

## ***Performance Analysis of the RBF-SOM Network for Iris Data Classification as an Effort to Overcome System Control Problems***

### **Analisa Performa Jaringan RBF-SOM dalam Kasus Pengenalan Data Iris Sebagai Upaya Menanggulangi Permasalahan Pengendalian Sistem**

Yoan Elviralita<sup>1\*</sup>, Asrul Hidayat<sup>2</sup>

#### **Abstract**

*One way to solve system control problems is by using pattern recognition. Many studies are related to pattern recognition, including artificial neural networks. This study develops an algorithm that combines artificial neural networks with Radial Basis Function (RBF) and Self-Organizing Maps (SOM). The proposed RBF-SOM algorithm was successfully realized with the MATLAB routine program and tested with the case of iris data recognition. The results of the recognition rate show that the developed artificial neural network has a good performance with an average of 98%.*

#### **Keywords**

*Artificial Neural Networks, self-organizing Maps, Radial Basis Function*

#### **Abstrak**

Salah satu upaya dalam menyelesaikan permasalahan pengendalian system adalah dengan melakukan pengenalan pola. Banyak penelitian yang terkait dengan pengenalan pola diantaranya dengan jaringan syaraf tiruan. Penelitian ini mengembangkan sebuah algoritma perpaduan antara jaringan saraf tiruan *Radial Basis Function* (RBF) dan *Self-Organizing Maps* (SOM). Algoritma RBF-SOM ini berhasil direalisasikan dengan program MATLAB dan diuji dengan kasus pengenalan data bunga iris. Hasil *recognition rate* menunjukkan bahwa jaringan saraf tiruan yang dikembangkan tersebut memiliki performa yang baik dengan rata-rata sebesar 98 %.

#### **Kata Kunci**

Jaringan saraf tiruan, self-organizing Maps, Fungsi Basis Radial

<sup>1</sup> Teknik Mekatronika, Politeknik Bosowa Makasar

Jl. Kapasa Raya No.23, Kota Makassar, Sulawesi Selatan 90245

<sup>2</sup> Prodi Perawatan dan Perbaikan Mesin, Politeknik Bosowa Makasar

Jl. Kapasa Raya No.23, Kota Makassar, Sulawesi Selatan 90245

\* [yoan.elviralita@politeknikbosowa.ac.id](mailto:yoan.elviralita@politeknikbosowa.ac.id)

Submitted : January 12, 2022. Accepted : February 25, 2022. Published : February 26, 2022.

## PENDAHULUAN

Proses pengendalian adalah sebuah proses yang terdiri dari masukan, sistem pemroses dan keluaran. Masukan biasanya adalah beberapa nilai parameter tertentu yang diperoleh dari sebuah sensor. Bagian keluaran juga sebuah nilai tertentu yang dihasilkan oleh pemroses. Biasanya pemroses memiliki struktur rumus / model matematis tertentu untuk menghasilkan keluaran. Keluaran yang dihasilkan bisa berupa nilai tertentu maupun respon tertentu yang bersifat biner.

*Artificial Neural Networks* atau lebih dikenal dengan jaringan syaraf tiruan (JST) merupakan sistem yang dapat mengubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir di dalam jaringan tersebut. JST banyak digunakan pada berbagai riset di berbagai bidang. Salah satunya dalam bidang pengendalian sistem [1]–[3]. JST dapat juga digunakan sebagai sebuah pendekatan dalam pengendalian sistem yang parameter-parameternya tidak diketahui atau sulit untuk ditentukan [4].

Banyak jenis dan konfigurasi dari JST. Pada penelitian ini peneliti mencoba untuk menggabungkan metode jaringan saraf tiruan *Radial Basis Function* (RBF) dan *Self-Organizing Maps* (SOM). Penggunaan dua metode ini akan menggabungkan cara *unsupervised* dari SOM dan *supervised* yang dimiliki RBF. Penggabungan dua metode ini kemudian diwujudkan dengan program MATLAB untuk dalam mengidentifikasi tiga jenis dari bunga Iris. Hal ini dilakukan sebagai langkah awal untuk melihat kemungkinan pengembangan jaringan saraf tiruan FBR untuk diterapkan pada penelitian untuk identifikasi dan prediksi pengendalian sistem.

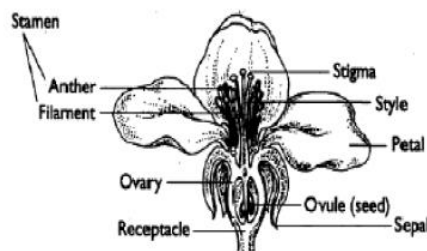
Tanaman Iris sengaja dipakai karena data sudah tersedia secara bebas (*open source*) di Internet dan banyak digunakan sebagai bahan untuk menguji pengklasifikasian objek. Data ini hasil penelitian Sir Ronald Aylmer Fisher pada tahun 1936 [5]. Pada penelitian ini digunakan 50 sampel data untuk masing-masing tiga jenis Iris. Tanaman Iris yang diuji mempunyai tampilan fisik seperti terlihat pada Gambar 1.



a. Iris setosa      b. Iris versicolor      c. Iris virginica

Gambar 1. Tiga jenis bunga Iris

Dari gambar 2, bagian atau fitur yang dipakai dalam pengklasifikasian berupa 1) lebar kelopak (*sepal width*), 2) panjang kelopak (*sepal length*), 3) lebar mahkota (*petal width*), dan 4) panjang mahkota (*petal length*). Pada proses pelatihan digunakan 70% dari 150 data Iris dan proses pengujian digunakan 30% dari 150 data tersebut.

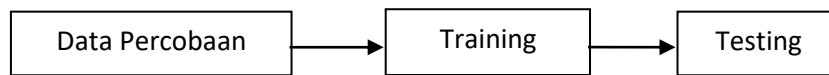


Gambar 2. Bagian yang diidentifikasi pada Iris

## METODE PENELITIAN

### Skema Penelitian Jaringan Saraf FBR Menggunakan SOM

Sistem pengenalan pola iris yang dikerjakan pada penelitian ini terbagi menjadi 3 tahapan, yang dijabarkan seperti gambar 3 berikut.



Gambar 3. Skema penelitian

#### Data Percobaan

Uji coba dilakukan pada komputer dengan prosesor Pentium 1.66 Ghz dan Memory 2 GB. Percobaan dilakukan terhadap data bunga Iris yang mengandung informasi dari tiga jenis atau kelas Bunga Iris yaitu Setosa, Versicolour dan Virginica. Setiap Iris dicirikan oleh empat atribut, yaitu: panjang *sepal* dalam cm, lebar *sepal* dalam cm, panjang *petal* dalam cm, dan lebar *petal* dalam cm. Perangkat lunak yang digunakan dalam percobaan ini adalah bahasa pemrograman MATLAB [6] dan dijalankan pada sebuah komputer.

#### Pelatihan atau Training

Setelah data percobaan didapatkan, tahap selanjutnya yang merupakan bagian utama sistem pengenalan pola adalah pelatihan pola atau training. Sistem dilatih untuk mengenali pola masukan agar menghasilkan keluaran yang sesuai dengan target tertentu. Awal pelatihan digunakan pelatihan tanpa arahan untuk melakukan pengelompokan data, kemudian setelah itu digunakan pelatihan dengan arahan untuk menggeneralisasi nilai masukan. Jaringan ini hanya mempunyai satu lapisan tersembunyi. Nilai bobot center, dan spread akan dimodifikasi terus menerus sampai sistem ini menemukan nilai akhir yang konvergen. Dalam penelitian ini, satu kali pelatihan dinyatakan sebagai satu *epoch*.

#### Pengujian atau Testing

Tahap pengujian ini berupa verifikasi suatu pola akan ditempatkan dalam kelas yang mana. Pada penelitian, tahap pengujian ini dilakukan dengan menggunakan pola Iris juga namun tidak sama dengan pola Iris yang digunakan pada pelatihan.

Dalam proese pengujian, setiap pola dijalankan ke dalam jaringan SOM dengan menggunakan nilai bobot akhir atau vektor perwakilan akhir yang telah disimpan sebelumnya untuk melakukan pengelompokan data, setelah itu dijalankan dalam jaringan backpropagation tetapi hanya sampai pada tahap propagasi maju saja dengan menggunakan nilai bobot akhir yang disimpan sebelumnya.

Untuk satu kali pengujian, persentase kesesuaian terhadap target yang disebut *recognition rate* pun dihitung.

#### Parameter yang Digunakan sebagai Variabel pada Jaringan Saraf FBR dengan Menggunakan SOM

Parameter yang digunakan dalam penelitian jaringan saraf FBR menggunakan SOM ini dan sekaligus menjadi variable bebas, antara lain:

**Inisialisasi bobot;** dalam penelitian ini ada dua macam inisialisasi bobot, yaitu inisialisasi bobot pada jaringan saraf swa organisasi dengan memilih salah satu dari vektor data dan inisialisasi bobot pada jaringan saraf propagasi balik yang ditentukan secara random dan metode *Nguyen-Widrow*. Inisialisasi ini banyak digunakan karena terbukti dapat meningkatkan waktu eksekusi [7]-[12].

**Banyaknya neuron yang digunakan pada lapisan masukan, lapisan tersembunyi, dan lapisan keluaran;** dalam Penelitian ini peneliti menggunakan neuron pada lapisan masukan sebanyak empat neuron disesuaikan dengan banyaknya atribut yang digunakan data

masukannya, neuron pada lapisan tersembunyi sebanyak tiga neuron, dan tiga neuron untuk lapisan keluaran.

**Besarnya epoch yang digunakan;** peneliti mencoba membandingkan hasil yang diperoleh ketika epoch yang ditentukan antara lain 100, 1000 dan 10000.

**Besarnya laju pembelajaran ( $\alpha$ );** laju pembelajaran untuk algoritma SOM sebesar 0.1 dan algoritma propagasi balik sebesar 0.2, laju pembelajaran untuk pembaharuan center sebesar 0.5 dan pembaharuan spread sebesar 0.8. Sementara itu, variabel terikat yang digunakan pada penelitian ini adalah nilai recognition rate training, recognition rate testing, waktu training, waktu testing, dan grafik *error* terhadap *epoch*. Berdasarkan hasil tersebut, akan ditentukan parameter-parameter yang dapat mengoptimalkan performa jaringan, yang selanjutnya disebut parameter optimalisasi.

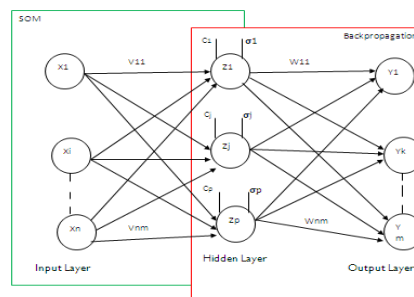
**Nilai Kesalahan Uji;** daerah nilai kesalahan uji yang digunakan dalam penelitian ini adalah 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, dan 0.49 untuk nilai 0 dan range 0.85, 0.80, 0.75, 0.70, 0.65, 0.60, dan 0.51 untuk nilai 1.

## HASIL DAN PEMBAHASAN

### Hasil

#### Proses Pelatihan (training) dan Pengujian (testing) Data

Susunan dari jaringan RBF secara umum terdiri dari tiga lapisan, yang samasekali berbeda. Lapisan (*layer*) masukan terbentuk dari titik sumber (*unit sensor*). Lapisan yang kedua adalah lapisan *hidden* yang terdiri dari dimensi yang cukup besar, yang memberikan maksud berbeda dari lapisan *hidden* pada *multilayer perceptron*. Lapisan keluaran memberikan tanggapan dari jaringan terhadap bentuk aktivasi lapisan masukan.



Gambar 4. Jaringan Fungsi Basis Radial

Pada pelatihan ini algoritma secara detail yang digunakan sebagai berikut:

#### Langkah 0: Inisialisasi

#### SOM

- Bobot awal  $V_{ji}$  di ambil dr salah satu vektor dalam setiap kelas
- Set parameter laju pembelajaran (*Learning Rate* ( $\alpha$ ))

#### Backpropagation

- Bobot awal  $w_{kj}$  menggunakan nguyen widrow.
- Bias awal  $w_{0k}$
- Set parameter laju pembelajaran (*Learning rate*)

#### Proses SOM:

#### Langkah 1: Clustering Data, menggunakan SOM

- Langkah a : Jika kondisi henti gagal, lakukan langkah b-g
- Langkah b : Untuk setiap vektor masukan  $x$ , lakukan langkah c sampai e
- Langkah c : Untuk setiap  $j$ , hitung:

$$D_i = \sqrt{\sum_{j=1}^m (V_{ji} - X_i)^2} \quad (1)$$

- Langkah d : Temukan indeks j sehingga D(j) minimum
- Langkah e : Untuk setiap neuron J meng-update bobotnya

$$v_{ji}(\text{new}) = v_{ji}(\text{lama}) + \alpha (x_i - v_{ji}(\text{lama})) \quad (2)$$

- Langkah f : Memodifikasi laju pemahaman
- Langkah g : melanjutkan iterasi

### Langkah 2: Inisialisasi center dan Spread

- Langkah a : menghitung jumlah/ banyaknya vektor data dalam kelas
- Langkah b : menghitung rata-rata vektor /center (c)

$$c_j = \frac{\sum_{i=1}^p x_{pj}}{N_j} \quad (3)$$

- Langkah c : menghitung nilai spread ( $\sigma$ )

$$\sigma = \frac{\text{jarak maksimum antara 2 center}}{\sqrt{\text{banyaknya center}}} \quad (4)$$

### Proses Backpropagation:

#### Langkah 3: Untuk masing masing vektor kelas j

##### Feedforward:

- Langkah a : menghitung nilai zj

$$z_j = \exp\left(-\frac{(x_i - c_j)^T (x_i - c_j)}{2(\sigma_j)^2}\right) = \exp(-z_{in}) \quad (5)$$

- Langkah b : menghitung nilai yink

$$y_{ink} = w_0(k) + \sum z_j \cdot w_{kj} \quad (6)$$

- Langkah c : menghitung nilai Yk

$$Y_k = f(-y_{ink})$$

$$Y_k = 1/(1 + \exp(-y_{ink})) \quad (7)$$

##### Backward:

- Langkah d : menghitung nilai  $\delta_k$

$$\delta_k = (T_k - Y_k) f'(y_{ink})$$

$$\delta_k = (T_k - Y_k) (Y_k)(1 - Y_k) \quad (8)$$

- Langkah e : Menghitung besarnya koreksi bobot unit output

$$\Delta w_{kj} = \alpha \delta_k \quad (9)$$

- Langkah f : Menghitung besarnya koreksi bias output

$$\Delta w_0_k = \alpha \delta_k \quad (10)$$

##### Perubahan Bobot

- Langkah g : Setiap unit output mengupdate bobotnya

$$w_{kj}(\text{new}) = w_{kj}(\text{lama}) + \Delta w_{kj} \quad (11)$$

- Langkah h : Setiap unit output mengupdate biasnya

$$w_0_k(\text{new}) = w_0_k(\text{lama}) + \Delta w_0_k \quad (12)$$

##### Perubahan center

- Langkah i : mengup-date center

$$c_j(\text{new}) = c_j(\text{lama}) + \alpha \left( \frac{do(k) \cdot w_{kj}}{\sigma_j^2} \right) \cdot z_j \cdot [X_i - c_j] \quad (13)$$

##### Perubahan nilai spread

- Langkah j : mengupdate nilai spread

$$\sigma_j(\text{new}) = \sigma_j(\text{lama}) + \alpha \left( \frac{do(k) \cdot w_{kj}}{\sigma_j^3} \right) \cdot z_j \cdot ((x_i - c_j)^T (x_i - c_j)) \quad (14)$$

- Langkah k : kondisi stopping

$$E_{\min} < 0.01 \quad (15)$$

Setelah melakukan pelatihan, dilakukan pengenalan pola dengan menggunakan 30 % dari 150 data Iris yaitu sebesar 45 data, agar seragam diambil pola dengan jumlah yang sama untuk setiap target, yaitu 15. Pada proses pengenalan pola ini, sistem akan memberi jawaban terhadap setiap pola, masuk ke cluster yang mana dari tiga cluster yang ada.

Apabila jawaban yang diberikan sesuai target, maka jaringan dianggap telah mengenali pola dengan benar. Namun bila jawaban yang diberikan tidak sesuai dengan target, maka jaringan dinilai salah dalam mengenali pola. Sistem tidak dapat menghasilkan jawaban bahwa suatu pola berada di luar target yang ditentukan. Dengan demikian, hanya ada dua pilihan, yaitu benar dan salah. Persentase jawaban yang benar terhadap seluruh pertanyaan mengacu kepada *recognition rate*.

## Pembahasan

### Pengaruh Parameter-Parameter pada Algoritma FBR Banyaknya *epoch*

Tabel 1 Recognition Rate dan Waktu dengan 100 Epoch

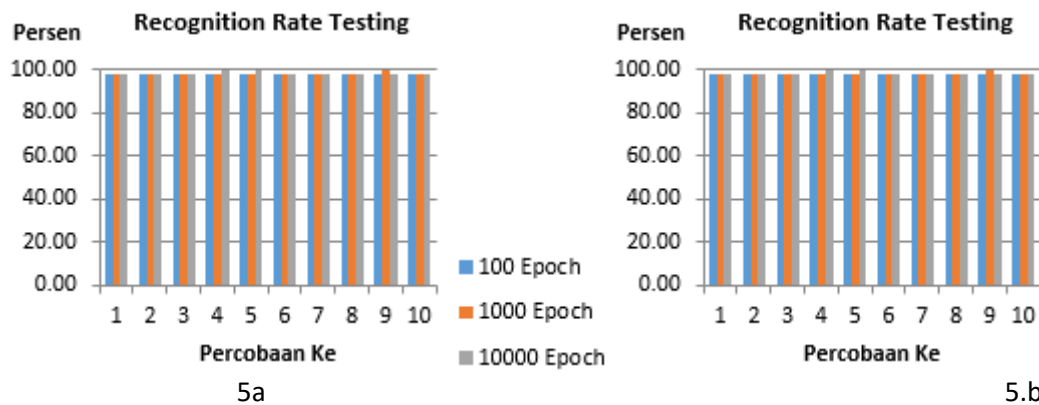
| Percobaan ke | Recognition Rate(%) |         | Waktu (detik) |         |
|--------------|---------------------|---------|---------------|---------|
|              | Training            | Testing | Training      | Testing |
| 1            | 96.19               | 97.78   | 4.71          | 0.0182  |
| 2            | 96.19               | 97.78   | 4.74          | 0.0183  |
| 3            | 96.19               | 97.78   | 4.74          | 0.0183  |
| 4            | 96.19               | 97.78   | 4.72          | 0.0182  |
| 5            | 96.19               | 97.78   | 4.77          | 0.0194  |
| 6            | 91.43               | 97.78   | 4.72          | 0.0185  |
| 7            | 91.43               | 97.78   | 4.73          | 0.0226  |
| 8            | 96.19               | 97.78   | 4.77          | 0.0183  |
| 9            | 96.19               | 97.78   | 4.75          | 0.0187  |
| 10           | 93.33               | 97.78   | 4.72          | 0.0182  |

Tabel 2 Recognition Rate dan Waktu dengan 1000 Epoch

| Percobaan ke | Recognition Rate(%) |         | Waktu (detik) |         |
|--------------|---------------------|---------|---------------|---------|
|              | Training            | Testing | Training      | Testing |
| 1            | 97.14               | 97.78   | 46.77         | 0.0186  |
| 2            | 97.14               | 97.78   | 46.39         | 0.0182  |
| 3            | 97.14               | 97.78   | 46.41         | 0.0182  |
| 4            | 95.24               | 97.78   | 46.68         | 0.0181  |
| 5            | 95.24               | 97.78   | 46.43         | 0.0182  |
| 6            | 97.14               | 97.78   | 46.33         | 0.0182  |
| 7            | 97.14               | 97.78   | 46.72         | 0.0185  |
| 8            | 97.14               | 97.78   | 47.43         | 0.0183  |
| 9            | 96.19               | 100.00  | 46.68         | 0.0185  |
| 10           | 97.14               | 97.78   | 46.80         | 0.0185  |

Tabel 3 Recognition Rate dan Waktu dengan 10000 Epoch

| Percobaan ke | Recognition Rate(%) |         | Waktu (detik) |         |
|--------------|---------------------|---------|---------------|---------|
|              | Training            | Testing | Training      | Testing |
| 1            | 98.10               | 97.78   | 475.28        | 0.0237  |
| 2            | 98.10               | 97.78   | 478.81        | 0.0184  |
| 3            | 98.10               | 97.78   | 498.35        | 0.0187  |
| 4            | 96.19               | 100.00  | 485.19        | 0.0188  |
| 5            | 96.19               | 100.00  | 476.40        | 0.0192  |
| 6            | 98.10               | 97.78   | 476.87        | 0.0187  |
| 7            | 96.19               | 97.78   | 475.97        | 0.0185  |
| 8            | 98.10               | 97.78   | 476.19        | 0.0185  |
| 9            | 98.10               | 97.78   | 476.95        | 0.0183  |
| 10           | 98.10               | 97.78   | 468.61        | 0.0321  |



Gambar 5. Grafik Recognition Rate dengan tiga variasi 100, 1000, 10000 Epoch

Dari tabel 1, 2, dan 3, recognition rate untuk pola-pola dari data training mencapai 91.43 % sampai 98.10 %, berarti sistem berhasil mengenali hampir semua pola di dalam data training dengan benar. Recognition rate untuk pola-pola data testing mencapai 97.78 % sampai 100 %, artinya sistem berhasil mengenali 44 atau 45 pola data testing dengan benar. Nilai ini cukup tinggi dan menunjukkan bahwa sistem sudah baik dalam mengenali pola setelah dilatih dengan algoritma FBR dengan menggunakan SOM dengan parameter-parameter standar yang ditentukan. Hal ini dapat diibaratkan dengan otak manusia yang dapat mengenali wajah seseorang meskipun dalam keadaan yang berbeda dari yang dikenalnya sebelumnya.

Pada variasi ini, yang menjadi perbedaan hanyalah waktu pembelajarannya. Semakin banyak iterasi yang dilakukan, semakin lama waktu yang dibutuhkan jaringan untuk melatih dirinya. Atas dasar recognition rate yang besarnya sama, maka dengan iterasi sebanyak 100 epoch sebenarnya sudah cukup baik untuk diterapkan pada sistem. Atau dapat disimpulkan, jika banyaknya iterasi yang ditetapkan sebagai syarat henti telah mencapai epoch atau batas toleransi *error*, sistem dapat mempelajari pola dan mengenali pola dengan baik.

### Besarnya Laju Pembelajaran ( $\alpha$ )

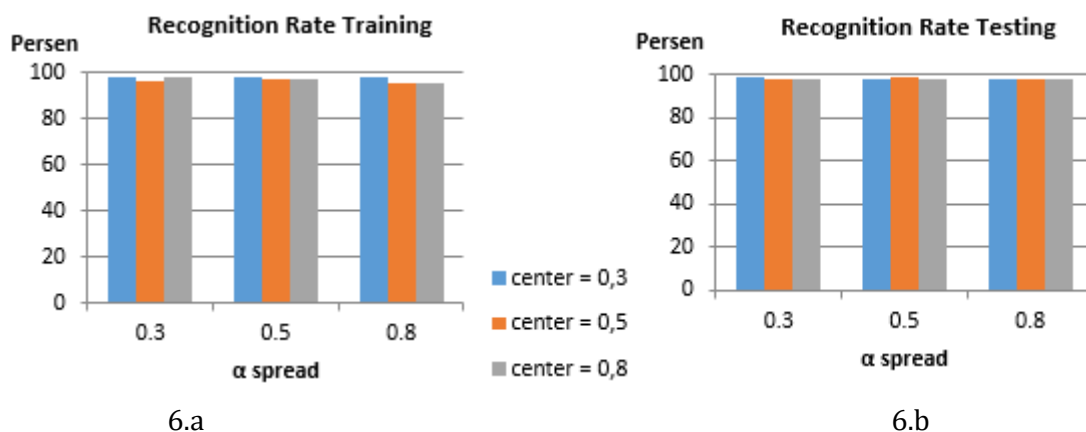
Besarnya laju pembelajaran mempengaruhi seberapa cepat jaringan akan mencapai konvergensi *error*. Pada percobaan, terdapat empat laju pembelajaran, yaitu laju pembelajaran untuk algoritma SOM, laju pembelajaran untuk algoritma BP, laju pembelajaran

untuk perubahan center, dan laju pembelajaran untuk perubahan spread. Laju pembelajaran untuk SOM dan BP sudah menggunakan standar seperti yang telah disebut pada bagian 2, yaitu 0.1 dan 0.2, sedangkan untuk laju pembelajaran center dan spread menggunakan range sebagai berikut: kecil dengan range antara 0.1 sampai 0.4, sedang dengan range antara 0.4 sampai 0.7, dan besar dengan range antara 0.7 sampai 0.9. Namun dari range-range tersebut diambil salah satu nilai untuk setiap rangenya, kecil dengan nilai 0.3, sedang dengan nilai 0.5, dan besar dengan nilai 0.8.

Tabel 4 Variasi Laju Pembelajaran ( $\alpha$ )

| Laju Pembelajaran ( $\alpha$ ) |     |        |        | FBR                  |         |          |         |
|--------------------------------|-----|--------|--------|----------------------|---------|----------|---------|
| SOM                            | BP  | Center | Spread | Recognition Rate (%) |         | Waktu    |         |
|                                |     |        |        | Training             | Testing | Training | Testing |
| 0.1                            | 0.2 | 0.3    | 0.3    | 98.38                | 99.33   | 46.56    | 0.0186  |
| 0.1                            | 0.2 | 0.3    | 0.5    | 98.19                | 98.00   | 46.55    | 0.0184  |
| 0.1                            | 0.2 | 0.3    | 0.8    | 98.19                | 97.78   | 46.60    | 0.0184  |
| 0.1                            | 0.2 | 0.5    | 0.3    | 96.86                | 99.78   | 46.61    | 0.0184  |
| 0.1                            | 0.2 | 0.5    | 0.5    | 96.95                | 98.89   | 46.58    | 0.0186  |
| 0.1                            | 0.2 | 0.5    | 0.8    | 95.81                | 97.78   | 46.61    | 0.0184  |
| 0.1                            | 0.2 | 0.8    | 0.3    | 96.76                | 98.44   | 46.76    | 0.0185  |
| 0.1                            | 0.2 | 0.8    | 0.5    | 97.14                | 98.22   | 46.66    | 0.0202  |
| 0.1                            | 0.2 | 0.8    | 0.8    | 95.90                | 98.00   | 47.24    | 0.0198  |

Gambar 6 menunjukkan *recognition rate* untuk data training mencapai 95.81 % sampai 98.38 %, dapat dilihat bahwa nilai laju pembelajaran tidak mempengaruhi tingkat akurasi pengenalan pola. Begitu juga *recognition rate* untuk data testing mencapai 97.78 % sampai 99.78 % dan tidak menunjukkan relasi antara *recognition rate* dengan nilai laju pembelajaran, hal ini disebabkan *recognition rate* untuk semua alpha center maupun alpha *spread* mempunyai nilai yang sama juga. maka semakin besar nilai laju pembelajaran, jaringan akan semakin cepat mencapai konvergensi dan waktu pembelajaran pun semakin singkat.



Gambar 6. Grafik Recognition Rate Training and Testing menggunakan  $\alpha$  center dan  $\alpha$  spread dengan ukuran kecil, sedang, dan besar



**Range Nilai Kesalahan Uji**

Parameter ini digunakan untuk mengamati besarnya nilai *recognition rate* yang dicapai. Apabila keluaran yang diberikan sesuai target, maka jaringan dianggap telah mengenali pola dengan benar. Namun bila jawaban yang diberikan tidak sesuai dengan target, maka jaringan dinilai salah dalam mengenali pola. Sistem tidak dapat menghasilkan jawaban bahwa suatu pola berada di luar target yang ditentukan, maka dari itu diberikan suatu nilai kebenaran pada sistem dengan memberikan batasan nilai, sedangkan nilai yang berada di luar batasan tersebut merupakan suatu nilai kesalahan. Dengan demikian, hanya ada dua nilai, yaitu benar dan salah.

*Tabel 5 Variasi Nilai Kebenaran Sistem*

| No Range | Range Keluaran (Y) |     |   | Recognition Rate (%) |         | Waktu    |         |
|----------|--------------------|-----|---|----------------------|---------|----------|---------|
|          |                    |     |   | Training             | Testing | Training | Testing |
| 1        | 0.85               | s/d | 1 | 66.67                | 66.67   | 46.00    | 0.0186  |
|          | 0.15               | s/d | 0 |                      |         |          |         |
| 2        | 0.80               | s/d | 1 | 70.48                | 77.78   | 46.02    | 0.0205  |
|          | 0.20               | s/d | 0 |                      |         |          |         |
| 3        | 0.75               | s/d | 1 | 76.19                | 82.22   | 45.99    | 0.0185  |
|          | 0.25               | s/d | 0 |                      |         |          |         |
| 4        | 0.70               | s/d | 1 | 83.81                | 88.89   | 45.72    | 0.0183  |
|          | 0.30               | s/d | 0 |                      |         |          |         |
| 5        | 0.65               | s/d | 1 | 88.57                | 93.33   | 45.85    | 0.0183  |
|          | 0.35               | s/d | 0 |                      |         |          |         |
| 6        | 0.60               | s/d | 1 | 90.48                | 97.78   | 45.85    | 0.0323  |
|          | 0.40               | s/d | 0 |                      |         |          |         |
| 7        | 0.51               | s/d | 1 | 94.29                | 97.78   | 47.96    | 0.0306  |
|          | 0.49               | s/d | 0 |                      |         |          |         |



7.a

7.b

Gambar 7 Variasi Nilai Kebenaran Sistem

## SIMPULAN DAN SARAN

### Simpulan

Sistem pengenalan Iris menggunakan metode FBR dengan algoritma SOM telah dilakukan dan dapat bekerja dengan efektif. Hasil-hasil eksperimen dari pengujian sistem menggunakan data uji menunjukkan tingkat akurasi yang cukup baik. Pada penelitian selanjutnya, sistem yang telah dibuat ini masih sangat dimungkinkan untuk dikembangkan lagi.

### Saran

Penelitian sistem pengenalan iris ini diharapkan dapat dilakukan dengan menggunakan data uji yang lain yang terkait dengan penanggulangan masalah sistem pengendali.

## DAFTAR RUJUKAN

- [1] G. Dewantoro and J. N. Sukamto, 'Implementasi Kendali PID Menggunakan Jaringan Syaraf Tiruan Backpropagation', *ELKHA*, vol. 11, no. 1, p. 12, Apr. 2019, doi: 10.26418/elkha.v11i1.29959.
- [2] L. Widaningrum, B. Setiyono, and M. A. Riyadi, 'PERANCANGAN KONTROLER JARINGAN SYARAF TIRUAN B-SPLINE BERBASIS MIKROKONTROLER ATMEGA16 SEBAGAI KENDALI KECEPATAN MOTOR BRUSHLESS DC (BLDC)', *Transient*, vol. 6, no. 3, p. 373, Nov. 2017, doi: 10.14710/transient.6.3.373-379.
- [3] M. Azmi, D. S. Putra, W. Purwanto, T. Sugiarto, and D. Fernandez, 'Implementasi Jaringan Syaraf Tiruan untuk Mengendalikan Lampu Sein Sepeda Motor', *1*, vol. 19, no. 2, Art. no. 2, Oct. 2019, doi: 10.24036/invotek.v19i2.622.
- [4] D. S. Putra, 'Pemodelan Sistem dengan Metode Neural Network Back Propagation Modeling System Using Neural Network Backpropagation', *Jurnal Ilmiah Poli Rekayasa*, vol. 11, no. 2, pp. 22–31, 2016.
- [5] E. Anderson, 'The Species Problem in Iris', *Annals of the Missouri Botanical Garden*, vol. 23, no. 3, pp. 457–509, 1936, doi: 10.2307/2394164.
- [6] 'MathWorks - Makers of MATLAB and Simulink'. <https://www.mathworks.com/> (accessed Feb. 25, 2022).
- [7] S. Aisyah, M. Harahap, A. Dharma, and M. Turnip, 'Implementation artificial neural network nguyen widrow algorithm for lupus prediction', *J. Phys.: Conf. Ser.*, vol. 1361, no. 1, p. 012067, Nov. 2019, doi: 10.1088/1742-6596/1361/1/012067.
- [8] R. Damanik, 'Analisis Penggunaan Algoritma Nguyen Widrow Dalam Backpropagation pada Penyakit Ginjal', 2013, Accessed: Feb. 25, 2022. [Online]. Available: <https://repositori.usu.ac.id/handle/123456789/43010>
- [9] I. Hakim, S. Efendi, and P. Sirait, 'Optimization of the Backpropagation Method with Nguyen-widrow in Face Image Classification', *Randwick International of Social Science Journal*, vol. 2, no. 2, Art. no. 2, Apr. 2021, doi: 10.47175/rissj.v2i2.226.
- [10] E. Kurniawan, H. Wibawanto, and D. A. Widodo, 'Implementasi Metode Backpropagation dengan Inisialisasi Bobot Nguyen Widrow untuk Peramalan Harga Saham', *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 1, Art. no. 1, Jan. 2019, doi: 10.25126/jtiik.201961904.
- [11] Rosmaliati, N. E. Setiawati, M. H. Purnomo, and A. Priyadi, 'Nguyen-Widrow Neural Network for Distribution Transformer Lifetime Prediction', in *2018 International Conference on Computer Engineering, Network and Intelligent Multimedia (CENIM)*, Nov. 2018, pp. 305–310. doi: 10.1109/CENIM.2018.8710815.
- [12] M. R. Wayahdi, M. Zarlis, and P. H. Putra, 'Initialization of the Nguyen-widrow and Kohonen Algorithm on the Backpropagation Method in the Classifying Process of Temperature Data in Medan', *J. Phys.: Conf. Ser.*, vol. 1235, no. 1, p. 012031, Jun. 2019, doi: 10.1088/1742-6596/1235/1/012031.