APLIKASI JARINGAN SYARAF TIRUAN PADA SISTEM KONTROL *ROBOT LINE FOLLOWER* DENGAN ALGORITMA*BACKPROPAGATION*

ARTIFICIAL NEURAL NETWORK APPLICATION IN LINE FOLLOWER ROBOT WITH BACKPROPAGATION ALGORITHM

Nur Yanti[1*], Taufik Nur[2]
[1]*Politeknik Negeri Balikpapan/Jln. Soekarno Hatta Balikpapan*
[2]*Universitas Muslim Indonesia /Jln. Urip Sumohardjo KM.5 Panakkukang Makassar*

[*]*E-mail: nur.yanti@poltekba.ac.id*

Diterima 09-10-2017	Diperbaiki 09-11-2017	Disetujui 16-11-2017

ABSTRAK

Dunia elektronika yang semakin berkembang pesat tidak terlepas dari sistem kendali otomatis, salah satunya adalah sistem robot. Robot tidak hanya dapat dikendalikan secara konvensional tetapi telah dikembangkan dengan kecerdasan buatan. Pada penelitian ini jaringan syaraf tiruan diaplikasikan pada robot line follower (LFR), dimana selama ini robot LFR dikendalikan dengan metode konvensional IF dan THEN, sehingga pembacaan pola garis yang berdampak pada gerakan roda robot tidak berjalan mengikuti jalur dengan baik. Sehingga penggunaan jaringan syaraf tiruan dengan algoritma backpropagation pada robot LFR bertujuan menghasilkan putaran motor yang sesuai dengan pembacaan sensor-sensor sehingga robot dapat berjalan mengikuti jalur dengan lebih baik atau gerakan nya lebih halus. Metodologi yang digunakan adalah eksperimental dengan cara melatih data input yang diberikan dari sensor infra merah pendeteksi garis pada robot. Selanjutnya mengatur PWM motor pada robot. Aturan besar PWM disesuaikan dengan pembacaan sensor. Hasil yang diperoleh dengan arsitektur jaringan 1 input layer, 4 neuron, 1 hidden layer, 6 neuron, 1 output layer dengan 2 neuron, Learning rate = 0.1, maksimum epoch = 5000, target error 0.02, yaitu delapan performance pengendalian motorpada robot terbaik dimana nilai validation mendekati target. Secara keseluruhan dengan menggunakan Jaringan Syaraf Tiruan dapat menghasilkan nilai data terlatih yang sudah mendekati target.

Kata kunci:*Backpropagation, jaringan syaraf tiruan, robot line follower*

ABSTRACT

The growing rapidly electronics world can't be separated from the automatic control system, which one is robot system. Robots can't be controlled conventionally only but have been developed with artificial intelligence. In this research, artificial neural network is applied to Line Follower Robot (LFR), where all this time controlled by conventional method : IF and THEN, so that the line pattern reading that impact on robot wheel motion doesn't follow the path well. So the use of artificial neural network with backpropagationalgoritm on LFR aims to produce motor rotation in accordance with the sensor readings so the robot can move to follow the path better or move smoother. The methodology used was experimental by training the input data provided from the line detection infrared sensors on the robot. Next set the PWM motor on the robot. The PWM set rules are based on the sensor readings. The results obtained by network architecture : 1 input layer, 4 neurons, 1 hidden layer, 6 neurons, 1 output layer with 2 neurons, Learning rate = 0.1, maksimum epoch = 5000, target error 0.02, which is eight motor control performance on the best robot where the validation value approaches the target. Overall using the Artificial Neural Network can generate trained data values that are already close to the target.

Keywords:*Backpropagation, artificial neural network, line follower robot*

PENDAHULUAN

Robotika merupakan suatu wujud implementasi dari perkembangan teknologi yang ada pada saat ini, sudah sewajarnya perkembangan keilmuan robotika perlu untuk ditekuni terutama bagi yang berkecimpung di bidang elektronika. Perlunya mengetahui dan menciptakan inovasi baru mengenai robot, dapat dilakukan dengan menerapkan berbagai cara baik dari segi*hardware* maupun *software*.

Salah satu cara untuk menciptakan inovasi dalam dunia robotika adalah dengan menanamkan kecerdasan buatan *(Artificial Intelegent)* pada robot yang kita desain.Dengan

menggunakan kecerdasan buatan diharapkan sistem pengendalian pada robot semakin baik dan dapat memperkecil nilai *error* yang dihasilkan dibandingkan dengan sistem kendali robot secara konvensional. Pada dasarnya kecerdasan buatan tidak hanya dapat dipergunakan untuk desain sistem kontrol, tetapi juga dapat dipergunakan untuk berbagai macam keperluan seperti analisis, diagnosis, memprediksi suatu kondisi yang akan datang berdasarkan data masa lampau, dan lainnya. Kecerdasan buatan dapat diimplementasikan dalam berbagai bidang antara lain elektronika, biomedis, ekonomi, kelautan, pertanian, geofisika dan sebagainya.

Dalam dunia robotika, kecerdasan buatan telah banyak menghasilkan berbagai macam robot dengan kemampuan yang sangat baik seperti mengenali suara, sentuhan, menghindari halangan, mengenali pola atau jalur yang dilaluinya. Pada penelitian ini kecerdasan buatan yaitu jaringan syaraf tiruan digunakan untuk menganalisis gerakanatau kendali pada *robot line follower* (penjejak garis).*Robot Line Follower* (LFR) adalah robot beroda yang dapat berjalan pada jalur hitam atau putih karena dilengkapi oleh sensor inframerah. Robot LFR sederhana biasanya dikendalikan dengan menggunakan pemrograman *IF* dan *THEN*. Penggunaan Jaringan Syaraf Tiruan (JST) menjadikan pengontrolan robot LFR dengan cara dilatih atau pembelajaran [1].

Analisa sistem kontrol dengan kecerdasan buatan untuk robot LFR ini diimplementasikan pada *robot line follower* 4 sensor, metode kontrol menggunakan Jaringan Saraf Tiruan (JST) dengan algoritma *backpropagation*. Dimana arstektur jaringan terdiri dari 1 *input layer* dengan 4 *neuron* yang merupakan keempat sensor robot (S1, S2, S3, S4), kemudian 1 *hidden layer* yang terdiri dari 6 *neuron*, serta 1 *output layer* yang terdiri dari 2 *neuron* yang merupakan PWM motor kiri dan kanan robot.

Tujuan dari penelitian ini menghasilkan *robot line follower* dengan putaran motor yang sesuai dengan nilai pembacaan sensor dan dapat berjalan mengikuti jalur dengan lebih baik atau gerakannya lebih halus dibanding hanya menggunakan kontrol konvensional

METODOLOGI

Penelitian ini dilakukan dengan menggunakan metode eksperimental dengan cara melatih data input yang diberikan dari sensor inframerah pendeteksi garis pada robot. Penggunaan algoritma jaringan dilakukan dengan beberapa tahapan. Tahap pertama dengan melihat pembacaan sensor-sensor robot berupa sensor inframerah sebanyak 4 buah. Hasil pembacaan dikonversikan dalam bentuk digital dengan ketentuan jika sensor mengenai garis maka akan bernilai 1, sedangkan jika sensor tidak membaca garis maka sensor bernilai 0. Untuk selanjutnya keempat sensor tersebut dijadikan sebagai input pada jaringan syaraf tiruan. Dimana sensor 1 terletak paling kiri dan secara berurutan hingga sensor 4 terletak paling kanan.

Tahap kedua adalah mengatur PWM motor pada robot. Adapun aturan besar PWM disesuaikan dengan pembacaan sensor. Dimana nilai-nilai PWM motor ini digunakan sebagai target dalam proses pelatihan data pada Jaringan Syaraf Tiruan. Jika sensor 4 mendeteksi garis sedangkan sensor 1 tidak membaca, artinya posisi robot berada lebih disebelah kiri dari garis untuk itu motor kiri akan aktif atau kita mengatur PWM yang lebih besar dibanding PWM kanan sehingga robot akan berbelok ke kanan. Untuk penelitian ini digunakan range PWM dari 0 sampai 350 rpm.

HASIL DAN PEMBAHASAN

Kondisi input sensor dengan *output* PWM motor telah didapatkan, maka mulai melakukan perhitungan algoritma jaringan syaraf tiruan. Namun, terlebih dahulu besaran PWM ditansformasikan agar dapat dilakukan perhitungan algoritma jaringan syaraf tiruan dengan metode *backpropagation*.Setelah output PWM dikonversi maka proses pelatihan data dengan Jaringan Syaraf Tiruan sudah dapat dilakukan. Selain dari pelatihan data secara manual juga digunakan *software* Matlab untuk melakukan pelatihan data. Data input dan target untuk beberapa kondisi robot ditunjukkan pada Tabel 1.

Tabel 1. Data input dan target untuk pelatihan JST

S1	S2	S3	S4	T1	T2
0	0	0	1	150	0
0	0	1	0	100	10
0	0	1	1	150	30
0	1	0	0	10	100
0	1	0	1	95	100
0	1	1	0	350	350
0	1	1	1	350	350
1	0	0	0	0	150
1	0	0	1	350	350
1	0	1	0	70	95
1	0	1	1	150	30
1	1	0	0	30	150

1	1	0	1	30	150
1	1	1	0	0	150
1	1	1	1	350	350

Setelah dilakukan tahapan penentuan input untuk JST dari sensor serta penentuan nilai target dari PWM motor maka selanjutnya dilakukan pelatihan data dengan JST menggunakan algoritma *backpropagation*.

Terdapat perubahan atau konversi nilai target agar dapat digunakan untuk prosespelatihan data. Adapun nilai target setelah transformasi ditunjukkan pada Tabel 2.

Tabel 2. Hasil transformasi nilai target

S1	S2	S3	S4	T1	T2
0	0	0	1	0,44	0,10
0	0	1	0	0,33	0,12
0	0	1	1	0,44	0,17
0	1	0	0	0,12	0,33
0	1	0	1	0,32	0,26
0	1	1	0	0,90	0,90
0	1	1	1	0,90	0,90
1	0	0	0	0,10	0,44
1	0	0	1	0,90	0,90
1	0	1	0	0,26	0,32
1	0	1	1	0,44	0,17
1	1	0	0	0,17	0,44
1	1	0	1	0,17	0,44
1	1	1	0	0,10	0,44
1	1	1	1	0,90	0,90

Adapun cara mentransformasikan data target adalah sebagai berikut [2]:

- Data terbesar : 350
- Data Terkecil : 0

- Data 1 Target 1: 150

$$\frac{0{,}8(Data\ 1\ Target\ 1 - DataTerkecil)}{(DataTerbesar - DataTerkecil)} + 0{,}1$$

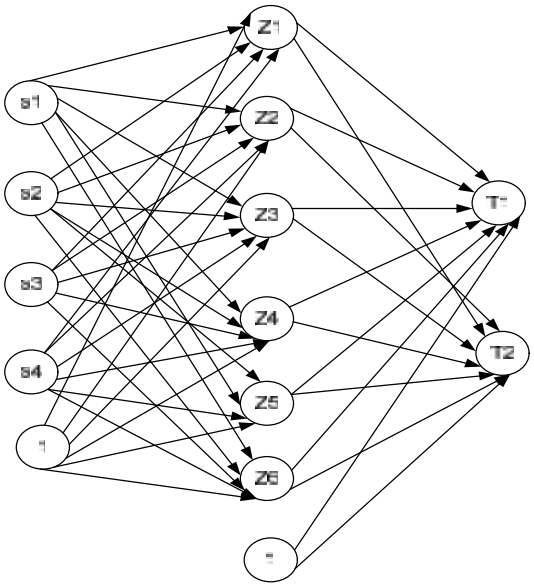
$$\frac{0{,}8(150-0)}{350-0} + 0{,}1 = 0{,}44$$

- Data 2 Target 1 : 100

$$\frac{0{,}8(100-0)}{350-0} + 0{,}1 = 0{,}33$$

Begitu seterusnya dilakukan proses transformasi data pada keseluruhan data target, baik target 1 maupun target 2.

Arsitektur jaringan syaraf tiruan yang digunakan seperti ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur Jaringan JST

Adapun ketentuan dari jaringan adalah:

- 1 input layer, 4 neuron
- 1 hidden layer, 6 neuron
- 1 output layer dengan 2 neuron
- Learning rate = 0,1
- Maksimum epoch = 5000
- Target error 0,02

1. Analisis Data Manual

Proses Training

Bobot - bobot awal sebagai berikut :

- Bobot awal input ke *hidden layer*, diperlihatkan pada Tabel 3.

Tabel 3. Bobot awal input ke *hidden layer*

Bobot (Vij)	Hidden Layer					
	1	2	3	4	5	6
V1	0.1	0.2	0.3	0.4	0.5	0.6
V2	0.1	0.2	0.3	0.4	0.5	0.6
V3	0.1	0.2	0.3	0.4	0.5	0.6
V4	0.1	0.2	0.3	0.4	0.5	0.6

- Bobot awal bias ke hidden layerditunjukkan pada Tabel 4.

Tabel 4. Bobot awal bias ke hidden layer

Bobot (Wk)	Hidden Layer	
	1	2
W1	0.1	0.1
W2	0.2	0.2
W3	0.3	0.3
W4	0.4	0.4
W5	0.5	0.5
W6	0.6	0.6

- Bobot awal hidden ke output layer ditunjukkan pada Tabel 5.

Tabel 5. Bobot awal hidden ke output layer

Bias (V0)	Hidden Layer					
	1	**2**	**3**	**4**	**5**	**6**
V0	0.1	0.2	0.3	0.4	0.5	0.6

- Bobot awal bias ke output layerdiperlihatkan pada Tabel 6.

Tabel 6. Bobot awal hidden ke output layer

Bobot	Nilai
W01	0.1
W02	0.2

Epoch 1

Data 1 :

S1 = 0 ; S2 = 0 ; S3 = 0 ; S4 = 1

T1 = 0,44 T2 = 0,10

Perambatan Maju

- Operasi pada hidden layer:

$Z_{in_1} = V_{01} + V_{11}.S1 + V_{21}.S2 + V_{31}.S3 + V_{41}.S4$

$= 0,1 + 0,1.0 + 0,1.0 + 0,1.0 + 0,1$

$= 0,2$

- Fungsi aktivasi:

$Z_1 = \frac{1}{1+e^{-Zin_1}} = \frac{1}{1+e^{-0,2}} = 0,5524$

- Operasi pada output layer

$Y_{in_1} = W_{01} + W_{11}.Z_1 + W_{21}.Z_2 + W_{31}.Z_3 + W_{41}.Z_4 + W_{51}.Z_5 + W_{41}.Z_4$

$= 0,1 + 0,1.0,5524 + 0,2.0,5988 + 0,3.0,6451 + 0,4.0,6896 + 0,5.0,7299 + 0,6.0,7692$

$= 1,57084$

$Y_2 = \frac{1}{1+e^{-Yin_2}} = \frac{1}{1+e^{-1,67084}} = 0,8403$

- Fungsi aktivasi:

$Y_1 = \frac{1}{1+e^{-Yin_1}} = \frac{1}{1+e^{-1,57084}} = 0,8264$

$Y_2 = \frac{1}{1+e^{-Yin_2}} = \frac{1}{1+e^{-1,67084}} = 0,8403$

- Cek error

Error 1 = $T_1 - Y_1$ = 0,44 – 0,8264 = - 0,3864

Error 2 = $T_2 - Y_2$ = 0,10 – 0,8403 = - 0,7403

- Kuadrat Error

MSE 1 = $(-0,3864)^2$ = 0,15

MSE 2 = $(-0,7403)^2$ = 0,55

Saat MSE lebih besar dari target *error*, yang diinginkan maka dilakukan proses propoagasi mundur (perambatan balik). Proses ini digunakan untuk memperbarui nilai bobot yang sudah diinisialisasi. Bobot yang sesuai akan menghasilkan *error* yang kecil [3].

Perambatan Balik

- *Error 1[4]:*

$\delta_1 = T_1 - Y_1 x(\frac{1}{1+e^{-Yin_1}})x [1-(\frac{1}{1+e^{-1.57084}})] = -0,05543$

$\Delta W_{11} = \alpha \times \delta_1 x Z_1 = -0,003061$

$\Delta W_{01} = \alpha \times \delta_1 = 0,1 \times -0,05543 = -0,005543$

$\delta_{in11} = \delta \times W_{11} = -0,05543 \times 0,1 = -0,005543$

$\delta_{11} = \delta_{in11} x (\frac{1}{1+e^{-zin_1}})x [1-(\frac{1}{1+e^{-zin_1}})]$

$= \delta_{in11} x (\frac{1}{1+e^{-0,2}})x [1-(\frac{1}{1+e^{-0,2}})] = -0,001370$

$\Delta V_{111} = \alpha \times \delta_{11} \times S_1 = 0$

$\Delta V_{211} = \alpha \times \delta_{11} \times S_2 = 0$

$\Delta V_{311} = \alpha \times \delta_{11} \times S_3 = 0$

$\Delta V_{411} = \alpha \times \delta_{11} \times S_4 =$

-0,000137

$\Delta V_{011} = \alpha \times \delta_{11} = -\ 0{,}00001370$

- *Error 2[4]:*

$\delta_2 = T_2 - Y_2 x(\frac{1}{1+e^{-Yin_2}}) x\ [1-(\frac{1}{1+e^{-1{,}67084}})] = -0{,}09934$

$\Delta W_{12} = \alpha \times \delta_2 x\ Z_1 = -0{,}005487$

$\Delta W_{02} = \alpha \times \delta_2 = 0{,}1 \times -0{,}09934 = -0{,}009934$

$\delta_{in12} = \delta \times W_{21} = -0{,}09934 x\ 0{,}1 = -0{,}00934$

$\delta_{12} = \delta_{in12} \times (\frac{1}{1+e^{-zin_1}}) x\ [1-(\frac{1}{1+e^{-zin_1}})] = \delta_{in12} \times (\frac{1}{1+e^{-0{,}2}}) x\ [1-(\frac{1}{1+e^{-0{,}2}})] = -0{,}002309$

$\Delta V_{112} = \alpha \times \delta_{12} \times S_1 = 0$

$\Delta V_{212} = \alpha \times \delta_{12} \times S_2 = 0$

$\Delta V_{312} = \alpha \times \delta_{12} \times S_3 = 0$

$\Delta V_{412} = \alpha \times \delta_{12} \times S_4 = -0{,}0002309$

$\Delta V_{012} = \alpha \times \delta_{12} = -0{,}0002309$

Tahap perubahan bobot dan bias [5]:

$V_{11}(baru) = V_{11}(lama) + \Delta V_{111} + \Delta V_{112} = 0{,}1$

$V_{01}(baru) = V_{01}(lama) + \Delta V_{011} + \Delta V_{012} = 0{,}1 + (-0{,}00001370) + (-0{,}0002309) = 0{,}0996$

$W_{11}(baru) = W_{11}(lama) + \Delta W_{11} = 0{,}1 + (-0{,}003061) = 0{,}0969$

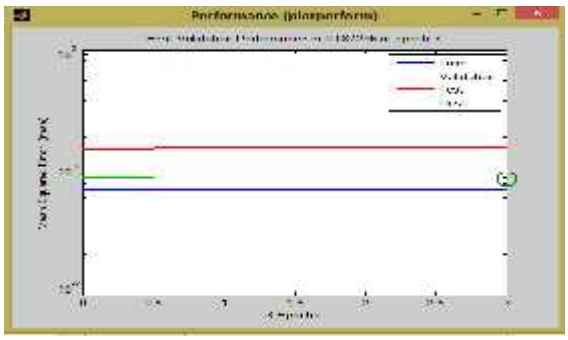
$W_{01}(baru) = W_{01}(lama) + \Delta W_{01} = 0{,}1 + (-0{,}005543) = 0{,}0944$

$W_{02}(baru) = W_{02}(lama) + \Delta W_{02} = 0{,}2 + (-0{,}009934) = 0{,}1900$

Perhitungan algoritma Jaringan Syaraf Tiruan diatas adalah *epoch* 1 jaringan dan hanya untuk data 1. Perhitungan pelatihan data dilakukan sampai kuadrat *error* ≤0,02 atau jika *epoch*-nya memenuhi 5000.

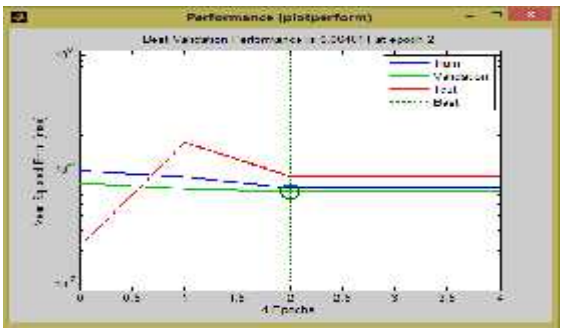
2. Analisis Data Dengan Matlab

Selain pelatihan data secara manual digunakan juga program matlab untuk menentuka nilai *output* yang mendekati target atau memiliki nilai *error* yang rendah. Berikut merupakan hasil *perform* dari masing-masing input dengan 2 *output*. Dimana terdapat 8 *performance*. Berikut merupakan grafik dari hasil *performace* pelatihan data. Gambar 2 menunjukkan hasil latih S1 dan T1.



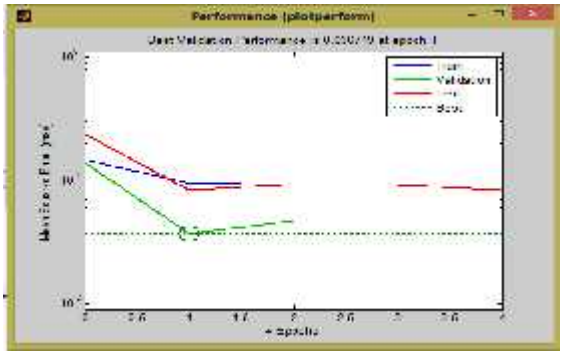
Gambar 2. Performance S1 dengan T1

Hasil latih S2 dengan T1 diperlihatkan pada Gambar 3.



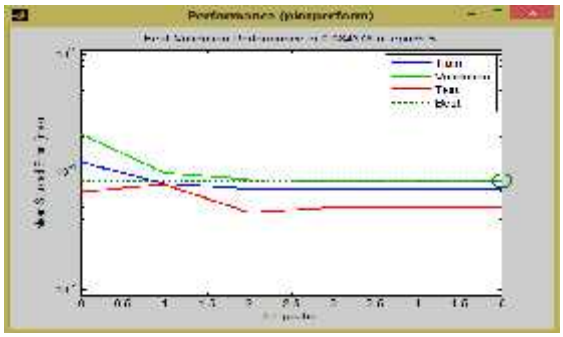
Gambar 3. Performance S2 dengan T1

Hasil latih S3 dengan T1 diperlihatkan pada Gambar 4.



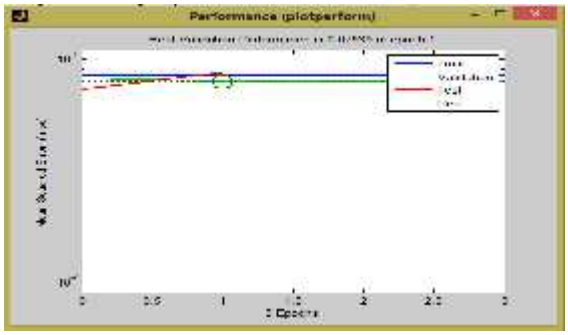
Gambar 4. Performance S3 dengan T1

Gambar 5 menunjukkan hasil latih S4 dan T1.



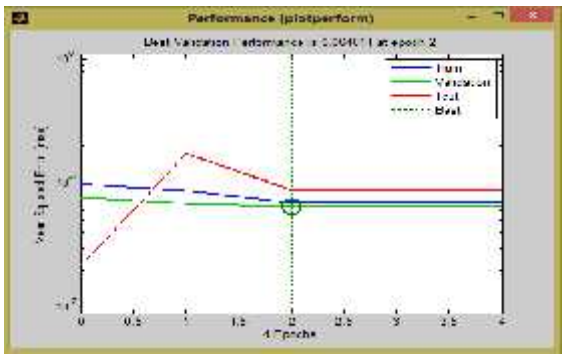
Gambar 5. Performance S4 dengan T1

Hasil latih S1 dengan T2 diperlihatkan pada Gambar 6.



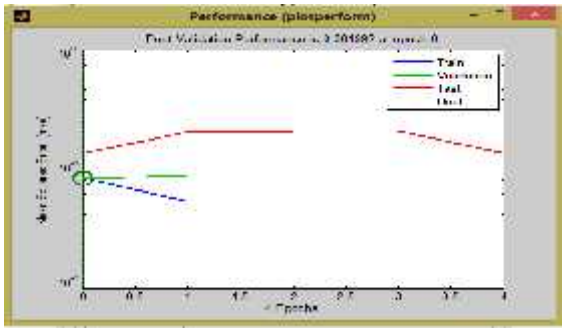
Gambar 6. Performance S1 dengan T2

Gambar 7 menunjukkan hasil latih S2 dan T2.



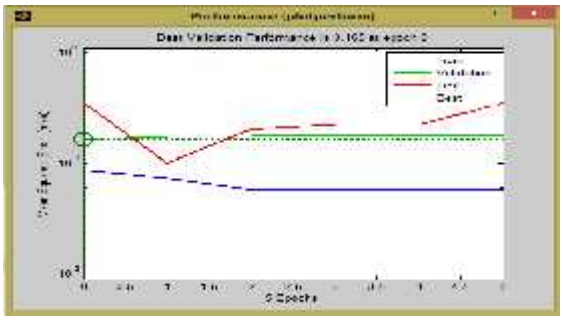
Gambar 7. Performance S2 dengan T2

Hasil latih S3 dengan T2 diperlihatkan pada Gambar 8.



Gambar 8. Performance S3 dengan T2

Hasil latih S4 dengan T2 diperlihatkan pada Gambar 9.



Gambar 9. Performance S4 dengan T2

Berdasarkan pelatihan data dengan menggunakan matlab didapatkan hasil 8 performance, yaitu : S1 dengan T1, S2 dengan T1, S3 dengan T1, S4 dengan T1, S1 dengan T2, S2 dengan T2, S3 dengan T2,serta S4 dengan T2. Dari kedelapan hasil *permorfance* merupakan tampilan *performance* terbaik dimana nilai *validation* mendekati target. Dan untuk setiap *performance* dihasilkan dari *epoch* yang berbeda beda.

KESIMPULAN

Berdasarkan percobaan pelatihan data dengan jaringan syaraf tiruan dimana arsitektur jaringan terdiri dari 1 *input layer* dengan 4 *neuron*, 1 *hidden layer* dengan 6 *neuron*, serta 1 *output layer* dengan 2 *neuron*. Didapatkan hasil dari pelatihan data secara manual pada *epoch* 1 untuk data 1. Dimana setelah melakukan tahap perambatan maju didapatkan nilai *kuadrat error* (MSE) yang lebih besar dari 0,02. Sehingga diperlukan tahap perambatan mundur.Setelah tahap perambatan mundur dilakukan lagi tahap perubahan bobot dan bias. Bobot dan bias baru ini kemudian digunakan untuk pelatihan data kedua. Pelatihan terus dilakukan untuk seluruh data sampai memenuhi nilai MSE yang ≤0,02 atau *epoch* maksimal 5000.

Secara keseluruhan dengan menggunakan Jaringan Syaraf Tiruan dapat menghasilkan nilai data terlatih yang sudah mendekati target.

SARAN

Sebagai pengembangan selanjutnya analisis sistem kontrol pada robot LFR dengan jaringan syaraf tiruan dapat menggunakan algoritma selain backpropagation atau dengan metode hyrid agar dapat dihasil performance robot yang lebih baik.

UCAPAN TERIMA KASIH

Ucapan terima kasih kepada P3M Politeknik Negeri Balikpapan yang telah memberikan pendanaan dalam pelaksanaan penelitian dan tim pelaksana penelitian sehingga penelitian ini berjalan baik dan lancar.

DAFTAR PUSTAKA

[1] A. Sholahuddin, S. Hadi. 2013. Penerapan Jaringan Syaraf Tiruan Pada Pengenalan Pola Robot Line Follower. Prosiding Seminar Nasional Sains dan Teknologi Nuklir PTNBR-BATAN Bandung: 381-386.

[2] N. Yanti, F.Z. Rachman. 2016. Neural Network Application For The Analysis Of The Nutrition Andenvironment Effect To Mirobial Growth Rate On Fermented Soybean Patty (Tempe) Fermentation. JTT Vol.4 no.2: 100-106

[3] R.A. Firmansyah, T.Odinanto. 2017. Algoritma Pencarian Jalur Terpendek Menggunakan Jaringan Syaraf Tiruan Pada Aplikasi Robot Penyelamat Kebakaran. TELCOMATICS, Vol.2 no.2: 1-7.

[4] Kusumadewi, D. 2004. Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB dan Excel Link. Graha Ilmu. Yogyakarta: 93-198

[5] T.Sutojo, E.Mulyanto, V.Suhartono. 2011. Kecerdasan Buatan. ANDI Yogyakarta: 362-374.

.