Modeling and Control of Poppy Humanoid Robot Arm used for Teeth-Brushing

Ahmed Mady, Amro Abdellatif, Marwan Sallam, Mohamad Magdy, and Mostafa Kashaf
Multi-Robot Systems Research Group(MRS)
Engineering and Material Science
German University in Cairo (GUC), Egypt
Emails: ahmed.alimady@student.guc.edu.eg, amro.abdellatif@student.guc.edu.eg, marwan.sallam@student.guc.edu.eg, mohamed.elsaidnada@student.guc.edu.eg, mostafa.kashaf@student.guc.edu.eg

***Abstract*—In this paper, an open-source Poppy humanoid robotic hand was implemented to help in service robotics. The application was to brush the teeth by implementing a sequence for the hand to move on. The robotic hand is implemented both in software and hardware. The software consists of SIMSCAPE multibody in SIMULINK and MATLAB, and the hardware is designed and implemented using SOLIDWORKS and 3D-printed using PLA-plus material and then actuated. Tests were performed on the robot including both open-loop testing and trajectory tracking to plan the motion of the hand or end-effector. The trajectory included 2 motions, forward and backward to brush teeth.**

***Keywords*-Robotics, Service Robotics**

I. Introduction

The area of service robots has steadily gained interest over the years as an attempt for deploying robots to tackle problems faced in everyday life. Robots are well-implemented in the industry for their speed, precision, stamina, and strength which allows them to perform many tasks better and faster than humans. They also allow humans to avoid many tasks that would be too dangerous or repetitive. In many applications, humans would benefit from direct interaction with robots. For instance, we could interact with the robot by taking advantage of our capabilities (capacity of decision) along with the robot's capabilities (strength, endurance, stamina) thus creating a synergy allowing us to accomplish tasks that were not possible for one or the other alone.

II. Literature Review

Service Robotics can be used in various fields including educational, entertainment, household, social, gaming, and security. Hentout [1] presented Kiva robots which are used in equipping Amazon online site warehouses to acquire and arrange pods autonomously so that pickers may complete orders more effectively. Zachiotis [2] presented iPal, which can serve as a teacher's assistant in such areas as taking a role and enhancing the education process. Under the supervision of a teacher, iPal can aid in lessons by presenting educational content in an engaging manner that supports social development and encourages interest in science and technology. Chunli [3] stated that most children with disabilities suffer from difficulties in social interaction and communication.

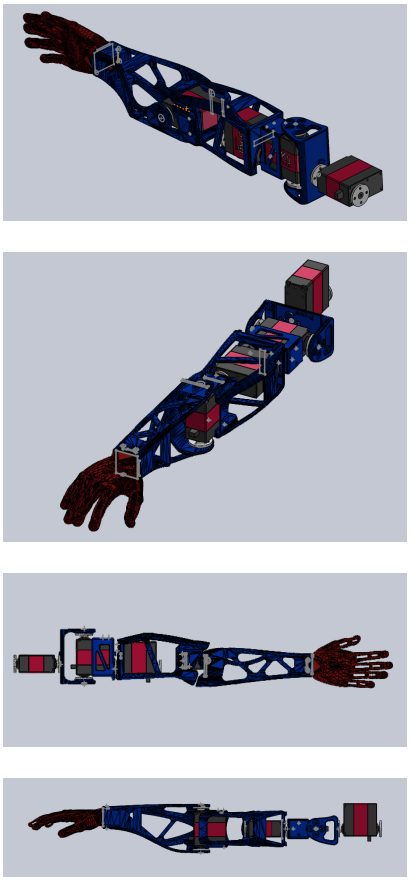
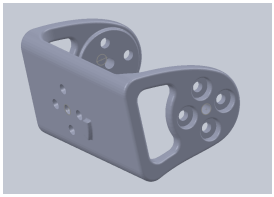
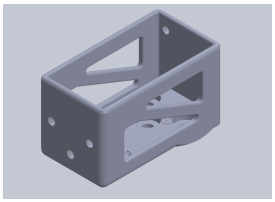


Figure 1: Poppy Robot Right Hand showing the isometric, side, and top views

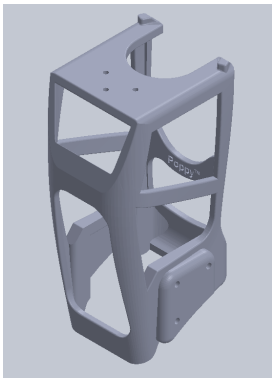
To ensure them a better life and promote their imitation skills, many studies used SPRITE. SPRITE is developed on a 6-DOF rotary Stewart platform mounted on a bracket holding a mobile phone to display the robot's face. Yaacoub [4] presented the WowWee Rovio which monitors homes with field recognition, motion detection, and person video capture. It was developed to monitor intrusion, detect smoke, monitor children, turn on/off home appliances, etc. Maibaum [5] combined a mobile robot with a smart environment to



(a) Shoulder

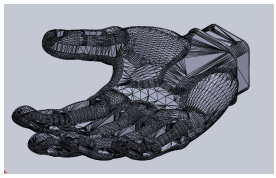


(b) Arm Connector



(c) Upper Arm

(d) Forearm



(e) Hand

Figure 2: Poppy Robot Right Hand parts 3D printed

assist the elderly with cognitive impairments living alone at home. The CompanionAble Robot provides various services such as appointment reminders, and frequent recommendations for specific activities listed by caregivers.

To assist the elderly, the poppy hand will be implemented as an end-effector used with a toothbrush to help in brushing teeth. The motion will be a trajectory as a straight line translating from left to right. Section I presents an introduction and section II literature review of service robotics. In section III, the hardware is implemented along with the DH convention. In section IV, the forward and inverse position kinematics are obtained. In section V, the software and hardware open-loop are tested by having a trajectory tracking algorithm. Section VI provides the conclusion and future recommendations.

III. Hardware Design and Implementation

A. Hardware Components

The hardware components used are presented in Table I. The arm consists of 5 3D-printed parts by PLA+, the shoulder, an arm connector, the upper arm, the forearm, and the hand.

Table I: Hardware Components Table

Part	Purchased loc.	Quantity	Price
Arduino UNO	Future Elect.	1	360 LE
Servo Motor 10kgcm dual-axis	Future Elect	4	475 LE
Power Source 5V, 10A	RAM Elect.	1	100 LE
3D printed parts	Local	5	200 LE

B. Simscape Simulation

The arm consists of 4 DOF or 4 motors. The first motor which consists of the fixed motor rotates around the z-axis and has a range of motion of $0°$ to $180°$. The second motor which connects the shoulder to the rest of the arm rotates around the x-axis and has a range of motion of $-90°$ to $90°$. The third motor which connects the arm connector to the upper arm rotates around the z-axis and has a range of motion of $-135°$ to $-45°$. The last motor which connects to the forearm rotates around the y-axis and has a range of $-150°$ to $0°$.

C. Circuit Diagram

Figure 3 includes the circuit diagram of the electrical components designed using Tinkercad software.

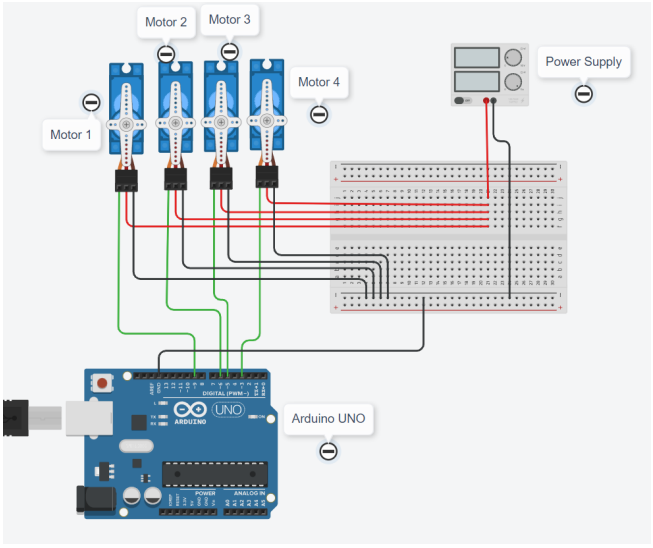


Figure 3: Assembly Side View

D. Hardware Assembly

In Figures 5 and 6, the full hardware is assembled which includes all motors, the 3D printed parts, the hardware fixation, and the enclosed box for circuitry. The enclosed box includes the hardware connections that include the power source, the Arduino controller, and a breadboard used for the ground and input voltage. The wires are color codes with yellow as the PWM signal from the Arduino, white as the input voltage, and purple as the ground. Below the Arduino is an insulator for protection from the power source. The

circuit contained inside the enclosed box is shown in Figure 4.

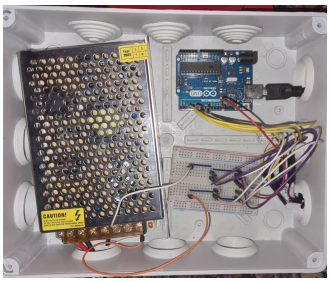


Figure 4: Hardware circuit contained inside the enclosed box.

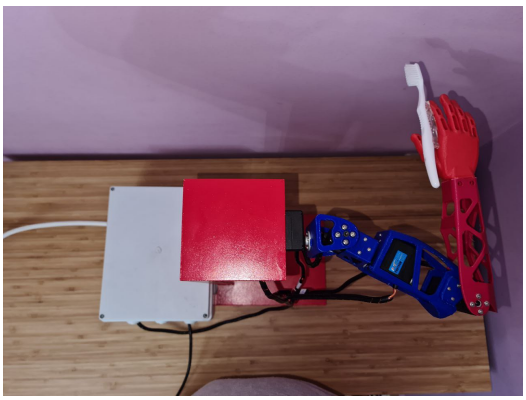


Figure 5: Assembly

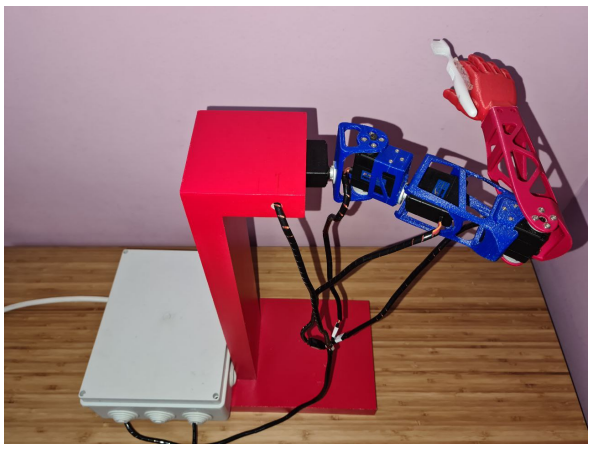


Figure 6: Assembly of the robotic hand

IV. Robot Kinematics

A. Robot's Frame Assignment

The frames applied on the robotic arm are shown in Figure 7.

Table II: DH-Parameters Table

Joints	θ	d	a	α
1	q_1	l_1	0	90^o
2	q_2	0	0	-90^o
3	$q_3 + 90^o$	$l_2 + l_3$	0	-90^o
4	$q_4 - 90^o$	0	l_4	0

Table III: Lengths of the links

l_1	2.7cm
l_2	3.5cm
l_3	10.57cm
l_4	15.07cm

B. DH Convention

The DH convention values are presented in Table II.

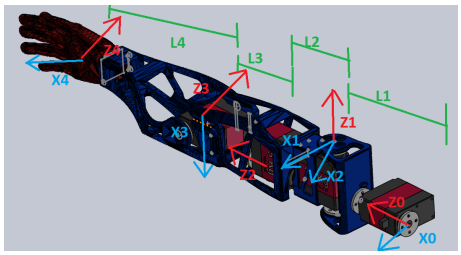


Figure 7: Frames Assignment

The total homogeneous matrix is shown in Figure 8 and the lengths of the links are presented in Table III

$$\begin{bmatrix} -c(q1)c(q4)s(q2) & -c(q4)s(q1)c(q3) & -c(q1)c(q2)c(q3) & -14c(q1)s(q2) \\ s(q1)c(q3)s(q4) & c(q1)c(q2)s(q3)c(q4) & s(q1)s(q3) & -14.8c(q1)c(q4)s(q2) \\ -c(q1)c(q2)s(q3)s(q4) & c(q1)s(q2)s(q4) & c(q1)s(q2)s(q4) & -14.8c(q3)s(q1)s(q4) \\ & & & -14.8s(q4)c(q1)c(q2)s(q3) \\ -c(q4)s(q1)s(q2) & c(q4)c(q1)c(q3) & -c(q2)c(q3)s(q1) & -14s(q1)s(q2) \\ c(q1)c(q3)s(q4) & -c(q4)c(q2)s(q1)s(q3) & c(q1)s(q3) & -14.8c(q4)s(q1)s(q2) \\ -c(q2)s(q1)s(q3)s(q4) & s(q1)s(q2)s(q4) & -s(q1)s(q2)s(q4) & 14.8c(q1)c(q3)s(q4) \\ & & & -14.8s(q4)c(q2)s(q1)s(q3) \\ c(q2)c(q4) & -c(q4)s(q2)s(q3) & -c(q3)s(q2) & 2.7 + 14c(q2) \\ s(q2)s(q3)s(q4) & -c(q2)s(q4) & c(q2)s(q4) & 14.8c(q2)c(q4) \\ & & & -14.8s(q2)s(q3)s(q4) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 8: Total Homogeneous Transformation Matrix

V. Simulation Results

A. Forward and Inverse Position Kinematics

This section tests the forward and inverse position kinematics in SIMSCAPE multibody and SIMULINK Environment. Figure 9 shows the robotic arm in SIMSCAPE multibody comparing the actual position sensed(transform sensor) of the end-effector, the hand, with the DH-parameters values, and Figure 10 shows the 3D simulation orientation from the Mechanics Explorer in SIMSCAPE. The inputs are given from the first joint respectively as $30^o, 45^o, 70^o, 70^o$. As shown the values measured are of a very small tolerance in cm. Figure 11 shows inputs given from the first joint

respectively as $0^o, 30^o, 45^o, 90^o$ and Figure 12 shows the 3D simulation orientation from the Mechanics Explorer in SIMSCAPE.

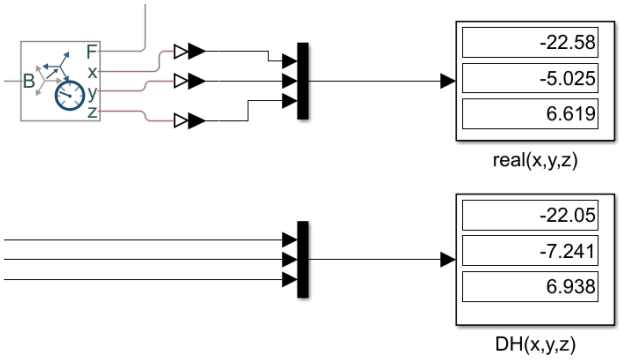


Figure 9: Forward Kinematics Simulation 1

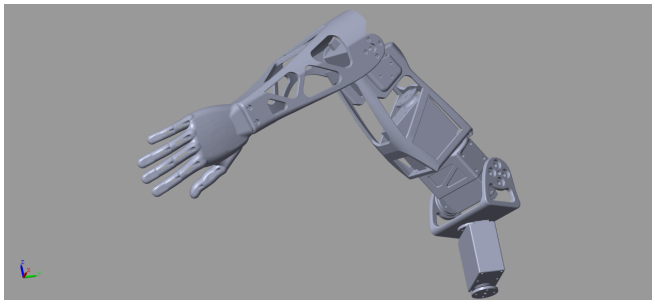


Figure 10: The arm configuration inputting joint angles as $30^o, 45^o, 70^o, 70^o$ from the first joint respectively

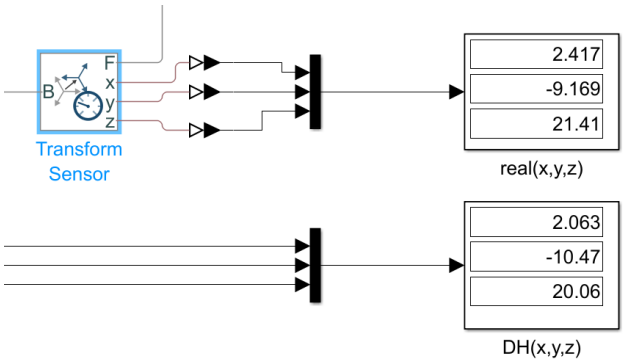


Figure 11: Forward Kinematics Simulation 2

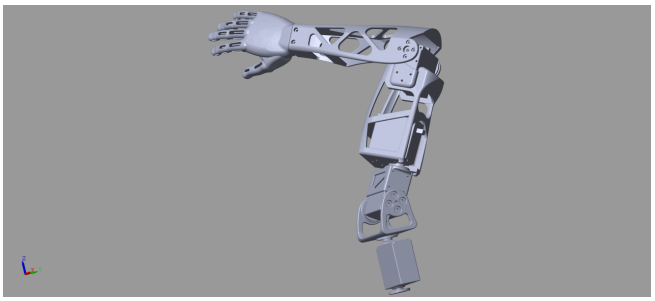


Figure 12: The arm configuration inputting joint angles as $0^o, 30^o, 45^o, 90^o$ from the first joint respectively

The Inverse position kinematics is calculated using the numerical approach Newton-Raphson. The method is a root-finding algorithm that produces successively better approximations to the roots (or zeroes) of a real-valued function until a certain tolerance is reached. Initially, a guess for the joint angles are input-ed along with the target x,y, and z positions of the task space or end effector. Next, the forward kinematics is evaluated for the initial joint space angles. The forward kinematics is then subtracted from the x,y, and z positions to measure the error, and this value is multiplied by the inverse jacobian matrix or the differentiation of the x,y, and z position equations updating the joint angles to a new value subtracted from the previous values. This process is repeated until the error in x,y, and z tends to 0.01. The inverse kinematics and problem may output different joint angles for the same target position. The position is validated by inserting the joint angles into the forward kinematics to be validated. Figure 13 shows the simulation result as the x,y,z from the inverse kinematics and forward kinematics positions equal to the input x,y,z positions.

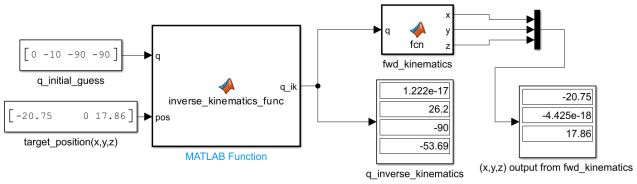


Figure 13: The inverse kinematics simulation in SIMULINK software having the initial guess of the angles, the target position, and the output of the function to the forward kinematics to validate the angles

B. Trajectory

Two Task-Space trajectories, both of straight lines, are implemented for the robot to traverse, one forward and another backward changing only the x position while keeping the y and z positions constant. The forward trajectory equation is presented in Equation 1 and the backward trajectory is presented in Equation 2. The forward trajectory starts from $x = -20.75cm$ to $-7.774cm$ while the backward trajectory

starts from $x = -7.774$ to $x = -20.75$ with having y and z positions as constants, $y = 0cm$ and $z = 17.86cm$. The traversed motion is presented in Figure 14. This is done by activation of only 2 motors to be variable, 2 and 4, and 2 actuated but constant, 1 and 3. $q_1 = 0°$ and $q_3 = -90°$. Figures show the change of angle in both q_2 and q_3. In the simulation, the trajectory is placed in a MATLAB function and outputs the position at every time step, sample time is discrete, and input. The output position is then placed in the inverse position kinematics to find the joint angles to actuate the robot. The range of angles for q_2 is 30^o to -25^o, and for q_4 is 112^o to 58^o. Ts is the sampling time and the final time for the forward trajectory is 5 seconds, and the backward motion is 5 to 10 seconds.

$$x = -7.774 - 2.5952 * Ts \tag{1}$$

$$x = -33.726 + 2.5952 * Ts \tag{2}$$

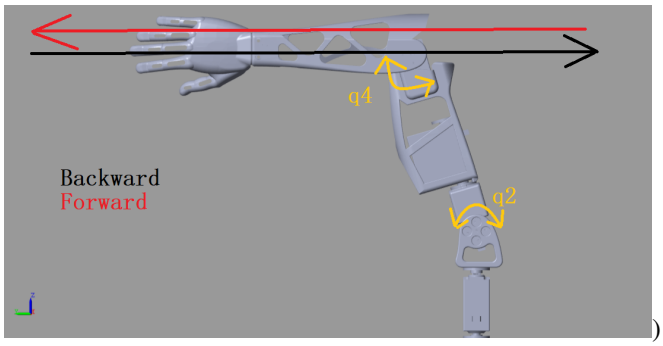


Figure 14: The traversed path by the robot

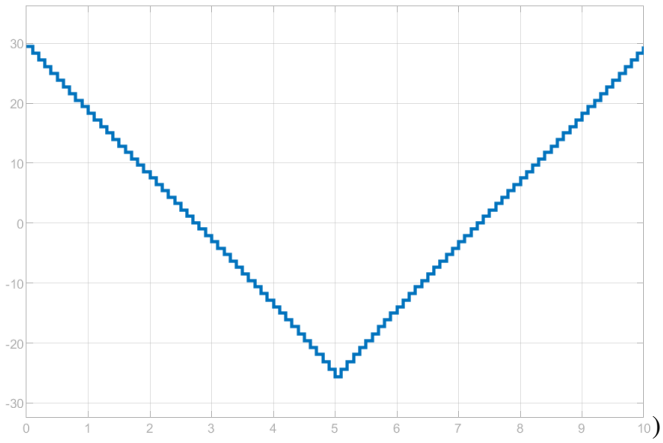


Figure 16: q_2 change

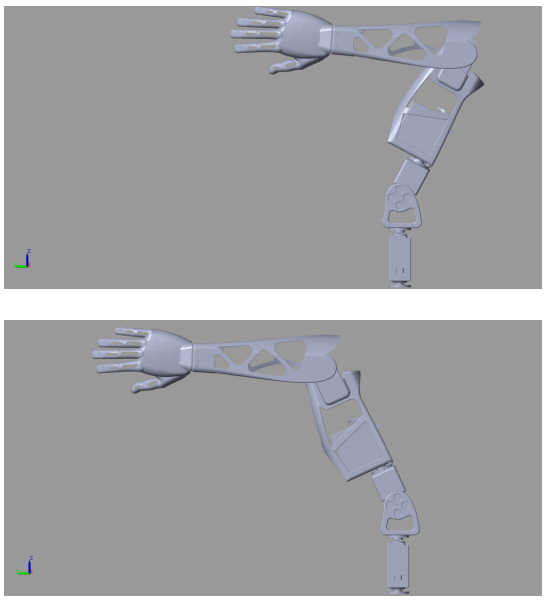


Figure 15: The maximum and minimum points traversed from the left view in the mechanics explorer on SIMSCAPE

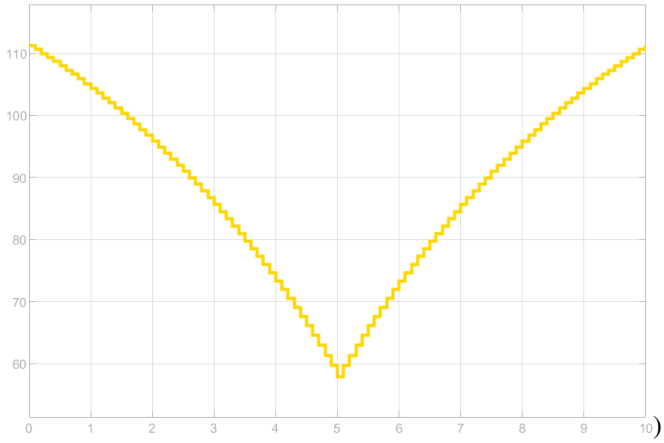


Figure 17: q_4 change

VI. Conclusions and Future Recommendations

An open-source Poppy humanoid robotic hand was created in this work to aid in service robotics. The application aimed to brush the teeth by developing a movement sequence for the hand. Both software and hardware are used to build the robotic hand. SIMSCAPE multibody in SIMULINK and MATLAB is used for the software, while the hardware is planned and implemented in SOLIDWORKS, 3D-printed in PLA-plus material, and then actuated. To design the movements of the hand or end-effector, tests were done on the robot. First, the forward position kinematics is obtained by using the DH convention, followed by the inverse position kinematics. Both were tested and compared to the actual values obtained from the

robot dimensions. Then test included both open-loop testing and trajectory tracking using the equation for a straight line for 2 movements, the first forward and the other backward.

Future recommendations include using a servo motor of stronger torque and using Feedback control on the robotic manipulator to represent trajectories more accurately.

REFERENCES

[1] A. Hentout, A. Maoudj, and A. Kouider, "A literature review of service robotics."

[2] G. A. Zachiotis, G. Andrikopoulos, R. Gornez, K. Nakamura, and G. Nikolakopoulos, "A survey on the application trends of home service robotics," in *2018 IEEE international conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2018, pp. 1999–2006.

[3] C. Liu, J. Li, S. BalaMurugan, and M. BalaAnand, "Cognitive computing for intelligent robots in assisting preschool children," *Intelligent Service Robotics*, pp. 1–11, 2020.

[4] J.-P. A. Yaacoub, H. N. Noura, O. Salman, and A. Chehab, "Robotics cyber security: Vulnerabilities, attacks, countermeasures, and recommendations," *International Journal of Information Security*, pp. 1–44, 2021.

[5] A. Maibaum, A. Bischof, J. Hergesell, and B. Lipp, "A critique of robotics in health care," *AI & SOCIETY*, vol. 37, no. 2, pp. 467–477, 2022.