_{M2}	θ_{M2}	0	0
	3	a_{M3}	θ_{M3}	0	0
Ring finger	0	a_{R0}	θ_R	d_R	0
	1	a_{R1}	θ_{R1}	0	0
	2	a_{R2}	θ_{R2}	0	0
	3	a_{R3}	θ_{R3}	0	0
Little finger	0	a_{L0}	θ_L	d_L	0
	1	a_{L1}	θ_{L1}	0	0
	2	a_{L2}	θ_{L2}	0	0
	3	a_{L3}	θ_{L3}	0	0

where d_i is the offset of the i^{th} coordinate system along the Z_{i-1} axis of the $(i\text{-}1)^{th}$ coordinate system; θ_i is the rotation about the Z_i axis so that the X_{i-1} axis is in the same direction as the X_i axis; a_i is the length of the common normal of the Z_{i-1} and Z_i axes; α_i is the rotation about the common normal of the Z_{i-1} and Z_i axes so that the Z_{i-1} axis is in the same direction as the Z_i axis. The position of the n^{th} link with

respect to the original coordinate system is obtained as the product of the matrices representing the pose of *i* from 1 to *n* as

$$^{0}A_n = \prod_{i=1}^{n} {}^{i-1}A_i \tag{2}$$

It was applied to all the links on the designed robot hand with the parameters in Table 1 used for calculation in the sensory glove and the simulation software. Then the transformation matrices for the arm and the palm from the $X_pY_pZ_p$ coordinate system to the $X_0Y_0Z_0$ coordinate system are obtained respectively as

$$^{0}A_a = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_p & -s\theta_p & 0 \\ 0 & s\theta_p & c\alpha_p & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$^{0}A_p = \begin{bmatrix} c\theta_p & -s\theta_p & 0 & 0 \\ s\theta_p c\alpha_L & c\theta_L c\alpha_p & -s\alpha_p & 0 \\ s\theta_p s\alpha_p & c\theta_p s\alpha_p & -s\alpha_p & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

where *c* and *s* stand for cos and sin functions. The transformation matrix for the phalanges of the fingers is written in general form as

$$^{0}A_{fk} = \begin{bmatrix} a_{f11}^{k} & a_{f12}^{k} & a_{f13}^{k} & a_{f14}^{k} \\ a_{f21}^{k} & a_{f22}^{k} & a_{f23}^{k} & a_{f24}^{k} \\ a_{f31}^{k} & a_{f32}^{k} & a_{f33}^{k} & a_{f34}^{k} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

where *f* stands for the name of a finger as T, I, M, R and L corresponding to the thumb, index, middle, ring, and little fingers. The letter *k* is the index of a link on the finger as shown in Table 1. Formulas of the items a_{Tij}^{k} in the matrix $^{0}A_{Tk}$ of the k^{th} link on the thumb are given in Table 2. Those of the items a_{fij}^{k} in the matrix $^{0}A_{fk}$ of the k^{th} link on the index, middle, ring, and little fingers are given in Table 3. Using the transformation matrices with the join parameters φ_p, θ_p, φ_T, δ_f, q_{f1}, q_{f2}, q_{f3}, (*f* is T, I, M, R and L) obtained from the human hand by the sensors, the humanoid robot hand will simulate the real hand.

Table 2

Formulas of the items in the matrix for the links of the thumb

Part	**Link**	$^{0}A_{Tk}$	a_{Tij}^{k}	
Thumb	0	$^{0}A_{T0}$	$a_{T11}^{0} = c\theta_p$	$a_{T13}^{0} = s\theta_T s\theta_p$
			$a_{T12}^{0} = -c\theta_T s\theta_p$	$a_{T14}^{0} = a_{T0}c\theta_p$
			$a_{T21}^{0} = c\alpha_p s\theta_p$	$a_{T23}^{0} = -s\theta_T c\alpha_p c\theta_p - c\theta_T s\alpha_p$
			$a_{T22}^{0} = c\theta_T c\alpha_p c\theta_p - s\theta_T s\alpha_p$	$a_{T24}^{0} = a_{T0}c\alpha_p s\theta_p - d_T s\alpha_p$
			$a_{T31}^{0} = s\alpha_p s\theta_p$	$a_{T33}^{0} = -s\theta_T s\alpha_p c\theta_p + c\theta_T c\alpha_p$

			$a^0_{T32} = c\theta_T s\alpha_p c\theta_p + s\theta_T c\alpha_p$ | $a^0_{T34} = a_{T0} s\alpha_p s\theta_p + d_T c\alpha_p$
	1	${}^0A_{T1}$	$a^1_{T11} = c\theta_p c\theta_{T1} - s\theta_T c\theta_p s\theta_{T1}$
			$a^1_{T12} = -c\theta_T s\theta_p$
			$a^1_{T13} = s\theta_T s\theta_p c\theta_{T1} + c\theta_p s\theta_{T1}$
			$a^1_{T14} = a_{T0} c\theta_p + a_{T1} s\theta_p s\theta_T$
			$a^1_{T21} = s\theta_{T1}(c\theta_T s\alpha_p + s\theta_T c\theta_L c\theta_p) + c\theta_{T1} c\alpha_p s\theta_p$
			$a^1_{T22} = c\theta_T c\alpha_p c\theta_p - s\theta_T s\alpha_p$
			$a^1_{T23} = -c\theta_{T1}(c\theta_T s\alpha_p + s\theta_T c\alpha_p c\theta_p) + s\theta_{T1} c\alpha_p s\theta_p$
			$a^1_{T24} = a_{T0} c\alpha_p s\theta_p - a_{T1}(c\theta_T s\alpha_p + s\theta_T c\alpha_p c\theta_p) - d_T s\alpha_p$
			$a^1_{T31} = -s\theta_{T1}(c\theta_T c\alpha_p - s\theta_T s\alpha_p c\theta_p) + c\theta_{T1} s\alpha_p s\theta_p$
			$a^1_{T32} = c\theta_T s\alpha_p c\theta_p + s\theta_T c\alpha_p$
			$a^1_{T33} = c\theta_{T1}(c\theta_T c\alpha_p - s\theta_T s\alpha_p c\theta_p) + s\theta_{T1} s\alpha_p s\theta_p$
			$a^1_{T34} = a_{T0} s\alpha_p s\theta_p + d_T c\alpha_p + a_{T1}(c\theta_T c\alpha_p s\theta_T s\alpha_p c\theta_p)$
	2	${}^0A_{T2}$	$a^2_{T11} = c\theta_{T2}(c\theta_p c\theta_{T1} - s\theta_T s\theta_p s\theta_{T1}) - s\theta_{T2}(c\theta_p s\theta_{T1} + s\theta_T s\theta_p c\theta_{T1})$
			$a^2_{T12} = -c\theta_T s\theta_p$
			$a^2_{T13} = c\theta_{T2}(c\theta_p s\theta_{T1} + s\theta_T s\theta_p c\theta_{T1}) + s\theta_{T2}(c\theta_p c\theta_{T1} - s\theta_T s\theta_p s\theta_{T1})$
			$a^2_{T14} = a_{T0} c\theta_p + a_{T1} s\theta_p s\theta_T + a_{T2}(c\theta_p s\theta_{T1} + s\theta_T s\theta_p c\theta_{T1})$
			$a^2_{T21} = s\theta_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p + s\theta_T c\theta_p s\theta_L) - c\alpha_p s\theta_p s\theta_{T1}\} + c\theta_{T2}\{s\theta_{T1}(c\theta_T s\alpha_p + s\theta_T c\theta_p c\alpha_p) + c\alpha_p s\theta_p c\theta_{T1}\}$
			$a^2_{T22} = -(s\theta_T s\alpha_p - c\theta_T c\theta_p c\alpha_p)$
			$a^2_{T23} = -c\theta_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p + c\theta_L s\theta_T c\alpha_p) - c\alpha_p s\theta_p s\theta_{T1}\} + s\theta_{T2}\{s\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p) + c\alpha_p c\theta_{T1} s\theta_p\}$
			$a^2_{T24} = a_{T0} c\alpha_p s\theta_p - a_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p) - a_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p) - c\alpha_p s\theta_p s\theta_{T1}\} - d_T s\alpha_p$
			$a^2_{T31} = -s\theta_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p - s\theta_T c\theta_p s\alpha_p) + s\alpha_p s\theta_p s\theta_{T1}\} - c\theta_{T2}\{s\theta_{T1}(c\theta_T c\alpha_p - s\theta_T c\theta_p s\alpha_p) - s\alpha_p s\theta_p c\theta_{T1}\}$
			$a^2_{T32} = s\theta_T c\alpha_p + c\theta_T c\theta_p s\alpha_p$
			$a^2_{T33} = c\theta_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p) + s\alpha_p s\theta_p s\theta_{T1}\} - s\theta_{T2}\{s\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p) - c\theta_{T1} s\alpha_p s\theta_p\}$
			$a^2_{T34} = a_{T0} s\alpha_p s\theta_p + a_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p) + a_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p) + s\alpha_p s\theta_p s\theta_{T1}\} + d_T c\alpha_p$
	3	${}^0A_{T3}$	$a^3_{T11} = -c\theta_{T3}\{s\theta_{T2}(c\theta_p s\theta_{T1} + c\theta_{T1} s\theta_T s\theta_p) - c\theta_{T2}(c\theta_p c\theta_{T1} - s\theta_T s\theta_p s\theta_{T1})\}$

			$-s\theta_{T3}\{c\theta_{T2}(c\theta_p + c\theta_{T1}s\theta_T s\theta_p)$ $+s\theta_{T2}(c\theta_p c\theta_{T1} - s\theta_T s\theta_p s\theta_{T1})\}$
			$a^3_{T12} = -c\theta_T s\theta_p$
			$a^3_{T13} = c\theta_{T3}\{c\theta_{T2}(c\theta_p s\theta_{T1} + c\theta_{T1}s\theta_T s\theta_p)$ $+s\theta_{T2}(c\theta_p c\theta_{T1} - s\theta_T s\theta_p s\theta_{T1})\} - s\theta_{T3}\{s\theta_{T2}(c\theta_p s\theta_{T1}$ $+c\theta_{T1}s\theta_T s\theta_p) - c\theta_{T2}(c\theta_p c\theta_{T1} - s\theta_T s\theta_p s\theta_{T1})\}$
			$a^3_{T14} = a_{T0}c\theta_p + a_{T1}s\theta_T s\theta_p + a_{T2}(c\theta_p s\theta_{T1} + c\theta_{T1}s\theta_T s\theta_p)$ $+a_{T3}\{c\theta_{T2}(c\theta_p s\theta_{T1} + c\theta_{T1}s\theta_T s\theta_p)$ $+s\theta_{T2}(c\theta_p c\theta_{T1} - s\theta_T s\theta_p s\theta_{T1})\}$
			$a^3_{T21} = c\theta_{T3}\{s\theta_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p)$ $-c\alpha_p s\theta_p s\theta_{T1}\} + c\theta_{T2}\{s\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p)$ $+c\theta_{T1}c\alpha_p s\theta_p\}\} + s\theta_{T3}\{c\theta_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p$ $+c\theta_p s\theta_T c\alpha_p) - c\alpha_p s\theta_p s\theta_{T1}\} - c\varphi_T s\theta_{T2}\{s\theta_{T1}(c\theta_T s\alpha_p$ $+c\theta_p s\theta_T c\alpha_p) + c\theta_{T1}c\alpha_p s\theta_p\}\}$
			$a^3_{T22} = c\theta_T c\theta_p c\alpha_p - s\theta_T s\alpha_p$
			$a^3_{T23} = s\theta_{T3}\{s\theta_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p)$ $-c\alpha_p s\theta_p s\theta_{T1}\} + c\theta_{T2}\{s\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p)$ $+c\theta_{T1}c\alpha_p s\theta_p\}\} - c\theta_{T3}\{c\theta_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p$ $+c\theta_p s\theta_T c\alpha_p) - c\alpha_p s\theta_p s\theta_{T1}\} - s\theta_{T2}\{s\theta_{T1}(c\theta_T s\alpha_p$ $+c\theta_p s\theta_T + c\theta_{T1}c\alpha_p s\theta_p\}\}$
			$a^3_{T24} = a_{T0}c\alpha_p s\theta_p - a_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p)$ $-a_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p) - c\alpha_p s\theta_p s\theta_{T1}\}$ $-a_{T3}\{c\theta_{T2}\{c\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p)$ $-c\alpha_p s\theta_p s\theta_{T1}\} - s\theta_{T2}\{s\theta_{T1}(c\theta_T s\alpha_p + c\theta_p s\theta_T c\alpha_p)$ $+c\theta_{T1}c\alpha_p s\theta_p\}\} - d_T s\alpha_p$
			$a^3_{T31} = -c\theta_{T3}\{s\theta_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p)$ $+s\theta_p s\theta_{T1}s\alpha_p\} + c\theta_{T2}\{s\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p)$ $-c\theta_{T1}s\theta_p s\alpha_p\}\} - s\theta_{T3}\{c\theta_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p$ $-c\theta_p s\theta_T s\alpha_p) + s\theta_p s\theta_{T1}s\alpha_p\} - s\theta_{T2}\{s\theta_{T1}(c\theta_T c\alpha_p$ $-c\theta_p s\theta_T s\alpha_p) - c\theta_{T1}s\theta_p s\alpha_p\}\}$
			$a^3_{T32} = s\theta_T c\alpha_p + c\theta_T c\theta_p s\alpha_p$
			$a^3_{T33} = c\theta_{T3}\{c\theta_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p)$ $+s\theta_p s\theta_{T1}s\alpha_p\} - s\theta_{T2}\{s\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p)$ $-c\theta_{T1}s\theta_p s\alpha_p\}\} - s\theta_{T3}\{s\theta_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p$ $-c\theta_p s\theta_T s\alpha_p) + s\theta_p s\theta_{T1}s\alpha_p\} + c\theta_{T2}\{s\theta_{T1}(c\theta_T c\alpha_p$ $-c\theta_p s\theta_T s\alpha_p) - c\theta_{T1}s\theta_p s\alpha_p\}\}$
			$a^3_{T34} = a_{T0}s\theta_p s\alpha_p + a_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p)$ $+a_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p) + s\theta_p s\theta_{T1}s\alpha_p\}$

			$+a_{T3}\{c\theta_{T2}\{c\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p)$ $+s\theta_p s\theta_{T1} s\alpha_p\} - s\theta_{T2}\{s\theta_{T1}(c\theta_T c\alpha_p - c\theta_p s\theta_T s\alpha_p)$ $-c\theta_{T1} s\theta_p s\alpha_p\}\} + d_c c\alpha_p$

Table 3
Formulas of the items in the matrix ${}^0A_{fk}$ for the links of the other fingers

Part	**Link**	${}^0A_{fk}$	a_{fij}^k	
Index, Middle, Ring, Little fingers	0	${}^0A_{f0}$	$a_{f11}^0 = c\theta_f c\theta_p$	$a_{f13}^0 = s\theta_f c\theta_p$
			$a_{f12}^0 = -s\theta_p$	$a_{f14}^0 = a_{f0} c\theta_p$
			$a_{f21}^0 = s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p$	$a_{f23}^0 = s\theta_f c\alpha_p s\theta_p - c\theta_f s\alpha_p$
			$a_{f22}^0 = c\theta_p c\alpha_p$	$a_{f24}^0 = a_{f0} c\alpha_p s\theta_p - d_f s\alpha_p$
			$a_{f31}^0 = c\theta_f s\theta_p s\alpha_p - s\theta_f c\alpha_p$	$a_{f33}^0 = c\theta_f c\alpha_p + s\theta_f s\theta_p s\alpha_p$
			$a_{f32}^0 = c\theta_p s\alpha_p$	$a_{f34}^0 = a_{f0} s\theta_p s\alpha_p + d_f c\alpha_p$
	1	${}^0A_{f1}$	$a_{f11}^1 = c\theta_f c\theta_p c\theta_{f1} - s\theta_p s\theta_{f1}$	
			$a_{f12}^1 = -c\theta_{f1} s\theta_p - c\theta_f c\theta_p s\theta_{f1}$	
			$a_{f13}^1 = c\theta_p s\theta_f$	
			$a_{f14}^1 = a_{f0} c\theta_p + a_{T1} c\theta_f c\theta_p$	
			$a_{f21}^1 = c\theta_p c\alpha_p s\theta_{f1} + c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p)$	
			$a_{f22}^1 = c\theta_p c\theta_{f1} c\alpha_p - s\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p)$	
			$a_{f23}^1 = s\theta_f c\alpha_p s\theta_p - c\theta_f s\alpha_p$	
			$a_{f24}^1 = a_{f0} c\alpha_p s\theta_p + a_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) - d_f s\alpha_p$	
			$a_{f31}^1 = c\theta_p s\theta_{f1} s\alpha_p - c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p)$	
			$a_{f32}^1 = s\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) + c\theta_p c\theta_{f1} s\alpha_p$	
			$a_{f33}^1 = c\theta_f c\alpha_p + s\theta_f s\theta_p s\alpha_p$	
			$a_{f34}^1 = a_{f0} s\theta_p s\alpha_p - a_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) + d_f c\alpha_p$	
	2	${}^0A_{f2}$	$a_{f11}^2 = -c\theta_{f2}(s\theta_p s\theta_{f1} - c\theta_f c\theta_p c\theta_{f1}) - s\theta_{f2}(c\theta_{f1} s\theta_p$ $+c\theta_f c\theta_p s\theta_{f1})$	
			$a_{f12}^2 = s\theta_{f2}(s\theta_p s\theta_{f1} - c\theta_f c\theta_p c\theta_{f1}) - c\theta_{f2}(c\theta_{f1} s\theta_p$ $+c\theta_f c\theta_p s\theta_{f1})$	
			$a_{f13}^2 = c\theta_p s\theta_f$	
			$a_{f14}^2 = a_{f0} c\theta_p + a_{f1} c\theta_f c\theta_p - a_{f2}(s\theta_{f1} s\theta_p - c\theta_{f1} c\theta_f c\theta_p)$	
			$a_{f21}^2 = c\theta_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1} c\theta_p c\alpha_p\}$ $-s\theta_{f2}\{s\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) - c\theta_{f1} c\theta_p c\alpha_p\}$	
			$a_{f22}^2 = -c\theta_{f2}\{s\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) - c\theta_{f1} c\theta_p c\alpha_p\}$ $-s\theta_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1} c\theta_p c\alpha_p\}$	
			$a_{f23}^2 = s\theta_f c\alpha_p s\theta_p - c\theta_f s\alpha_p$	

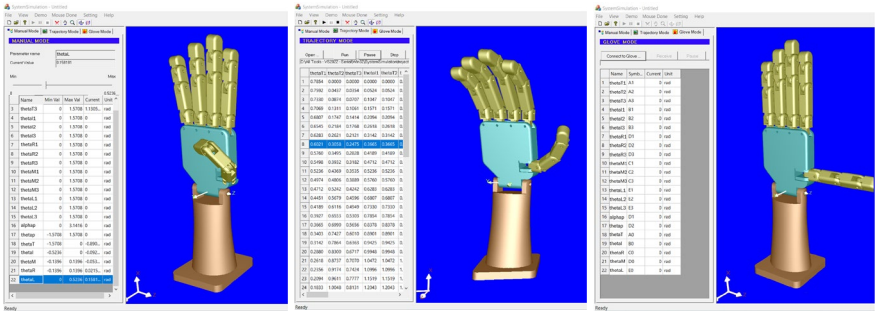
			$a_{f24}^2 = a_{f0}c\alpha_p s\theta_p + a_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + a_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1}c\theta_p c\alpha_p\} - d_f s\alpha_p$
			$a_{f31}^2 = s\theta_{f2}\{s\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) + c\theta_{f1}c\theta_p s\alpha_p\} - c\theta_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\}$
			$a_{f32}^2 = c\theta_{f2}\{s\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) + c\theta_{f1}c\theta_p s\alpha_p\} + s\theta_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\}$
			$a_{f33}^2 = c\theta_f c\alpha_p + s\theta_f s\theta_p s\alpha_p$
			$a_{f34}^2 = a_{f0}s\theta_p s\alpha_p - a_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - a_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\} + d_f c\alpha_p$
	3	${}^0A_{f3}$	$a_{f11}^3 = -c\theta_{f3}\{c\theta_{f2}(s\theta_{f1}s\theta_p - c\theta_{f1}c\theta_f c\theta_p) + s\theta_{f2}(c\theta_{f1}s\theta_p + s\theta_{f1}c\theta_f c\theta_p)\} - s\theta_{f3}\{c\theta_{f2}(c\theta_{f1}s\theta_p + s\theta_{f1}c\theta_f c\theta_p) - s\theta_{f2}(s\theta_{f1}s\theta_p - c\theta_{f1}c\theta_f c\theta_p)\}$
			$a_{f12}^3 = s\theta_{f3}\{c\theta_{f2}(s\theta_{f1}s\theta_p - c\theta_{f1}c\theta_f c\theta_p) + s\theta_{f2}(c\theta_{f1}s\theta_p + s\theta_{f1}c\theta_f c\theta_p)\} - c\theta_{f3}\{c\theta_{f2}(c\theta_{f1}s\theta_p + s\theta_{f1}c\theta_f c\theta_p) - s\theta_{f2}(s\theta_{f1}s\theta_p - c\theta_{f1}c\theta_f c\theta_p)\}$
			$a_{f13}^3 = c\theta_p s\theta_f$
			$a_{f14}^3 = a_{f0}c\theta_p + a_{f1}c\theta_f c\theta_p - a_{f2}(s\theta_p s\theta_{f1} - c\theta_{f1}c\theta_f c\theta_p) - a_{f3}\{c\theta_{f2}(s\theta_p s\theta_{f1} - c\theta_{f1}c\theta_f c\theta_p) + s\theta_{f2}(c\theta_{f1}s\theta_p + s\theta_{f1}c\theta_f c\theta_p)\}$
			$a_{f21}^3 = c\theta_{f3}\{c\theta_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1}c\theta_p c\alpha_p\} - s\theta_{f2}\{s\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) - c\theta_{f1}c\theta_p c\alpha_p\}\} - s\theta_{f3}\{c\theta_{f2}\{s\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) - c\theta_{f1}c\theta_p c\alpha_p\} + s\theta_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1}c\theta_p c\alpha_p\}\}$
			$a_{f22}^3 = -c\theta_{f3}\{c\theta_{f2}\{s\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) - c\theta_{f1}c\theta_p c\alpha_p\} + s\theta_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1}c\theta_p c\alpha_p\}\} - s\theta_{f3}\{c\theta_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1}c\theta_p c\alpha_p\} - s\theta_{f2}\{s\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) - c\theta_{f1}c\theta_p c\alpha_p\}\}$
			$a_{f23}^3 = s\theta_f c\alpha_p s\theta_p - c\theta_f s\alpha_p$
			$a_{f24}^3 = a_{f0}c\alpha_p s\theta_p + a_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + a_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1}c\theta_p c\alpha_p\} + a_{f3}\{c\theta_{f2}\{c\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) + s\theta_{f1}c\theta_p c\alpha_p\} - s\theta_{f2}\{s\theta_{f1}(s\theta_f s\alpha_p + c\theta_f c\alpha_p s\theta_p) - c\theta_{f1}c\theta_p c\alpha_p\}\} - d_f s\alpha_p$
			$a_{f31}^3 = s\theta_{f3}\{c\theta_{f2}\{s\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) + c\theta_{f1}c\theta_p s\alpha_p\}$

			$+s\theta_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\}\}$ $-c\theta_{f3}\{c\theta_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\}$ $-s\theta_{f2}\{s\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) + c\theta_{f1}c\theta_p s\alpha_p\}\}$
			$a^3_{f32} = s\theta_{f3}\{c\theta_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\}$ $-s\theta_{f2}\{s\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) + c\theta_{f1}c\theta_p s\alpha_p\}\}$ $+c\theta_{f3}\{c\theta_{f2}\{s\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) + c\theta_{f1}c\theta_p s\alpha_p\}$ $+s\theta_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\}\}$
			$a^3_{f33} = c\theta_f c\alpha_p + s\theta_f s\theta_p s\alpha_p$
			$a^3_{f34} = a_{f0}s\theta_p s\alpha_p - a_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p)$ $-a_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\}$ $-a_{f3}\{c\theta_{f2}\{c\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\theta_p s\alpha_p) - s\theta_{f1}c\theta_p s\alpha_p\}$ $-s\theta_{f2}\{s\theta_{f1}(s\theta_f c\alpha_p - c\theta_f s\varphi_p s\alpha_p) + c\theta_{f1}c\theta_p s\alpha_p\}\}$ $+d_f c\alpha_p$

3 Simulation Software

In the field of mechanics, there are many software that allow users to build models and perform simulations with 3D models. However, the simulation modules are usually given with the special models of robot, and difficult to customize the software with other robot models or the input parameters. In this study, a 3D simulation software of the robot hand has been built by the authors in Visual C++ with the OpenGL graphics library, where the software interfaces are shown in Fig. 2 with the 3D model of the designed robot hand presented on the right side by OpenGL environment. The movement of the parts in the model is controlled by the transformation matrices presented in the previous section. There are three modes in the software as Manual Mode (Fig. 2a), Trajectory Mode (Fig. 2b) and Glove Mode (Fig. 2c). The use of this simulation software is briefly described by the flowchart in Fig. 3(a), where Glove Mode works with the glove by the process in Fig. 3(b). Manual Mode allows the users viewing the movement of each link by changing the parameters manually. In this mode, there is a grid showing the list of the joint variables with their minimum, maximum and current values. The user could select a parameter in this list and then move the slider to change its value as well as the movement of the 3D hand model. Trajectory Mode gives the movement view of the robot hand with the given joint variables loaded from a text file, where the values of the joint variables could be calculated from an intended trajectory by using software such as Matlab, Mathematica, ect., and saved to ASCII text file. Glove Mode is used to connect the sensory glove to the software for simulation of the actual hand's movement, where the data transmission is performed through a serial port. To using this mode, the sensory glove, which is going to be presented in the next section should be connected to the computer via an USB cable. Then, the user

should press the “Connect to Glove …” button to set the port parameters and connect the sensory glove to the software. If the connection is successful, the 3D model will move following the human hand’s gesture.



(a) Manual Mode (b) Trajectory Mode (c) Glove Mode

Figure 2
Simulation software interface

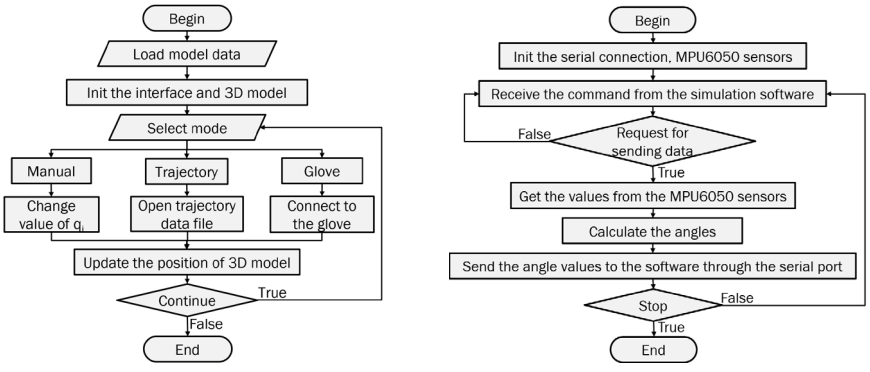
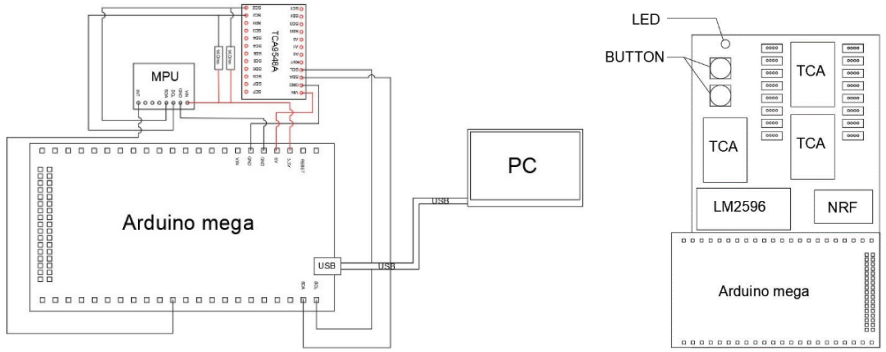


Figure 3
Flowcharts of (a) using the simulation software and (b) working of the glove

4 Sensory Glove

The sensory glove in this study uses a Arduino Mega2560 microcontroller board based on the ATmega2560 for data acquisition from the sensors attached to the phalanges by using a glove. Arduino is selected because of its large RAM capacity, many input/output (I/O) pins, and easy to program. The sensors named as GY-521 6DOF IMU MPU6050 are used to measure the angle between phalanges because of low-cost solution, where each sensor is attached to a phalange. With 22 links of the robot hand as shown in Table 1, 17 sensors are used to collect fully the movements of a human hand, which lead over the number of I/O ports of Arduino board.

In order to overcome this problem, I2C communication was used for multiple sensors with same address, where TCA9548A Address Conversion Module was selected because of its low cost and easy programming. The connection of the modules, sensors and Arduino board is shown by the diagram in Fig. 4(a). The glove is connected to the software by using an USB cable connecting from the USB port of the Arduino board to that of the computer. The Arduino board and the TCA modules are distributed on a base integrated with control buttons that turn on and off receiving data as shown in Fig. 4(a).



(a) (b)

Figure 4

Diagrams of (a) connection of the modules and (b) circuit distribution on the glove

The user will put on gloves and tie the base to the arm as illustrated in Fig. 5(a) and connect the USB cable to the computer as shown in Fig. 5(b). The control program uploaded to Arduino board was coded with Arduino IDE tool, where the algorithm is shown in Fig. 3(b). Then, the angle between the two adjacent phalanges is calculated by the absolute value of the difference of two adjacent sensors along the axis of rotation when folding the fingers. This means that the value of the angles θ_i or α_i in Table 1 is calculated from the $(i\text{-}1)^{th}$ and i^{th} sensors. The Arduino board will send those angle values following the notation of the θ_i and α_i parameters in Table 1 to the simulation software on the computer as shown in the grid in Fig. 2(c). The software receives the values of the angles and displays the 3D robot hand model on the screen following correctly the gesture of the human hand as shown in Fig. 5(b) by using the transformation matrices in the previous section.

(a) (b)

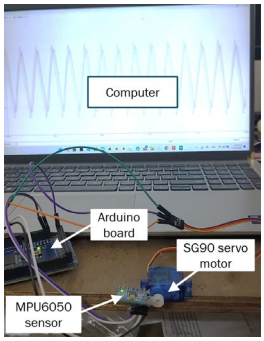
Figure 5
Sensory glove with (a) completed product and (b) connection with the simulation software

5 Result and Discussion

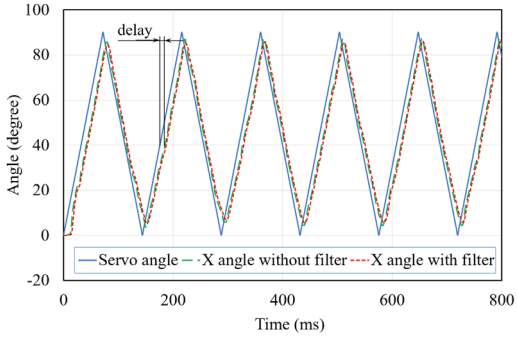
When applying the MPU6050 sensors, one problem with the collected data from the sensors is that the noise from the hand shaking phenomenon for example, which needs to be removed [25]. In order to reduce the noise of the measured data from the MPU6050 sensors, Kalman filter [25-27] was applied by using open library [28]. The definition of this filter as

SimpleKalmanFilter (float mea_e, float est_e, float q);

where mea_e is measurement uncertainty, est_e is estimation uncertainty and q is the process variance. The suitable values for these parameters were found by experimental setup as shown in Fig. 6(a), where the values of mea_e, est_e and q have been set in this study as 0.1, 0.1 and 0.01, respectively. The setup used a MPU6050 sensor held by an arm assembled on the shaft of the SG90 servo motor. The Arduino Mega2560 board was used to control the servo motor and get the return data from the sensor. The motor was controlled to cyclically rotate from 0 to 90 degrees with the velocity of 0.5 degree/ms. The X-axis angle values without and with Kalman filter were calculated and sent to the computer together with the angle value of the motor by the Arduino board. The test result is shown in Fig. 6(b), where the countinuous, long-dash and short-dash curves show the angles of the servo motor shaft, the X-angle of the sensor without and with applying Kalman filter, respectively. It could be seen that the curve of the angle with Kalman filter is smoother than the case without the filter. Then the angular velocity and acceleration were also calculated as plotted in Fig. 7, where the noise (see the green long-dash line) could be seen clearly in the measured data which induce the bad influence in the control process of the robot hand. When applying the Kalman filter (see the red short-dash line), the amplitude of the noise was significantly reduced about 50%.



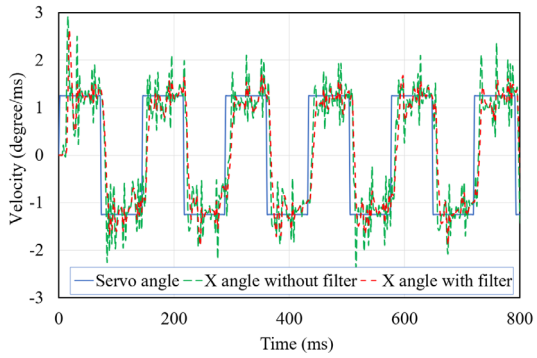
(a) Experimental setup



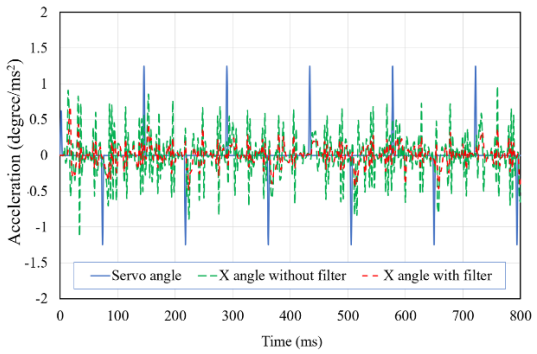
(b) The angle values of the motor and the sensor

Figure 6

The test to measure the difference between the cases of without and with Kalman filter



(a) Angular velocity

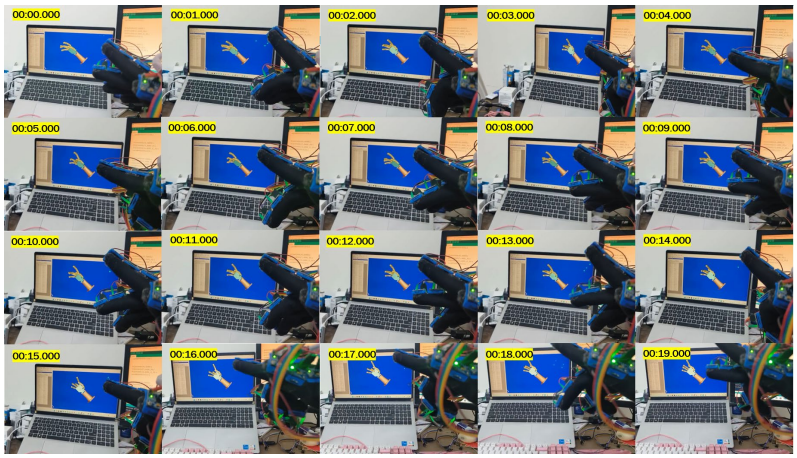


(b) Angular acceleration

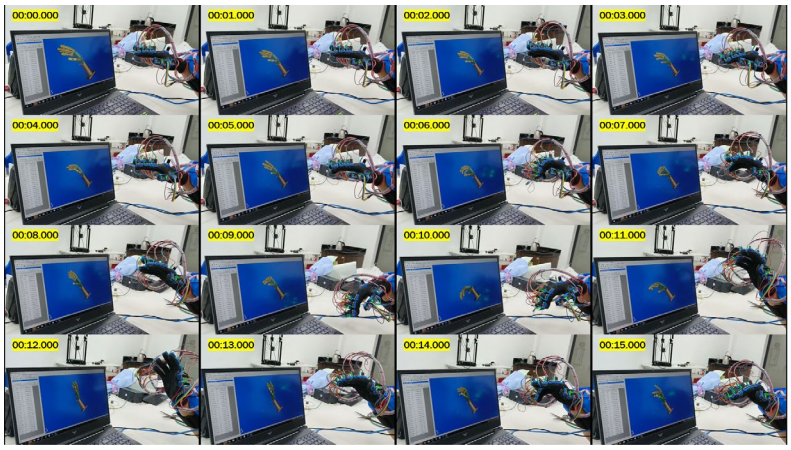
Figure 7

Results of angular velocity and acceleration

From the above test, the delay time was also evaluated for the delay time to calculate and transmit the data to the computer for one sensor to be about 4.3 ms and 5.8 ms for the cases of without and with applying Kalman filter, respectively. Therefore, the case with the filter made the delay about 1.5 ms in comparison to the case without the filter. When the sensory glove used with the simulation software, the time delay should be much larger. Figure 8 shows the frames cropped from the two videos recording the using the sensory glove to control the 3D robot hand model on the simulation software. The 3D model followed well the gestures of the human hand, however, there was a time delay measured about 350 ms. This time delay could come from both the transmission and calculation of data as well as the performance of drawing the 3D model on the computer. The lager delay is a limitation that needs to be overcome for the purpose of performing real-time tasks.



(a) Video 1



(b) Video 2

Figure 8

Frames of videos showing the tests

Conclusions

This paper presented our study where a design of system with a sensory glove and software to simulate fully movements of humanoid robot hand was successfully developed. The 3D model of the robot hand was designed according to the structure of the human hand, however, with 22 DOF of rotation. The kinematic problem was then established as foundation of the simulation software built with Visual Studio C++. The movement data of phalanges and carpal of the 3D model hand in the software controlled by three different modes which are Manual, Trajectory and Glove Modes for different missions. In the Glove Mode, the sensory glove was connected to the simulation software on the computer through the serial port to collect the movement data of the human hand. The time delay in using the MPU6050 sensor without and with applying the Kalman filter was estimated. The 3D robot hand model followed well the gesture of the human hand but there was large time delay. Therefore, further development is required for many applications with real-time tasks.

References

[1] H. Kawasaki, T. Mouri: Humanoid Robot Hand and its Applied Research. Journal of Robotics and Mechatronics, Vol. 31, No. 1, 2019

[2] M. S. Han, C. K Harnett: Journey from human hands to rbot hands: biological inspiration of anthropomorphic robotic manipulators. Bioinspir. Biomim. 19, 2024, 021001

[3] S. Pohtongkam, J. Srinonchat: Object Recognition for Humanoid Robots Using Full Hand Tactile Sensor. IEEE Access, Vol. 11, 2023, pp. 20284-20297

[4] J. Zhao, E. H. Adelson: GelSight Svelte Hand: A Three-finger, Two-DoF, Tactile-rich, Low-cost Robot Hand for Dexterous Manipulation. arXiv:2309.10886 [cs.RO], 2023

[5] Shuang Li a, Qiang Li b, Jianwei Zhang: Chapter 12 - A hand-arm teleoperation system for robotic dexterous manipulation. Tactile Sensing, Skill Learning, and Robotic Dexterous Manipulation, Academic Press, 2022, pp. 227-245

[6] A. Hussein, C. Tarry, C. Lambert, S. Barreca, O. B. Allen: Results of Clinicians Using a Therapeutic Robotic System in an Inpatient Stroke. Journal of Neuro Engineering (BioMed Central), 8, No. 50, 2011

[7] Y. Fu, Q. Zhang, F. Zhang, Z. Gan: Design and development of a hand rehabilitation robot for patient-cooperative therapy following stroke. 2011 IEEE International Conference on Mechatronics and Automation, 2011, pp. 112-117

[8] P. Polygerinos, Z. Wang, K. C. Galloway, R. J. Wood, C. J. Walsh: Soft robotic glove for combined assistance and at-home rehabilitation. Robotics and Autonomous Systems, Vol. 73, 2015, pp. 135-143

[9] A. Nair, P. L. Nair, G. Mohan, S. J. Nair, J. Baby: E-gloves: Wearable assistance in sports training. International Journal of Physiology, Nutrition and Physical Education, SP2: 01-06, 2019

[10] A. L. Angulo: Design and Development of a Glove with Remote controller function for video games. Final Degree Work Bachelor's Degree in Video Game Design and Development Universitat Jaume I, 2020

[11] A. A. Deshpande, G. V. Keerthi, N. Ramya, N. S. Suchitra: Gesture Controlled Robotic Arm for Testing and Diagnosis of Novel CORONA-VIRUS. International Journal of Engi-neering Research in Electronics and Communication Engineering (IJERECE), Vol. 8 (5), 2021, pp. 44-50

[12] Azhari, T. I. Nasution, P. F. A. Azis: MPU-6050 Wheeled Robot Controlled Hand Gesture Using L298N Driver Based on Arduino. Journal of Physics: Conference Series, Vol. 2421, 2023, 012022

[13] H. Wang, Z. Feng, J. Tian, X. Fan: MFA: A Smart Glove with Multimodal Intent Sensing Capability. Computational Intelligence and Neuroscience, Vol. 2022, Article ID 3545850

[14] Z. R. Saeed, Z. B. Zainol, B. B. Zaidan, A.H. Alamoodi: A Systematic Review on Systems-Based Sensory Gloves for Sign Language Pattern Recognition: An Update From 2017 to 2022, IEEE Access, Vol. 10, 2022, pp. 123358-123377

[15] V. T. Minh, N. Katushin, J. Pumwa: Motion tracking glove for augmented reality and virtual reality, J. Behav. Robot., Vol. 10, 2019, pp. 160-166

[16] M. Kumar, S. Shelji, P. Varghese, P. S. Shilpa, D. Thomas, A. Paul: Gyro Glove: Stabilizing the Lives of Those with the Hand Tremor. International Journal of Engineering Science and Computing, Vol. 10, No. 7, 2020, pp. 26839-26843

[17] D. D. Khoa, N. N. Vinh, N. M. Hieu: Development of Data Gloves for Humanoid Robot Hand Simulation and Hand Posture Recognition. Advances in Asian Mechanism and Ma-chine Science, ASIAN MMS 2021, Mechanisms and Machine Science, Vol. 113, 2021

[18] T. Bright, S. Adali, G. Bright: Low-Cost Sensory Glove for Human–Robot Collaboration in Advanced Manufacturing Systems. Robotics, 11, 56, 2022

[19] P. K. Surbhi, M. Kevin: Skeletal Anatomy of the Hand. Hand clinics, 29(4), 2013, pp. 459-471

[20] T. Kang, H. Kaminaga, Y. Nakamura: A Robot Hand Driven By Hydraulic Cluster Actuators. 2014 IEEE-RAS International Conference on Humanoid Robots, Madrid, Spain, 2014, pp. 39-44

[21] M. K. Prabhu, P. Sivaraman, M. Vishnukaarthi, V. Shankara Narayanan, S. Raveen, G. Keerthana: Humanoid gesture control arm with manifold actuation using additive manufacturing. Materials Today: Proceedings, Vol. 37, Part 2, 2021, pp. 717-722

[22] H. Ikeda, T. Saeki: Transformation of foldable robotic hand to scissor-like shape for pinching based on human hand movement. Scientific Reports, 13, 2023, 19150

[23] M. Faisal, F. Laamarti, A. El Saddik: Digital Twin Haptic Robotic Arms: Towards Handshakes in the Metaverse. Electronics, 12(12), 2023, 2603

[24] J. Denavit, R. S. Hartenberg: A kinematic notation for lower-pair mechanisms based on matrices. Journal of Applied Mechanics, Vol. 22 (2), 1955, pp. 215-221

[25] A. Ma'arif, Iswanto, A. A. Nuryono, R. I. Alfian: Kalman Filter for Noise Reducer on Sensor Readings. Signal and Image Processing Letters, Vol. 1, No. 2, 2019, pp. 10-21

[26] B. Alsadik: Chapter 10 - Kalman Filter. In Computational Geophysics, Adjustment Models in 3D Geomatics and Computational Geophysics, Elsevier, Vol. 4, 2019, pp. 299-326

[27] F. Winjaya, D. A. Feryando, N. A. Azizah: Accuration of Measurement Snake Motion Based on Kalman Filter Sensor Fusion. Proceedings of the 2nd International Conference on Railway and Transportation 2023 (ICORT 2023), Advances in Engineering Research 231, 2023, pp. 300-312

[28] Simple Kalman Filter Library. Available online: https://github.com/denyssene/SimpleKalmanFilter (accessed on 15 March 2024)