

## **IMPLEMENTASI PROSES *LEARNING* DAN *RUNNING* SISTEM KECERDASAN BUATAN (JARINGAN SYARAF TIRUAN) BERBASIS MIKROKONTROLER**

**Joko Subur**

Program Studi Teknik Elektro, Universitas Hang Tuah Surabaya  
joko.subur@hangtuah.ac.id

**Abstrak:** Kecerdasan buatan (*Artificial Intelligent*) merupakan sistem kecerdasan yang dibuat oleh manusia dengan menggunakan pengolahan komputasi menggunakan komputer. Salah satu sistem kecerdasan buatan yang biasa diaplikasikan adalah kecerdasan buatan Jaringan Syaraf Tiruan. Proses komputasi pada jaringan syaraf tiruan pada umumnya menggunakan media komputer, dikarenakan saat proses komputasi tersebut diperlukan ruang memori yang cukup besar. Mikrokontroler adalah sebuah chip mikroprosesor yang memiliki ruang memori yang terbatas. Sehingga proses komputasi kecerdasan buatan tidak dapat sepenuhnya dijalankan di sistem mikrokontroler. Pada penelitian ini sudah dapat merancang dan membangun sebuah sistem mikrokontroler yang mampu menjalankan proses komputasi kecerdasan buatan jaringan syaraf tiruan sepenuhnya, yaitu proses *learning* dan *running*. Dengan menambahkan memori eksternal berupa SD Card pada sistem mikrokontroler, untuk menyimpan parameter pengaturan yang digunakan serta untuk menyimpan nilai bobot akhir dari proses *learning* dan mengambil data bobot akhir untuk digunakan proses *running*. Hasil ujicoba sistem kecerdasan buatan yang dibuat memiliki tingkat akurasi sebesar 71,42 %, untuk implementasi mengenali jenis warna objek.

**Kata kunci:** Kecerdasan buatan, Jaringan syaraf tiruan, mikrokontroler, proses komputasi, memori, bobot.

### **PENDAHULUAN**

Kecerdasan buatan (*Artificial Intelligence* atau AI) didefinisikan sebagai kecerdasan yang ditunjukkan oleh suatu entitas buatan. Kecerdasan diciptakan dan dimasukkan ke dalam suatu mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan oleh manusia. Salah satu sistem kecerdasan buatan adalah jaringan syaraf tiruan (*Artificial Neural Network*) atau biasa disingkat JST.

Jaringan syaraf tiruan merupakan salah satu representasi buatan dari otak manusia yang selalu mencoba untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan disini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran [Kusumadewi, 2006]. Seperti halnya otak manusia, jaringan syaraf juga terdiri dari beberapa neuron, dan ada hubungan antara neuron-neuron tersebut. Neuron-neuron tersebut akan mentransformasikan informasi yang diterima melalui sambungan keluarnya menuju ke neuron-neuron yang lain. Pada jaringan syaraf tiruan penghubung antar neuron tersebut dikenal dengan nama bobot. Nilai bobot itu yang nantinya menentukan hasil keluaran dari proses komputasi system JST. Dimana nilai bobot optimal dihasilkan dari proses pembelajaran (*learning*) terlebih dahulu. Jadi sebelum JST dijalankan (*running*) untuk proses suatu data, maka pada JST tersebut dilakukan proses *learning* terlebih dahulu, proses ini bertujuan untuk mendapatkan nilai bobot yang optimal pada tiap-tiap layer. Salah satu metode *learning* yang digunakan dalam sistem JST adalah *backpropagation*.

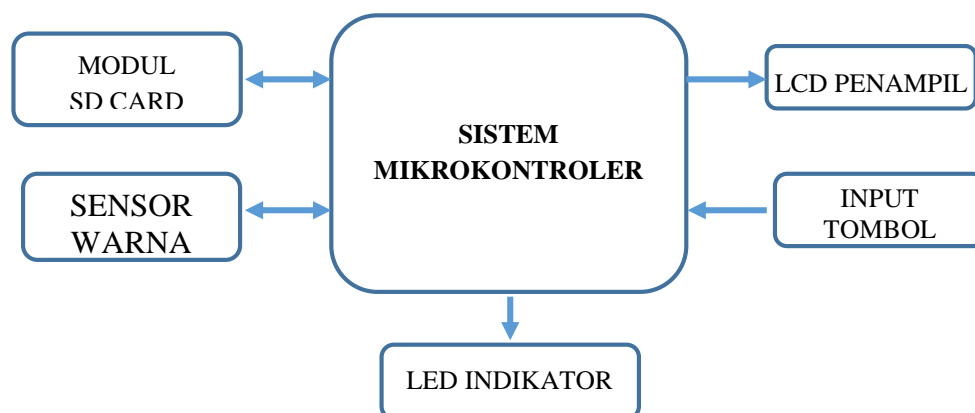
Pada umumnya proses perhitungan pada sistem JST dibuat dan dijalankan pada media komputer atau *Personal Computer* (PC), dikarenakan saat proses komputasi *learning* pada sistem jaringan syaraf tiruan tersebut diperlukan ruang (*space*) *memory Random Access Memory* (RAM) yang cukup besar. Dimana dari hasil proses *learning* akan dihasilkan nilai nilai bobot yang selanjutnya digunakan untuk perhitungan saat proses *running*. Nilai bobot merupakan suatu variable yang memerlukan ruang memori untuk penyimpanan, adapun nilai bobot yang dihasilkan dari proses *learning* bisa banyak sekali tergantung jumlah neuron yang digunakan, sehingga diperlukan ruang memori yang besar. Dikarenakan perlunya ruang memori yang cukup besar maka proses *learning* pada sistem jaringan syaraf tiruan yang dibuat akan kesulitan apabila dijalankan langsung pada media mikrokontroler secara langsung. Mikrokontroler merupakan sebuah chip IC (*Integrated Circuits*) yang didalamnya terdapat mikroprosesor dan dilengkapi dengan memori serta jalur port untuk komunikasi dengan perangkat lain.

Beberapa penelitian yang sudah dilakukan menggunakan sistem jaringan syaraf tiruan berbasis mikrokontroler secara *off-line*. Seperti yang dilakukan oleh [Jaka Prayudha, dkk, 2017] membuat sistem kecerdasan berbasis arduino, namun proses *learning* dilakukan terlebih dahulu menggunakan PC, setelah didapat nilai nilai bobot optimal, kemudian nilai bobot tersebut dimasukkan pada program editor mikrokontroler. Sehingga nilai nilai bobot tersebut bersifat statis atau tetap, dan akan berakibat sistem jaringan syaraf tiruan yang dibuat bersifat kaku. Begitu juga yang dilakukan oleh [Setya Ardhi, 2015] dalam membuat sistem JST untuk sistem kontrol umpan balik pada *buck converter switching power supply*, masih bersifat *off-line*.

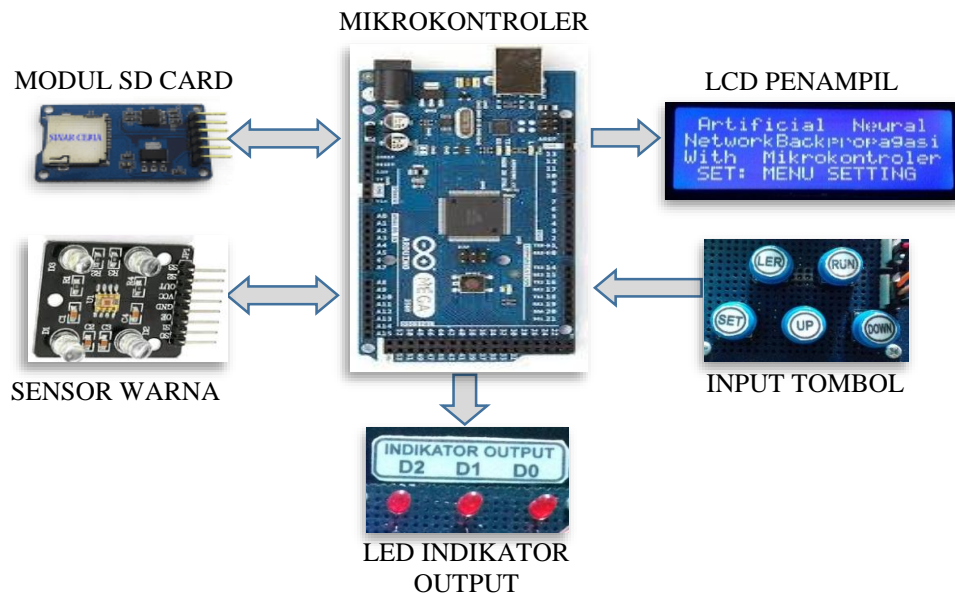
Berdasarkan permasalahan tersebut pada penelitian ini dirancang dan membangun penerapan proses *learning* dan *runing* pada sistem kecerdasan buatan (jaringan syaraf tiruan) berbasis mikrokontroler, solusi untuk menyediakan ruang memori yang besar, akan disediakan memori tambahan diluar yaitu kartu memori SD Crad. Dari penelitian ini didapatkan suatu cara bagaimana memaksimalkan kinerja dari mikrokontroler untuk proses sistem jaringan syaraf tiruan yang meliputi proses *learning* dan *running*. Untuk pengujian implementasi sistem yang dibuat, diujicoba pada sistem untuk mengenali warna kertas yang didekatkan pada sensor. Pengerjaan penelitian ini dilakukan di Laboratorium Elektronika Prodi Teknik Elektro Fakultas Teknik dan Ilmu Kelautan, Universitas Hang Tuah Surabaya.

## METODE PENELITIAN

Dari hasil studi literatur dan penelitian yang sudah dilakukan maka telah didapatkan sebuah rancangan blok sistem sistem mikrokontroler untuk bisa ditanamkan sistem kecerdasan buatan (jaringan syaraf tiruan), seperti terlihat pada Gambar 1.



**Gambar 1.** Diagram blok sistem Jaringan Syaraf Tiruan berbasis Mikrokontroler



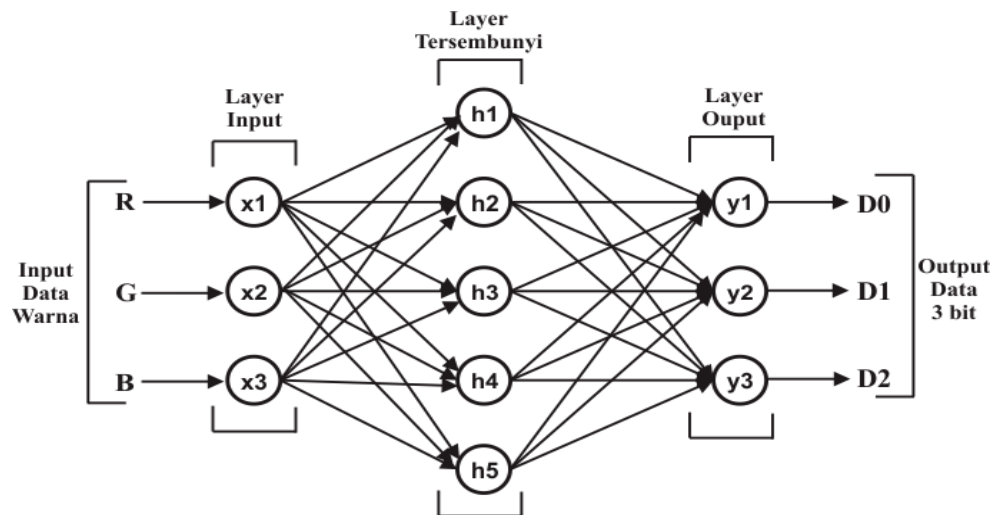
**Gambar 2.** Realisasi dalam bentuk hardware

Uraian penjelasan dari gambar diagram blok sistem diatas adalah sistem utama berada pada mikrokontroler, yang dihubungkan dengan input berupa modul SD Card, sensor warna TCS3200 dan input tombol. Sedangkan untuk output terhubung ke LCD penampil dan LED indikator.

Mikrokontroler yang digunakan adalah mikrokontroler tipe Arduino Mega 2560, dengan alasan arduino tipe tersebut memiliki kapasitas memori lebih besar jika dibandingkan dengan tipe Uno maupun Nano [Heri Andrianto, dkk. 2016], dengan harapan dapat menampung program yang lebih banyak. Namun kapasitas memori tersebut masih belum mampu untuk menampung data saat proses sistem JST, oleh karena itu ditambahkan memori eksternal berupa SD card. Dimana memori SD card tersebut sebagai media penyimpan nilai parameter pengaturan dan juga menyimpan nilai bobot yang digunakan pada proses sistem JST. Data input untuk proses sistem JST didapat dari sensor warna, dimana data yang dihasilkan dari sensor tersebut adalah nilai intensitas dari warna objek yang di *sensing*, yaitu parameter nilai R (*Red*), G (*Green*) dan B (*Blue*). Nilai RGB dari intensitas warna tiap objek dikumpulkan dan dipolakan untuk dijadikan data input saat proses *learning* dan saat proses *running* pada sistem JST yang dibuat. Untuk pengaturan parameter pada sistem JST digunakan tombol input, dan proses dari sistem JST yang dibuat dapat dipantau atau diamati pada media penampil LCD, dan hasil identifikasi warna saat proses *running* dari sistem JST ini direpresentasikan dalam kode bilangan biner yang ditunjukkan pada penyalan led indikator data output.

Sebelum dilakukan proses *running* pada jaringan syaraf tiruan (JST), maka pada JST harus dilakukan proses *learning* (pembelajaran) terlebih dahulu. Proses pembelajaran dilakukan dengan memasukkan data input yang telah didapat dari proses pembacaan data sensor input beserta data target yang diinginkan dari masing-masing data pelatihan tersebut. Proses pembelajaran dilakukan terus-menerus sampai didapat nilai *Mean Square Error (MSE)* lebih kecil dari nilai *error minimum* yang telah ditetapkan. Ketika nilai *MSE* masih lebih besar dari nilai *error minimum* maka dilakukan proses mundur (*backward*) dari *layer output* ke *layer input* untuk merubah nilai bobot pada tiap *neuron* pada masing-masing layer. Kemudian dilakukan proses maju kembali (*forward*) dan dihitung kembali nilai *MSE*-nya. Ketika nilai *MSE* sudah lebih kecil dari nilai *error minimum* maka proses pelatihan akan berhenti, dan hasil dari proses pelatihan tersebut didapatkan nilai-nilai bobot tiap-tiap *neuron*.

Berikut ini struktur topologi Jaringan Syaraf Tiruan (JST) yang dibuat, yang terdiri dari: Layer Input (3 *neuron*), layer tersembunyi (5 *neuron*) dan layer output (3 *neuron*). Untuk lebih jelasnya dapat diamati pada Gambar 3.



**Gambar 3.** Topologi struktur jaringan JST yang digunakan

Gambar topologi struktur jaringan JST pada Gambar 3 dapat diuraikan dengan penjelasan sebagai berikut:

- Pada topologi tersebut terdapat 3 layer, yaitu layer input, layer tersembunyi dan layer output.
- Pada layer input terdapat 3 node neuron, jumlah 3 neuron tersebut disesuaikan dengan jumlah parameter inputnya, dimana ada 3 parameter input dari sensor warna yang dipakai yaitu parameter nilai R, G dan B dari hasil pembacaan sensor warna terhadap objek sebuah warna.
- Pada layer tersembunyi ditentukan sebanyak 5 node neuron, jumlah neuron pada layer tersembunyi sebenarnya bebas tidak ditentukan. Namun jumlah neuron pada layer akan mempengaruhi kapasitas memori RAM pada mikrokontroler yang dipakai saat proses komputasi. Semakin banyak jumlah neuron yang dipakai maka akan semakin banyak juga jumlah memori yang digunakan, sehingga bisa berpengaruh dengan performa proses pembelajaran (*learning*).
- Kalau tadi kita amati pada Gambar 3 dimana pada tiap layer terdapat node neuron-neuron, dan neuron-neuron tersebut dihubungkan dengan sebuah garis antar neuron lain pada layer sebelum atau sesudahnya. Perlu diketahui bahwa garis-garis penghubung antar neuron tersebut adalah representasi dari sebuah nilai bobot atau *weight*. Nilai bobot didapat dari proses komputasi saat proses pembelajaran (*learning*) dari sistem Jaringan Syaraf Tiruan, dimana dari proses pembelajaran tersebut akan didapat nilai bobot yang optimal (*konvergen*) untuk nantinya digunakan dalam proses saat dijalankan (*running*) sistem Jaringan Syaraf Tiruan dalam hal ini untuk aplikasi pengenalan jenis warna.
- Pada layer output ditentukan 3 node neuron, jumlah 3 neuron tersebut diibaratkan 3 bit data. Maka dari 3 bit data tersebut dapat terbentuk 8 variasi kombinasi, dari data  $000_b - 111_b$  (bilangan biner). Karena bisa dihasilkan data output sebanyak 8 variasi maka bisa digunakan untuk penentuan 8 jenis warna, dimana setiap jenis warna diwakili 1 jenis variasi. Penentuan jenis 1 variasi dalam mewakili setiap 1 warna tersebut dalam sistem JST ini pada proses pembelajaran nantinya disebut target. Penentuan target dalam proses pembelajaran untuk tiap jenis warna dapat dilihat pada Tabel 1.

**Tabel 1.** Penentuan data target tiap jenis warna untuk proses pembelajaran

Jenis Warna	Data Target		
	Y3 (D2)	Y2 (D1)	Y1 (D0)
Hitam	0	0	1
Coklat	0	1	0
Merah	0	1	1
Kuning	1	0	0
Hijau	1	0	1
Biru	1	1	0
Putih	1	1	1

Dari proses pembelajaran didapat nilai bobot optimal tiap neuron dalam setiap layer. Nilai bobot tersebut yang nantinya akan digunakan untuk melakukan perhitungan pada proses *running* JST dalam melakukan pembacaan data dan pengambilan keputusan berdasarkan kondisi dari data yang masuk. Nilai bobot optimal yang dihasilkan dari proses *learning* juga perlu disimpan dimemori untuk nanti bisa diambil kembali dan digunakan untuk proses *running*. Media penyimpanan data memori eksternal yang digunakan adalah memori SD Card.

Pada memori eksternal SD Card data nilai bobot disimpan dalam bentuk file extension \*.txt. Sehingga dapat dipastikan bahwa tipe data yang digunakan untuk menyimpan data nilai bobot tersebut adalah tipe data *String*, sedangkan nilai bobot akhir yang dihasilkan dari proses *learning* adalah tipe data bilangan pecahan (*float*). Oleh karena itu maka untuk menyimpan data nilai bobot akhir tersebut maka perlu proses konversi dari tipe data bilangan *float* menjadi tipe data *String*. Begitu juga saat nilai bobot akhir yang tersimpan pada memori eksternal tersebut akan digunakan pada proses sistem JST, maka perlu juga dilakukan konversi dari tipe data *String* menjadi bilangan tipe data *float*.

Pengujian awal dilakukan dengan membaca nilai intensitas warna R, G dan B dari objek yang akan dibaca menggunakan sensor warna. Dalam pengujian ini objek yang dibaca adalah kertas karton dengan warna warna yang berbeda, yaitu: hitam, coklat, merah, kuning, hijau, biru, dan putih. Nilai data hasil pembacaan sensor yang dilakukan harus dapat disimpan di memori *eksternal*, dalam hal ini digunakan modul memori SD Card sebagai media penyimpanan nilai data intensitas tiap warna.

Data nilai intensitas warna R, G dan B yang tersimpan pada memori eksternal SD card tersebut nantinya akan dibaca/diambil untuk dijadikan nilai data input pada proses pembelajaran. Data nilai intensitas tiap warna yang tersimpan pada SD card disimpan pada sebuah file dengan tipe \*.txt (text). Perlu diketahui bahwa data nilai RGB merupakan nilai dengan rentang nilai bilangan bulat (*byte*) dengan jangkauan antara 0-255, namun ketika disimpan pada SD card pada file tipe \*.txt, maka nilai dengan bilangan bulat tersebut dirubah terlebih dahulu dalam bentuk *char* (karakter). Sehingga ketika data dibaca/diambil kembali untuk proses JST maka nilai RGB yang dalam bentuk *char* tersebut harus diubah/dikonversi terlebih dahulu menjadi bentuk nilai bilangan, tanpa melalui proses konversi bilangan maka proses JST tidak bisa jalan dengan baik.

Dari hasil pengujian yang dilakukan, sistem sudah berhasil menyimpan data nilai RGB tiap warna sesuai nama file yang sudah ditentukan pada memori eksternal SD card dengan baik pada tiap nama file. Sistem yang dibuat juga sudah mampu membaca data nilai RGB yang tersimpan pada memori eksternal SD card pada file file data nilai RGB sesuai warna yang ditentukan. Pada tabel 2 merupakan data nilai intensitas warna R, G dan B pada warna warna yang diambil sample datanya Meliputi data warna: Hitam, Coklat, Merah, Kuning, Hijau, Biru dan Putih. Dimana tiap sample warna diambil data sebanyak 10 sample data, tiap sample data terdiri dari parameter intensitas warna R, G dan B.

**Seminar Nasional Kelautan XIV**

" Implementasi Hasil Riset Sumber Daya Laut dan Pesisir Dalam Peningkatan Daya Saing Indonesia"  
 Fakultas Teknik dan Ilmu Kelautan Universitas Hang Tuah, Surabaya 11 Juli 2019

**Tabel 2.** Data nilai RGB intensitas warna hasil pembacaan sensor warna TCS3200

Data ke	R	G	B
warna HITAM			
1	191	198	156
2	191	198	156
3	191	198	156
4	191	198	155
5	191	198	156
6	191	198	156
7	190	197	156
8	190	198	156
9	190	198	157
10	190	198	156
warna COKLAT			
1	111	143	132
2	111	142	132
3	111	143	132
4	111	143	132
5	111	142	132
6	111	143	132
7	111	143	132
8	110	141	132
9	110	141	131
10	109	140	131
warna MERAH			
1	80	189	148
2	80	189	148
3	81	189	148
4	80	188	148
5	80	188	148
6	80	189	147
7	80	187	147
8	80	188	147
9	80	188	147
10	80	186	147
warna KUNING			
1	54	64	88
2	54	65	88
3	54	65	88
4	53	65	88
5	53	65	88
6	53	65	88
7	53	65	88
8	53	65	88
9	53	65	88
10	53	65	88

**Seminar Nasional Kelautan XIV**

" Implementasi Hasil Riset Sumber Daya Laut dan Pesisir Dalam Peningkatan Daya Saing Indonesia"  
 Fakultas Teknik dan Ilmu Kelautan Universitas Hang Tuah, Surabaya 11 Juli 2019

Data ke	R	G	B
warna HIJAU			
1	127	98	103
2	127	98	103
3	127	99	103
4	127	97	102
5	126	98	102
6	126	98	102
7	126	98	102
8	126	98	102
9	126	99	102
10	125	98	102
warna BIRU			
1	135	112	69
2	135	112	69
3	135	112	69
4	135	112	69
5	135	113	69
6	136	113	70
7	135	111	70
8	135	112	70
9	135	112	70
10	135	112	70
warna PUTIH			
1	49	50	39
2	49	50	39
3	48	50	39
4	48	50	39
5	48	50	39
6	48	50	39
7	48	49	39
8	49	49	39
9	49	50	39
10	49	50	39

Sample data pada Tabel 2 tersebut dijadikan data input pada proses *learning* pada sistem JST. Dimana nilai parameter R, G dan B menjadi data input neuron pada layer input, seperti yang terlihat pada topologi jaringan JST yang digunakan seperti yang terlihat pada Gambar 3. Dan pada saat proses *learning* tiap warna akan diberi target sesuai dengan target yang sudah ditentukan pada Tabel 1.

Sebelum dijalankan proses *learning* perlu dilakukan pengaturan (*setting*) penentuan nilai parameter yang digunakan pada proses pembelajaran. Nilai parameter tersebut meliputi: banyak pola data, banyak data tiap pola, banyak neuron pada hidden layer, nilai laju pembelajaran (*learning rate*), *momentum* dan penentuan nilai minimal *error* yang diinginkan. Nilai parameter tersebut yang mempengaruhi kecepatan proses pembelajaran untuk mendapatkan kondisi konvergen. Kondisi konvergen adalah ketika proses *learning* pada JST nilai *error* sudah lebih

kecil dari nilai *minimal error* yang kita tentukan, dan sudah didapat nilai bobot bobot optimal tiap neuron yang terhubung pada tiap layer.

**Tabel 3.** Keterangan nilai parameter pengaturan proses pembelajaran

Nilai	Keterangan Parameter
7	Jumlah jenis pola data input
10	Banyak sample data tiap jenis pola
5	Jumlah neuron pada layer tersembunyi
0.9	Nilai laju pembelajaran
0.75	Nilai momentum (update bobot)
0.001	Nilai minimal error yang diinginkan

Pada Tabel 3 merupakan uraian keterangan parameter nilai yang tersimpan pada hasil seting. Dimana nilai nilai parameter tersebut akan digunakan dalam proses pembelajaran, yaitu untuk menentukan seberapa banyak pola data input yang digunakan, seberapa banyak sample data tiap jenis pola data dan berapa neuron yang terdapat pada layer tersembunyi. Hal ini untuk menentukan data larik (*array*) dari nilai bobot yang akan digunakan dalam proses pembelajaran. Untuk nilai laju pembelajaran dan nilai momentum adalah nilai parameter yang mempengaruhi kecepatan proses pembelajaran untuk mendapatkan kondisi konvergen.

Hasil dari proses pembelajaran yang berhasil mencapai kondisi konvergen didapatkan nilai bobot yang optimal, nilai bobot optimal tersebut adalah nilai bobot akhir yang digunakan sebagai penghubung antar neuron pada setiap layer. Selain didapat nilai bobot optimal, juga didapatkan nilai bias optimal, berbeda dengan nilai bobot yang digunakan sebagai nilai penghubung antar neuron, untuk nilai bias bukan sebagai nilai penghubung antar neuron, akan tetapi digunakan sebagai nilai penimbang sebelum proses aktivasi pada masing masing neuron di layer tersembunyi dan layer output. Pada tabel 4 berikut adaah hasil nilai bobot dan nilai bias yang didapat dari proses pembelajaran.

**Tabel. 4** Hasil nilai bobot dan nilai bias optimal

		Layer Tersembunyi				
		Neuron 1 (h1)	Neuron 2 (h2)	Neuron 3 (h3)	Neuron 4 (h4)	Neuron 5 (h5)
Layer Input	Neuron 1 (x1)	-20.918587000000	-3.583618600000	2.883760700000	-2.665881400000	-0.199999990000
	Neuron 2 (x2)	12.279148000000	1.237825800000	-0.877337750000	-0.232536760000	-0.410000000000
	Neuron 3 (x3)	-0.099999994000	0.149999980000	0.420000020000	0.839215700000	0.870588240000
Layer Output	Neuron 1 (y1)	-20.657904000000	-18.307606000000	11.012403000000	-16.788485000000	0.081132166000
	Neuron 2 (y2)	-20.566587000000	23.495193000000	-13.922636000000	30.139328000000	0.246398370000
	Neuron 3 (y3)	-0.210000010000	-0.470000000000	0.069999993000	0.490000010000	0.232547970000
Bias		0.000075332246	-0.000261362760	0.000146170470	-0.000315230400	0.000000000000
		Layer Output				
Bias	Neuron 1 (y1)	Neuron 2 (y2)		Neuron 3		
		-0.000000000000	-0.000045146746	0.000000000000		



**HASIL DAN PEMBAHASAN**

Untuk menguji sistem JST yang dibuat maka akan dilakukan percobaan proses running atau menjalankan sistem JST untuk mengenali input objek berupa kertas berwarna sesuai warna saat *learning*, kemudian meng analisa seberapa akurat sistem JST dalam mengenali warna objek input. Pada tabel 5 adalah hasil dari percobaan sistem JST yang dibuat untuk mendeteksi warna dari kertas yang dibaca oleh sensor.

**Tabel 5.** Data hasil percobaan sistem JST mendeteksi warna objek

No	Percobaan	Warna Objek (kertas)	Hasil Pembacaan Alat Sistem JST	Data Output (y3 y2 y1)	Keterangan
1	1	Hitam	Hitam	0 0 1	Benar
2	2	Hitam	Hitam	0 0 1	Benar
3	3	Hitam	Hitam	0 0 1	Benar
4	4	Hitam	Hitam	0 0 1	Benar
5	5	Hitam	Hitam	0 0 1	Benar
6	1	Cokelat	Hitam	0 0 1	Salah
7	2	Cokelat	Hitam	0 0 1	Salah
8	3	Cokelat	Hitam	0 0 1	Salah
9	4	Cokelat	Hitam	0 0 1	Salah
10	5	Cokelat	Hitam	0 0 1	Salah
11	1	Merah	Merah	0 1 1	Benar
12	2	Merah	Merah	0 1 1	Benar
13	3	Merah	Merah	0 1 1	Benar
14	4	Merah	Merah	0 1 1	Benar
15	5	Merah	Merah	0 1 1	Benar
16	1	Kuning	Putih	1 1 1	Salah
17	2	Kuning	Putih	1 1 1	Salah
18	3	Kuning	Putih	1 1 1	Salah
19	4	Kuning	Putih	1 1 1	Salah
20	5	Kuning	Putih	1 1 1	Salah
21	1	Hijau	Hijau	1 0 1	Benar
22	2	Hijau	Hijau	1 0 1	Benar
23	3	Hijau	Hijau	1 0 1	Benar
24	4	Hijau	Hijau	1 0 1	Benar
25	5	Hijau	Hijau	1 0 1	Benar
26	1	Biru	Biru	1 1 0	Benar
27	2	Biru	Biru	1 1 0	Benar
28	3	Biru	Biru	1 1 0	Benar
29	4	Biru	Biru	1 1 0	Benar
30	5	Biru	Biru	1 1 0	Benar
31	1	Putih	Putih	1 1 1	Benar
32	2	Putih	Putih	1 1 1	Benar
33	3	Putih	Putih	1 1 1	Benar
34	4	Putih	Putih	1 1 1	Benar
35	5	Putih	Putih	1 1 1	Benar

Jika diamati hasil uji coba yang tertera pada Tabel 5, dari data percobaan diatas sistem JST yang dibuat hanya mampu mengenali 5 warna yang benar, dari 7 warna yang dideteksi.

## Seminar Nasional Kelautan XIV

"Implementasi Hasil Riset Sumber Daya Laut dan Pesisir Dalam Peningkatan Daya Saing Indonesia"  
Fakultas Teknik dan Ilmu Kelautan Universitas Hang Tuah, Surabaya 11 Juli 2019

Terdapat 2 warna yang salah dari hasil pembacaan alat, yaitu warna Cokelat dibaca Hitam dan warna Kuning dibaca Putih.

Dengan demikian tingkat akurasi sistem yang diperoleh dapat dihitung sebagai berikut:

$$\begin{aligned} \text{Akurasi} &= \frac{\text{Jumlah benar}}{\text{Jumlah percobaan}} \times 100\% \\ &= \frac{35}{25} \times 100\% \\ &= 71,42 \%. \end{aligned}$$

Jadi tingkat akurasi sistem JST yang dibuat sebesar **71,42 %**

## KESIMPULAN

Sistem JST berbasis mikrokontroler yang dibuat hanya mampu memiliki tingkat akurasi sebesar 71,42 %. Hasil tingkat akurasi yang didapat kurang baik dikarenakan faktor hasil konversi tipe data nilai bobot dari *string* ke tipe data *float*. Perlu diperhatikan proses konversi tipe data, saat proses penyimpanan nilai bobot akhir dari tipe data *float* menjadi *string*. Begitu juga konversi dari *string* menjadi *float*, pada saat pembacaan data nilai bobot akhir dari SD Card. Hal ini berpengaruh terhadap hasil konversi, karena bilangan *float* berupa bilangan pecahan dan nilai pecahan tersebut bisa sampai 12 digit dibelakang koma. Sedangkan kemampuan konversi dari program mikrokontroler Arduino hanya bisa sampai 10 digit dibelakang koma. Sehingga berpengaruh dari nilai bobot yang didapat untuk proses *running* dari JST.

## UCAPAN TERIMA KASIH

Terima kasih disampaikan penulis kepada Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) Universitas Hang Tuah, yang telah mendanai penelitian internal ini. Semoga hasil penelitian ini dapat sebagai sarana sumbang sih penulis dalam ikut pengembangan ilmu pengetahuan dan teknologi, khususnya dibidang teknik elektro konsentrasi elektronika.

## DAFTAR PUSTAKA

- Heri Andrianto, & Aan Darmawan, 2016, "*ARDUINO Belajar Cepat dan Pemrograman*", Informatika, Bandung
- Jaka Prayudha, Darjat Saripurna, Nurcahyo Budi Nugroho, 2017, "*Implementasi Backpropagation Untuk Pengenalan Warna Garis Lintasan Robot Maze Solving Berbasis Arduino*", Jurnal SAINTIKOM Vol. 16, No. 2, Mei 2017
- Kusumadewi, S, Hartati, S., (2006), "*Integrasi Sistem Fuzzy dan Jaringan Syaraf Tiruan*". ISBN: 979-756-124-1, Graha Ilmu, Yogyakarta.
- Setya Ardhi, 2015, "*Penerapan Pelatihan Jaringan Syaraf Tiruan Untuk Sistem Kontrol Umpan Balik Pada Buck Converter Switching Power Supply*", Seminar Nasional "Inovasi dalam Desain dan Teknologi" - IDEaTech 2015