

## Research Article

# Implementation and Comparison of Wearable Exoskeleton Arm Design with Fuzzy Logic and Machine Learning Control

Çağatay Ersin <sup>1</sup> and Mustafa Yaz <sup>2</sup>

<sup>1</sup>Electronics and Automation Department, Vocational School, Çankırı Karatekin University, Çankırı, Türkiye

<sup>2</sup>Electrical and Electronics Engineering Department, Bozok University, Yozgat, Türkiye

Correspondence should be addressed to Çağatay Ersin; [cagatayersin@karatekin.edu.tr](mailto:cagatayersin@karatekin.edu.tr)

Received 24 February 2023; Revised 28 December 2023; Accepted 27 February 2024; Published 3 April 2024

Academic Editor: Rajkishor Kumar

Copyright © 2024 Çağatay Ersin and Mustafa Yaz. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this study, a wearable exoskeleton arm was designed and controlled with different control methods to help people with muscle disorders in their arms and support treatment. The developed robot arm was transferred to Simulink software with the Simmechanics application. Two electromyography (EMG) muscle sensors and the ADXL335 position and acceleration sensors attach to the human arm's biceps and triceps muscle areas. As the human moved the arm, data were obtained from the EMG muscle sensors and the ADXL335 position and acceleration sensor. The received data were first trained with the fuzzy logic algorithm. The same data were then trained with machine learning algorithms in Simulink software. It has been determined that the best result is the quadratic support vector machine (SVM) algorithm. The fuzzy logic algorithm trained with the PID controller block and the received sensor data have been added to the degrees of freedom regions that will enable rotation in the block diagram of the previously exported system. Later, the fuzzy logic block was removed and the machine learning algorithm, the quadratic SVM algorithm, was added. The designed system was operated with two different control systems, and the control algorithm closest to the human arm movement was determined. In addition, each part of the system, whose design was prepared, was removed and assembled separately with a 3D printer. ESP32 microcontroller development board was used to control the system, and it was run in real-time with EMG muscle sensors and position sensors.

## 1. Introduction

People working in high-risk jobs may have accidents and damage their limbs [1, 2]. Depending on the increasing average age, cerebrovascular and neuromuscular diseases are also increasing. The use of wearable biomedical sleeves has also increased in patients who have lost limb skills due to these diseases and accidents [2, 3]. Since the arms are used extensively in daily life, the quality of life of the people is adversely affected when any discomfort occurs in the arms due to the muscles. Clinically, upper extremity disability of neurology or orthopaedic origin is characterised by insufficient muscle strength, altered muscle group firing pattern, or inability to control the joint voluntarily. This problem can worsen pain, stiffness, or shoulder impingement syndrome. Clinical studies show that long-term repetitive rehabilitation can help these patients restore motor function and prevent complications

[4]. However, some individuals with muscle loss in their arm muscles do not recover with rehabilitation. In these cases, wearable robotic arms are needed to increase power [1, 3]. Today, with the widespread use of robotics, many studies have been carried out on exoskeleton robot designs and the limb designs of these robots [1]. The exoskeleton robot is a type of wearable robot that incorporates many disciplines, such as bionics and ergonomics, and can appropriately perform a combination of human and machine functions.

The purpose of biomedical devices' production and control phase is to help people regain their lost physical functions or improve their rehabilitation processes. Wearable exoskeleton robot arms interact between the human and the mechatronic system and should perform human movements as similarly as possible [5, 6]. However, if the joint angle of the wearable exoskeleton is greater than the joint range, this dangerous situation should be prevented by making necessary

restrictions [3]. Human upper limbs consist of six joints. These six joints are common in both arms. Motors are used to bring mobility to these joint areas of the wearable robot arm mechanism. The first movement of the wearable exoskeleton arm is the angular movement of the arm to the right, left, forward (flexion), and backward (extension) on the axis parallel to the ground. With this movement, the human arm can move angularly  $120^\circ$ . These are  $70^\circ$ – $75^\circ$  forward and  $40^\circ$ – $45^\circ$  back [7, 8]. The second movement is the rotation of the arm around its axis. This movement takes place up to  $90^\circ$ . The third movement is parallel to the ground, sideways opening (abduction) and approaching the arm body (adduction). The human arm can rotate  $120^\circ$  at the shoulder joint and be brought up to  $15^\circ$  to the body. Limb-joint models are preferred for biomechanical human body modelling [6, 9].

Wearable robot arm works should have a control mechanism. Many types of controls can be used to control a system. One of these control systems is the proportional integral derivative (PID) controller. PID control consists of proportional, integral, and differential [10]. The control body (controller)'s task in the closed-loop control system is to compare the output size (speed, rotational speed, position, etc.) fed back from the measurement with the reference input size and calculate the difference. The system is controlled by generating the control signal and giving it to the output again. A PID controller compares the input (reference) signal with the feedback from the output, and an error occurs due to the difference [11]. PID control has a simple structure for three gains. PID algorithms control most robot manipulators independently at each joint [12]. PID controllers are frequently used in speed and position control of robot arm mechanisms [13].

Another method used to control robot arm mechanisms is the fuzzy logic controller. Fuzzy input, represented by an error in the fuzzy logic controller, regulates the performance of a system. The fuzzy controller uses variable input–output information to control the joint speed and position of the robot mechanisms [8, 14]. Using the advantageous features of the fuzzy controller scheme, it is also possible to control systems with a wider range of reference and interference signals [15]. A fundamental problem in robot control is following the desired trajectory with sufficient precision. Controllers are tuned by changing variables in classic robot manipulator control. However, in real applications, control systems are exposed to external effects such as noise and distortion. The fuzzy control algorithm allows the controlled variables to be controlled over a wider range to eliminate the undesirable effects of the inputs [16].

Machine learning, also used as another robot control method, is an algorithm that teaches machines how to use data more efficiently. Learning with machine learning takes place thanks to the data received from the external environment. The more sensor data received, the better the machine learns and applies. The machine learning algorithm is a preferred algorithm for making sense of data. With the abundance of available datasets, the demand for machine learning is also increasing. Many robotic systems apply machine learning algorithms to make sense of data [17]. In a machine learning algorithm, a computer programme performs some task. It

is stated that the machine learns from experience whether its measurable performance in these tasks improves as it gains more experience in performing these tasks. Thus, the machine learning algorithm makes decisions and forecasts based on data [18].

It has been seen in the reviewed literature that different control methods were used to control the exoskeleton robot arm. In the literature studies, it is seen that ready-made robot arms are used and controlled only with EMG signals or only with position and acceleration sensors [19–27]. In some recent studies in the literature, it is seen that only PID controllers [28–31] control robot arms. Again, in some recent studies in the literature, it is seen that the data obtained from EMG muscle signals only work with fuzzy controllers or machine learning algorithms in this direction [32–37]. Many studies have compared EMG signal data with different control algorithms. When the studies conducted with fuzzy controllers are scanned, fuzzy logic is used for rehabilitation. It is used to control the robot's movement to rehabilitate patients. The system can effectively adapt to the situation using the fuzzy logic approach [38–40]. There are studies in which machine learning algorithms are compared, and the algorithm that gives the best result is determined. In the literature, there is no study in which the wearable robot arm, which has an original design, is controlled and compared with a fuzzy logic controller and machine learning algorithms as in this study.

The study is quite original regarding the design and comparison of current control methods. Contrary to the studies in the literature, a suitable upper extremity robot arm was designed, and necessary kinematic analyses and angle analyses were performed in the previous study [41]. The human anatomy carried out the designed robot arm's angle limitation and kinematic analysis. The comparison of control methods not included in the literature compared with the SVM algorithm, which gives more stable results than both fuzzy logic algorithms and machine learning algorithms. Unlike the literature, the study includes control with PID controller, fuzzy logic control (EMG muscle sensor, position, and acceleration sensor) trained with actual sensor data, and control with quadratic support vector machine (SVM), a machine learning algorithm. The purpose of using a PID controller in this study is to bring the motor angles to the joint position fast and stable with the robot arm's real muscle and position data. In addition, this study contributes to the literature by controlling the upper extremity robot arm with actual data. It contains information about which widely used fuzzy logic algorithms and machine learning algorithms can be preferred and it leads to similar studies in the future. The study was also carried out for the benefit of humanity by providing original design, real muscle and position data, and appropriate control methods.

## 2. Materials and Methods

Wearable exoskeleton robot arms, including interdisciplinary studies, emerged as devices that interact between humans and machines [6]. This study designed the wearable exoskeleton arm according to the user, and the necessary kinematic analyses and angle limitations were made. The wearable

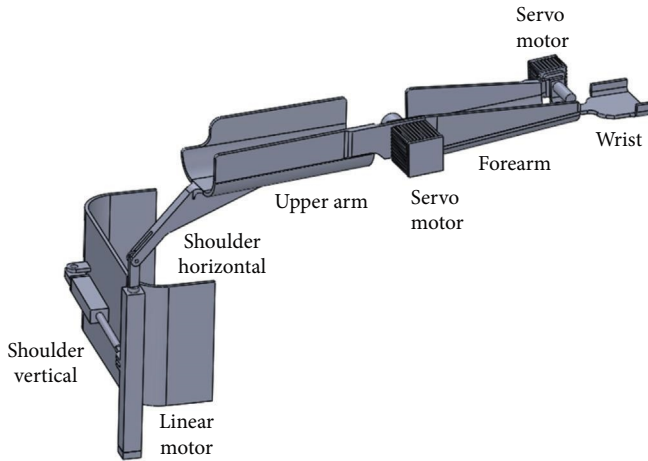


FIGURE 1: Wearable exoskeleton robot design [41].

exoskeleton arm was transferred to Simulink software with Simmechanics application, and control was realised with PID controller, machine learning, and fuzzy logic algorithms. For the exoskeleton robot, machine learning, and fuzzy logic algorithm, data were taken from the user's arm via sensors, and the system was run and compared with these two algorithms.

**2.1. Design of Wearable Exoskeleton Robot Arm.** Wearable exoskeleton robot arm structures can be designed in different types according to the work to be done [41]. While there are robot studies that can move in one direction, there are also robot arm studies with very different mobility [42, 43]. The robot arm, made in the previous study, is ergonomically designed so that the person with a loss of strength in his arm can perform the activities of daily living (the safety coefficients of the arm, hand, and shoulder apparatus are calculated for a maximum load of 250 N). Necessary kinematic analyses of the wearable robot arm were designed according to user dimensions and angle limitations suitable for human anatomy. Maximum displacements were determined by making force distributions for the system's arm, hand, and shoulder parts. SolidWorks 3D design software was used for the design and angle analysis of the wearable exoskeleton robot arm [41]. The designed wearable exoskeleton robot arm is given in Figure 1.

The angle analysis of the wearable exoskeleton robot arm structure was made by considering the parts that make up the system individually. The maximum range of motion of the exoskeleton robot arm and the joints of the human arm are limited by values. The motors placed in the three joint areas of the exoskeleton robot arm structure realise these movements. Motors have been added to the wrist, elbow, and shoulder parts. The motion angle values of the wrist, elbow, and shoulder joints of the realised exoskeleton robot arm design are given in Figure 2. The wrist angle is given an angle value similar to the human wrist motion angle in the design. It has been observed that the backward movement of the wrist of healthy individuals is between  $\pm 70^\circ$  and  $80^\circ$  without difficulty, and the wrist movement of the mechanism is limited to  $\pm 70^\circ$ . The vertical movement angle of the arm at the shoulder joint of the exoskeleton robot arm

is  $90^\circ$ . Horizontal angles of shoulder movement are limited to  $15^\circ$  in the negative direction and  $75^\circ$  in the positive direction. The maximum vertical movement angle on the front of the designed and analysed exoskeleton robot system was determined as  $110^\circ$  [41].

**2.2. Control of Exoskeleton Robot Arm with PID Controller.** Control is an important issue in automatic control systems. Many types of controls can be used to control a system. One of these control systems is the PID controller. The closed-loop task of the control system is to compare the output size (speed, rotational speed, position, etc.) fed back from the measurement with the reference input size and calculate the difference. In other words, the difference between the effect of the control system on the system and the error value of the system is found. In this way, the system is controlled by generating and representing a control signal to the output. *P* is proportional; *I* is integral, and *D* is derivative in PID control. Most control actions combine to form the *PI*, *PD*, and PID control structures. A PID controller compares the input (reference) signal with the feedback from the output, and an error occurs due to the difference. *P* is the current error, *I* is the sum of past errors, and *D* is an estimate of future errors. With the weighted sum of these three actions, the system is controlled and brought to the desired state [44]. In the PID control block diagram given in Figure 3,  $(r)$  is the set point,  $e(t)$  error,  $u(t)$  is the controller output, and  $y(t)$  is the process variable. There are two types of structures commonly used for PID controllers. These structures are serial and parallel-type structures. In proportional *P*, integral *I*, and derivative *D* parallel-type structures, the action takes place in separate equations, and their combined effect produces the total value. In the parallel type, each parameter is independent of the others, and the corresponding control structure is given in Equation (1).

$$u(t) = K_c \left( e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right). \quad (1)$$

In the equation, the proportional gain is  $K_p = K_c$ , the integral time is  $T_i$  and  $k_i = \frac{K_c}{T_i}$  the derivative time is  $T_d$  and  $k_d = K_c \times T_d$ . This is also known as an ideal representation.

The serial-type structure, on the other hand, is derived from the serial or interactive equation, mainly from the pneumatic and analogue electronic circuit properties. Like an ideal PID equation, change in  $K_c$  affects all three actions, but both derivative and integer constants affect proportional action. PID serial type equation is given in Equations (2) and (3).

$$e_1(t) = e(t) + T_d \frac{de(t)}{dt}, \quad (2)$$

$$u(t) = K_c \left( e_1(t) + \frac{1}{T_i} \int e(t) dt \right). \quad (3)$$

In this equation, the  $K_c$  gain affects all three parts of the PID structure. On the other hand, the values of the integral

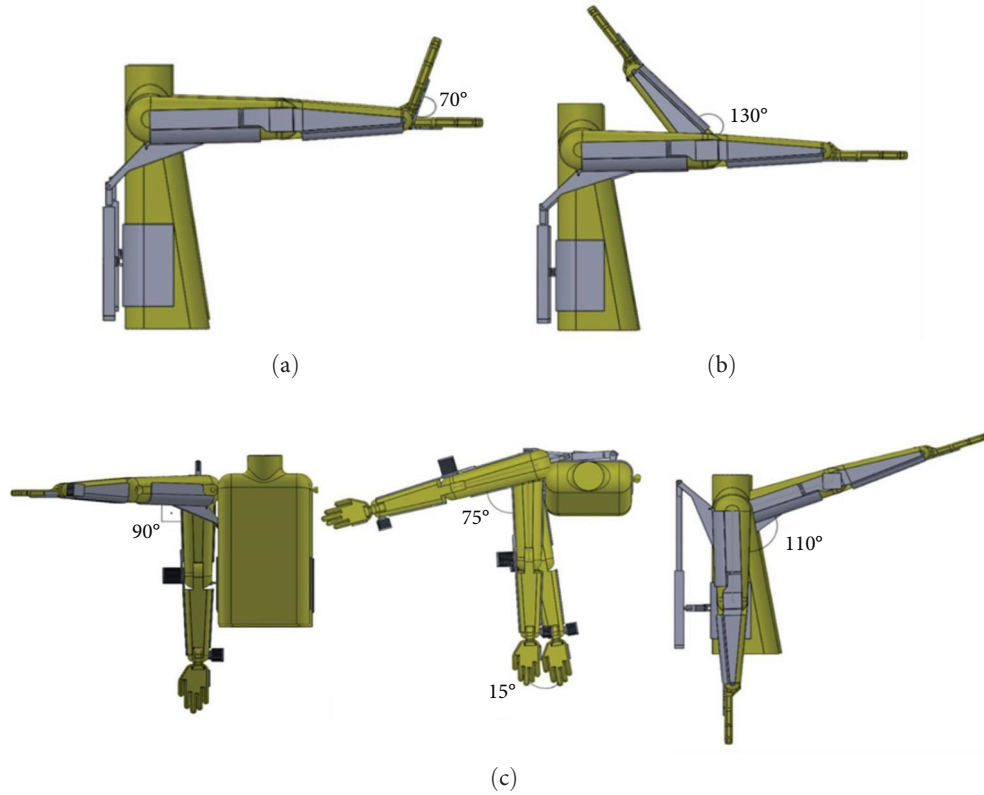


FIGURE 2: Robot arm mechanism positions and angle limitations: (a) wrist movement; (b) elbow movement; (c) shoulder movement [41].

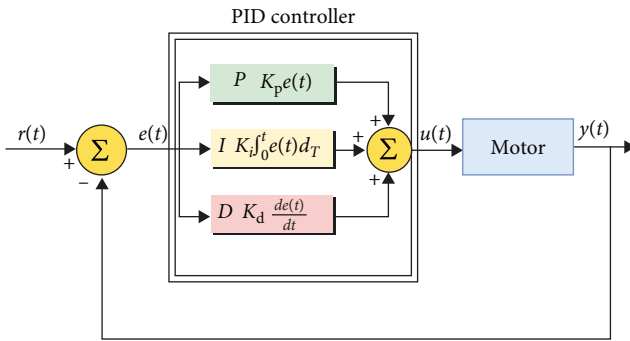


FIGURE 3: Block diagram of the system controlled by PID.

and derivative tuning parameters  $T_d$  and  $T_i$  also affect the proportional term. Thus, changing  $T_d$  tends towards effects and actions, both by changing the effect on actions and the effects of  $K_c$  on all three steps [45]. In this study, the block diagram of the system controlled by PID is shown in Figure 3.

The block diagram of the robot arm, designed and kinematically analysed in the Solidworks programme, was created with the simmechanics plugin [41]. The designed wearable robot arm system has four motion zones. These are wrist, elbow, and horizontal and vertical shoulder movements. PID controller blocks have been added to the rotary blocks of the design, the block diagram of which has been drawn to provide a more stable movement. PID blocks have constant

blocks to provide a continuous angle of motion to the system, PID and feedback blocks for stable system operation, common actuators to provide movement, and standard sensor blocks to receive feedback. PID and feedback blocks added for stable operation of the system are given in Figure 4.

With PID control, different  $P$ ,  $I$ , and  $D$  values are entered separately for four movements. For the PID control block, the system's response time and reference traces are set on the chart, and the values of each PID control unit are assigned according to this reference chart. Reference tracking and response time graphs of wrist, elbow, and shoulder movements are shown in Figure 5. The graph shows that the same amplitude values for shoulder vertical, shoulder horizontal, and elbow and wrist movements were reached in different time intervals. This is due to the angle limitation. Each joint has a different angle value.

**2.3. Acquiring Muscle Data.** Electromyography (EMG) is a method of measuring the electrical activity of muscle fibres during activation. EMG signals are analogue signals that are not periodic or deterministic. EMG involves measuring this action potential, which passes through the muscle fibre at a speed of 2–6 m/s [46]. EMG signals are not repeated at certain time intervals, and a single mathematical expression cannot represent EMG signals obtained during a recording period. EMG signals are non-stationary signals. The characteristics of the EMG signals differ according to the position of the muscle group being measured [47]. During the

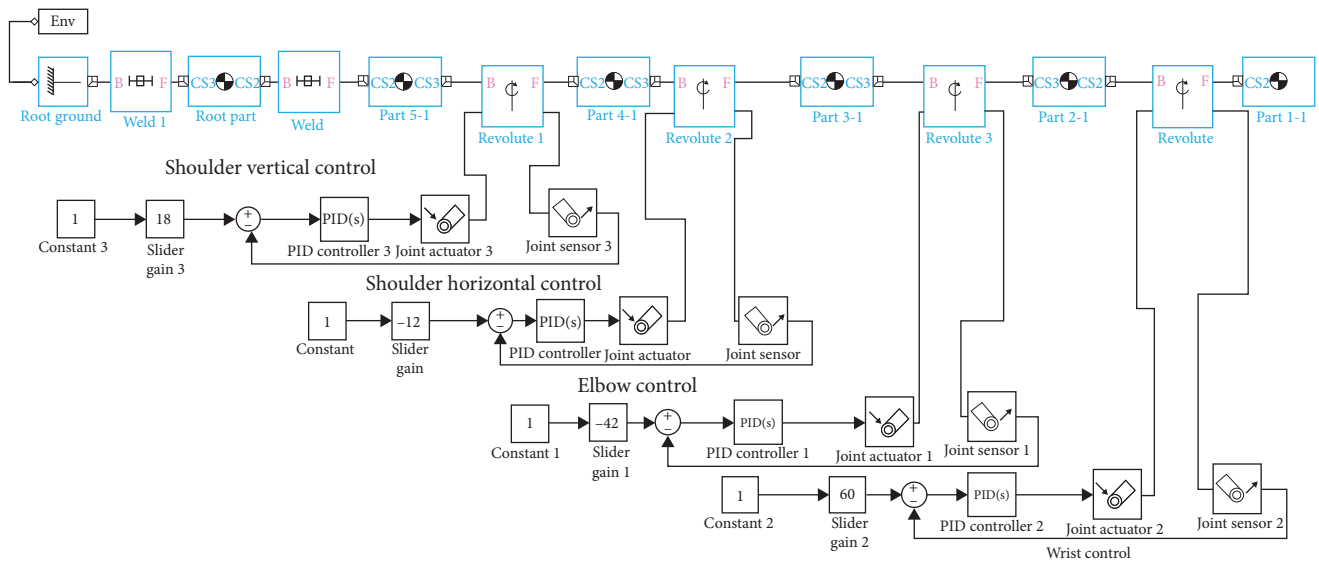


FIGURE 4: Block diagram of a system built with PID controller and added feedback.

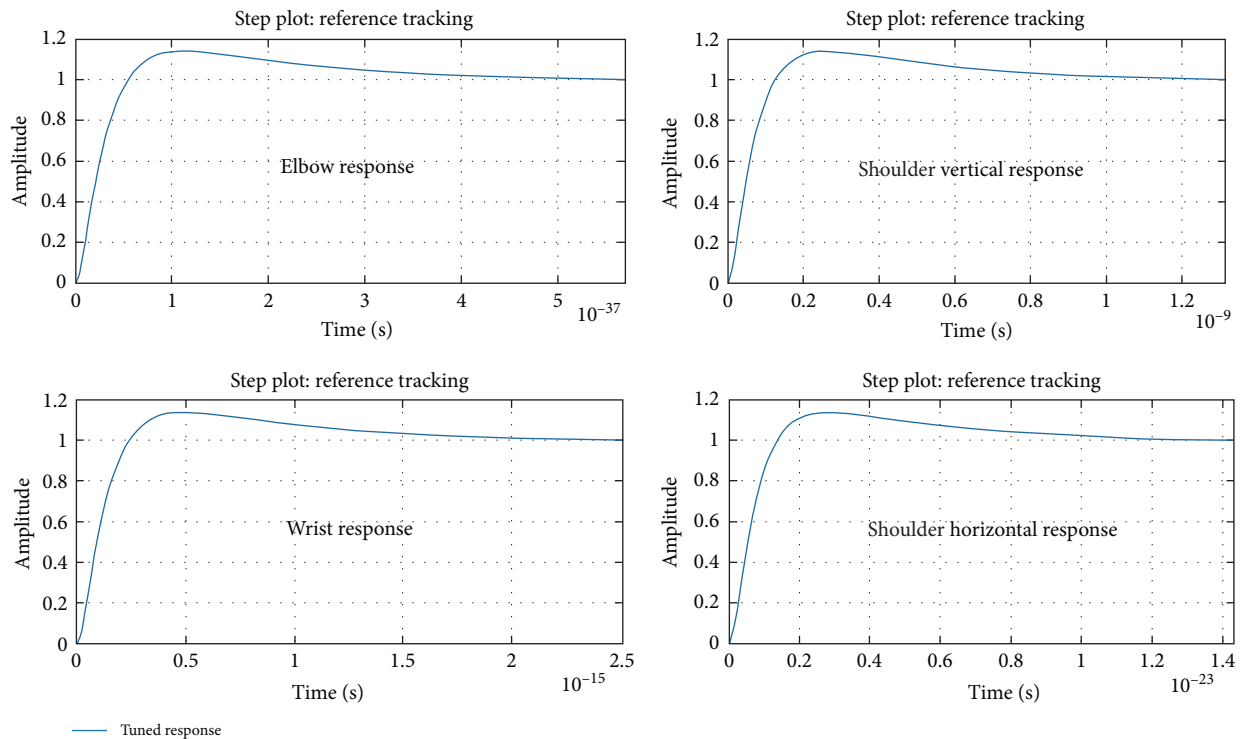


FIGURE 5: Reference tracking graph for wrist, elbow, movement shoulder, horizontal, and vertical movement.

recording of the EMG signal, ambient sound, magnetic effects, and vibration in the electronic recorder create noise sources that affect the characteristics of the raw EMG signal. To minimise this situation and improve the sensor’s performance, it may be necessary to use self-assembling monolayers, porous materials, and hydrogel or polymeric coatings. Probe materials support various sensor construction components and provide flexibility, stability, durability, lightness, and anatomical compatibility with the human body. The sensitivity and selectivity of the sensor are directly dependent on the surface of the

sensor structure [48, 49]. Studies also show that developing sensor probes can help obtain more accurate data [50]. In some recent studies, probe designs that affect the detection time of the sensor, the detection limit, and the sensor’s linearity have been carried out. This allows us to obtain data with a high accuracy rate [51, 52]. High-order statistics, spectral and wavelet analysis, etc., for interpreting signal-processing techniques for EMG signals. The process of interpretation of EMG signals is carried out in three stages. These stages are preprocessing, feature extraction, and classification [49]. This study

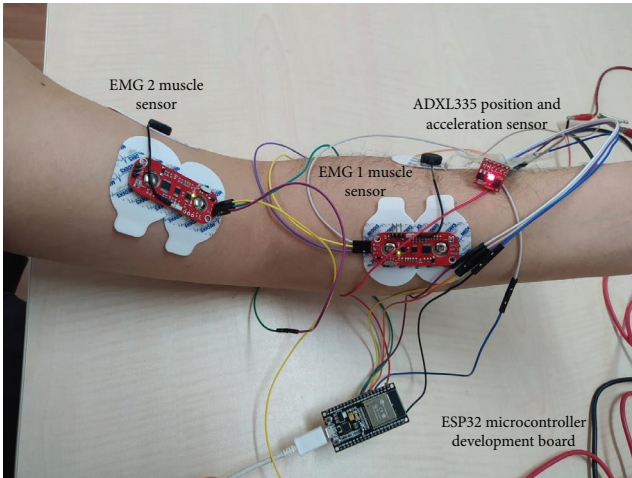


FIGURE 6: Data acquisition process image with sensors.

placed EMG muscle biomedical sensor pads on the upper arm and forearm. In obtaining data from the sensor, a more precise and uninterrupted data collection process was carried out by applying gel to the probes. During the data collection, the person moved his arm and recorded the data with ESP32. Fuzzy control and machine learning algorithms were used to classify the data.

The system simulation with the PID controller was first performed with fixed values. The position of the joint angles of the system was determined with fast and stable responses. Then, two EMG sensors are connected to the ESP32 microcontroller development board so that the robot arm can work with real wrist and elbow movement muscle data. The first EMG sensor is attached to the forearm to receive data from wrist movement, and the second EMG sensor is attached to the upper arm for elbow movement. During the contraction of the muscles, the electric potential difference between the inside and outside of the muscle cells was measured with the pads. Two pads of the EMG sensor were connected side by side, and the other pad was attached to the outer part of the arm as a reference pad. Two EMG probes detected arm contraction and compared it with the reference probe to generate data. The data were taken with these EMG sensors fitted to the user. The system has also added the ADXL335 position and acceleration sensor to keep the arm at a certain angle. With this sensor, when the user wants to move his wrist and elbow, the EMG muscle data are changed, and the angle of the user's arm is taken by fixing the position and acceleration sensor in three axes:  $x$ ,  $y$ , and  $z$ . arm. The data obtained with two EMG sensors and position and acceleration sensors were transferred to MATLAB software. Data between 0 and 4,096 values were obtained with the ESP32 microcontroller development board with 12-bit analogue digital converter channels. The data recording process with the sensors installed on the user is given in Figure 6.

The electronic circuit connection of the system was established in the Proteus ISIS programme, and the printed circuit board was removed. The electronic circuit drawing of the study is also shown in Figure 7.

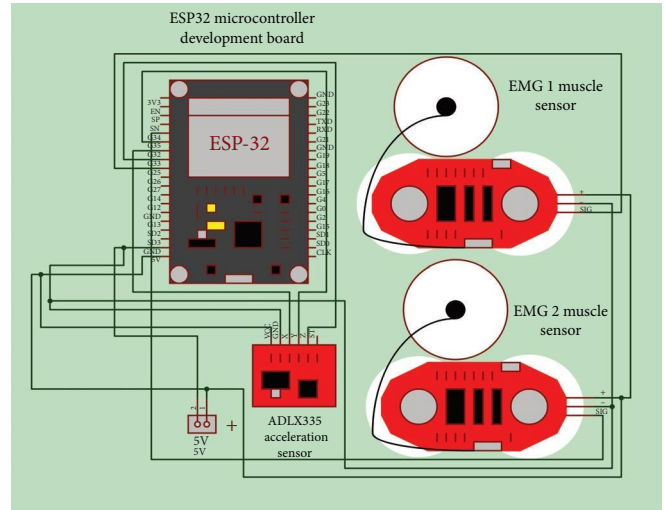


FIGURE 7: Electronic circuit diagram of the system.

#### 2.4. Integration of Fuzzy Logic Algorithm into the System.

Fuzzy logic applications consist of computer-assisted artificial intelligence applications that can predict human behaviour and the functioning of nature in a way that mimics it. Fuzzy logic has high, medium, and low values. It also includes intermediate values such as very low, medium, and high. A fuzzy set is the foundation of fuzzy logic. In the fuzzy logic classical set approach, the elements belong either to that set (1) or (0) [53]. This is called the degree of membership and is represented by Equation (4).

$$A_i\{(x, \mu_A(x)) | x \in A_i, \mu_A(x) \in [0, 1]\}. \quad (4)$$

Structures called fuzzy controllers are used in fuzzy control systems. Fuzzy controllers use a collection of rules called fuzzy rules to produce results. Fuzzy controllers can also use different membership functions. Membership functions related to fuzzy logic are necessary for fuzzy controllers to produce results. The most known fuzzy logic membership functions are triangle, trapezoid, sinusoid, Cauchy, bell, sigmoid, and Gaussian. Thanks to the feedback mechanism in fuzzy logic, it can be seen which rule system and which membership function gives more efficient results [54]. Fuzzy Logic is used to control situations that are nonlinear, complex, and difficult to model, with ambiguous information properties [55].

A fuzzy control algorithm was added for the motors in the wrist and elbow joints. For wrist movement of the wearable exoskeleton robot arm, the EMG sensor is attached to the forearm, and the position and acceleration sensor that receives position information in three axes are determined as inputs. The output parameter for wrist movement was the wrist angles of the system trained with fuzzy control. For elbow movement, the data received with the EMG sensor and position sensor attached to the upper arm became the input parameter, and the angle values of the trained data became the output parameter. The fuzzy control block diagram for wrist and elbow movement is shown in Figures 8 and 9.

There are various inference methods in fuzzy logic control systems. The most widely used are the Mamdani, Larsen,

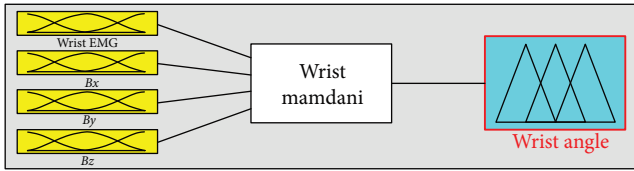


FIGURE 8: Fuzzy control block diagram for wrist movement.

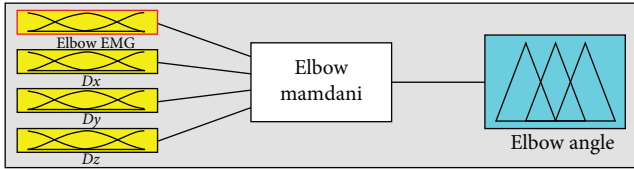


FIGURE 9: Fuzzy control block diagram for elbow movement.

Tsukamoto, and Takagi–Sugeno methods. Mamdani method was used in this study. The Mamdani method is the most widely used fuzzy inference method. The main reason is that the Mamdani inference is more appealing to human perception because it is relatively easy to design and more interpretable. In Mamdani inference, the inputs and outputs are fuzzy values. Membership values are calculated based on rules triggered by input values. The calculated values are then given to the maximum or minimum operator according to the rules and/or their logical connectors. If the facts in the rule are connected with “and,” the calculated membership values are given to the minimum operator; If bound with “or,” the maximum is given to the operator. As the name suggests, these operators return the most minor or most significant of the multiple values they take. The minimisation process is given in Equation (5) [56].

$$R_c = \mu_A(U_1) \wedge \mu_B(U_2). \quad (5)$$

To obtain data from different angles in the system, the data received with two EMG sensors and 3-axis ( $x, y, z$ ) ADXL335 position and acceleration sensors were trained with seven rules for wrist movement and nine rules for elbow movement. According to the data obtained, the system training image for the wrist and elbow is given in Figure 7. The Gaussian membership function is preferred while the data are trained with rules. It is defined by the parameters  $c$  and  $\sigma$  in the Gaussian membership function. The Gaussian membership function used in the system is shown in Equation (6).

Function  $G: x \rightarrow [0,1]$ ;

$$G(x; c; \sigma) = e^{-\frac{(x-c)^2}{2\sigma^2}}. \quad (6)$$

Two EMG sensors and 3-axis ( $x, y, z$ ) ADXL335 position and acceleration sensors were used to obtain data from different angles in the system. The received data were recorded and trained by creating a function with fuzzy logic. The image of the system training for the wrist and elbow according to the obtained data is given in Figure 10.

Here, the system is determined by four different input value rules. As a result of this rule, the angle values of elbow and wrist movement are reported as output. The input parameters seen in Figure 7 are the elbow EMG ( $Dx, Dy, Dz$ ), wrist EMG ( $Bx, By, Bz$ ), and the position sensor's  $x, y, z$  data. Angle values resulting from the rules of the fuzzy logic algorithm applied to the robot arm are given in Figure 11.

In the fuzzy logic controller created, position and EMG sensors are determined as inputs, and elbow and wrist motion angles are shown according to these variables. The maximum and minimum angle values according to the values taken by the position sensor in three axes ( $x, y, z$ ) are also seen in the graph.

The system was trained with the rules, and the imported data were used in the previously exported system with the Simmechanic plugin in Simulink. A fuzzy logic control algorithm has been added to the block diagram that comes with the Simmechanic application. The chart with the fuzzy logic block added to the system is given in Figure 12. The block diagram extracted in Simulink software was run with fuzzy logic blocks trained with muscle and position data and a PID controller.

In the system block diagram, a fuzzy block trained with PID controller and EMG, position, and acceleration sensors has been added to four rotating blocks. Wrist and elbow blocks are also added for shoulder movement. The motion angles and durations of the wearable robot arm were determined by running the simulation. The image of the running simulation is given in Figure 13.

**2.5. Integrating Quadratic SVM Algorithm into Robot Arm Mechanism.** The quadratic SVM algorithm, a regression analysis algorithm, is highly accurate. SVM algorithm is a two-class classifier that fits into a hyperplane that distinguishes between two classes [57]. Classification methods aim to maximise the perpendicular distances of these examples to the separating surface by finding the closest examples of the courses while classifying the data. This method is based on structural risk minimisation and statistical learning theory. This algorithm can also define nonlinear decision boundaries in high-dimensional variable space by solving the second-order optimisation problem. SVM is a supervised learning method generally used in classification problems. Draws a line to separate points placed on a plane. This aims to have the line at the maximum distance for the points of both classes. Basic SVM algorithm theory states that an infinite number of lines are dividing the classes for a nonlinearly separable dataset containing scores from two courses. A line that best separates the two classes (i.e., the decision boundary) is selected using a subset of only training samples known as support vectors. It suits complex but small and medium-sized datasets [11, 58]. It has been shown to produce lower estimation errors than classifiers based on other methods, such as artificial neural networks, especially considering many features for sample identification. The working state of the SVM algorithm is shown in Figure 14.

A margin (separation, gap, separation, distance) is defined as the distance between two classes defined by a hyperplane. Geometrically, it corresponds to the shortest distance between

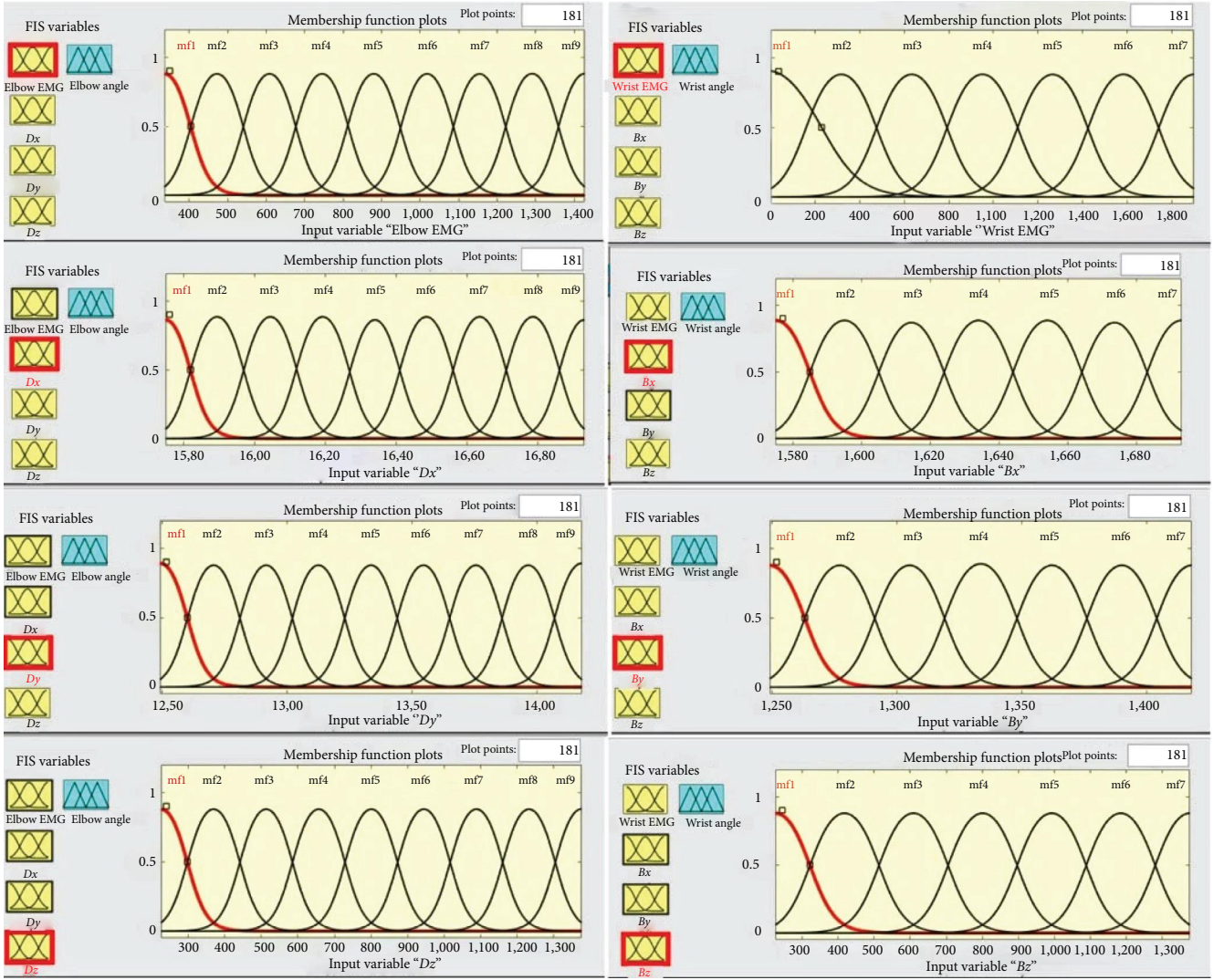


FIGURE 10: Image of the trained system for wrist and elbow movement.

the data points closest to the margin and any point on the hyperplane. Figure 14 shows a geometric structure of the respective optimal hyperplane under the above conditions for a two-dimensional input space. Let's assume that  $w$  and  $b$  are the weight vector and bias in the hyperplane, respectively. The related hyperplane can be defined as in Equation (7).

$$w^T x + b = 0. \quad (7)$$

As a result, SVM aims to find the parameters  $w$  and  $b$  to maximise the separation margin ( $H$ ) for the optimal hyperplane; this value is found by the shortest geometric distances  $d_+$  and  $d_-$  from the two classes, respectively, thus the SVM is also referred to as the "maximal margin classifier" is named. It is corrected to have a functional margin equal to 1 without loss of generality. Here, Equations (8) and (9) are obtained.

$$w^T x_i + b \geq +1 \text{ when } y_i = +1, \quad (8)$$

$$w^T x_i + b \leq -1 \text{ when } y_i = -1. \quad (9)$$

Equations (10) and (11) describe the  $H_1$  and  $H_2$  planes.

$$H_1 : w^T x_i + b = +1, \quad (10)$$

$$H_2 : w^T x_i + b = -1. \quad (11)$$

The points in the  $H_1$  and  $H_2$  planes are the ends of the support vectors; the  $H_0$  plane is the median in between, denoted  $w^T x_i + b = 0$ .  $d_+$  is the shortest distance from the nearest positive point and  $d_-$  is the shortest from the nearest negative point. The margin of a separating hyperplane is  $d_+ + d_-$ . The form of the equation separating the classes and defining the decision surface is a hyperplane of this form:

$$w_x^T + b \geq 0 \text{ for } d_i = +1, \quad (12)$$



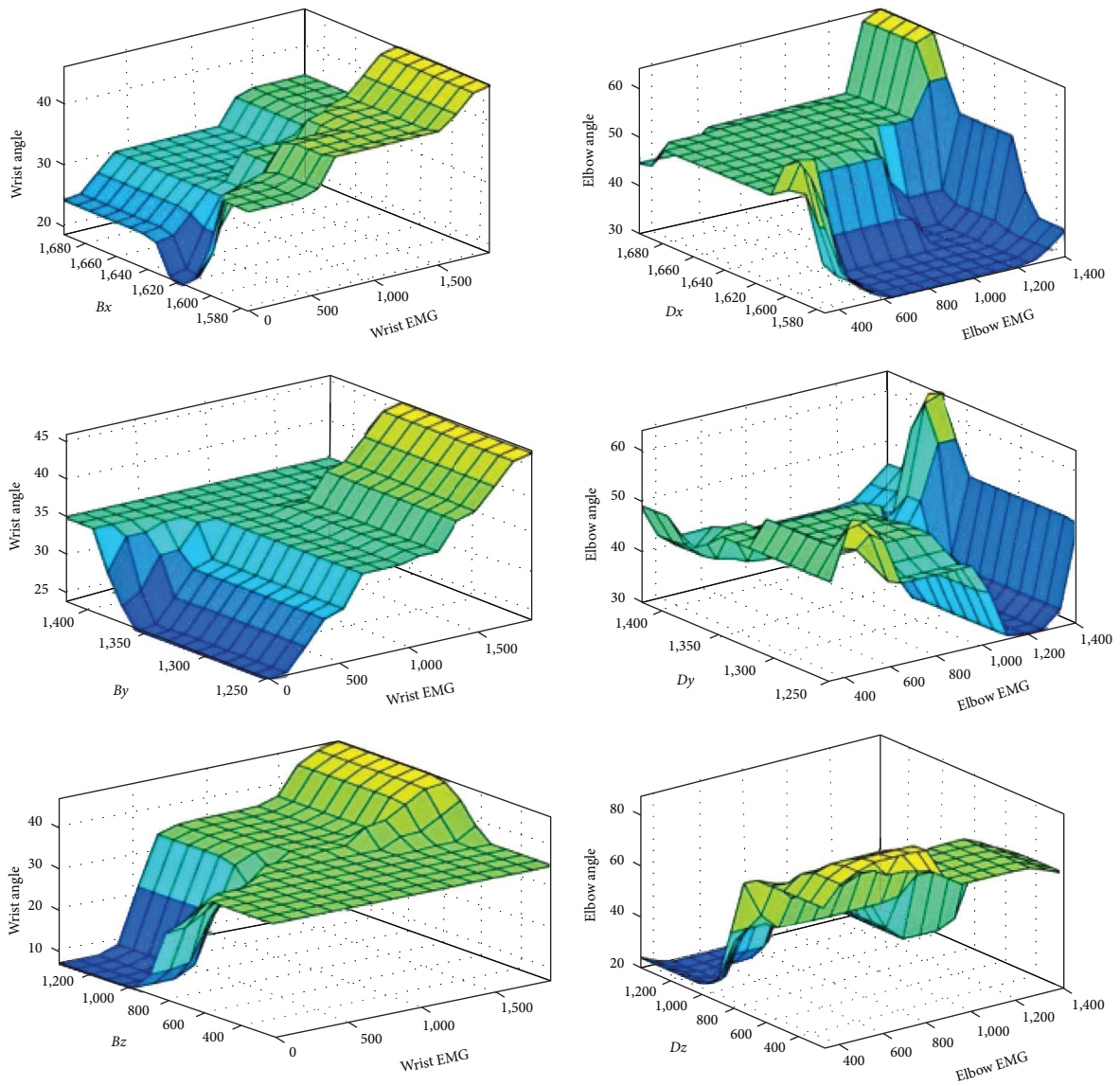


FIGURE 11: Data and angle graph for wrist and elbow movement.

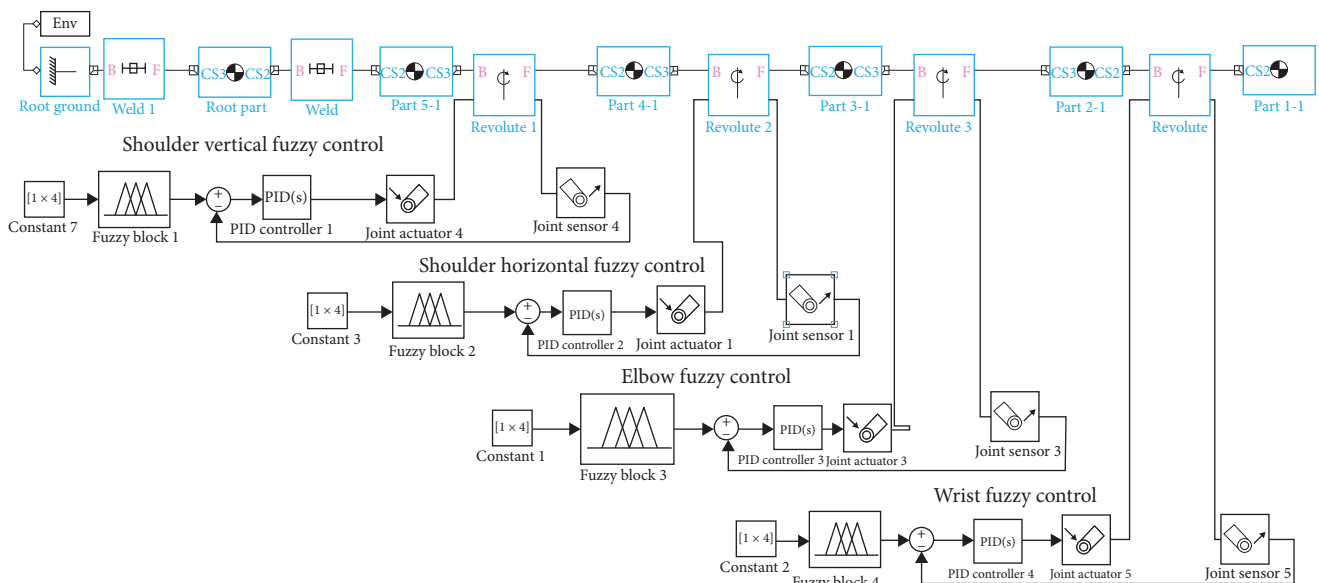


FIGURE 12: Image of the system with fuzzy logic added.

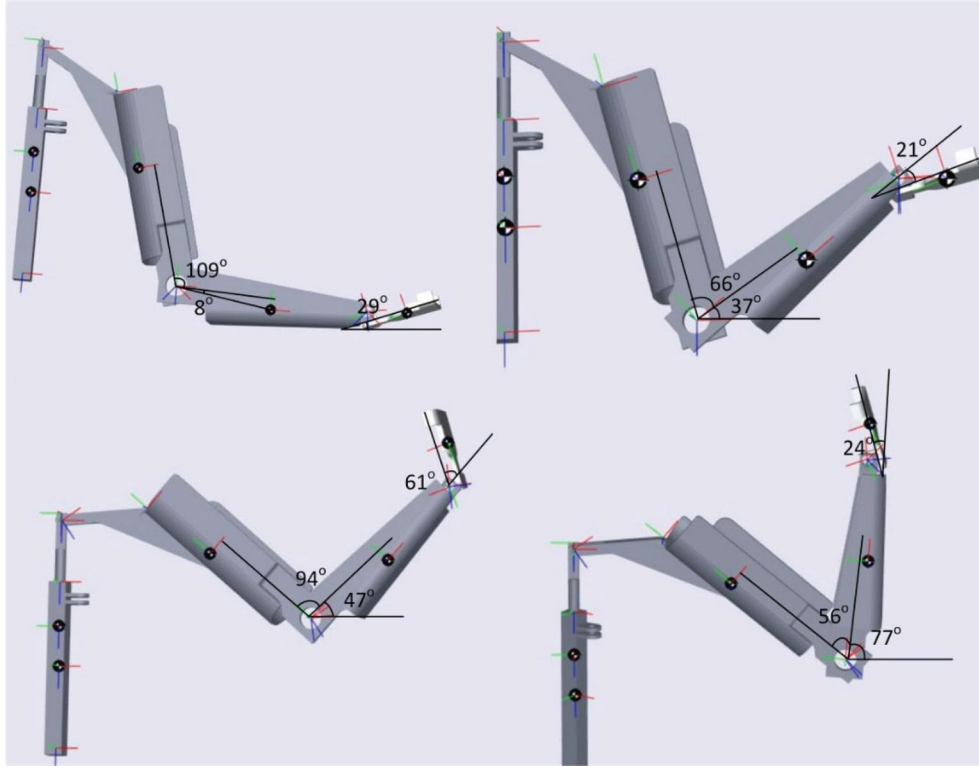


FIGURE 13: Image of the simulation run with fuzzy logic.

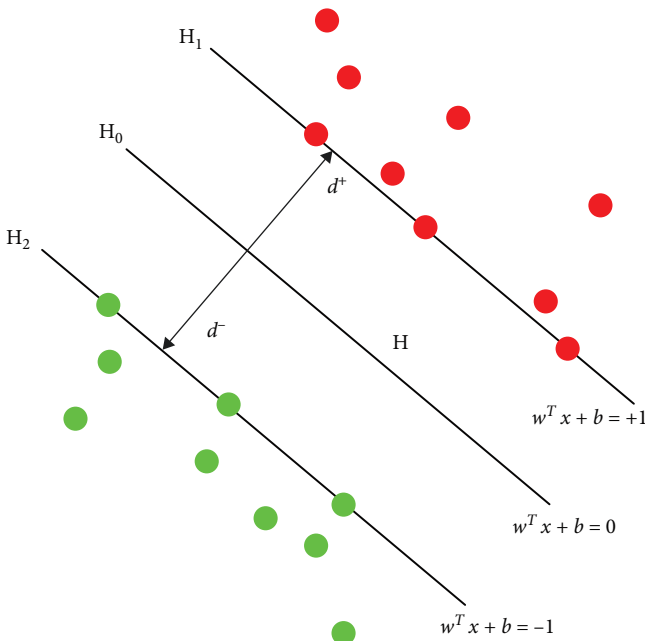


FIGURE 14: SVM algorithm state.

$$w_x^T + b < 0 \text{ for } d_i = -1. \quad (13)$$

The working state of the SVM algorithm is given in Equations (14) and (16).

$$d = \frac{2}{\sqrt{w \cdot w}} f \min(w) \frac{w \cdot w}{2}, \quad (14)$$

$$di(w \times x_i + b) \geq 1, 1 \leq i \leq N, \quad (15)$$

$$L(w, b, \alpha) = \frac{w \cdot w}{2} - \sum \alpha_i, \quad (16)$$

where  $d_i$  is calculated from the equation  $((w^T x_i + b) - 1)$  [59, 60].

Elbow and wrist angle values were obtained from the data obtained with two EMG sensors: position and acceleration sensors. Angle values are limited to the motion angles of the human anatomical structure specified in the literature. An angle-finding function was created in MATLAB software, and the angle value was calculated according to the received data. System angle values trained with the quadratic SVM algorithm were transferred to Simulink software as output. The training of muscle and position data for wrist and elbow movements with quadratic SVM and the graph of response functions are given in Figures 15 and 16. It is seen that the prediction and response points of the data obtained here show a normal distribution. In addition, it was determined that the validity estimation for wrist and elbow motion settled in the linear direction.

Many machine learning algorithms were tested in the system for the joint angular movements of the wearable exoskeleton robot arm, which was trained with the data obtained from the EMG muscle and the ADXL335 position and acceleration sensor. It has been determined that the quadratic SVM algorithm gives the most stable result. Therefore, the

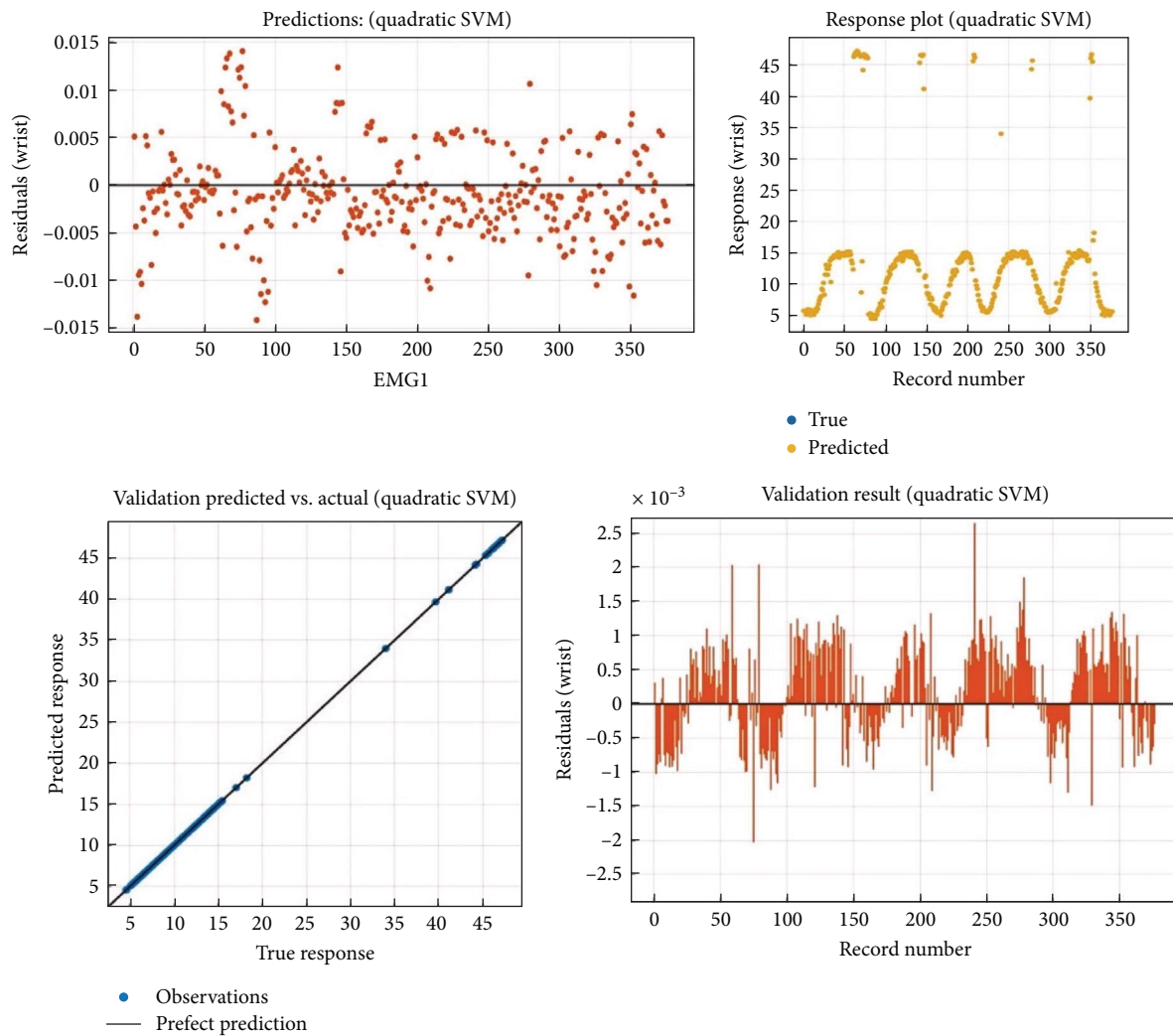


FIGURE 15: Quadratic SVM for wrist movement.

system was trained with the quadratic SVM algorithm and added to the block diagram. In addition to the PID control in the system, which was previously designed and whose block diagram was extracted in Simulink, the system was operated in line with the data received with the quadratic SVM algorithm. With its original design and kinematic analysis, the system works both stable and with the PID controller and feedback blocks. The instantaneous muscle can work at the closest right angle to the human arm and position data with the quadratic SVM algorithm. The block diagram of the system realised with the quadratic SVM algorithm in Simulink is given in Figure 17.

Many machine learning algorithms were tested in the system for the joint angular movements of the wearable exoskeleton robot arm, which was trained with the data obtained from the EMG muscle and the ADXL335 position and acceleration sensor. It has been determined that the quadratic SVM algorithm gives the most stable result. The accuracy rate was 99% with the data trained on the quadratic SVM algorithm. Therefore, the system was trained with the quadratic SVM algorithm and added to the block diagram. In

addition to the PID control in the system, which was previously designed and whose block diagram was extracted in Simulink, the system was operated in line with the data received with the quadratic SVM algorithm. With its original design and kinematic analysis, the system works both stable and with the PID controller and feedback blocks. The instantaneous muscle can work at the closest right angle to the human arm and position data with the quadratic SVM algorithm. The block diagram of the system realised with the quadratic SVM algorithm in Simulink is given in Figure 18.

In the MATLAB software, a function called angular find was created for the system, and the fuzzy logic algorithm in the wrist and elbow rotary blocks was removed and replaced with the quadratic SVM algorithm with the PID controller. The system was run in a simulation environment, and angle values were determined with EMG and position sensor data. The system's Angle values realised with the fuzzy logic algorithm and the quadratic SVM algorithm were compared. Figure 19 shows the angle values of wrist movement and the angle values obtained in the simulation from these two

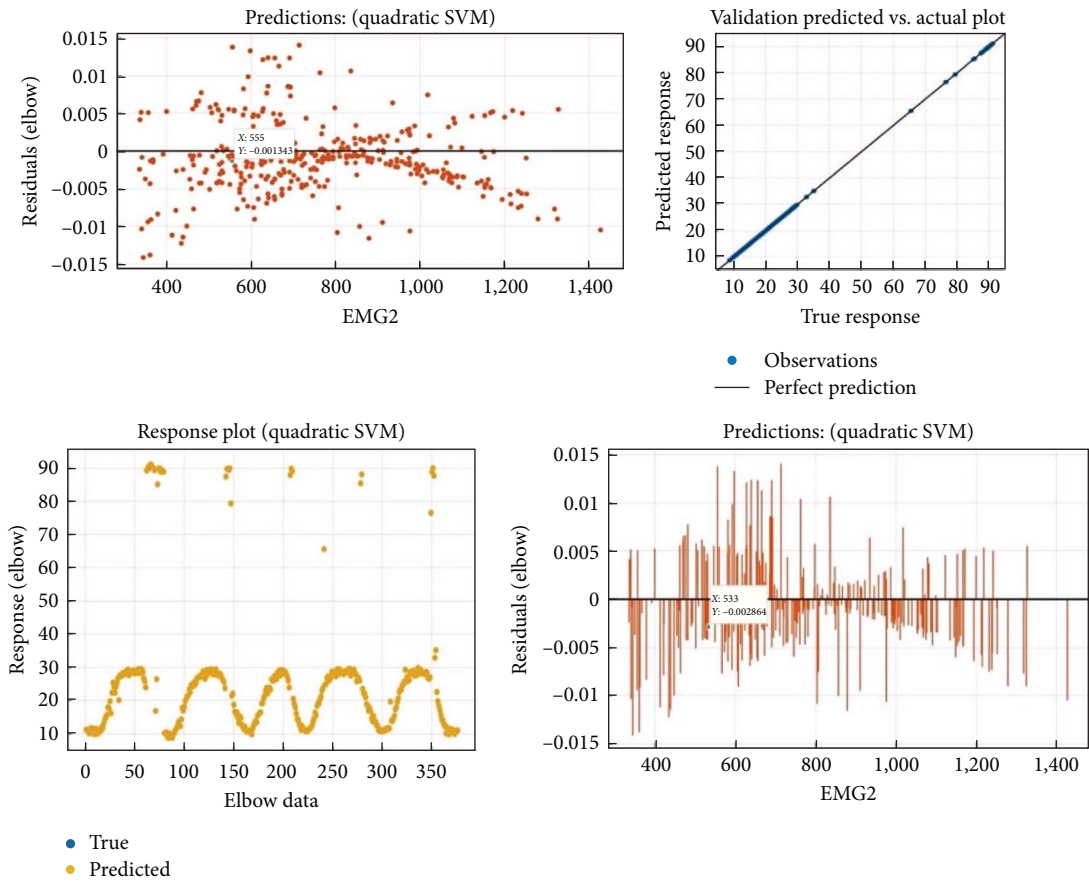


FIGURE 16: Quadratic SVM for elbow movement.

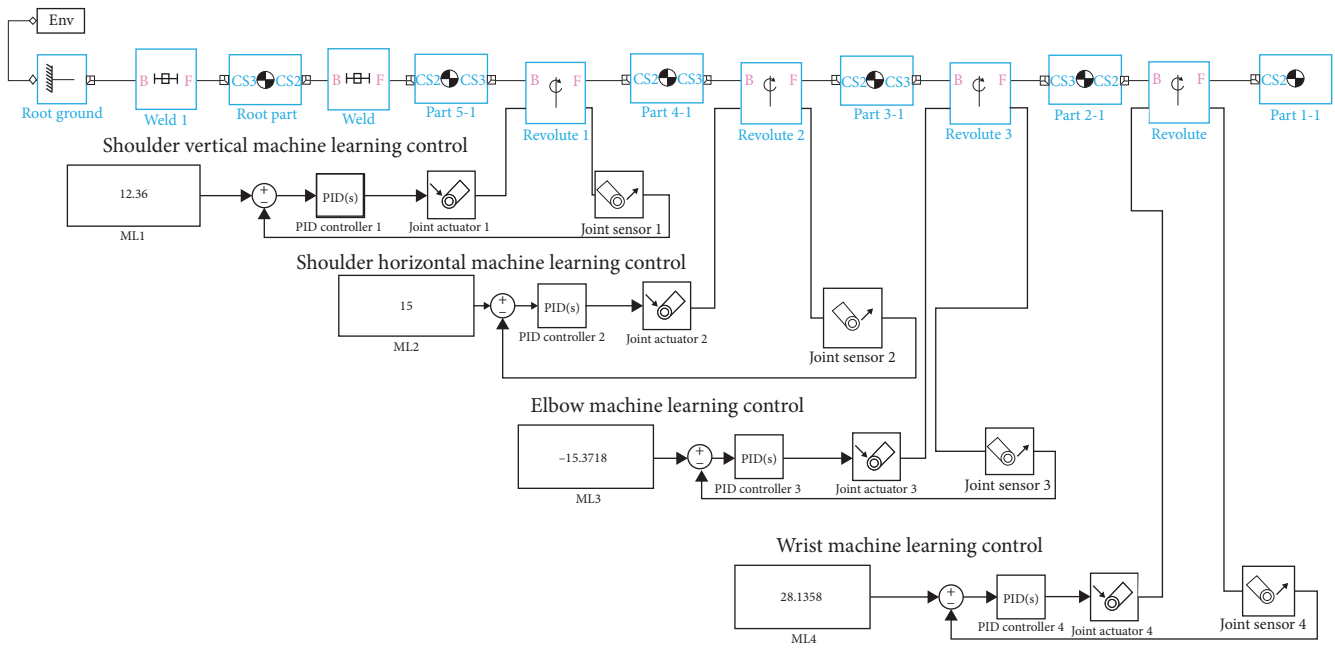


FIGURE 17: Simulink model of the system working with quadratic SVM algorithm and PID controller.

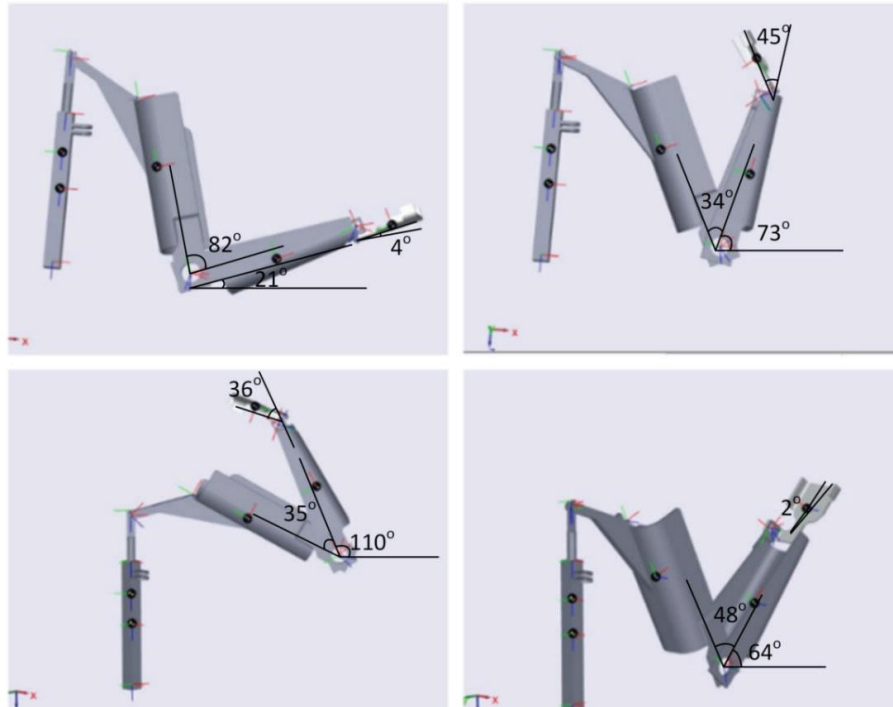
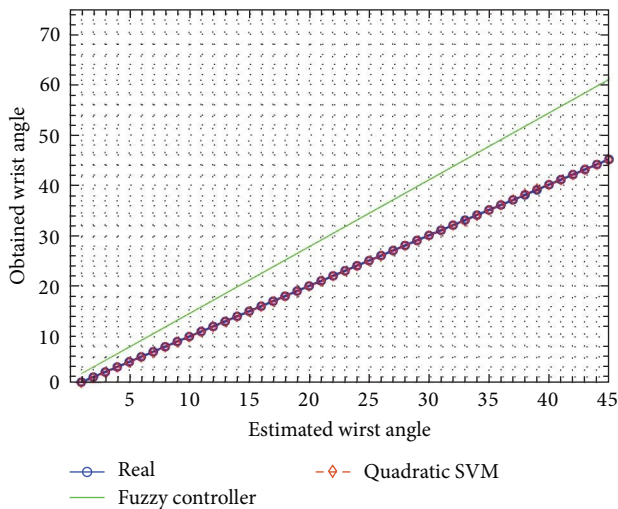
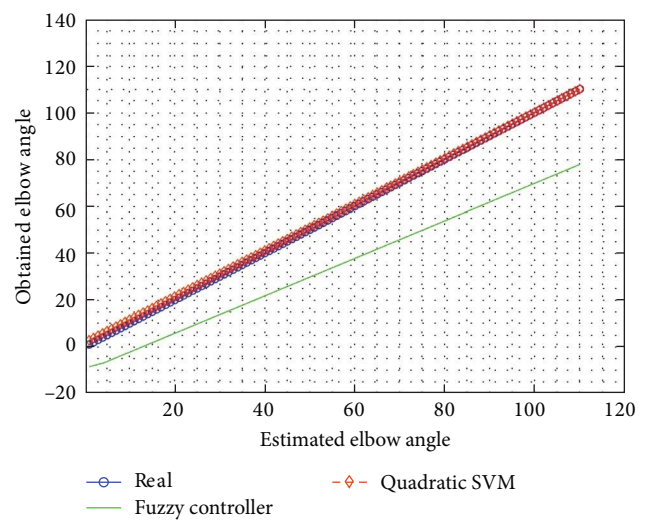


FIGURE 18: Simulation image of the system with angle value with quadratic SVM.



(a)



(b)

FIGURE 19: (a) Graphical comparison of fuzzy logic algorithm and quadratic SVM algorithm for wrist movement with actual angle values. (b) Graphical comparison of fuzzy logic algorithm and quadratic SVM algorithm for elbow movement with actual angle values.

algorithms. When the human anatomy is examined, the maximum angle of the wrist movement to the horizontal is  $45^\circ$ . In the system controlled by the fuzzy logic algorithm, the maximum angle is determined as  $61^\circ$ . The graph concludes that the angle values of the system realised with the quadratic SVM algorithm are very close to the actual angle values. The same procedure was repeated for the elbow movement, and the angle values were found.

The maximum angle of elbow movement in human anatomy is determined as  $110^\circ$ . Here, the maximum angle value of the system controlled by fuzzy logic has been determined as  $77^\circ$  in the horizontal. Again, in this algorithm, the elbow angle, which should be  $1^\circ$ , is in the negative direction. It is seen that the system controlled by the quadratic SVM algorithm is very close to the actual angle values. Error percentages of wrist and elbow movement angle values were calculated.

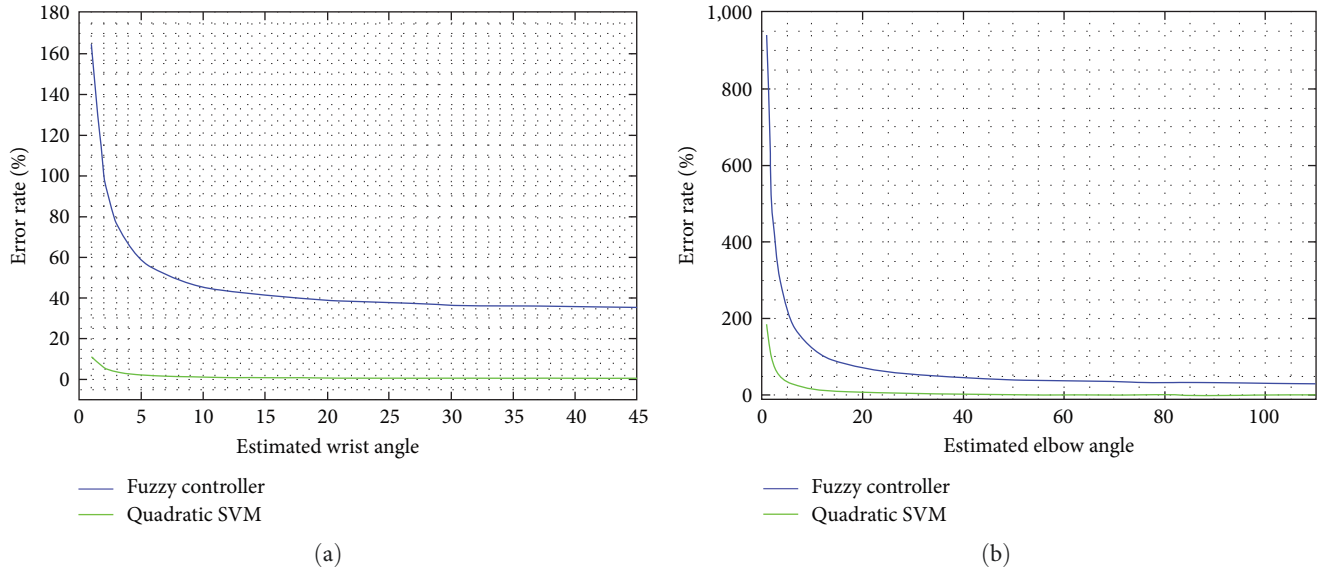


FIGURE 20: (a) Error percentage of fuzzy logic algorithm and quadratic SVM algorithm for wrist movement. (b) Error percentage of fuzzy logic algorithm and quadratic SVM algorithm for elbow movement.

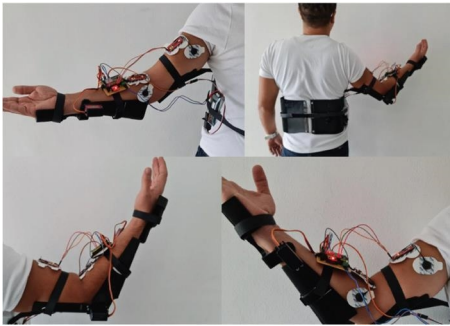


FIGURE 21: Image of the wearable exoskeleton robot arm.

The graph of fuzzy logic control and quadratic SVM control error percentages is shown in Figure 20. Here, the error percentage for the smallest value of wrist movement is 164.25% for fuzzy logic and 10.9% for quadratic SVM. As the angle value approaches the maximum angle, the error percentage of the fuzzy logic controller is 35.90%, and 0.21% for the quadratic SVM controller. This ratio of elbow movement is 29.27% in the Fuzzy Logic controller for the maximum angle value and 0.0006% in the quadratic SVM controller. The error rate decreases for both controllers as the wrist and elbow angles increase.

The simulated robot arm design was controlled with a fuzzy logic algorithm and quadratic SVM algorithm, which is a machine-learning algorithm. These two algorithms were compared with the human arm angle values analysed in the literature before, and it was determined that the closest angle value was realised by machine learning. In vertical motion, the human arm angle value was simulated as  $70^\circ$  from the wrist,  $130^\circ$  from the elbow, and  $110^\circ$  from the shoulder. The simulation's actual human arm angle values were the same as those obtained from the 1,880 EMG sensors and the position sensor. In addition, the parts of the designed system (hand, forearm, shoulder) were printed separately

from the 3D printer. EMG muscle sensors and position sensors were added to the realised design and placed on the previously designed electronic circuit. With the design tested on the user's arm, the motion angles have been determined. Servo and linear motors placed in the design were operated according to the EMG muscle sensors and position sensor values. When the user wears the designed upper limb robot arm and tries to lift a load (max 25 kg), the arm muscles contract, and the load is lifted easily by working the motors. The images of the system performing the design are shown in Figure 21.

### 3. Conclusions

In this study, the wearable exoskeleton robot arm designed for individuals with arm disorders was controlled in a simulation environment, and the designed robot arm was realised. The designed robotic arm was transferred to Simulink software with the Simmechanics application, and the block diagram was created. When the robot arm is simulated, it has been observed to work unbalanced and move very fast. PID controller blocks have been added to the block diagram to provide stability and balance in elbow and wrist movements. The  $P$ ,  $I$ , and  $D$  values of the PID controller are entered separately for the movement of the simulated wearable exoskeleton robot arm. It was determined that the robot arm in the simulation reached the manually entered angle values. For the system to work stably with real muscle and position data, a lot of data were taken from the user's arm at different angles for wrist and elbow movement with EMG muscle sensors and position sensors. The system was first run with a fuzzy logic algorithm, and angle analysis was done. It was determined that some angle values could not be reached in the system realised with fuzzy logic. Then, the system was tested with machine learning algorithms in MATLAB software with the same data. It was determined that the quadratic SVM algorithm gave the best

accuracy among the machine learning algorithms. Angle analysis was performed by training the same data with the quadratic SVM algorithm. It has been determined that the system trained with quadratic SVM in the simulation works very stable and gives the same results when the angle values are taken. The angle analysis performed with two different algorithms determined that the system trained with the second-order SVM algorithm gave movement angles closer to human limb movements compared to the fuzzy logic algorithm. In addition, when the simulation image is examined, it is observed that the system with the quadratic SVM algorithm is less jittery and stable than the fuzzy logic system. This study compares the fuzzy logic and quadratic SVM algorithms in robotic arm simulation. The study also shows the literature that the quadratic SVM algorithm gives the best results for robotic arm operation. In addition, the reality of the robot arm designed and controlled in the simulation environment has been created. According to the sensor data, it has been observed that it works stably in daily work.

### Data Availability

The (EMG sensor data, ADXL335 position, and acceleration sensor data) data used to support the findings of this study are available from the corresponding author upon request.

### Conflicts of Interest

The authors of this publication certify that they have no affiliations with any organization in which they have any financial interest (such as honoraria, educational grants, participation in speakers' bureaus, membership, employment, consultancy, stock ownership) or other equity interest.

### Authors' Contributions

Design was contributed by Çağatay Ersin; electronic circuit connection was contributed by Çağatay Ersin and Mustafa Yaz; software was contributed by Çağatay Ersin; data acquisition was contributed by Çağatay Ersin and Mustafa Yaz; creating the design was contributed by Çağatay Ersin; original draft writing was contributed by Çağatay Ersin; review writing and editing was contributed by Mustafa Yaz.

### Acknowledgments

This study was supported by the Bozok University Scientific Research Projects Coordination Unit (project no: 6601b-FBE/21-440).

### References

- [1] N. G. Adar, H. Ören, and R. Kozan, "5 Serbestlik dereceli robot kolunun modellenmesi ve kontrolü," *Sakarya Üniversitesi Fen Bilimleri Dergisi*, vol. 17, no. 1, pp. S155–160, 2013.
- [2] Ç. Ersin and M. Yaz, "Comparison of KNN and random forest algorithms in classifying EMG signals," *European Journal of Science and Technology*, vol. 51, pp. 209–216, 2023.
- [3] S. Kocaoğlu and E. Akdoğan, "Exoskeleton robots for rehabilitation of the upper limb," in *Otomatik Kontrol Ulusal Toplantısı*, pp. 10–12, TOK, 2015.
- [4] J. L. Ren, Y. H. Chien, E. Y. Chia, L. C. Fu, and J. S. Lai, "Deep learning based motion prediction for exoskeleton robot control in upper limb rehabilitation," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 5076–5082, IEEE, 2019.
- [5] J. S. Artal-Sevil, A. Acon, J. L. Montanes, and J. A. Dominguez, "Design of a low-cost robotic arm controlled by surface EMG sensors," in *2018 XIII Technologies Applied to Electronics Teaching Conference (TAAE)*, pp. 1–8, IEEE, 2018.
- [6] Ö. Başer, B. Şekerci, H. Kizilhan, and E. Kılıç, "İnsan ve alt ekstremite dış iskelet robotunun matlab simmechanics ortamında modellenmesi ve etkileşim kuvvetlerinin minimize edilmesi üzerine kontrol çalışması," *Mühendislik Bilimleri ve Tasarım Dergisi*, vol. 6, no. 3, pp. 365–374, 2018.
- [7] C. Közcut and M. Soyaşlan, "Euler bilek scara robot kolu için kinematik analiz yazılımının geliştirilmesi," in *6th International Advanced Technologies Symposium (IATS'11)*, 2011.
- [8] Y. Bouteraa, I. B. Abdallah, A. ElMogy, A. Ibrahim, U. Tariq, and T. Ahmad, "A fuzzy logic architecture for rehabilitation robotic systems," *International Journal of Computers Communications & Control*, vol. 15, no. 4, 2020.
- [9] S. Gürses, Ö. Başer, E. ve Kılıç, and U. Tilki, "Biyomimetik alt ekstremite dış iskelet robotunun tasarımı ve kontrolü" 2017.
- [10] M. Bi, "Control of robot arm motion using trapezoid fuzzy two-degree-of-freedom PID algorithm," *Symmetry*, vol. 12, no. 4, Article ID 665, 2020.
- [11] U. Aydın, *Buck-boost çevirici devresinin pid ve floating mode kontrol ile kontrolü*, Yüksek Lisans Tezi, Fen Bilimleri Enstitüsü, 2019.
- [12] N. M. Ghaleb and A. A. Aly, "Modeling and control of 2-DOF robot arm," *International Journal of Emerging Engineering Research and Technology*, vol. 6, no. 11, pp. 24–31, 2018.
- [13] S. M. Attya, "Robot arm Ijournals: international journal of software & hardware research in engineering (Ijshre)," 2022.
- [14] A. M. Abdul-Sadah, K. M. Raheem, and M. M. S. Altufaili, "A fuzzy logic controller for a three links robotic manipulator," in *AIP Conference Proceedings*, AIP Publishing LLC, Article ID 050026, 2022.
- [15] M. Namazov, "Fuzzy logic control design for 2-link robot manipulator in MATLAB/Simulink via robotics toolbox," in *2018 Global Smart Industry Conference (GloSIC)*, pp. 1–5, IEEE, 2018.
- [16] T. Hussein Elmenfy and M. Mohamed, "Robot arm control using fuzzy logic design integrated with genetic algorithm optimization," in *The 7th International Conference on Engineering & MIS 2021*, pp. 1–10, 2021.
- [17] B. Mahesh, "Machine learning algorithms—a review," *International Journal of Science and Research (IJSR)*, vol. 9, pp. 381–386, 2020.
- [18] S. Ray, "A quick review of machine learning algorithms," in *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, pp. 35–39, IEEE, 2019.
- [19] T. Triwiyanto, I. P. A. Pawana, B. G. Irianto, T. B. Indrato, and I. D. G. H. Wisana, "Embedded system for upper-limb exoskeleton based on electromyography control," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 17, no. 6, pp. 2992–3002, 2019.
- [20] R. A. R. C. Gopura, K. Kiguchi, and Y. Li, "SUEFUL-7: A 7DOF upper-limb exoskeleton robot with muscle-model-oriented EMG-based control," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1126–1131, IEEE, 2009.

- [21] A. Sappat, K. Mahaphonchaikul, M. Sangworasil, C. Pintavirooj, and A. Tuantranont, "Real-time identification of electromyographic signals from hand movement," in *2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*, pp. 1–4, IEEE, 2012.
- [22] H. Mizuno, N. Tsujiuchi, and T. Koizumi, "Forearm motion discrimination technique using real-time EMG signals," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4435–4438, IEEE, 2011.
- [23] L. D. L. da Silva, T. F. Pereira, V. R. Q. Leithardt, L. O. Seman, and C. A. Zeferino, "Hybrid impedance-admittance control for upper limb exoskeleton using electromyography," *Applied Sciences*, vol. 10, no. 20, Article ID 7146, 2020.
- [24] M. Oghogho, M. Sharifi, M. Vukadin, C. Chin, V. K. Mushahwar, and M. Tavakoli, "Deep reinforcement learning for emg-based control of assistance level in upper-limb exoskeletons," in *2022 International Symposium on Medical Robotics (ISMR)*, pp. 1–7, IEEE, 2022.
- [25] S. Gowtham, K. A. Krishna, T. Srinivas, R. P. Raj, and A. Joshuva, "EMG-based control of a 5 DOF robotic manipulator," in *2020 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, pp. 52–57, IEEE, 2020.
- [26] Z. Song, Z. Luo, G. Wei, and J. Shang, "Self-adaptive obstacle crossing of an AntiBot from reconfiguration control and mechanical adaptation," *Journal of Mechanisms and Robotics*, vol. 16, no. 2, Article ID 021002, 2024.
- [27] I. Kang, R. R. Peterson, K. R. Herrin, A. Mazumdar, and A. J. Young, "Design and validation of a torque-controllable series elastic actuator-based hip exoskeleton for dynamic locomotion," *Journal of Mechanisms and Robotics*, vol. 15, no. 2, Article ID 021007, 2023.
- [28] N. Y. Sattar, U. A. Syed, S. Muhammad, and Z. Kausar, "Real-time EMG signal processing with implementation of PID control for upper-limb prosthesis," in *2019 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 120–125, IEEE, 2019.
- [29] H. B. Kazemian, "The SOF-PID controller for the control of a MIMO robot arm," *IEEE Transactions on Fuzzy Systems*, vol. 10, no. 4, pp. 523–532, 2002.
- [30] H. Aksoy, *Engelliler için dış iskelet tipi robot kol tasarımı, analizi ve kontrolü*, Master's thesis, Balıkesir Üniversitesi Fen Bilimleri Enstitüsü, 2023.
- [31] S. Müftü, *İki uzuvlu bir robot kolunun pozisyon kontrolü için optimize edilmiş PID kontrolcü tasarımı*, Master's thesis, Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü, 2023.
- [32] K. S. Sree, T. Bikku, S. Mounika, N. Ravinder, M. L. Kumar, and C. Prasad, "EMG controlled bionic robotic arm using artificial intelligence and machine learning," in *2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pp. 548–554, IEEE, 2021.
- [33] P. W. Laksono, T. Kitamura, J. Muguro, K. Matsushita, M. Sasaki, and M. S. A. bin Suhaimi, "Minimum mapping from EMG signals at human elbow and shoulder movements into two DoF upper-limb robot with machine learning," *Machines*, vol. 9, no. 3, Article ID 56, 2021.
- [34] B. Lento, Y. Aoustin, and T. Zielinska, "The control concept for upper limb exoskeleton," 2021.
- [35] M. Rahmani and M. H. Rahman, "An upper-limb exoskeleton robot control using a novel fast fuzzy sliding mode control," *Journal of Intelligent & Fuzzy Systems*, vol. 36, no. 3, pp. 2581–2592, 2019.
- [36] J. Tang, J. Cao, M. Wu, L. Zhao, and F. Zhang, "Position control of electro-hydraulic servo system using active disturbance rejection control for upper-limb exoskeleton," *Journal of Vibroengineering*, vol. 25, no. 3, 2023.
- [37] Y. Bouteraa, I. Ben Abdallah, and K. Boukthir, "A new wrist-forearm rehabilitation protocol integrating human biomechanics and SVM-based machine learning for muscle fatigue estimation," *Bioengineering*, vol. 10, no. 2, Article ID 219, 2023.
- [38] B. B. Nair and N. R. Sakthivel, "A deep learning-based upper limb rehabilitation exercise status identification system," *Arabian Journal for Science and Engineering*, vol. 48, no. 2, pp. 1237–1271, 2023.
- [39] Y. Bouteraa, I. B. Abdallah, and A. Elmogy, "Design and control of an exoskeleton robot with EMG-driven electrical stimulation for upper limb rehabilitation," *Industrial Robot: the International Journal of Robotics Research and Application*, vol. 47, no. 4, pp. 489–501, 2020.
- [40] B. Chen, Y. Zhou, C. Chen et al., "Volitional control of upper-limb exoskeleton empowered by EMG sensors and machine learning computing," *Array*, vol. 17, Article ID 100277, 2023.
- [41] Ç. Ersin, M. Yaz, and H. Gökçe, "Upper limb robot arm system design and kinematic analysis," *El-Cezeri*, vol. 7, no. 3, pp. 1320–1331, 2020.
- [42] M. Bozdemir and E. Adıgüzel, "Puma tipi robot ve insan kol hareketlerinin karşılaştırılması; alternatif bir robot kol omuz tasarımı," *Mühendislik Bilimleri Dergisi*, pp. S1057–1061, 1999.
- [43] A. F. Şenel and B. Çetişli, "Beş Eksenli Robot Kolu ile Üretim Bandından Görüntü İşleme ve Nesne Kontrolü," in *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, pp. S158–161, 2015.
- [44] S. Bereketoğlu, H. ve Açıkgöz, and M. Shekkeli, "Fırçasız doğru akım motorunun kesintili tip-2 bulanık mantık denetleyici tabanlı hız kontrolü," *Emo Bilimsel Dergi*, vol. 10, no. 2, pp. 31–40, 2020.
- [45] R. P. Borase, D. K. Maghade, S. Y. Sondkar, and S. N. Pawar, "A review of PID control, tuning methods and applications," *International Journal of Dynamics and Control*, vol. 9, no. 2, pp. 818–827, 2021.
- [46] D. A. Artemyev and I. I. Bikmullina, "Combination of myoware muscle sensor, bluetooth module and analog receiver," *International Journal of Engineering Research and Technology*, vol. 13, no. 11, pp. 3519–3523, 2020.
- [47] E. Altan, K. Pehlivan, and E. Kaplanoğlu, "Comparison of EMG based finger motion classification algorithms," in *2019 27th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, IEEE, 2019.
- [48] B. Kaur, S. Kumar, and B. K. Kaushik, "Novel wearable optical sensors for vital health monitoring systems—a review," *Biosensors*, vol. 13, no. 2, Article ID 181, 2023.
- [49] B. Taşar, "Emg sinyallerinin anlamlandırılması için radyal fonksiyonlu yapay sinir ağı tasarımı," *International Journal of Engineering Research and Development*, vol. 10, no. 2, pp. 153–159.
- [50] X. Liu, R. Singh, M. Li et al., "Plasmonic sensor based on offset-splicing and waist-expanded taper using multicore fiber for detection of aflatoxins B1 in critical sectors," *Optics Express*, vol. 31, no. 3, pp. 4783–4802, 2023.
- [51] R. Singh, Z. Wang, C. Marques, R. Min, B. Zhang, and S. Kumar, "Alanine aminotransferase detection using TIT assisted four tapered fiber structure-based LSPR sensor: from healthcare to



- marine life,” *Biosensors and Bioelectronics*, vol. 236, Article ID 115424, 2023.
- [52] S. Kumar, Y. Wang, M. Li et al., “Plasmon-based tapered-in-tapered fiber structure for p-cresol detection: from human healthcare to aquaculture application,” *IEEE Sensors Journal*, vol. 22, no. 19, pp. 18493–18500, 2022.
- [53] M. F. Tosun, A. A. Gençkal, and R. Şenol Isubu, “Modern kontrol yöntemleri ile bulanık mantık tabanlı oda sıcaklığı kontrolü,” *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, vol. 23, no. 3, pp. 992–999, 2019.
- [54] O. Özdemir and Y. Kalınkara, “Bulanık mantık: 2000–2020 yılları arası tez ve makale çalışmalarına yönelik bir içerik analizi,” *Acta Infologica*, vol. 4, no. 2, pp. 155–174, 2020.
- [55] A. E. Tiryaki and R. ve Kazan, “Bulanık mantık ile bulaşık makinesinin modellenmesi,” *Mühendis Ve Makine*, vol. 48, no. 565, pp. 3–8, 2007.
- [56] G. Petriçli and G. G. Emel, “Bulanık mantık ve kontrol sistemlerinin bireysel kredilendirmelerin nihai karar aşamasında kullanımı,” *International Journal of Social Inquiry*, vol. 13, no. 2, pp. 725–764, 2020.
- [57] Ö. Arslan and M. Karhan, “Effect of Hilbert–Huang transform on classification of PCG signals using machine learning,” *Journal of King Saud University—Computer and Information Sciences*, vol. 34, no. 10, pp. 9915–9925, 2022.
- [58] E. Byvatov and G. Schneider, “Support vector machine applications in bioinformatics,” *Applied Bioinformatics*, vol. 2, no. 2, pp. 67–77, 2003.
- [59] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, “A comprehensive survey on support vector machine classification: applications, challenges and trends,” *Neurocomputing*, vol. 408, pp. 189–215, 2020.
- [60] Y. Zhang, “Support vector machine classification algorithm and its application,” in *Information Computing and Applications: Third International Conference, ICICA 2012*, vol. 3, pp. 179–186, Springer, Chengde, China, Berlin Heidelberg, 2012.