Design And Control Of Robotic Arm For Object Picker Using Artificial Neural Network

Pranika Walde [1], Prof A M Jain [2]
[1,2] Dept of Electrical Engineering (CS),
[1,2] K. K. Wagh Institute of Engineering Education & Research, Nashik.

Abstract- *Artificial Neural Network (ANN) is a powerful and one of the most effective learning techniques and is inspired by biological neural networks. This paper is focus on training a robotic arm to accurately pick an object within the arm's range. This will be accomplished by creating a system with the implementation of ANN the object's location will be fed as an input to the neural network. The neural network will process this data such that it will output a set of two angles at which each joint angle of the robotic arm should move to accurately reach the aforementioned object. Methodology focuses on train a robotic arm to accurately locate and pick an object within the arm's range. This will be accomplished by creating a system with the implementation of ANN. The ANN output angles after it has learned and been trained are compared to the actual output angles that were manually measured.*

The performance of the system is measured by calculating the error or the difference between the output angles of the neural network and the measured angles. Although there are no straight forward relationships between the input and outputs, the neural network was able to fit reasonable output angles with corresponding input coordinates to properly locate the object. The neural network nonlinearly predicted the appropriate outputs for a given set of inputs. The outputs of the neural network as it was fed with the input training data are compared against the output training data to measure its accuracy. The designed ANN was able to learn the relationship between the input coordinates and the output angles as shown by the results. However, there are some inputs to the neural network that it cannot accurately process to give a reasonable output. This can be improved by adding more training data to the system so that the calculated weights by the neural network can be corrected. These additional training data can help the neural network to decide on which weights to converge to will be correct. Also, adding more hidden layers and more neurons to each layer can help solve this. Using the artificial neural network, the machine will learn by itself how it would move based on the position of the object. Hence, less human involvement and intervention will be needed to control the machine. The normalization of Distance from robotic arm data for the trained neural network using the sensors input can be implemented for analysis of workplace to improve the performance of robotic arm.

Keywords- *Artificial Neural Network (ANN), robotic arm, Object picker.*

I. INTRODUCTION

Artificial Neural Network (ANN) is a powerful and one of the most effective learning techniques and is inspired by biological neural networks. The processing elements of the ANN are analogous to a neuron in a human brain.

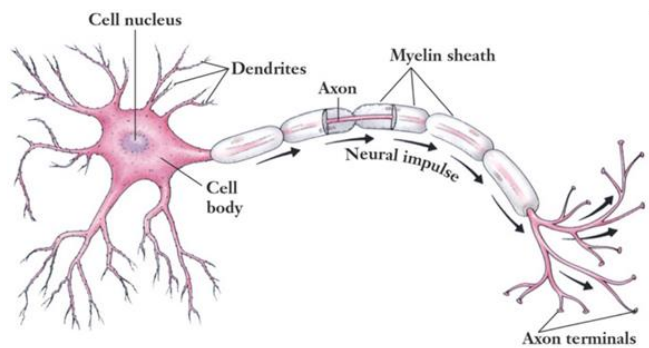


Fig 1. Biological Neuron

These processing elements receive input and multiply it by the weight strength of the connection between successive elements. These are then added in the body of processing element and passed on to the next element. The processing element is the building block of an artificial neural network. This report is focus on training a robotic arm to accurately locate and pick an object within the arm's range. This will be accomplished by creating a system with the implementation of ANN the object's picker will be fed as an input to the neural network. The neural network will process this data such that it will output a set of two angles at which each joint angle of the robotic arm should move to accurately reach the afore mentioned object. Methodology focus on training a robotic arm to accurately picked an object within the arm’s range.

II. ARTIFICIAL NEURAL NETWORK

The ANN output angles after it has learned and been trained are compared to the actual output angles that were manually measured. The performance of the system is measured by calculating the error or the difference between the output angles of the neural network and the measured angles. Although there are no straight forward relationships between the input and outputs, the neural network was able to fit reasonable output angles with corresponding input coordinates to properly locate the object. The neural network nonlinearly predicted the appropriate outputs for a given set of inputs. The outputs of the neural network as it was fed with the input training data are compared against the output training data to measure its accuracy.

In ANN implementations, the signal at a connection between artificial neurons are a real number and the output of each artificial neuron is calculated by a non-linear function of the sum of its inputs. Artificial neurons and connections typically have a weight that adjusts as learning proceeds. The weight increases or decreases the strength of the signal at a connection. Artificial neurons may have a threshold such that only if the aggregate signal crosses that threshold is the signal sent. Typically, artificial neurons are organized in layers. Different layers may perform different kinds of transformations on their inputs. Signals travel from the first (input), to the last (output) layer, possibly after traversing the layers multiple times.

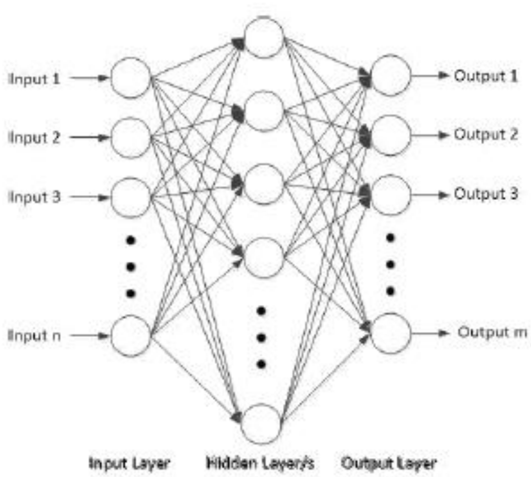


Fig 2. Artificial neural networks model

The designed ANN was able to learn the relationship between the input coordinates and the output angles as shown by the results. Articulated robots resemble the human arm in their 3D motion (they are anthropomorphic). They have three joints, with three variable angles θ1, θ2 and θ3 representing the human body waist, 1-dof shoulder, and elbow joints. They are versatile robots, but have more difficult kinematics and dynamics control equations than other serial robots. All of these robot architectures may be used with variety of robot wrists to provide the orientation degree of freedom. A wrist pitch, with variable angle, is also shown with the articulated robot below. Artificial neural network model train a robotic arm to accurately locate and pick an object within the arm's range. This will be accomplished by creating a system with the implementation of artificial neural networks.

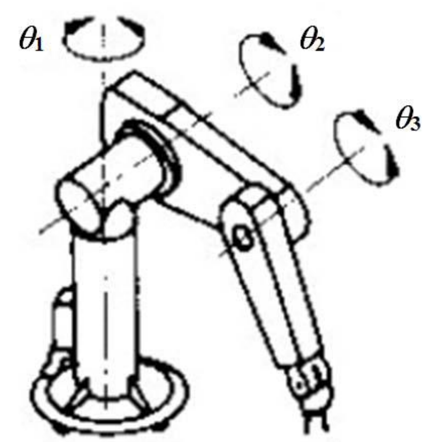


Fig. 3 Articulating robot with three joints (with three variable angles θ1, θ2 and θ3)

III. LEARNING ARCHITECTURE OF ANN

Learning process is a method or a mathematical logic which improves the artificial neural network's performance and usually this rule is applied repeatedly over the network. It is done by updating the weights and bias levels of a network when a network is simulated in a specific data environment. A learning rule may accept existing condition (weights and bias) of the network and will compare the expected result and actual result of the network to give new and improved values for weights and bias. Depending on the complexity of actual model, which is being simulated, the learning rule of the network can be as simple as an XOR gate or Mean Squared Error or it can be the result of multiple differential equations. The learning rule is one of the factors which decides how fast or how accurate the artificial network can be developed.
Depending upon the process to develop the network there are three main models of machine learning:

a) Unsupervised learning.
b) Supervised learning.
c) Reinforcement learning.

Training of ANN using Back propagation neural network .

One of the widely used artificial neural network topologies for classification is back propagation neural network (BPNN). It has the ability to classify data according to similarities in its pattern. BPNN is a popular classifier along with other classifiers like decision trees and Bayesian classifier. BPNN to classify robotic environments using time series data collected from these environments. Back propagation neural network (BPNN) is used in this work for classifying robotic scenarios.

Back propagation is a form of supervised learning for multi-layer nets, also known as the generalized delta rule. The back propagation algorithm has been widely used as a weight adaptation learning algorithm in feed forward multilayer neural networks. The output of the net is compared with the expected output and the difference or error is calculated. Then this error value is propagated backwards to the previous layers so that the incoming weights to these layers can be updated. The overall goal of the learning process is to minimize the total squared error of the signals at the output layer.

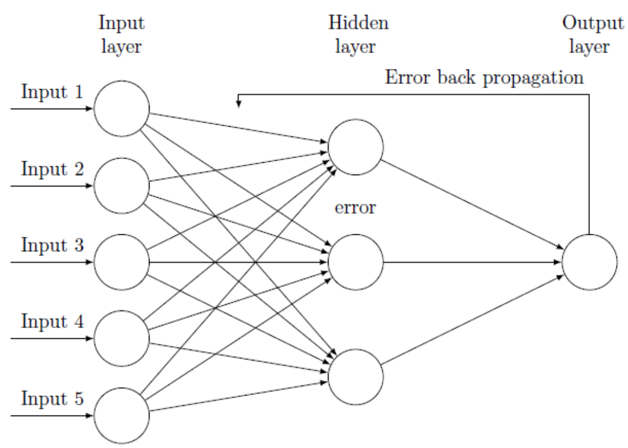


Fig 4. Back propagation neural network (BPNN)

It is basically a gradient descent method. One of the challenges faced until the development of back propagation algorithm was to develop a suitable mechanism to calculate the errors at intermediate levels. Since one does not know the expected output at hidden layers, error calculation is not simple. BPN algorithm calculates error at the hidden layers that will cause minimization of the output error. The back propagation algorithm is an involved mathematical tool; however, execution of the training equations is based on iterative processes, and thus is easily implementable on a computer.

Different types of sensors mounted on robots are often used to collect information about robotic environments. The sensor data could include measurements that provide insights into objects in these environments such as distance to objects, intensity, shapes etc. Artificial neural networks (ANN) offer tremendous opportunities for performing data mining activities, in particular problems pertaining to data classification and clustering. ANN has the ability to learn even from noisy data. It also has ability for reducing the dimensionality of multi-dimensional data.

IV. METHODOLOGY FOR DESIGN OF ROBOTIC ARMS USING ANN

Artificial neural network model train a robotic arm to accurately locate and pick an object within the arm's range. This will be accomplished by creating a system with the implementation of artificial neural networks. The input layer is a layer of processing elements that receives the data receives the data entering the neural network. The hidden layers are all the layers between the input and output layers. These contain elements with varying weights and have highly parallel connections between other hidden layers and the input and output layers. The hidden layers can be considered as a black box, creating a nonlinear relation between the inputs and the outputs. Lastly, the output layer is where the output responses exit the neural network. In a supervised neural network, the system is trained with a set of inputs and outputs. The neural network is then tasked to map the inputs to the outputs by inferring the relationship between the inputs and outputs from given training data

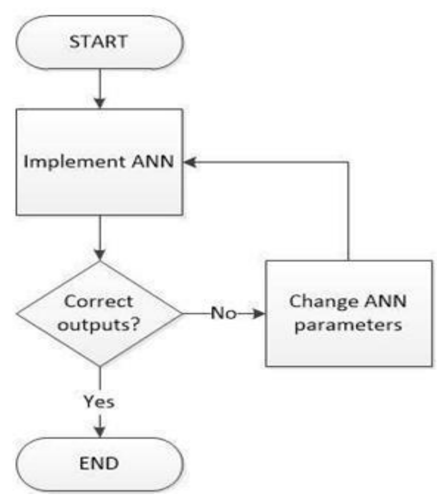


Fig 5. Flow chart of neural network model to train a robotic arm

V. DESIGN AND CONTROL OF ROBOTIC ARMS USING ANN

Artificial neural network model train a robotic arm to accurately locate within the arm's range. This will be accomplished by creating a system with the implementation of artificial neural networks. The figure below shows the robotic arm and the location of each joint along the arm.

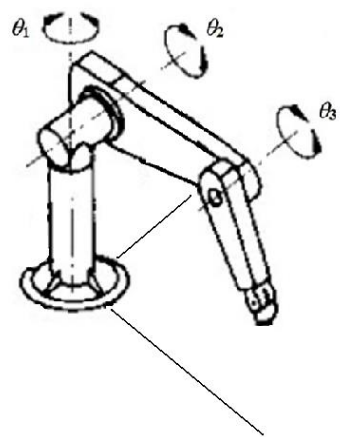


Fig.6 Model of Robotic arm use for implementation.

The hardware mentioned (i.e Sensors for detection of object from X-Y axis respectively) will be the source of the data for the inputs and the desired outputs of robot will be the deflection of links of robot to the desired value (i.e Angle made between the link).

Internal component of model
Input : Input to the network (X1& X2.)
X1- Distance from robot in x-axis.
X2- Distance from robot in y-axis.
Output : Output angle of Robot (θ1, θ2 and θ3).
ϴ1- Angle made with respect to X-axis.
ϴ2- Angle made with respect to link-2.
ϴ3- Angle made with respect to link-3.

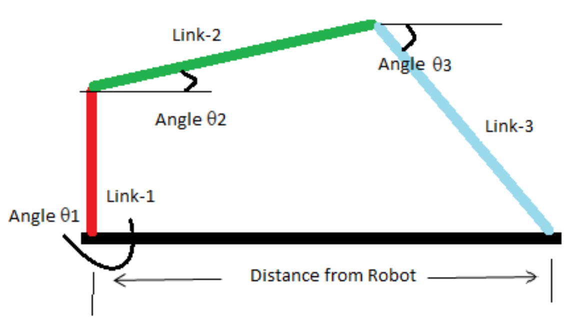


Fig 7 Mathematical model of robotic arm

Design of robot arm for work palace of area 20 x 20 Sq cm
Longest distance detected by sensors (D max)
Note for the entire project all length are in cm (i.e centimeter)

Parameters of robot under study

Height of First Arm H (i.e Link 1) is taken as 5cm
Length of Second Arm L1 (i.e Link 2) is taken as15cm
Length of Third Arm L2 (i.e Link 3) is taken as14cm

Sr No	x1	x2	D	Angle ϴ1	Angle ϴ2	Angle ϴ3
1	0.10	0.10	0.1421	45	-19.47	180
3	1.35	1.35	1.9233	45	0	159.07
3	8.48	8.48	12.0000	45	36.86	90
4	19.85	19.85	28.0767	45	0	-20.92
5	20.80	20.80	29.4200	45	9.92	9.92
6	19.90	19.90	28.1420	45	-19.47	0

Distance from robot vs Deflection of Robot

For analysis of robotic arm, distance is to be calculated for all probable angles of robotic arm. For smooth flow of output angle definite interval is chosen for Angle ϴ3 i.e from 0 to 180 and feed in the above table.

VI. IMPLEMENTATION OF NEURAL MODEL

Implementation of neural network model for control of robotic arm with respect to input signal provided by the sensors.
Input : Input to the network (X1& X2).
X1- Distance from robot in x-axis.
X2- Distance from robot in y-axis.
Output : Output angle of Robot (θ1, θ2 and θ3).
ϴ1- Angle made with respect to X-axis.
ϴ2- Angle made with respect to link-2.
ϴ3- Angle made with respect to link-3
Weight : There is weight association with each input (W0, W1, W2,..).
Bias : Associated with a bias value of X0.
Summing Unit: A summation unit which provide weight sum of the inputs.

$$S = W0X0+W1X1+W2X2$$

Activation Function: Sigmoidal

Steps for implementation of artificial neural network control

- Assign random synaptic weight value (range from -1 to 1) to w1,w2, w3, w4 and wo (range from 1 to 0).
- Apply the selected input for desired output to the neural model
- Based on the input signals output is calculated.
- Due to the difference in desired output and actual output, the error signal is generated and error signal is back propagated to just the each weight based on the proceeding equation, so that error is reduced.
- Repeat the process until the error became zero.
- Repeat the process for next set of inputs.
- After learning allow the robot to perform and monitor the results.

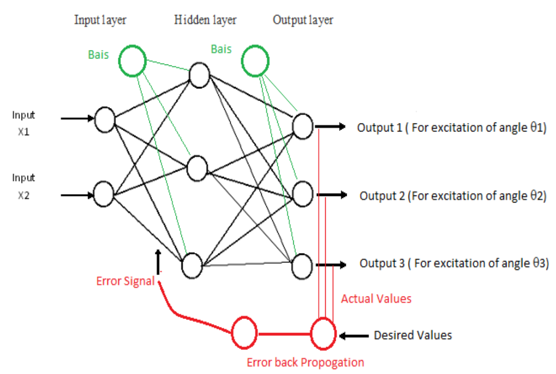
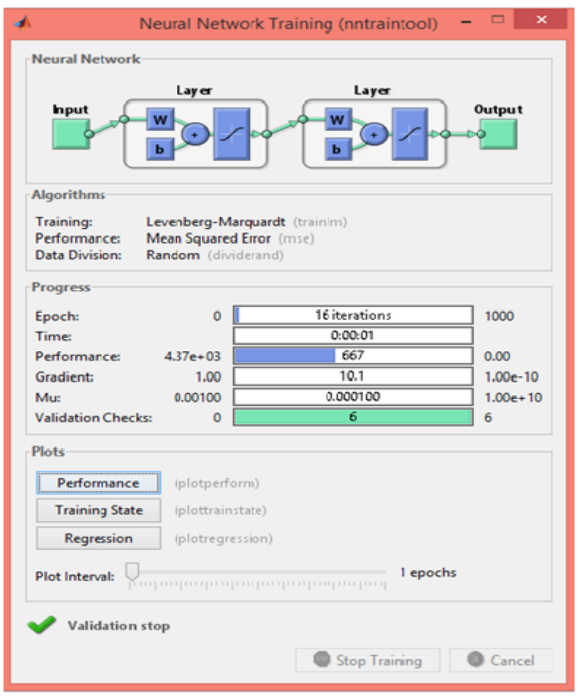
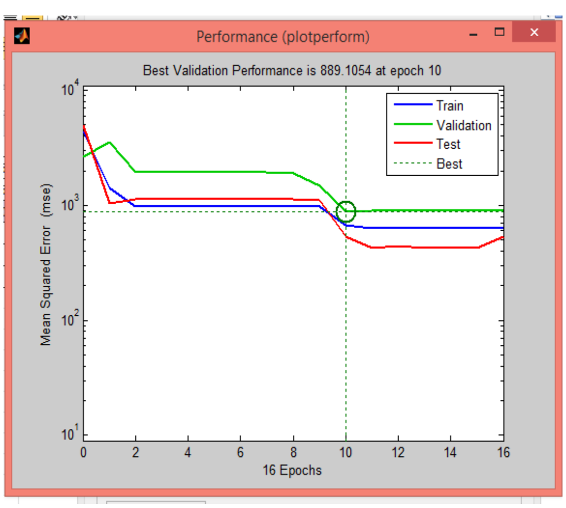


Fig 8 Artificial Neural model for the robotic arm under study

Simulation of neural model in MATLAB.





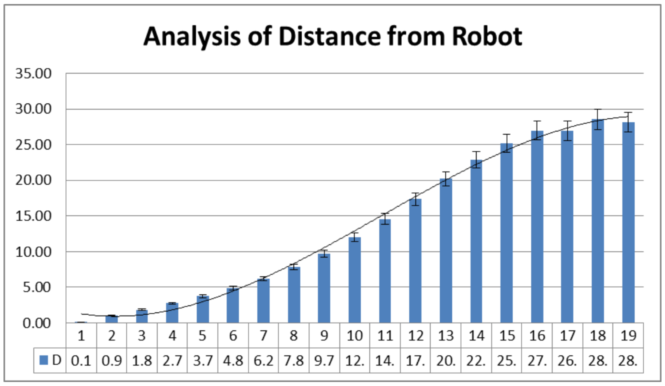
The artificial neural network's output angles after it has learned and been trained are compared to the actual output angles that were manually measured. The performance of the system is measured by calculating the error or the difference between the output angles of the neural network and the measured angles.

Variation of distance from robot and angle relationship						
Sr No	x1	x2	D	Angle ϴ1	Angle ϴ2	Angle ϴ3
1	0.10	0.10	0.1420	45.00	-19.47	180.00
2	0.70	0.70	0.9908	45.00	-9.86	170.00
3	1.30	1.30	1.8427	45.00	-0.80	160.00
4	1.94	1.94	2.7485	45.00	7.64	150.00
5	2.64	2.64	3.7360	45.00	15.42	140.00
6	3.44	3.44	4.8667	45.00	22.43	130.00
7	4.38	4.38	6.2000	45.00	28.35	120.00
8	5.52	5.52	7.8015	45.00	32.93	110.00
9	6.88	6.88	9.7256	45.00	35.86	100.00
10	8.49	8.49	12.0000	45.00	36.86	90.00
11	10.31	10.31	14.5864	45.00	35.86	80.00
12	12.29	12.29	17.3775	45.00	32.93	70.00
13	14.28	14.28	20.2000	45.00	28.35	60.00
14	16.17	16.17	22.8623	45.00	22.43	50.00
15	17.81	17.81	25.1840	45.00	15.42	40.00
16	19.09	19.09	26.9965	45.00	7.64	30.00
17	19.05	19.05	26.9415	45.00	-0.80	20.00
18	20.20	20.20	28.5652	45.00	-9.86	10.00
19	19.90	19.90	28.1420	45.00	-19.47	0.00

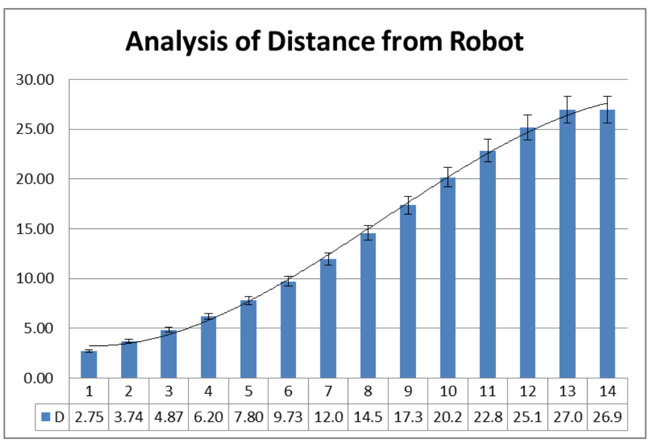
Table of Distance from robot and Deflection of Robot using matlab by providing sample inputs

VII. RESULTS AND DISCUSSIONS

As the data are collected from matlab and arrange in increasing order of Distance form robot i.e D in the table with respect to angles i.e ϴ1, ϴ2 and ϴ3. It is observe that variation is smooth but within a range, so additional error limit is introduce to the final value of D. For all practical application 5% tolerance is added.



Even after addition of tolerance the smooth flow is achievable at the particular values and high degree of error at extreme values due to mechanical constrains. So by investing heavily and making thing more complicated it is better to limit the robot arm with in the range of smooth variation. In this project it is clearly visible that by limiting the robot arm to a distance from 1.94 to 26.94 it is achievable as shown in figure



The designed artificial neural network was able to learn the relationship between the input coordinates and the output angles as shown by the results. However, there are some inputs to the neural network that it cannot accurately process to give a reasonable output. This can be improved by adding more training data to the system so that the calculated weights by the neural network can be corrected. These additional training data can help the neural network to decide on which weights to converge to will be correct. Also, adding more hidden layers and more neurons to each layer can help solve this. However, adding hidden layers should be done with precaution as having too many hidden layers for a given set of training data might result to the system becoming under-characterized since there are no enough training data to set the parameters of the increased number of hidden layers. Hence it is more advisable to increase the number of neurons for the each hidden layer instead.

In addition to that, This paper about the normalization of Distance from robotic arm data for the trained neural network using the sensors input can be implemented for analysis. The inputs should have boundary values to improve the performance of the robotic arm within the workplace.

REFERENCES

[1] Object Locator and Collector Robotic Arm Using Artificial Neural Network. 8th IEEE International Conference Humanoid, Nanotechnology, Information Technology.

[2] Neural Network Control of a Flexible Robotic Manipulator Using the Lumped Spring Mass Model. IEEE transactions on systems, man and Cybernetics: systems, vol. 47, no. 8, August 2017.

[3] Intelligent Control of Industrial Robotic Three Degree of Freedom Crane Using Artificial Neural Network. IEEE: 978-1-/14673-9191/$31.00 © 2016 IEEE.

[4] Speech Control of Robotic Hand Augmented with 3DAnimation Using Neural Network. 2016 IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES)

[5] Object Locator and Collector Robotic Arm Using Artificial Neural Network. 8th IEEE International Conference Humanoid, Nanotechnology, Information Technology

[6] Robotic Arm Joint Recognition in Space Using a Neural Network Trained with Pattern Matching in Lab Windows/CVI with Vision Development Module. IEEE: 978-1-5090-3748-3/16/$31.00 c2016 IEEE.

[7] Modeling and Control of One-Link Robotic Manipulator Using Neural Network Based PID Controller. 2016 Intl. Conference on Advances in Computing, Communications and Informatics (ICACCI), Sept. 21-24, 2016, Jaipur, India

[8] Classification of Robotic Data using Artificial Neural Network 2013 IEEE Recent Advances in Intelligent Computational Systems (RAICS)

[9] Robotic Position/Orientation Control Using Neural Networks Khalid Youssef, Member, IEEE, Peng-Yung Woo, Senior Member, IEEE

[10] Robotic Arm Control By Fine-Tuned Convolutional Neural Network Model 978-1-5090-6494-6/17/$31.00 c 2017 IEEE.

[11] Intelligent control techniques for humanoid Robots 2003 European control conference (ECC)

[12] Creating Graphical User Interfaces with MALTAB, The Math Works Inc., R2010a

[13] MATLAB R2010a User Guide \Create a Simple UI Using GUIDE".