

Pengenalan Pola Sidik Jari Berbasis Jaringan Syaraf Tiruan Perambatan Balik

Rahmawati Husen¹, Tole Sutikno², Ardi Pujianta³

^{1,3}Program Studi Teknik Informatika, Universitas Ahmad Dahlan

²Program Studi Teknik Elektro, Universitas Ahmad Dahlan

Abstrak

Pengenalan huruf, tanda tangan, ucapan, objek gambar dan sidik jari merupakan masalah kompleks yang sulit ditelusuri algoritmanya. Hal ini dikarenakan masalah tersebut tidak memiliki model matematis yang jelas dan pasti. Untuk itu dibutuhkan suatu teknik komputasi yang memiliki kemampuan untuk mempelajari dan mengenali sesuatu berdasarkan karakteristik yang dimiliki oleh masing-masing benda yang akan dikenali. Jaringan syaraf tiruan (JST) adalah teknik komputasi yang tepat karena teknik komputasi ini dikembangkan berdasarkan cara kerja sistem syaraf biologis manusia sehingga memiliki kemampuan untuk mempelajari dan mengenali sesuatu, sekalipun terdapat penyimpangan (noise). Subjek penelitian ini adalah pengenalan pola sidik jari dengan menggunakan jaringan syaraf tiruan perambatan balik. Perangkat lunak (software) yang digunakan dalam pembuatan aplikasi ini adalah borland delphi 7.0. Proses pengenalan yaitu dengan melatih pola sidik jari yang didapat dari hasil scan berukuran 80 * 80 piksel diambil dari sidik jari jempol tangan kanan. Pengenalan pola dilakukan per piksel pada citra pola input dan output. Prosentase yang diperoleh dari pelatihan jaringan syaraf tiruan untuk file citra yang sudah dilatih di atas 75 % sedangkan untuk file citra yang belum dilatih di bawah 75 %.

Kata Kunci: pengenalan pola sidik jari, jaringan syaraf tiruan perambatan balik

1. Pendahuluan

Jaringan syaraf tiruan adalah teknik komputasi yang tepat karena teknik komputasi ini dikembangkan berdasarkan cara kerja sistem syaraf biologis manusia sehingga memiliki kemampuan untuk mempelajari dan mengenali sesuatu, sekalipun terdapat penyimpangan (noise). Selanjutnya istilah jaringan syaraf tiruan ini disebut dengan JST.

Penelitian mengenai cara kerja sistem syaraf biologis manusia ini terus dilakukan sehingga tercipta berbagai macam model JST. Salah satunya adalah model jaringan propagasi balik atau perambatan balik [1-13].

Gambar pola yang akan dipakai dalam penelitian ini adalah pola sidik jari. Sidik jari adalah pola kulit pada ujung jari, sidik jari dapat digunakan untuk mengenali identitas orang yang memilikinya karena bentuknya khas untuk setiap orang, tidak berubah seumur hidup seseorang dan sangat sulit membuat tiruannya, kecuali mendapatkan kecelakaan yang serius. Cara untuk mendapatkan pola sidik jari, yaitu dengan mengusapkan tinta pada ujung jari dan menempelkannya pada sehelai kertas, sehingga pola sidik jari dapat diperiksa dengan lebih jelas.

2. Perancangan Antarmuka

Perancangan antarmuka (*interface*) yang digunakan dalam program aplikasi pengenalan pola sidik jari berbasis jaringan syaraf tiruan adalah antarmuka berbasis interaksi grafis yang berfungsi untuk memudahkan pengguna dalam hal pengoperasiannya diperlihatkan pada Gambar 1 dan Gambar 2.

Menu			
Citra Pola <i>Input</i>	Citra Pola <i>Output</i>	Hasil Pengujian dgn JST	Variabel Pelatihan Pola
Tombol Kontrol – Learn, Run, Reset, Load, Simpan, Keluar			

Gambar 1. Rancangan Tahap Pelatihan Pada *Form* Utama

Keterangan
<input type="button" value="OK"/>

Gambar 2. Rancangan *Form* Keterangan

3. Analisis Kebutuhan








Analisis kebutuhan merupakan suatu tahapan dimana terjadi komunikasi antara pengguna program (*user*) dengan komputer. Keberhasilan dari program tergantung dari beberapa hal, antara lain komponen untuk pembelajaran, kemampuan untuk mengenali sebuah pola dan kemampuan untuk mencocokkan pola. Hasil yang diharapkan pada penelitian ini adalah jaringan syaraf tiruan dapat mengenali pola sidik jari dengan menggunakan metode *perambatan balik*.









Fungsi-fungsi yang dibutuhkan pada analisis kebutuhan ini adalah analisis sistem yang dalam pembuatan perangkat lunaknya meliputi perancangan sistem seperti:









a) Pasangan *Input* Pelatihan








Pasangan *input* untuk pelatihan berupa data sidik jari sebanyak 30 *sample* citra sidik jari. *File* citra untuk *input* dan *output* yang akan dilatih adalah sama.

Tabel 1. *Input Pelatihan*

No	Nama File Citra (.bmp)	Bentuk Sidik Jari
1	Alan.bmp	
2	Amat.bmp	
3	Ana.bmp	
4	Ani.bmp	
5	Ema.bmp	
6	Feri.bmp	
7	Feti.bmp	

8	<i>Fia.bmp</i>	
9	<i>Fitria.bmp</i>	
10	<i>Ika.bmp</i>	
11	<i>Iksan.bmp</i>	
12	<i>Ina.bmp</i>	
13	<i>Inaya.bmp</i>	
14	<i>Iwan.bmp</i>	
15	<i>Leli.bmp</i>	








16	Lia.bmp	
17	Mery.bmp	
18	Nia.bmp	
19	Nuri.bmp	
20	Rina.bmp	
21	Rio.bmp	
22	Ririn.bmp	
23	Rizky.bmp	









24	Tiwi.bmp	
25	Try.bmp	
26	Wahyu.bmp	
27	Wati.bmp	
28	Wawan.bmp	
29	Wulan.bmp	
30	Yudi.bmp	






b) Pasangan Input Pengujian

Pasangan input untuk pengujian berupa data sidik jari sebanyak 50 *sample* citra sidik jari, yaitu 30 citra sidik jari yang sudah dilatih sebelumnya (pada Tabel 1 diatas) dan 20 citra sidik jari yang belum dilatih sebelumnya (pada Tabel 2).

Tabel 2. *Input* Pengujian

No	Nama <i>File</i> Citra (.bmp)	Bentuk Sidik Jari
1	Andi.bmp	
2	Bambang.bmp	
3	Damar.bmp	
4	Della.bmp	
5	Dian.bmp	
6	Didit.bmp	
7	Dika.bmp	

8	Dita.bmp	
9	Dodi.bmp	
10	Doni.bmp	
11	Ima.bmp	
12	Ira.bmp	
13	Ita.bmp	
14	Jodi.bmp	
15	Lutfi.bmp	

16	Mia.bmp	
17	Ria.bmp	
18	Tika.bmp	
19	Tina.bmp	
20	Tuti.bmp	

4. Implementasi Perangkat Lunak

Implementasi perangkat lunak merupakan tahap dimana sistem sudah jadi dan siap digunakan, sehingga pada tahap ini sistem sudah harus dapat dioperasikan sesuai dengan fungsi dan tujuan dibuatnya perangkat lunak tersebut. Apabila terdapat kesalahan-kesalahan pada sistem dapat diketahui pada saat implementasi perangkat lunak ini. Kesalahan sistem dapat terjadi karena kesalahan pada penulisan program, kesalahan logika ataupun kesalahan pada sistem operasi yang digunakan.

Perangkat lunak yang dibuat memiliki batasan-batasan dalam implementasinya yang lebih mengarahkan bentuk program aplikasi sesuai dengan rancangan program. Dalam pengembangan pembuatan perangkat lunak ini, diasumsikan bahwa perangkat lunak yang dibuat merupakan suatu program aplikasi (*software*) yang dapat digunakan untuk penelitian jaringan syaraf tiruan model propagasi balik (*perambatan balik*) dalam kasus pengenalan pola sidik jari dengan batasan-batasan seperti yang dijelaskan pada batasan masalah. Program ini dibangun dengan menggunakan tampilan yang diharapkan dapat mempermudah pengguna dalam pengoperasian program. Dengan asumsi bahwa setiap pengguna yang mengoperasikan program tidak harus orang yang terbiasa menggunakan komputer. Implementasi antarmukanya adalah sebagai berikut:

a) **Form Pembuka**

Form pembuka merupakan awal dari program aplikasi yang dibuat, *form* ini berisi judul program dan tombol-tombol untuk melanjutkan ke *form* berikutnya atau keluar dari program aplikasi ini (Gambar 3).



Gambar 3. Form Pembuka

b) **Form Utama**

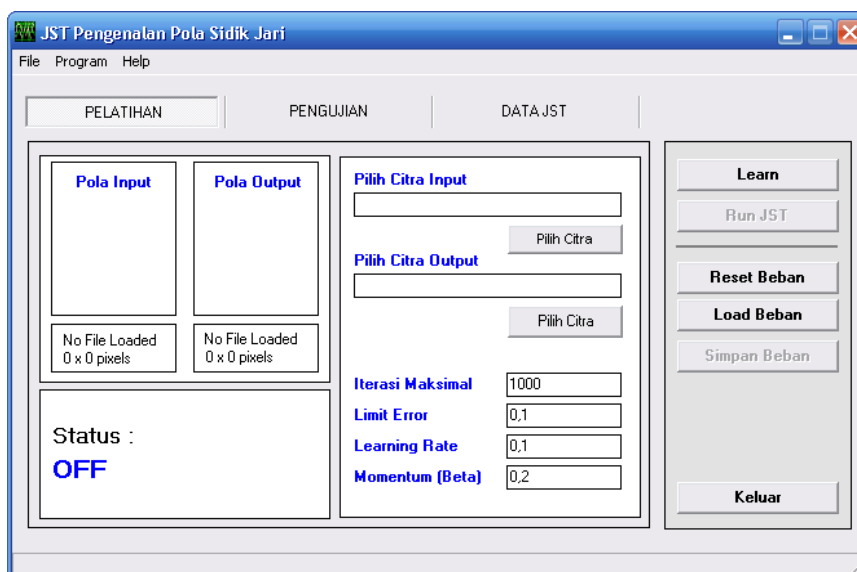
Form utama berisi pilihan menu yang berfungsi untuk membantu pengguna dalam mengoperasikan aplikasi, dan sebagai tempat dilakukannya pelatihan dan pengujian.

a. Menu

Adapun kontrol menu yang disediakan antara lain:

1. *File*, terdiri dari 3 sub menu yaitu open citra *input*, open citra *output*, dan *exit*.
2. *Program*, terdiri dari 5 sub menu yaitu mode pelatihan, mode pengujian, latihan, *run* JST, dan *reset* beban.
3. *Help*, terdiri dari 1 sub menu yaitu *about*

b. Pelatihan



Gambar 4. Form Utama - Pelatihan

Pelaksanaan pelatihan dalam aplikasi dilakukan oleh penggalan program dari prosedur yang berada pada tombol learn:

```

procedure TFormUtama.btnLearnClick(Sender: TObject);
var
  iterate : integer;
  i,j : integer;
  weightIn : double;
  weightH : double;
  inJST : double;
  outJST : double;

begin
  learning_rate:= strtfloat(EditLearningRate.Text);
  limitError := strtfloat(EditLimitError.Text);
  iterasiMax := strtoint(EditIterasiMaks.Text);
  beta := strtfloat(EditBeta.Text);
  lblProses.Caption := 'Pelatihan sedang dilakukan';
  SetBebanAwal(FormUtama);
  for i := 0 to 79 do
    for j := 0 to 79 do
      begin
        weightIn := bebanIn [i,j];
        weightH := bebanH [i,j];
        inJST := inputJST [i,j];
        outInput [i,j]:= inJST * weightIn;
        outJST := outputJST [i,j];
        iterate := 0;
        repeat
          net1:=(1/(1+exp((weightIn+outInput
            [i,j]) *-1)));
          net2:=(1/(1+exp((weightH+(net1*
            weightH))*-1)));
          delta := outJST - net2;
          weightIn:=weightIn+(learning_rate*inJST
            *delta)+(beta*delta);
          weightH:=weightH+(learning_rate*net1*
            delta);
          bebanIn [i,j] := weightIn;
          bebanH [i,j] := weightH;
          iterate := iterate + 1;
        until (iterate > iterasiMax) or (delta <
          limitError);
        lblProses.Caption:='Pelatihan Selesai';
        btnSimpanBeban.Enabled := true;

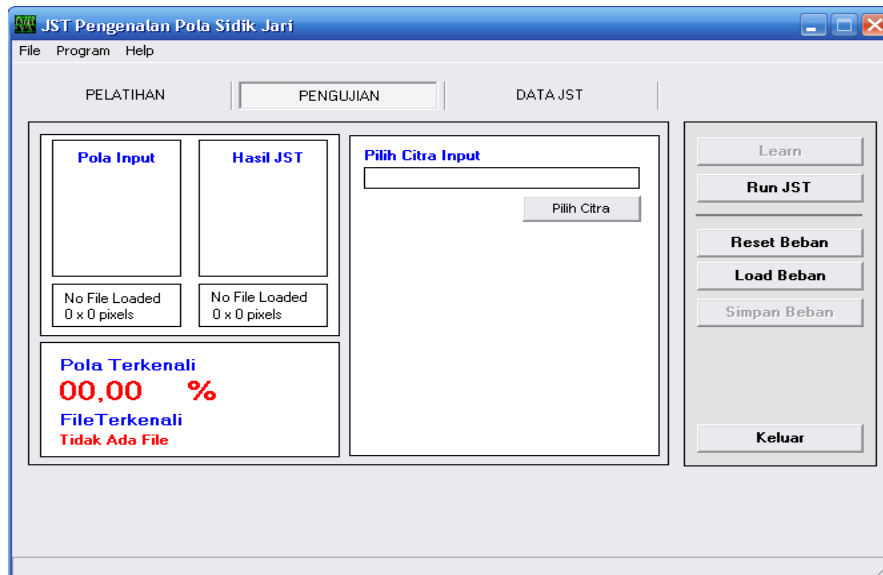
      end;
    SimpanCitra(FormUtama);
  end;
end;

```

keterangan kode program pada proses pelatihan:

1. Prosedur untuk tombol *learn* diberi nama dengan *procedure learn*.
2. Variabel-variabel yang dibutuhkan didefinisikan terlebih dahulu.
3. Nilai *learning rate*, *limit error*, jumlah iterasi maksimal dan *momentum* diambil.
4. *Net1* merupakan perhitungan nilai di *layer input*.
5. *Net2* merupakan perhitungan nilai di *layer hidden*.

6. *Delta* merupakan perhitungan nilai *error*
 7. Setelah proses pelatihan dilakukan maka hasil pelatihan tersebut tersimpan secara otomatis.
- c. Pengujian



Gambar 5. Form Utama - Pengujian

Pelaksanaan pengujian dalam aplikasi dilakukan oleh panggilan program dari prosedur yang berada pada tombol run JST:

```

OpenCitra(FormUtama);
for i := 0 to 79 do
  for j := 0 to 79 do
    begin
      weightIn := bebanIn [i,j];
      weightH := bebanH [i,j];
      inJST := inputJST [i,j];
      net1 := (1/(1+exp(( weightIn + (inJST * weightIn))*-
1)));
      net2 := (1/(1+exp(( weightH + (net1 * weightH))*-1)));
      hasilJST[i,j] := net2;
      stgHasil.Cells[i,j]:=formatfloat('#0.00',hasilJST[i,j]);
      deltaHasil := inJST - limitError;
      if (net1 >= deltaHasil) then
        begin
          dataHasil[i,j] :=0;
        end
      else
        begin
          dataHasil[i,j] :=1;
        end;
    end;
  end;

```

```

{ Pengenalan pola dan hitung prosentase pola yang terkenali }
nilaiRGB := imageJST.Canvas.Pixels[i,j];
dataAkurasi[i,j].in1 := GetRValue(nilaiRGB);
dataAkurasi[i,j].in2 := GetGValue(nilaiRGB);

```

```

dataAkurasi[i,j].in3 := GetBValue(nilaiRGB);
if ( dataAkurasi[i,j].in1 >= 255 )
  and ( dataAkurasi[i,j].in2 >= 255 )
  and ( dataAkurasi[i,j].in3 >= 255 ) then
begin
  if dataHasil[i,j] = mCitra[i,j] then
  begin
    polaCount := polaCount+1;
  end
  else
  begin
    polaCount := polaCount;
  end;
end;
end;
countPct1 := (polaCount/6400)*100;

{ Cari prosentase pola terkenali yang terbesar }
if countPct1 > countPct2 then
begin
  countPct2 := countPct1;
  noCitra2 := nomorCitra;
end;
end;
lblCountPCT.Caption := formatfloat('#0.00',countPct2);

{ Tampilkan hasil citra sesuai dengan prosentase pola yang terkenali }
nomorCitra := noCitra2;
OpenCitra(FormUtama);
lblFileKenalJST.Caption := namaFileKenal;
for i:=0 to 79 do
  for j :=0 to 79 do
  begin
    if mCitra[i,j] = 1 then
    begin
      ImageJST.Canvas.Pixels[i,j] := clBlack;
    end
    else
    begin
      ImageJST.Canvas.Pixels[i,j] := clWhite;
    end;
  end;
end;

```

c) Form About

Form about berisi judul program beserta data pembuat program.



Gambar 6. Form About

5. Pembahasan

Beberapa parameter jaringan syaraf tiruan *perambatan balik* yang akan mempengaruhi jaringan agar dapat bekerja dengan baik dalam mengenali pola masukan ditunjukkan dalam Tabel 3.

Tabel 3. Parameter Jaringan Syaraf Tiruan

Parameter
Iterasi maksimal (N) <i>Learning rate</i> <i>Momentum</i> (β)

Sesuai dengan analisis kebutuhan antarmuka yang telah dilakukan, maka aplikasi yang dibuat memiliki beberapa kontrol yang berfungsi untuk membantu dan memudahkan pengguna dalam mengoperasikan aplikasi yang telah dibuat. Adapun kontrol yang terdapat pada aplikasi yang dibuat meliputi:

1. Masukan (*input*)

Kontrol masukan yang disediakan berupa kontrol yang berfungsi untuk menerima data yang nantinya akan menjadi masukan untuk aplikasi dalam melakukan pelatihan dan pengujian pola. Beberapa kontrol masukan yang disediakan oleh aplikasi yang dibuat antara lain:

- Pilih citra *input*, berfungsi untuk memasukkan pola citra sidik jari yang ingin dikenali.
- Pilih citra *output*, berfungsi untuk memasukkan pola citra sidik jari yang akan menjadi output pada mode pelatihan.
- Iterasi maksimal, merupakan kontrol masukan untuk menentukan banyaknya jumlah pelatihan maksimal yang akan dilakukan.
- Limit error*, merupakan kontrol masukan untuk menentukan besar limit *error* yang digunakan.
- Learning rate*, merupakan kontrol masukan untuk menentukan nilai *learning rate* yang digunakan.
- Momentum* (*beta*), merupakan kontrol masukan untuk menentukan nilai *momentum* (*beta*) yang digunakan.

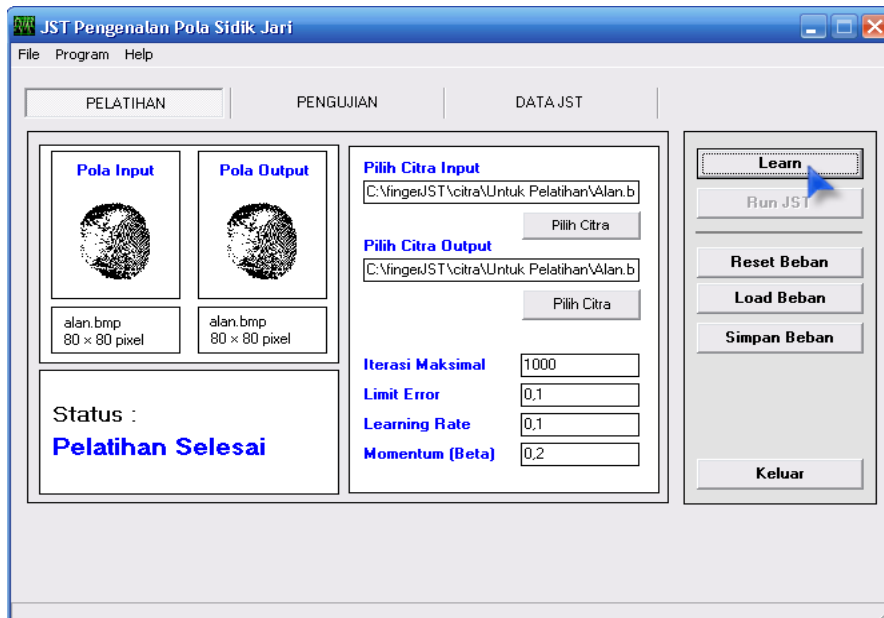
2. Keluaran

Untuk menampilkan keluaran yang dihasilkan oleh aplikasi tersebut, disediakan juga kontrol keluaran yang menampilkan hasil pengenalan pola sidik jari yang dikenali.

Untuk melihat unjuk kerja aplikasi berbasis jaringan syaraf tiruan yang telah dibuat, dilakukan proses pelatihan dan pengujian terhadap aplikasi yang telah dibuat. Hal ini dilakukan untuk mengetahui seberapa besar tingkat akurasi aplikasi dalam melakukan pengenalan pola sidik jari manusia.

a) Proses Pelatihan

Proses tahap belajar atau pelatihan merupakan tahap penyesuaian bobot-bobot jaringan. Dalam proses ini, data masukan (berupa pola citra *input* dan pola citra *output*, yang keduanya mempunyai *file* citra sama) diolah dan dilatih untuk melakukan pengenalan pola. Hasil keluaran pada proses ini adalah bobot-bobot koneksi antar *sel* dalam jaringan syaraf tiruan. Seperti tampak pada Gambar 7 dan Tabel 4.



Gambar 7. Hasil Pelatihan

Tabel 4. Data Hasil Pelatihan

No	Nama <i>File</i> Citra yg dilatih (.bmp)		File Hasil Pelatihan (tersimpan dgn nama)
	Input	Output	
1	Alan.bmp	Alan.bmp	Sidik01
2	Amat.bmp	Amat.bmp	Sidik02
3	Ana.bmp	Ana.bmp	Sidik03
4	Ani.bmp	Ani.bmp	Sidik04
5	Ema.bmp	Ema.bmp	Sidik05
6	Feri.bmp	Feri.bmp	Sidik06
7	Feti.bmp	Feti.bmp	Sidik07
8	Fia.bmp	Fia.bmp	Sidik08

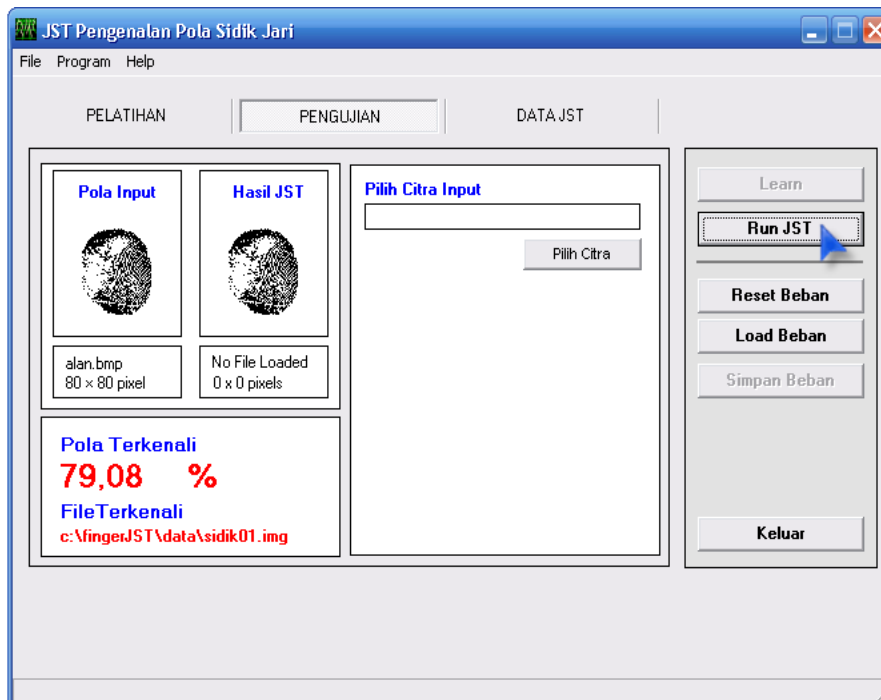
9	Fitria.bmp	Fitria.bmp	Sidik09
10	Ika.bmp	Ika.bmp	Sidik010
11	Iksan.bmp	Iksan.bmp	Sidik011
12	Ina.bmp	Ina.bmp	Sidik012
13	Inaya.bmp	Inaya.bmp	Sidik013
14	Iwan.bmp	Iwan.bmp	Sidik014
15	Leli.bmp	Leli.bmp	Sidik015
16	Lia.bmp	Lia.bmp	Sidik016
17	Mery.bmp	Mery.bmp	Sidik017
18	Nia.bmp	Nia.bmp	Sidik018
19	Nuri.bmp	Nuri.bmp	Sidik019
20	Rina.bmp	Rina.bmp	Sidik020
21	Rio.bmp	Rio.bmp	Sidik021
22	Ririn.bmp	Ririn.bmp	Sidik022
23	Rizky.bmp	Rizky.bmp	Sidik023
24	Tiwi.bmp	Tiwi.bmp	Sidik024
25	Try.bmp	Try.bmp	Sidik025
26	Wahyu.bmp	Wahyu.bmp	Sidik026
27	Wati.bmp	Wati.bmp	Sidik027
28	Wawan.bmp	Wawan.bmp	Sidik028
29	Wulan.bmp	Wulan.bmp	Sidik029
30	Yudi.bmp	Yudi.bmp	Sidik030

b) Proses Pengujian

Proses pengujian atau pengenalan pola merupakan tahap dimana aplikasi melakukan pengenalan pola sidik jari sesuai dengan pola *input* yang dimasukkan. Pola *input* yang dimasukkan diolah dan diproses, kemudian dibandingkan dengan pola yang telah dilatih sebelumnya yang hasilnya berupa bobot-bobot koneksi antar *sel*, sehingga memberikan keluaran akhir berupa pola sidik jari yang dikenali, prosentase kebenarannya, dan nama *file* citra yang terkenali.

a. Pola citra sidik jari yang sudah pernah dilatih

Dari pelatihan yang telah dilakukan maka didapatkan hasil pengujian pola citra sidik jari seperti tampak pada Gambar 8 dan Tabel 5.



Gambar 8. Hasil Pengujian
(Untuk File Citra Yang Sudah Dilatih)

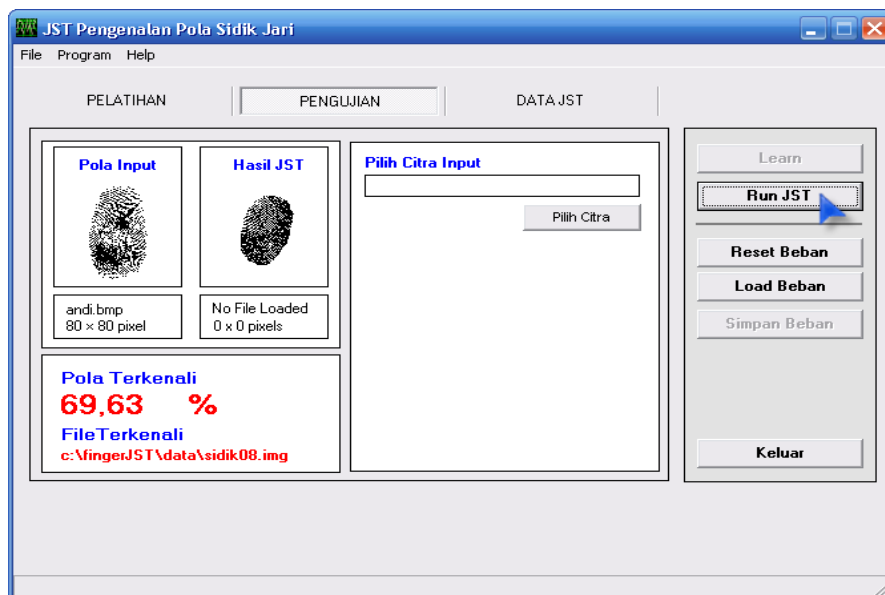
Tabel 5. Data Hasil Pengujian
(Untuk File Citra Yang Sudah Dilatih)

No	Nama File Citra (.bmp)	Prosentase	Nama File Yang Terkenali
1	Alan.bmp	79,08 %	Sidik01
2	Amat.bmp	80,22 %	Sidik02
3	Ana.bmp	77,31 %	Sidik03
4	Ani.bmp	82,97 %	Sidik04
5	Ema.bmp	77,31 %	Sidik05
6	Feri.bmp	83,77 %	Sidik06
7	Feti.bmp	77,22 %	Sidik07
8	Fia.bmp	80,39 %	Sidik08
9	Fitria.bmp	80,22 %	Sidik09
10	Ika.bmp	79,97 %	Sidik010
11	Iksan.bmp	77,03 %	Sidik011
12	Ina.bmp	81,83 %	Sidik012
13	Inaya.bmp	77,22 %	Sidik013
14	Iwan.bmp	89,73 %	Sidik014
15	Leli.bmp	82,97 %	Sidik015
16	Lia.bmp	83,77 %	Sidik016
17	Mery.bmp	76,81 %	Sidik017
18	Nia.bmp	78,56 %	Sidik018
19	Nuri.bmp	76,28 %	Sidik019
20	Rina.bmp	80,95 %	Sidik020
21	Rio.bmp	81,50 %	Sidik021
22	Ririn.bmp	80,95 %	Sidik022
23	Rizky.bmp	89,78 %	Sidik023

24	Tiwi.bmp	82,97 %	Sidik024
25	Try.bmp	80,39 %	Sidik025
26	Wahyu.bmp	89,78 %	Sidik026
27	Wati.bmp	81,73 %	Sidik027
28	Wawan.bmp	84,72 %	Sidik028
29	Wulan.bmp	80,67 %	Sidik029
30	Yudi.bmp	81,73 %	Sidik030

Dari hasil pengujian untuk *file* citra yang sudah dilatih sebelumnya diatas, maka dapat disimpulkan sebagai berikut:

1. Program mampu mengenali pola sidik jari yang sudah dilatih sebelumnya.
 2. Prosentase pengujiannya diatas 75 %.
 3. Program mampu menampilkan nama *file* terkenal.
- b. Pola citra sidik jari yang belum pernah dilatih
Untuk pengujian pola citra sidik jari yang belum pernah dilatih sebelumnya, hasilnya tampak pada Gambar 9 dan Tabel 6.



Gambar 9. Hasil Pengujian
(Untuk *File* Citra Yang Belum Dilatih)

Tabel 6. Data Hasil Pengujian
(Untuk *File* Citra Yang Belum Dilatih)

No	Nama <i>File</i> Citra (.bmp)	Prosentase	Nama <i>File</i> Yang Terkenali
1	Andi.bmp	69,63 %	Sidik08
2	Bambang.bmp	71,19 %	Sidik030
3	Damar.bmp	70,98 %	Sidik030
4	Della.bmp	72,92 %	Sidik019
5	Dian.bmp	73,97 %	Sidik030
6	Didit.bmp	72,00 %	Sidik030
7	Dika.bmp	71,45 %	Sidik030

8	Dita.bmp	72,55 %	Sidik08
9	Dodi.bmp	70,23 %	Sidik08
10	Doni.bmp	66,75 %	Sidik08
11	Ima.bmp	71,39 %	Sidik019
12	Ira.bmp	74,30 %	Sidik019
13	Ita.bmp	68,98 %	Sidik08
14	Jodi.bmp	73,52 %	Sidik08
15	Lutfi.bmp	72,55 %	Sidik027
16	Mia.bmp	72,58 %	Sidik019
17	Ria.bmp	72,11 %	Sidik019
18	Tika.bmp	71,00 %	Sidik019
19	Tina.bmp	70,58 %	Sidik08
20	Tuti.bmp	70,91 %	Sidik019

Dari hasil pengujian untuk *file* citra yang belum pernah dilatih sebelumnya diatas, maka dapat disimpulkan sebagai berikut:

1. Program mampu melakukan pengenalan pola yang belum pernah dilatih sebelumnya tetapi hasil pengenalan pola yang dilakukan adalah pengenalan pola berdasarkan pelatihan yang telah dilakukan sebelumnya.
2. Prosentase pengujiannya dibawah 75 %.
3. Nama *file* yang terkenali adalah *file-file* hasil pelatihan yang telah dilakukan sebelumnya.

c) Pengujian

Pengujian sistem yang dilakukan menggunakan *black box test*. Pengujian ini untuk memperoleh gambaran kesesuaian antara *input* dan *output*. Dalam pengujian ini, sistem dianggap sebagai kotak hitam yang memiliki pintu *input* dan *output*. Untuk keperluan pengujian, unjuk kerja sistem diberikan masukan berupa data yang dibutuhkan (dalam penelitian ini adalah *file* citra sidik jari dengan ukuran 80 x 80 *pixel*) dan diamati hasilnya. Adapun hasil pengujiannya dengan menggunakan *black box test* terdapat pada Tabel 7.

Tabel 7. Hasil Pengujian *Black Box Test*

No	Fungsi	Penilaian	
		Ya	Tidak
1	Program dapat mengenali pola dan masukan pelatihan	X	
2	Program dapat mengenali target dalam pelatihan	X	
3	Laju belajar dan <i>momentum</i> dapat mempercepat proses pelatihan	X	
4	Program dapat mengenali sidik jari yang sudah dilatih sebelumnya	X	
5	Program mampu melakukan pengenalan pola yang belum pernah dilatih sebelumnya tetapi hasil pengenalan pola yang dilakukan adalah pengenalan pola berdasarkan pelatihan yang telah dilakukan sebelumnya.	X	
6	Program dapat menampilkan prosentasi pengujian dan nama <i>file</i> terkenali	X	

Dari hasil pengujian *black box test* didapat hasil penelitiannya adalah:

1. Program mampu mengenali pola, masukan dan target dalam pelatihan.
2. Program mampu mengenali pola sidik jari yang sudah dilatih terlebih dahulu dalam pengujian.
3. Program dapat menampilkan prosentasi pengujian dan nama *file* terkenal.

6. Simpulan

Jaringan syaraf tiruan model perambatan balik dapat dijadikan sebagai alternatif pengenalan pola sidik jari. Hasil penelitian menunjukkan bahwa hasil prosentase pengenalan dari sistem yang dibangun dapat mengenali pola sidik jari di atas 75 % untuk file citra yang sudah dilatih. Kelebihan dari program aplikasi pengenalan pola sidik jari ini adalah program mampu mengenali pola sidik jari sangat baik, program bekerja dalam sistem operasi windows atau berbasis grafis sehingga interface lebih menarik dan proses kerjanya sangat cepat, dan program dapat menampilkan nama file terkenal. Kekurangan dari program aplikasi pengenalan pola sidik jari ini adalah program belum dapat menentukan kepemilikan pola sidik jari secara akurat, hanya berdasarkan prosentase kemiripan pola sidik jari dan belum bisa menampilkan secara langsung nama file atau pemilik dari sidik jari yang diuji.

Daftar Pustaka

- [1] Alam, M.A.J., 2003, Mengolah Database dengan Borland Delphi 7, PT Elex Media Komputindo, Jakarta.
- [2] Amanda, Said F., 2004, JST Hopfield Untuk Pengenalan Pola Sidik Jari Manusia, Skripsi S-1, Universitas Islam Indonesia, Yogyakarta.
- [3] Hapnaswati, Riska Riyani., 2004, Koreksi Galat Tunggal Sandi Siklis Dengan Jaringan Syaraf Tiruan Backpropagation, Skripsi S-1, Universitas Ahmad Dahlan, Yogyakarta.
- [4] Haykin, S., 1994, Neural Networks: A Comprehensive Foundation, Macmillan College Publishing Co., New York.
- [5] Kadir, Abdul., 2001, Dasar Pemrograman Delphi 5.0, Andi, Yogyakarta.
- [6] Kristanto, Andri., 2004, Jaringan Syaraf Tiruan, Gava Media, Yogyakarta.
- [7] Kusumadewi, Sri., 2003, Artificial Intellegence (Teknik dan Aplikasi), Graha Ilmu, Yogyakarta.
- [8] Kusumadewi, Sri., 2004, Membangun Jaringan Syaraf Tiruan Menggunakan Matlab & Excel Link, Graha Ilmu, Yogyakarta.
- [9] Muslim, 2004, Implementasi Jaringan Syaraf Tiruan Dengan metode Backpropagation Untuk Mendeteksi Penyakit THT, Skripsi S-1, Universitas Ahmad Dahlan, Yogyakarta.
- [10] Sutikno, Tole; Fathurrakhman, Andhy. Pengenalan Pola Alphabet Tulisan Tangan Secara On-line dengan JST-BP. TELKOMNIKA. 2005; 3(2): 65-71.
- [11] Putranto, Yulianto Tejo., 1998, Pengenalan Huruf Dengan Jaringan Syaraf Tiruan Tak Ubah Terhadap Translasi, Rotasi dan Skala, Tesis S-2, Universitas Gadjah Mada, Yogyakarta.
- [12] Risnely, Dian., 2004, Klasifikasi Karakteristik Konsumen Dengan Jaringan Syaraf Tiruan Backpropagation, Skripsi S-1, Universitas Ahmad Dahlan, Yogyakarta.
- [13] Wahono, Fery., 2001, Pengenalan Dan Perbandingan Pola Sidik Jari Menggunakan Metode Deteksi Minutiae, Skripsi S-1, Universitas Kristen Duta Wacana, Yogyakarta.