

## DETEKSI PENYAKIT PADA TANAMAN JERUK PONTIANAK DENGAN METODE JARINGAN SARAF TIRUAN BACKPROPAGATION

<sup>[1]</sup>Fitrajaya Nugraha, <sup>[2]</sup>Beni Irawan, <sup>[3]</sup>Dwi Marisa Midyanti

<sup>[1][3]</sup>Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura

<sup>[2]</sup>Jurusan Sistem Informasi, Fakultas MIPA Universitas Tanjungpura

Jl. Prof. Dr. H. Hadari Nawawi, Pontianak

Telp./Fax.: (0561) 577963

e-mail:

<sup>[1]</sup>fitrajaya@ymail.com, <sup>[2]</sup>benicsc@yahoo.com,

<sup>[3]</sup>marisa.untan@gmail.com

### Abstrak

*Perkembangan teknologi komputer yang begitu canggih dan mutakhir mampu membantu manusia dalam berbagai hal, tak terkecuali dalam mendeteksi penyakit. Penelitian ini untuk mendeteksi penyakit pada tanaman jeruk Pontianak dan mengetahui keakuratan hasil deteksi dengan jaringan saraf tiruan. Metode yang digunakan adalah Backpropagation. Gejala – gejala penyakit yang terjadi di lapangan merupakan masukan yang menjadi data input pada sistem yang terdiri dari 23 variabel. Jaringan terdiri dari 4 lapisan, 1 lapisan masukan, 2 lapisan tersembunyi dan 1 lapisan keluaran, dimana ada 15 lapisan tersembunyi pertama dan 15 lapisan tersembunyi kedua serta 4 sel lapisan keluaran. Maksimum iterasi pada jaringan ini berjumlah 35000 , learning rate 0,0001 , dan target error 0,0001. Hasil keakuratan yang didapat dari penelitian ini mencapai 83,33% dari 24 data pengujian.*

**Kata Kunci :** Jaringan Saraf Tiruan, Penyakit Jeruk, *Backpropagation*

### 1. PENDAHULUAN

Komputer tidak hanya digunakan untuk sekedar membantu pekerjaan manusia, tetapi sudah mulai dioperasikan untuk mengganti hampir sebagian besar pekerjaan manusia. Pengembangan teknologi membuat kerja sistem komputer menjadi lebih cepat, teliti dan akurat dibandingkan kerja manusia. Salah satu penerapan bidang ilmu yang membuat komputer dapat menimbang dan mengambil keputusan sendiri seperti manusia yaitu jaringan saraf tiruan.

Jaringan saraf tiruan ini mulai diterapkan diberbagai bidang kehidupan salah satunya bidang pertanian. Pada bidang pertanian jaringan saraf tiruan dapat mendeteksi penyakit tanaman dengan mempelajari masukan yang diberikan dan kemampuan untuk mengolah data. Salah satu metode dalam jaringan saraf tiruan adalah *backpropagation*.

Metode *backpropagation* dapat lebih mendekati target output yang diinginkan dan ketepatan dalam melakukan pengujian

karena terjadi penyesuaian nilai bobot dan bias yang semakin baik pada proses pelatihan. Metode *backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang dipakai selama pelatihan.

Metode *backpropagation* pada jaringan saraf tiruan ini digunakan dalam mendeteksi penyakit jeruk yaitu jeruk Pontianak. Jeruk Pontianak merupakan komoditas unggulan di Kabupaten Sambas. Pada perkembangan tanaman jeruk Pontianak ada dua faktor yang mempengaruhi tumbuh kembangnya tanaman tersebut, yaitu hama dan penyakit. Penyakit jeruk Pontianak ini dijadikan sebagai keluaran dari metode *backpropagation* sedangkan masukannya adalah gejala-gejala dari penyakit jeruk tersebut. Dengan mengetahui penyakit jeruk tersebut dapat dilakukan langkah

pengecanaan maupun pengobatan agar jeruk tersebut dapat diselamatkan ataupun disembuhkan.

Penelitian yang di lakukan oleh Titania Dwi Andini,S.Kom dan Jaenal Arifin [1] tentang identifikasi jenis penyakit pada tanaman jeruk ini menggunakan sistem pakar berbasis wireless aplication protocol (wap) dengan *wireless markup language* (wml) dan php hypertext preprocessor (php). Penelitian yang dilakukan memberikan infomasi mengenai jenis-jenis penyakit jeruk teknik pencarian dengan tehnik heuristik dan metode inferensi *forward* dan *backward chaining*.

Berdasarkan hal tersebut dibuat penelitian tentang penyakit jeruk dengan metode yang berbeda yaitu jaringan saraf tiruan serta metode *backpropagation*. Penelitian dengan jaringan saraf tiruan ini juga memberikan informasi jenis-jenis penyakit jeruk pontianak.

## 2. TINJAUAN PUSTAKA

### 2.1 Jaringan Saraf Tiruan

Jaringan saraf tiruan bisa dibayangkan seperti otak buatan. Otak buatan ini dapat berpikir seperti manusia,dan juga sepandai manusia dalam menyimpulkan suatu potongan-potongan informasi yang diterima. [2]

#### A. Pengertian Jaringan Saraf Tiruan

Jaringan syaraf tiruan (JST) adalah sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan saraf biologi. [3]

Jaringan saraf tiruan adalah generalisasi dari pemodelan saraf biologi dengan asumsi-asumsi antara lain :

Pemrosesan informasi terletak pada sejumlah komponen yang dinamakan neuron-neuron lainnya melalui jalur penghubung. Tiap jalur penghubung memiliki bobot dan mengalikan besar nilai singal yang masuk (jenis neuron tertentu). Tiap neuron menerapkan fungsi aktivasi (biasanya nonlinear) yang menjumlahkan semua masukan untuk menentukan sinyal keluarannya. Tiap jaringan di tentukan oleh arsitektur pola jaringan , bobot pada koneksi dan fungsi aktivasi.

Sistem jaringan saraf tiruan disebut juga *brain metaphor*, *computational neuroscience* atau *parallel distributed processing* serta *connection*. Jaringan saraf tiruan tersusun dari sejumlah besar elemen yang melakukan kegiatan yang analog dengan fungsi-fungsi biologis neuron yang paling elementer. Elemen-elemen ini terorganisasi sebagaimana layaknya anatomi otak walaupun tidak persis. Jaringan saraf tiruan dapat belajar dari pengalaman, melakukan generalisasi atas contoh-contoh yang diperolehnya dan mengabstraksi karakteristik esensial input bahkan untuk data yang tidak relevan. [4]

#### B. Fungsi Aktivasi dan Pelatihan

Fungsi aktivasi yaitu fungsi Sigmoid Biner. Fungsi ini digunakan untuk jaringan syaraf tiruan yang dilatih dengan menggunakan metode *backpropagation*. Fungsi sigmoid biner memiliki nilai range 0 sampai 1 . Oleh karena itu, fungsi ini sering digunakan untuk jaringan syaraf yang membutuhkan nilai output yang terletak pada interval 0 sampai 1. Namun, fungsi ini bisa juga digunakan oleh jaringan syaraf yang nilai outputnya 0 atau 1. [5]

Fungsi sigmoid biner dirumuskan sebagai:

$$y = f(x) = \frac{1}{1+e^{-\sigma x}}$$

$$f'(x) = f(x)[1-f(x)]..... (1)$$

Pada jaringan saraf tiruan, berdasarkan memodifikasi bobotnya, ada 2 macam pelatihan yaitu dengan supervisi/terawasi (supervised) dan tanpa supervisi/tidak terawasi (unsupervised).

#### 1. Pelatihan dengan supervisi/terawasi (supervised)

Dalam pelatihan ini terdapat sejumlah pasangan data yang digunakan untuk melatih jaringan hingga diperoleh bobot yang diinginkan. Pasangan data tersebut menjadi pengarah untuk melatih jaringan hingga diperoleh bentuk terbaik.

Untuk pelatihan terawasi kumpulan input yang digunakan, output-outputnya telah diketahui. Perbedaan antara output-output aktual dengan output-output yang diinginkan

digunakan untuk mengkoreksi bobot jaringan syaraf tiruan agar dapat menghasilkan jawaban semirip mungkin.

Proses yang terjadi pada setiap pelatihan ini yaitu, suatu input diberikan ke jaringan, dimana jaringan akan memproses dan mengeluarkan keluaran. Selisih antara keluaran jaringan dengan target merupakan kesalahan yang terjadi. Jaringan nantinya akan memodifikasi bobot sesuai dengan kesalahan tersebut.

2. Pelatihan tanpa supervisi/tidak terawasi (unsupervised)

Pada pelatihan ini tidak ada pengarahan dalam pelatihan. Dalam pelatihannya, perubahan bobot jaringan dilakukan berdasarkan parameter tertentu dan jaringan dimodifikasi menurut parameter tersebut.

Proses pelatihan ini jaringan syaraf tiruan mengorganisasi dirinya sendiri untuk membentuk contoh-contoh pelatihan. Struktur menggunakan dasar data atau korelasi antar pola-pola data yang dieksplorasi. Paradigma pembelajaran ini mengorganisasi pola-pola ke dalam kategori-kategori berdasarkan korelasi yang ada.

**2.2 Jaringan Saraf Tiruan**

**Backpropagation**

Backpropagation ini merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada ada lapisan tersembunyinya.

Pada algoritma backpropagation, algoritmanya menggunakan error output untuk mengubah nilai bobot-bobotnya dalam arah mundur (backward). Untuk mendapatkan error ini, tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu. [kusumadewi]

Adapun algoritma pelatihan untuk jaringan dengan satu lapisan tersembunyi (dengan fungsi aktivasi sigmoid biner) adalah sebagai berikut. [6]

1. Inisialisasi bobot (ambil bobot awal dengan nilai random yang cukup kecil).

2. Tetapkan: Maksimum Epoch, Target Error, dan *Learning Rate* ( ).

3. Inisialisasi: Epoch = 0, MSE = 1

4. Kerjakan langkah-langkah berikut ini selama kondisi bernilai false :

1. Epoch = Epoch + 1

2. Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan :

*Feedforward:*

- a. Tiap-tiap masukan ( $X_i$   $i= 1,2,3,\dots,n$ ) menerima sinyal  $x_i$  dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (lapisan tersembunyi)

- b. Tiap-tiap unit tersembunyi ( $Z_j$   $j= 1,2,3,\dots,p$ ) menjumlahkan sinyal-sinyal masukan terbobot:

$$z_{in_j} = v_{oj} + \sum_{i=0}^n x_i v_{ij} \dots\dots\dots (2)$$

gunakan fungsi aktivasi untuk menghitung sinyal *output*-nya

$$z_j = f(z_{in_j}) = \frac{1}{1+e^{-z_{in_j}}} \dots\dots\dots (3)$$

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit *output*nya).

- c. Tiap-tiap unit *output* ( $Y_k$ ,  $k= 1,2,3,\dots,m$ ) menjumlahkan sinyal-sinyal masukan terbobot.

$$Y_{in_k} = w_{ok} + \sum_{j=0}^n z_j w_{jk} \dots\dots\dots (4)$$

Gunakan fungsi aktivasi untuk menghitung sinyal keluarannya:

$$y_j = f(y_{in_j}) = \frac{1}{1+e^{-y_{in_j}}} \dots\dots\dots (5)$$

Dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit keluaran).

*Backpropagation*

- d. Tiap- tiap unit keluaran ( $Y_k$   $k= 1,2,3,\dots,m$ ) menerima target pola yang berhubungan dengan pola masukan pembelajarannya, hitung informasi error-nya:

$$e_k = (t_k - y_k) f'(y_{in_k}) \dots\dots\dots (6)$$

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $W_{jk}$ ):

$$\Delta W_{jk} = e_k \dots\dots\dots (7)$$

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $W_{ok}$ ):

- $W_{0k} = \dots \dots \dots (8)$   
 Kirimkan  $x_k$  ini ke unit-unit yang ada dilapisan dibawahnya.  
 e. Tiap- tiap unit tersembunyi ( $Z_j, j=1,2,3,\dots,p$ ) menjumlahkan delta masukannya (dari unit- unit yang berada pada lapisan di atasnya):  

$$z_{in_j} = \sum_{k=1}^n \delta w_{jk} \dots \dots \dots (9)$$

kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi error :

$$v_j = z_{in_j} f'(z_{in_j}) \dots \dots \dots (10)$$

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{ij}$ ):

$$v_{jk} = v_j x_k \dots \dots \dots (11)$$

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{0j}$ ):

$$v_{0j} = v_j \dots \dots \dots (12)$$

- f. Tiap-tiap unit keluaran ( $Y_k, k=1,2,3,\dots,p$ ) memperbaiki bias dan bobotnya ( $j=0,1,2,\dots,p$ ):

$$W_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \mu (w_{jk}(\text{lama}) - w_{jk}(\text{lama-1})) \dots \dots (2.13)$$

Tiap-tiap unit tersembunyi ( $Z_j, j=1,2,3,\dots,p$ ) memperbaiki bias dan bobotnya ( $i=0,1,2,3,\dots,n$ ):

$$W_{ij}(\text{baru}) = w_{ij}(\text{lama}) + v_{ij} + \mu (w_{ij}(\text{lama}) - w_{ij}(\text{lama-1})) \dots \dots (2.14)$$

3. Hitung MSE

$$MSE = \frac{1}{n} \sum_{i=1}^n e_i^2 \dots \dots \dots (15)$$

Keterangan :

- N : Jumlah data pembelajaran
- $e_i^2$  : Selisih antara nilai target dengan nilai keluaran prediksi
- $X_i$  : Unit masukan i
- $x_i$  : Sinyal masukan dan keluaran pada unit masukan
- $Z_j$  : Unit tersembunyi j
- $z_{in_j}$  : Sinyal masukan terboboti/masukan untuk unit tersembunyi  $Z_j$
- $z_j$  : Sinyal keluaran/aktivasi dari unit tersembunyi  $Z_j$
- $v_{0j}$  : Bias pada unit tersembunyi j
- $v_{ij}$  : Bias pada unit tersembunyi j
- $Y_k$  : Unit keluaran k

- $Y_{in_k}$  : Sinyal masukan terboboti/masukan untuk unit keluaran  $Y_k$
- $y_k$  : Sinyal keluaran/aktivasi dari unit keluaran  $Y_k$
- $w_{0k}$  : Bias pada unit keluaran k
- $w_k$  : Informasi error/galat pada unit keluaran  $Y_k$  yang di propagasi balik ke unit tersembunyi
- $z_{in_j}$  : Jumlah delta masukan pada lapisan tersembunyi dari unit pada lapisan di atasnya/lapisan keluaran  $Y_k$
- $z_j$  : Informasi error/galat pada unit tersembunyi  $Z_j$
- $W_{jk}$  : Koreksi bobot antara lapisan keluaran  $Y_k$  dengan lapisan tersembunyi  $Z_j$
- $W_{0k}$  : Koreksi bias antara lapisan keluaran  $Y_k$  dengan lapisan tersembunyi  $Z_j$
- $v_{ij}$  : Koreksi bobot antara lapisan tersembunyi  $Z_j$  dengan lapisan masukan  $X_i$
- $W_{jk}$  : Bobot antara lapisan keluaran  $Y_k$  dengan lapisan masukan  $Z_j$  yang sudah disesuaikan
- $v_{ij}$  : Bobot antara lapisan tersembunyi  $Z_j$  dengan lapisan masukan  $X_i$  yang sudah disesuaikan
- $\alpha$  : Laju pembelajaran
- $\mu$  : Momentum

3. METODOLOGI PENELITIAN

Penelitian ini memiliki beberapa tahap penelitian. Proses penelitian yang pertama dilakukan adalah tahap studi literatur yaitu mengumpulkan materi-materi yang berkaitan dengan judul yang diambil. Kemudian tahap pengumpulan data, dimana tahap ini dilakukan pengumpulan data yang terdapat di lapangan. Kemudian tahap selanjutnya adalah pengolahan data, pada tahap ini data yang didapat diolah dan dimasukkan kedalam sistem yang diinginkan. Selanjutnya tahap implementasi yang kemudian masuk ke tahap pengujian untuk menguji keakuratan dari sistem jaringan syaraf tiruan yang telah dibuat,

apabila hasil keluaran yang diinginkan telah sama dengan target, maka sistem tersebut berhasil.

#### 4. PERANCANGAN SISTEM

##### 4.1 Perancangan Model JST

Perancangan model jaringan saraf tiruan adalah perancangan yang berisikan, penetapan masukan, penetapan keluaran dan menggambarkan arsitektur jaringan saraf tiruan yang akan digunakan. Pada perancangan model jaringan saraf tiruan akan dijelaskan nilai-nilai masukan, nilai target keluaran, dan konfigurasi arsitektur jaringan. Berikut ini penjelasan dari perancangan model jaringan saraf tiruan yang dilakukan.

##### A. Penetapan Masukan

Masukan yang digunakan di dalam aplikasi ini adalah sebanyak 23 buah. Masukan merupakan gejala-gejala penyakit yang terdapat pada tanaman jeruk pontianak. Selanjutnya adalah menentukan nilai-nilai dari masing-masing gejala. Nilai terhadap masukan yang diberikan 0 dan 1, dikondisikan bahwa tanaman mengalami gejala atau tidak. Nilai 0 berarti tidak mengalami gejala sedangkan nilai 1 berarti tanaman jeruk Pontianak mengalami gejala yang merupakan masukan dari sistem aplikasi ini. Pada penelitian ini, hanya ditetapkan bernilai 0 dan 1 karena tidak ada tingkatan kerusakan per gejala. Gejala bersifat terjangkau dan tidak terjangkau atau mengalami atau tidak mengalami. Berikut ini adalah tabel masukan masing-masing gejala penyakit.

Tabel 1. Tabel Variabel dan Nilai

Gejala-gejala	Variabel	Nilai
X1	Tidak	0
	Ya	1
X2	Tidak	0
	Ya	1
X3	Tidak	0
	Ya	1
X4	Tidak	0
	Ya	1

Tabel 1. Tabel Variabel dan Nilai  
(Lanjutan)

Gejala-gejala	Variabel	Nilai
X5	Tidak	0
	Ya	1
X6	Tidak	0
	Ya	1
	Ya	1
X9	Tidak	0
	Ya	1
X10	Tidak	0
	Ya	1
X11	Tidak	0
	Ya	1
X12	Tidak	0
	Ya	1
X13	Tidak	0
	Ya	1
X14	Tidak	0
	Ya	1
X15	Tidak	0
	Ya	1
X16	Tidak	0
	Ya	1
X17	Tidak	0
	Ya	1
X18	Tidak	0
	Ya	1
X19	Tidak	0
	Ya	1
X20	Tidak	0
	Ya	1
X21	Tidak	0
	Ya	1
X22	Tidak	0
	Ya	1
X23	Tidak	0
	Ya	1

##### B. Penetapan Keluaran

Setelah menetapkan variabel masukan selanjutnya menetapkan nilai target keluaran yang akan dihasilkan dari pendeteksian penyakit jeruk pontianak. Nilai target keluaran antara lain CVPD dengan nilai biner 0000, CPVD dan Tristeza dengan nilai biner 0001, Tristeza dengan nilai biner 0010, Diploida dengan nilai biner 0011, Diploida dan Embun

Tepung dengan nilai biner 0100, Embun Tepung dengan nilai biner 0101, Jamur Upas dengan nilai biner 0110, Jamur Upas dan Kanker dengan nilai biner 0111, Kanker dengan nilai biner 1000, Woody gall dengan nilai biner 1001, Woody gall dan Psoriasis dengan nilai biner 1010, Psoriasis dengan nilai biner 1011, Busuk Pangkal Batang dengan nilai biner 1100, Busuk Pangkal Batang dan Psorossi dengan nilai biner 1101, CVPD dan Diplodia dengan nilai biner 1110, serta Tristeza dan Diplodia dengan nilai biner 1111. Keluaran akan dibuat dalam 4 node biner. Tabel 2 berikut adalah tabel dari target nilai keluaran sistem.

Tabel 2. Tabel Nilai Target Keluaran

No	Nilai (Biner)	Jenis Penyakit
1	0000	CVPD
2	0001	CVPD dan Tristeza
3	0010	Tristeza
4	0011	Diploida
5	0100	Diploida dan Embun Tepung
6	0101	Embun Tepung
7	0110	Jamur Upas
8	0111	Jamur Upas dan Kanker
9	1000	Kanker
10	1001	Woody gall (puru berkayu)
11	1010	Woody gall dan psoriasis
12	1011	Psoriasis
13	1100	Busuk Pangkal Batang
14	1101	Busuk pangkal batang dan Psoriasis
15	1110	CVPD dan Diploida
16	1111	Tristeza dan Diploida

### C. Arsitektur Jaringan

Arsitektur jaringan pada jaringan saraf tiruan memiliki beberapa lapisan,

sesuai dengan jumlah neuron yang dibutuhkan. Sebelum membuat arsitektur jaringan, yang pertama harus ditentukan adalah konfigurasi dari jaringan syaraf tiruan tersebut. Untuk lebih jelasnya dapat dilihat pada tabel 3.

Tabel 3 Konfigurasi jaringan saraf tiruan

Parameter	Nilai
Jumlah masukan sistem	23
Jumlah lapisan tersembunyi	2
Jumlah sel lapisan tersembunyi	15,15
Jumlah keluaran sistem	4
Besar Galat	0,0001
Fungsi Aktivasi	<i>Sigmoid biner</i>
<i>Learning rate</i>	0,0001
Maksimum Iterasi	35000

Pada Tabel 3 dapat dijelaskan bahwa pada perancangan ini akan dibuat dengan 23 buah masukan dan arsitektur pertama dan kedua menggunakan 3 lapisan tersembunyi yang masing-masing terdiri dari 12, 24 dan 24 buah neuron. Fungsi aktivasi menggunakan sigmoid biner karena menggunakan rentang nilai sampai dengan 1. Besarnya galat = 0,0001, learning rate = 0,0001, dan maksimum iterasi = 35000

### 4.2. Perancangan Prosedural

Pada algoritma backpropagation terdapat 2 tahap pemrosesan, yaitu tahap pemrosesan pelatihan dan tahap pemrosesan pengujian. Untuk tahap pelatihan terdiri atas fase *feedforward*, fase *backpropagation* dan perhitungan MSE. Pada tahap pelatihan akan menghasilkan bobot-bobot baru yang nantinya akan digunakan untuk tahapan selanjutnya, yaitu tahap pengujian. Tahap pengujian hanya menggunakan fase *feedforward*.

#### A. Tahap Pelatihan

Pada tahap pelatihan proses pertama adalah menginisialisai bobot, menetapkan konfigurasi dari jaringan saraf tiruan yaitu maksimum epoch, target error, learning rate. Proses selanjutnya masuk ke alur maju atau *feedforward*, kemudian dilanjutkan

pada tahap *backpropagation*, dan menghitung MSE yang didapat, apabila nilai error yang didapat lebih besar dari target error yang telah ditentukan, maka proses pelatihan tersebut akan kembali pada tahap inisialisasi bobot, dan seterusnya sampai didapat nilai error yang lebih kecil dari target error yang telah ditentukan sebelumnya. Jika nilai yang diinginkan telah didapat, bobot tersebut tersimpan untuk dimasukkan pada tahap pengujian.

#### B. Tahap Pengujian

Pada tahap pengujian bobot yang didapat dari tahap pelatihan akan digunakan untuk pengujian. Kemudian data penyakit jeruk akan digunakan untuk menguji sistem yang telah dibuat. Tahap pelatihan ini hanya menggunakan fase feedforward, dan hasil yang didapat merupakan keluaran berupa deteksi penyakit.

#### 4.3. Perancangan Pengolahan Data

Jumlah data penyakit jeruk yang didapat saat penelitian sebanyak 87 buah data. Data-data tersebut untuk digunakan sebagai data pelatihan dan data pengujian. Berikut akan dijelaskan pembagian data pelatihan dan pengujian, data tersebut telah disesuaikan dengan nilai variabel penyakit jeruk pontianak.

##### A. Data Pelatihan

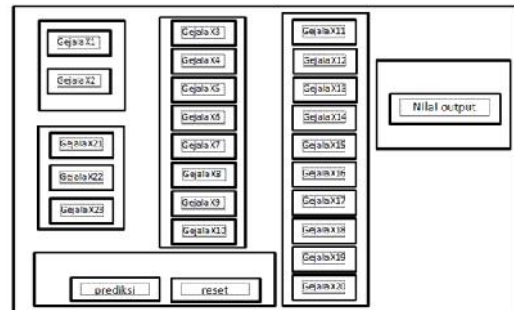
Data pelatihan merupakan data yang digunakan untuk melatih sistem JST yang telah dibuat, pada data pelatihan telah ditetapkan nilai target, jika hasil yang diperoleh pada pelatihan tidak sesuai dengan target, maka akan dilakukan perulangan pada sistem JST tersebut sehingga hasil pelatihan dapat mencapai target yang diinginkan.

##### B. Data Pengujian

Data pengujian merupakan, data yang digunakan untuk menguji yang telah dibangun sebelumnya dengan menggunakan data. Data pengujian berfungsi untuk menguji ke akuratan sistem JST yang telah dibuat. Data yang dipergunakan untuk tahap pengujian ini berjumlah 24 data agar kita dapat melihat hasil yang didapat selama pengujian yang dilakukan.

#### 4.4. Perancangan Penggunaan Antarmuka

Perancangan antarmuka dibuat dengan menggunakan GUI pada Matlab. Interface aplikasi dapat dilihat pada Gambar 2



Gambar 2. Tampilan Utama Aplikasi

Interface memiliki 1 buah form, pada form utama digunakan untuk mengisi data gejala penyakit jeruk pontianak yang ingin didiagnosa, dimana akan disikan 23 gejala penyakit. Berikut penjelasan tombol-tombol pada aplikasi tersebut, lihat pada Tabel 4.

Tabel 4. Tombol Fungsi

Tombol	Fungsi
Reset	Tombol ini digunakan untuk membersihkan <i>box</i> agar dapat dimasukkan data baru kembali.
Panel Output	Untuk mengetahui hasil kode penyakit yang dihasilkan
Input Data	Beberapa <i>box</i> yang harus diisi yaitu gejala-gejala penyakit jeruk pontianak
Prediksi	Hasil deteksi penyakit jeruk Pontianak

### 5. PENGUJIAN DAN ANALISIS HASIL

#### 5.1. Proses Pelatihan dan Pengujian

Proses pelatihan dan pengujian dilakukan dengan menggunakan 87 data gejala-gejala penyakit pada tanaman jeruk pontianak, dimana pada proses ini diambil 63 data untuk proses pelatihan dan 24 data untuk proses pengujian. Pada data-data tersebut menggunakan arsitektur jaringan saraf tiruan dengan 2 buah lapisan tersembunyi.

### A. Pelatihan Jaringan Saraf Tiruan

Pada program ini keluaran yang dihasilkan berupa kode dari deteksi penyakit jeruk pontianak, kode status gizi tersebut berupa 4 buah node, yaitu Y1, Y2, Y3 dan Y4 sebagai hasil keluaran. Selanjutnya menentukan formula dari jaringan saraf tiruan dari feedforward dengan cara mengatur fungsi aktivasi antara masukan ke lapisan tersembunyi dengan menggunakan logsig (sigmoid biner) dan fungsi aktivasi dari lapisan tersembunyi ke keluaran sistem menggunakan fungsi aktivasi logsig. Kemudian dimasukkan ke program dalam bentuk

Menentukan jumlah maksimum Epoch sangat berpengaruh terhadap kerja pada pelatihan. Nilai maksimum iterasi pada penelitian ini adalah 35000. Menentukan nilai target error yang menjadi nilai ukur untuk pemberhentian suatu proses pelatihan, proses pelatihan akan berhenti jika telah memenuhi target error yaitu 0,0001. Menentukan nilai learning rate agar proses dapat menjadi lebih cepat. Pada program ini nilai learning rate ditentukan sebesar 0,0001.

Menentukan nilai momentum, nilai momentum ini berfungsi untuk menyesuaikan nilai bobot yang lebih besar selama proses koreksi bobot, agar bobot dapat menyesuaikan dengan pola matriks masukan dan target yang ada. Setelah dilakukan beberapa percobaan penelitian

untuk nilai momentum, didapat nilai 0,9 sebagai nilai momentum terbaik. Kemudian di dalam Matlab akan dimasukkan dengan bentuk

```
jst.trainParam.epochs=35000;  
%maksimum epoch  
jst.trainParam.goal=0.0001; %target  
error  
jst.trainParam.lr=0.0001; %learning rate  
jst.trainParam.mc=0.9; %momentum  
jst.trainParam.show=1000; %tunjukkan  
perubahan setiap epoch 1000
```

Selanjutnya adalah melakukan pelatihan backpropagation dan mensimulasikan hasil dari pelatihan, pada tahap ini dilakukan coding untuk

melakukan pelatihan backpropagation dan coding untuk mensimulasikan hasil pelatihan di dalam perangkat lunak Matlab. Coding tersebut akan ditulis dalam bentuk

```
jst=train(jst, masukan, target);  
%melakukan pelatihan terhadap input  
dan target  
  
%melakukan simulasi  
keluaran = sim (jst, masukan);
```

Selanjutnya adalah menghitung MSE antara Target dan Hasil keluaran dengan memasukkan data coding untuk menghitung MSE antara target dan keluaran. Program tersebut ditulis dalam bentuk :

```
input_mse= y - target1; %koding  
untuk menampilkan nilai MSE  
MSE = mse(input_mse);  
fprintf('MSE_train=%12.8f\n',MSE)  
%menampilkan nilai MSE
```

Perubahan pada lapisan tersembunyi berpengaruh terhadap hasil prediksi yang dihasilkan, semakin banyak lapisan tersembunyi maka hasil prediksi semakin baik, tapi semakin banyak lapisan tersembunyi ini berpengaruh terhadap kinerja dari program tersebut, semakin banyak lapisan tersembunyi maka kerja program semakin lambat.

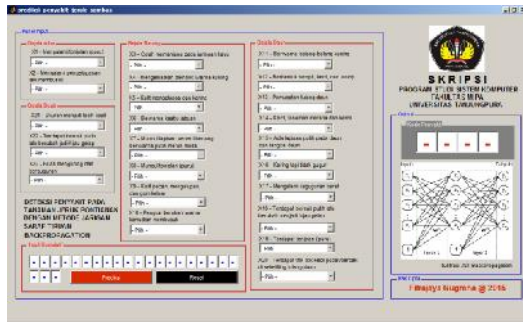
Pada proses pelatihan jaringan syaraf tiruan ini akan berhenti jika iterasi telah sampai pada batas maksimum yaitu 35000 iterasi, atau pelatihan akan berhenti jika nilai MSE (Mean Square Error) telah berada dibawah target error, disini diketahui target error sebesar 0,0001.

### B. Tampilan Aplikasi

Setelah melalui tahap pelatihan, dibuat interface untuk melakukan pengujian. Pembuatan interface menggunakan Matlab R2013a. Untuk lebih jelas dapat dilihat pada Gambar 5.1, terdapat beberapa bagian box yang mempunyai fungsi masing-masing. Beberapa box input data digunakan untuk memasukkan data gejala penyakit jeruk pontianak. Panel Kumulatif



digunakan sebagai parameter pengisian gejala penyakit jeruk pontianak. Box prediksi digunakan sebagai hasil dari deteksi penyakit jeruk pontianak, sedangkan output panel menampilkan kode penyakit jeruk pontianak. Jika hasil  $y < 0,5$  maka nilai binernya 0 sedangkan jika nilai  $y > 0,5$  maka nilai binernya 1.



Gambar 5. Tampilan Aplikasi

### C. Pengujian Aplikasi

Pengujian dilakukan dengan menggunakan 24 data pengujian. Dapat dilihat pada Tabel 5.1 hasil pengujian 26 data pengujian didapat keakuratan sebesar 83,33% dari data penelitian.

Tabel 6. Hasil Pengujian

Uji Ke-	Target Keluaran	Keluaran	Hasil
1	0000	0000	Sesuai
2	0001	0001	Sesuai
3	0010	0010	Sesuai
4	0011	0011	Sesuai
5	0011	0011	Sesuai
6	0100	0100	Sesuai
7	0100	0100	Sesuai
8	0101	0101	Sesuai
9	0110	0110	Sesuai
10	0111	0111	Sesuai
11	1000	1100	Tidak Sesuai
12	1001	1001	Sesuai
13	1001	1001	Sesuai
14	1010	1011	Tidak Sesuai
15	1010	1011	Tidak Sesuai
16	1011	1011	Sesuai

Tabel 6. Hasil Pengujian (Lanjutan)

Uji Ke-	Target Keluaran	Keluaran	Hasil
17	1100	1100	Sesuai
18	1100	1100	Sesuai
19	1101	1101	Sesuai
20	1101	1101	Sesuai
21	1110	1110	Sesuai
22	1110	1010	Tidak Sesuai
23	1111	1111	Sesuai
24	1111	1111	Sesuai

Dari hasil pengujian diatas nilai uji keluaran sesuai dengan nilai target yang diinginkan. Seperti yang sudah dijelaskan nilai target keluaran antara lain CVPD, CPVD dan trizteza, tristeza, diploidia, diploidia dan embun tepung, embun tepung, jamur upas, jamur upas dan kanker, kanker, woody gall, woody gall dan psorosis, psorosis, busuk pangkal batang, busuk pangkal batang dan psorosis, CVPD dan diploidia, tristeza dan diploidia jika disesuaikan dengan hasil uji dapat terlihat hasil keluaran adalah 83,33% keakuratannya dari data pengujian, sehingga dapat dikatakan jaringan saraf tiruan cukup dapat mendeteksi penyakit jeruk pontianak.

## 6. KESIMPULAN DAN SARAN

### 6.1 Kesimpulan

Dari pembahasan dapat disimpulkan bahwa tugas akhir penyakit jeruk pontianak menggunakan jaringan saraf tiruan backpropagation ini adalah :

1. Telah dibangun sebuah aplikasi untuk mendeteksi penyakit jeruk pontianak.
2. Jaringan syaraf tiruan backpropagation mampu mendeteksi penyakit jeruk pontianak dengan keakuratan 83,33% dari 24 data pengujian dari maksimum iterasi.
3. Jaringan syaraf tiruan metode backpropagation ini dapat digunakan untuk mendeteksi penyakit jeruk pontianak. Nilai tersebut diperoleh dari pelatihan 63 data dan pengujian 24 data menggunakan 2 buah hidden

layer, kemudian menggunakan target error sebesar 0,0001, dan learning rate = 0,0001 dengan MSE\_train= 0.0000999988

## 6.2 Saran

Saran untuk pengembangan sistem penyakit jeruk pontianak menggunakan jaringan syaraf tiruan backpropagation ini selanjutnya agar lebih baik lagi, yaitu :

1. Untuk pengembangan selanjutnya jumlah data pelatihan diperbanyak agar jika aplikasi ini digunakan pada deteksi penyakit jeruk dapat mendiagnosa penyakit jeruk pontianak.
2. Pengembangan selanjutnya sistem deteksi ini bukan hanya untuk penyakit tetapi juga hama yang terdapat pada jeruk.

## DAFTAR PUSTAKA

- [1] Dwi Andini, Titania, S.Kom dan Arifin, Jaenal. 2008. *Membangun Sistem Pakar Untuk Mengidentifikasi Jenis Penyakit pada Tanaman Jeruk Berbasis Wireless Application Protocol (WAP) Dengan Wireless Markup Language (WML) Dan Php Hypertext Preprocessor (PHP)* (Jurnal Ilmiah). Malang: Sekolah Tinggi Manajemen Informatika dan Komputer ASIA Malang.
- [2] Puspitaningrum, D., 2006. *Pengantar Jaringan Saraf Tiruan*. Yogyakarta : Penerbit Andi.
- [3] Prabowo, P.W., Rahmadya T.H., 2012. *Penerapan Soft Computing dengan MATLAB*. Bandung : Penerbit Rekayasa Sains.
- [4] Hermawan, A., 2006. *Jaringan Saraf Tiruan, Teori dan Aplikasi*. Yogyakarta : Penerbit Andi. Kusumadewi, S., 2004. *Membangun Jaringan Syaraf Tiruan (Menggunakan Matlab dan Excel Link)*. Yogyakarta : Penerbit Graha Ilmu.
- [5] Kusumadewi, S., 2004. *Membangun Jaringan Syaraf Tiruan (Menggunakan Matlab dan Excel Link)*. Yogyakarta : Penerbit Graha Ilmu.(5)
- [6] Siang, J.J., 2009. *Jaringan Saraf Tiruan dan Pemrogramannya Menggunakan Matlab*. Yogyakarta : Penerbit Andi.