

BAB IV

HASIL DAN PENGUJIAN

Bab ini menjelaskan tentang seluruh hasil pengujian dari aplikasi yang telah dibuat. Pengujian yang dilakukan adalah menguji proses *region of interest* (ROI) yaitu suatu proses membuang bagian dari citra yang tidak diperlukan untuk diolah menjadi input, supaya fokus ekstraksi hanya pada bagian yang terpenting dari citra saja. Menguji proses *resize* yaitu proses mengubah ukuran citra hasil ROI menjadi citra baru berukuran 64x64 piksel. Menguji hasil ekstraksi citra menggunakan *diagonal distance feature extraction* yaitu ekstraksi yang dilakukan dengan cara menghitung jarak dari sudut siku citra hingga bertemunya piksel berwarna hitam. Menguji hasil ekstraksi citra menggunakan *longest run feature extraction* yaitu ekstraksi yang dilakukan dengan cara membagi citra menjadi sembilan sub area citra, kemudian dari tiap sub area citra dicari jumlah dari piksel terpanjang serta berturut-turut di seluruh baris, kolom dan dua diagonalnya. Terakhir dilakukan pengujian mengenai hasil pengenalan Angka Jawa menggunakan sampel *training* dan sampel *testing*.

4.1 Pengujian *Region Of Interest* (ROI)

Pengujian dilakukan setelah *participant* menggambarakan sebuah citra berbentuk pola Angka Jawa pada kanvas virtual berukuran 256x256 piksel. Kemudian *scanning* pada citra dilakukan untuk mencari area piksel yang akan di crop.

4.1.1 Tujuan

Pengujian ini bertujuan untuk mengetahui apakah area yang di *crop* adalah bagian terpenting dari citra dan juga apakah hasil *cropping* citra sudah berbentuk persegi.

4.1.2 Peralatan Yang Digunakan

A. Perangkat Keras

Peralatan yang digunakan untuk menguji *region of interest (ROI)* adalah sebuah *personal computer* yang memiliki spesifikasi seperti berikut:

- Processor : Intel, Core i3 @2.10Ghz (4CPUs)
- Memory : 4 GB
- *Operating System* : Windows 7 Ultimate – 64 Bit

B. Perangkat Lunak

Perangkat lunak pendukung dalam menguji *region of interest (ROI)* adalah sebagai berikut:

- *Project Tugas Akhir* penulis digunakan untuk menghasilkan dan mengetahui ukuran *cropping* citra *ROI*.
- Photoshop cs6 digunakan untuk mengetahui resolusi hasil *crop* secara manual.

4.1.3 Pengujian Region Of Interest (ROI)

A. Prosedur pengujian

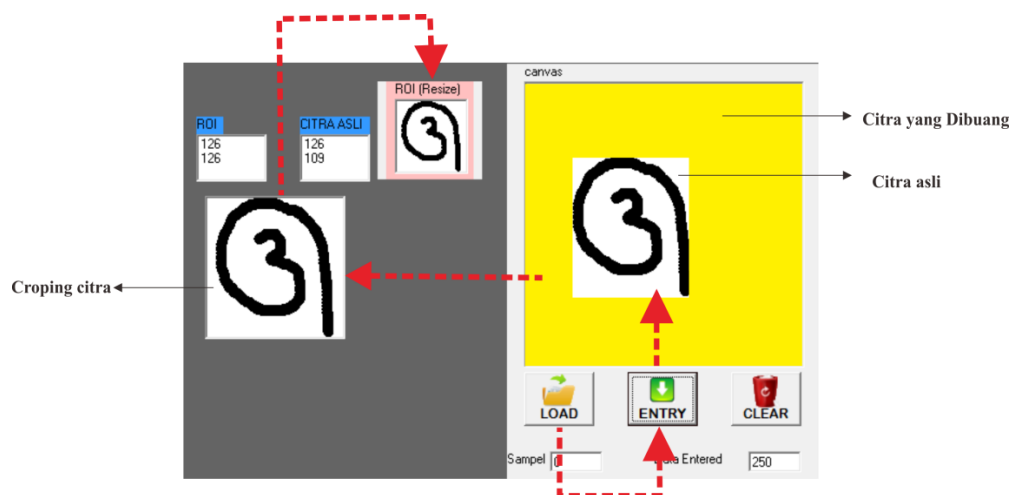
1. Jalankan *project* tugas akhir *.vbp
2. Masukkan (*Load*) citra Angka Jawa pada direktori, seperti contoh gambar dibawah ini:



Gambar 4.1 Citra Angka Jawa

3. Klik *entry*

B. Hasil pengujian:



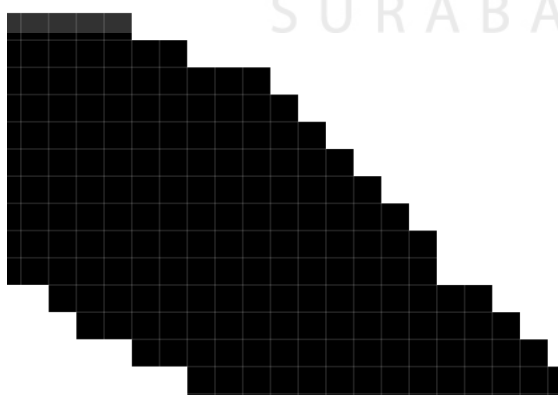
Gambar 4.2 Hasil uji ROI pada aplikasi

Area kuning pada gambar 4.2 adalah area yang dibuang. Citra asli menunjukkan berukuran 109x126 piksel. Setelah didapatkan citra asli selanjutnya dilakukan *crop* secara persegi. Sehingga ROI melakukan *crop* dengan ukuran 126x126 piksel. Sampai disini citra telah siap untuk diproses pada tahap *resize*.

4.1.4 Pengujian Region Of Interest (ROI) Secara Manual

A. Prosedur pengujian

1. Jalankan aplikasi adobe photoshop cs6
2. Klik *file* kemudian pilih *open*
3. Masukkan gambar yang sama seperti yang digunakan untuk pengujian pada sub bab 4.1.3
4. Lakukan *zooming* sampai maksimal dengan menekan tombol “CTRL dan + “ pada keyboard secara bersamaan sampai piksel citra terlihat jelas seperti gambar dibawah ini:

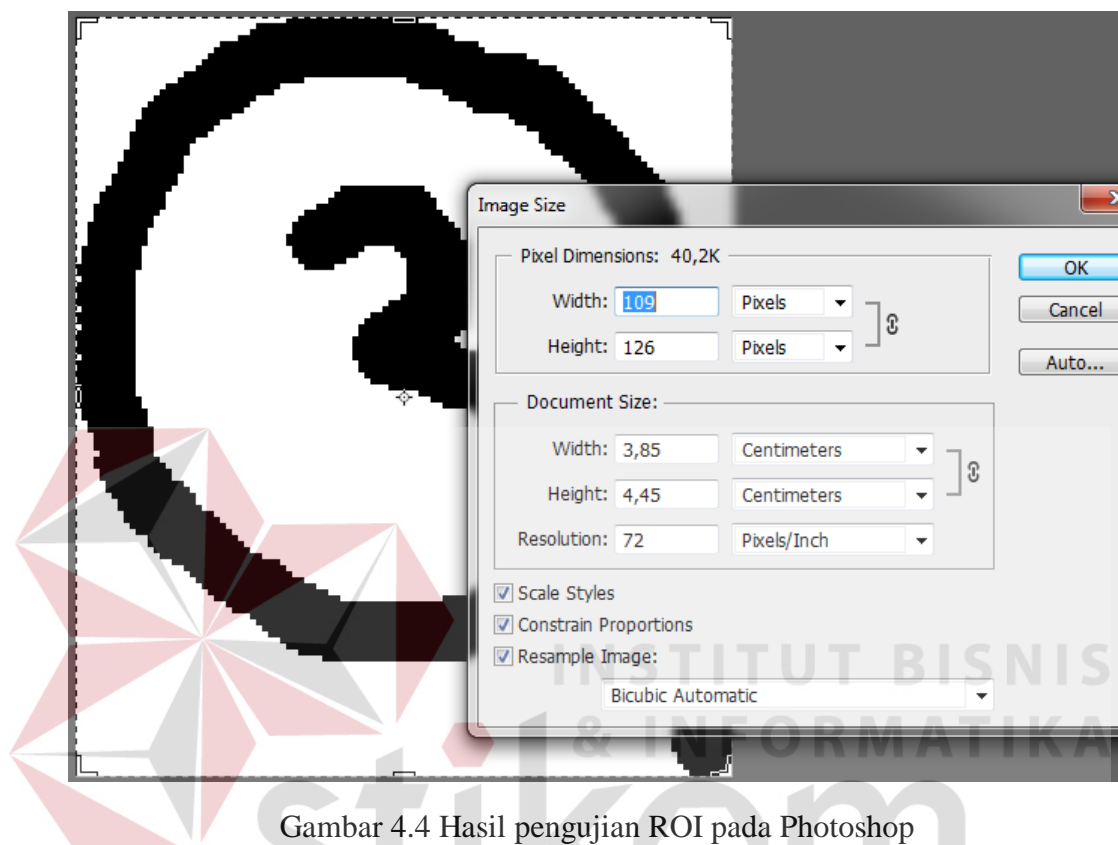


Gambar 4.3 *Zooming* citra pada photoshop

4. Lakukan *crop* secara manual menggunakan *tool crop* dengan melihat batas-batas piksel terluar pada citra

5. Kemudian klik pada menu *image*, pilih *image size*

B. Hasil pengujian:



Gambar 4.4 Hasil pengujian ROI pada Photoshop

Hasil *crop* secara manual menggunakan photoshop menunjukkan bahwa *crop* menghasilkan citra berukuran 109 x 126 piksel. Dari hasil pengujian ini dapat diambil kesimpulan bahwa *Region Of Interest (ROI)* telah menghasilkan *crop* citra dengan benar.

4.2 Pengujian *Resize*

Penentuan ukuran hasil *resize* akan berpengaruh pada hasil fitur yang akan didapatkan. Jika citra *dirresize* dengan ukuran terlalu kecil maka akan menyebabkan perubahan bentuk karakter yang sangat jauh perbedaannya dari bentuk karakter yang semestinya. Hal ini juga dapat menyebabkan kemiripan fitur antara satu Angka Jawa dengan Angka Jawa yang lainnya menjadi lebih banyak. Namun jika *resize* terlalu besar maka akan menyebabkan proses ekstraksi akan berjalan semakin lama karena terlalu banyak piksel yang harus di *scanning*. Pemilihan hasil *resize* citra menjadi berukuran 64x64 piksel ditentukan berdasarkan penelitian yang pernah dilakukan (Nibaran dan dkk, 2006). Citra yang sudah di *ROI* kemudian *dirresize* menjadi berukuran 64 x 64 piksel. Proses *resize* dilakukan agar semua citra sampel nantinya memiliki ukuran sama saat diekstrak.

4.2.1 Tujuan

Pengujian ini bertujuan untuk mengetahui resolusi citra yang telah *dirresize*. *Resize* harus menghasilkan output citra dengan resolusi 64x64 piksel.

4.2.2 Peralatan Yang Digunakan

A. Perangkat Keras

Peralatan yang digunakan untuk menguji *resize* adalah sebuah *personal computer* yang memiliki spesifikasi seperti berikut:

- Processor : Intel, Core i3 @2.10Ghz (4CPUs)
- Memory : 4 GB

- *Operating System* : Windows 7 Ultimate – 64 Bit

B. Perangkat Lunak

Perangkat lunak pendukung dalam menguji *resize* citra adalah sebagai berikut:

- Tugas akhir penulis digunakan untuk menghasilkan dan mengetahui resolusi citra yang telah di *resize*.
- Photoshop cs6 digunakan untuk mengetahui resolusi citra hasil *resize* secara manual.

4.2.3 Pengujian *Resize*

A. Prosedur Pengujian

Prosedur pengujian adalah seperti langkah-langkah berikut ini:

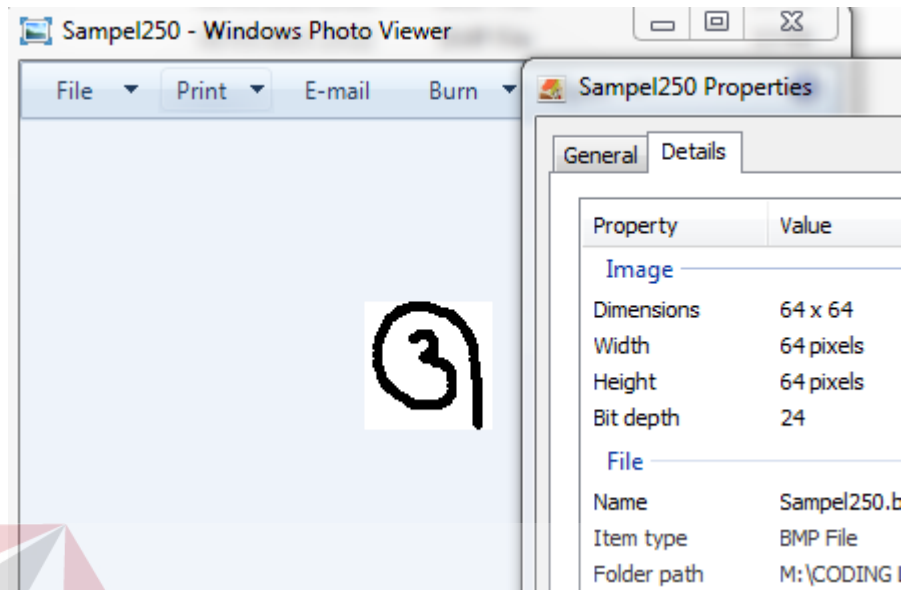
1. Jalankan *project* tugas akhir*.vbp
2. Klik *load* untuk masukkan citra yang akan diuji



Gambar 4.5 Citra uji

3. Klik tombol *entry*
4. Buka file citra yang dihasilkan *resize* aplikasi pada direktori yang telah ditentukan
5. Klik kanan, pilih *properties*, kemudian pilih menu *detail*

B. Hasil Pengujian



Gambar 4.6 Hasil pengujian *resize*

Pada gambar diatas menunjukkan *dimensions* atau resolusi citra berukuran 64x64 piksel, ini menunjukkan bahwa *Resize* telah menghasilkan citra dengan resolusi yang benar. Sebagai perbandingan, citra yang sama diujikan secara manual menggunakan adobe photoshop cs6 dibawah ini.

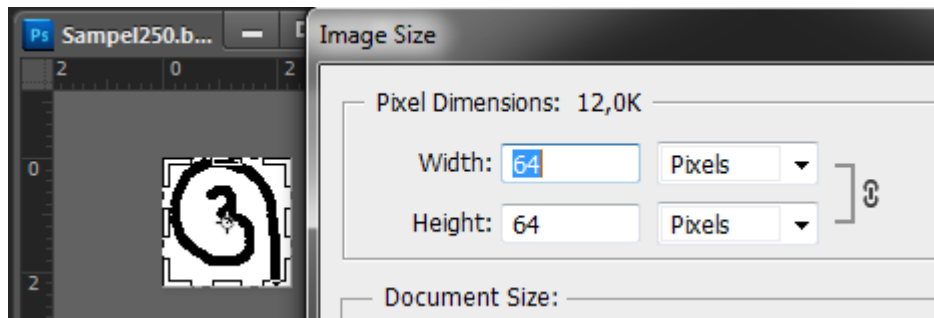
4.2.4 Pengujian *Resize* Secara Manual

A. Prosedur Pengujian

Prosedur pengujian adalah seperti langkah-langkah dibawah ini:

1. Buka Aplikasi Adobe photoshop cs6
2. Klik *file* kemudian *open*
3. Masukkan citra yang akan diujikan
4. Pilih menu *image* , kemudian klik *image size*

B. Hasil Pengujian



Gambar 4.7 Hasil pengujian *resize* pada photoshop

Hasil pengujian secara manual menggunakan adobe photoshop menunjukkan hasil yang sama seperti pada pengujian pada bab 4.2.3. Citra telah diresize dengan resolusi 64x64 piksel. Dapat disimpulkan bahwa proses *resize* citra pada aplikasi tugas akhir telah menghasilkan output citra dengan resolusi yang benar.

4.3 Pengujian *Longest Run Feature Extraction*

Citra yang telah melalui *ROI* dan *resize* selanjutnya akan diolah untuk mendapatkan fitur-fiturnya. Nilai fitur yang dihasilkan dari metode *longest run feature* berjumlah 36 fitur. Fitur didapatkan dengan cara mencari jumlah piksel hitam terpanjang serta berurutan di seluruh baris, kolom serta dua diagonal dari sub area citra.

4.3.1 Tujuan

Pengujian ini bertujuan untuk mengetahui apakah ekstraksi *longest run feature* telah menghasilkan *output* fitur yang benar.

4.3.2 Peralatan yang digunakan

A. Perangkat Keras

Peralatan yang digunakan untuk menguji *longest run feature extraction* adalah sebuah *personal computer* yang memiliki spesifikasi seperti berikut:

- Processor : Intel, Core i3 @2.10Ghz (4CPUs)
- Memory : 4 GB
- *Operating System* : Windows 7 Ultimate – 64 Bit

B. Perangkat Lunak

Perangkat lunak pendukung dalam menguji metode *longest run feature extraction* adalah sebagai berikut:

- *Project* tugas akhir penulis digunakan untuk menghasilkan fitur citra melalui aplikasi yang telah dibuat.
- Matlab digunakan untuk membuat citra menjadi biner.
- Microsoft Excel 2010 digunakan untuk menyimpan dan menghitung nilai biner dari matlab, serta digunakan juga untuk mencari nilai fitur *longest run* secara manual.

4.3.3 Pengujian Ekstraksi *Longest Run*

A. Prosedur Pengujian

Prosedur pengujian adalah seperti langkah-langkah berikut ini:

1. Jalankan *project* tugas akhir*.vbp

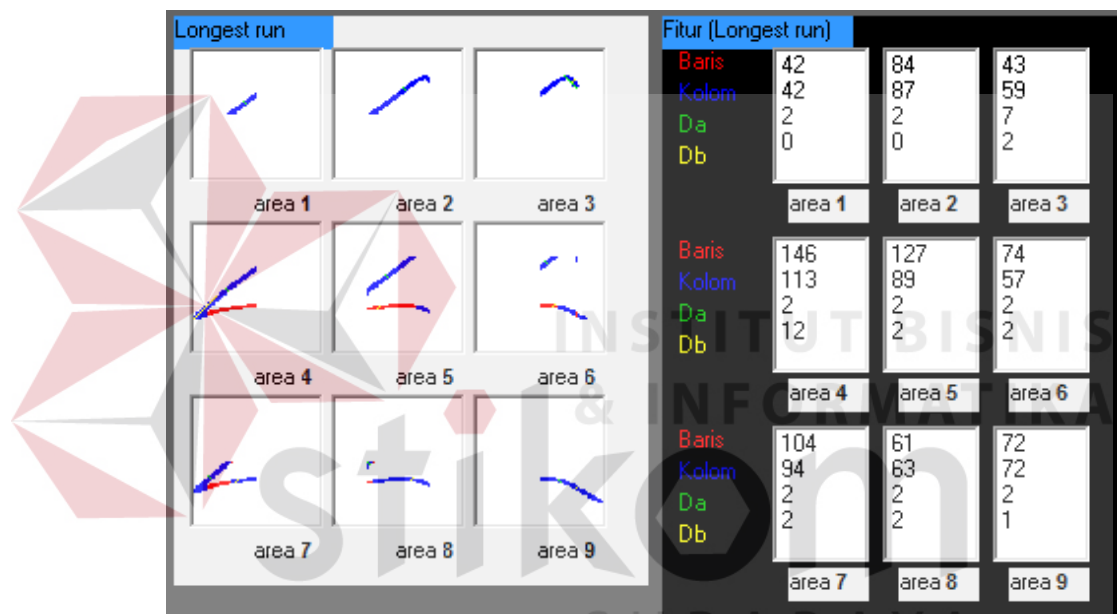
- Masukkan (*load*) sampel citra yang akan di uji, seperti gambar dibawah ini:



Gambar 4.8 Sampel uji *Longest Run Feature*

- Klik *entry*

B. Hasil Pengujian



Gambar 4.9 Hasil uji ekstraksi *longest run*

Seluruh hasil ekstraksi dari tiap sub area citra ditampilkan pada gambar 4.9.

Selanjutnya, dengan menggunakan sampel uji yang sama akan dilakukan pengujian secara manual menggunakan matlab dan microsoft excel.

4.3.4 Pengujian Ekstraksi *Longest Run* Secara Manual

A. Prosedur Pengujian

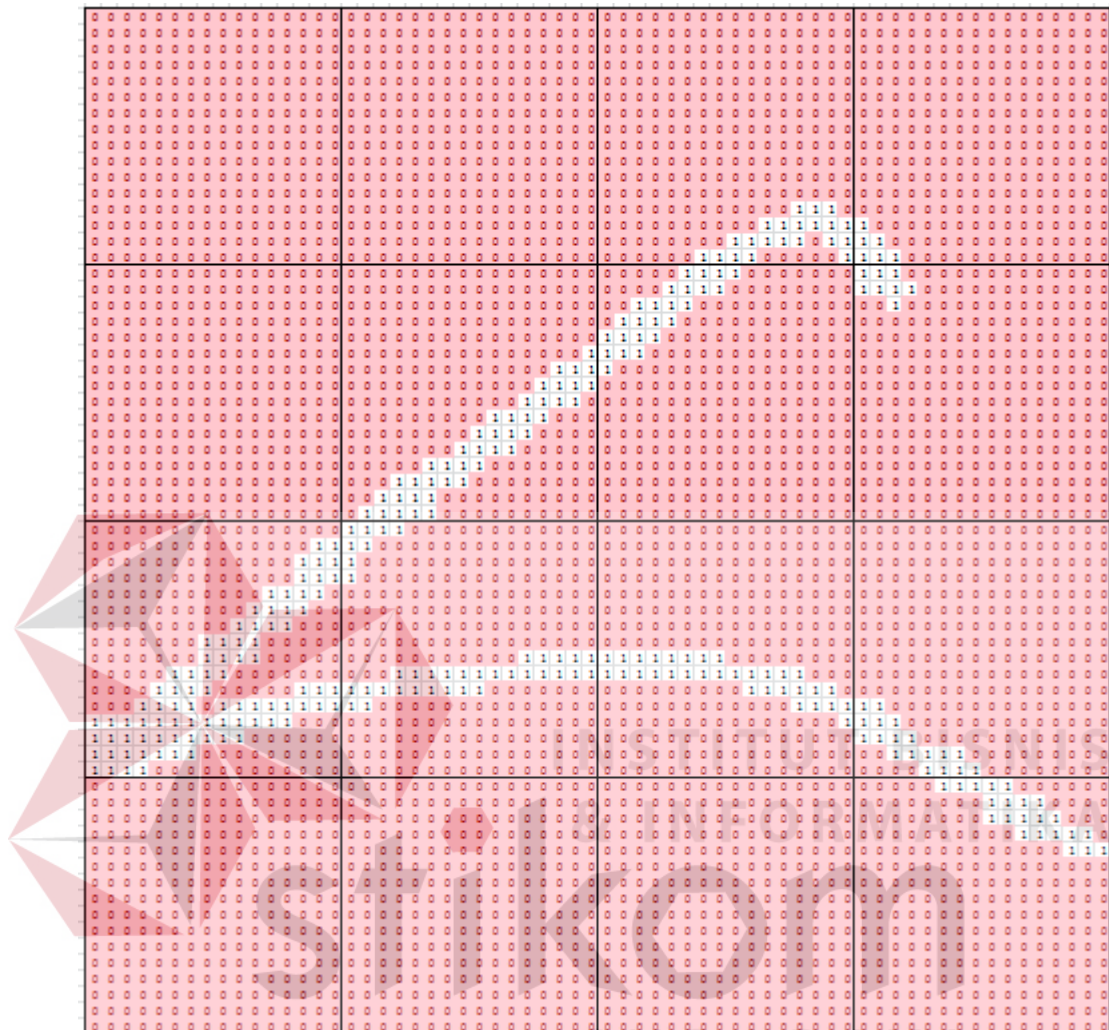
Prosedur pengujian adalah seperti langkah-langkah berikut ini:

1. Buka Matlab
2. Buat projek baru (pilih *file* kemudian *new project*)
3. Tuliskan *syntax* dibawah ini:

```
clc
I=imread('D:\1 Tugas Akhir\Coding TA\angka sampel\Angka6.bmp');
level = graythresh(I);
BW = im2bw(I,level);
imshow(BW);
```

4. Angka6.bmp diatas adalah citra yang sama, yang digunakan untuk pengujian *ekstraksi longest run* melalui *project* tugas akhir penulis
5. *Syntax* diatas akan mengubah sampel uji menjadi bentuk biner, kemudian ditampilkan pada *array editor*
6. *Copy* data biner di array editor ke microsoft excel
7. Lakukan ekstraksi secara manual untuk mencari nilai ekstraksi dari tiap sub area citra

B. Hasil Pengujian



Gambar 4.10 Pengujian ekstraksi *longest run* secara manual

Gambar diatas adalah tampilan saat nilai biner dari *array editor* matlab diduplikat ke microsoft excel. Untuk memudahkan perhitungan area yang tidak dihitung atau angka-angka nol diberi warna merah muda. Gunakanlah fungsi *sum* pada excel untuk mempermudah perhitungan. Hasil uji secara manual ditunjukkan pada Tabel 4.1.

Tabel 4.1 Hasil uji fitur *longest run* secara manual

Area	1	2	3	4	5	6	7	8	9
Baris	42	84	43	146	127	74	104	61	72
Kolom	42	87	59	113	89	57	94	63	72
Da	2	2	7	2	2	2	2	2	2
Db	0	0	2	12	2	2	2	2	1

Tabel diatas menunjukkan bahwa hasil ekstraksi menggunakan *project* tugas akhir penulis dan hasil ekstraksi secara manual menunjukkan hasil yang sama. Sehingga dapat di simpulkan bahwa ekstraksi *longest run* telah menghasilkan fitur yang benar.

4.4 Pengujian *Diagonal Distance Feature Extraction*

Citra yang telah melalui *ROI* dan *resize* selanjutnya akan di cari nilai fitur dari ekstraksi *diagonal distance*. Nilai fitur yang dihasilkan dari ekstraksi *diagonal distance feature* berjumlah 4 fitur. Fitur didapatkan dengan cara menghitung jarak diagonal dari sudut siku citra hingga piksel hitam.

4.4.1 Tujuan

Pengujian ini bertujuan untuk mengetahui apakah hasil ekstraksi *diagonal distance* pada aplikasi tugas akhir telah menghasilkan fitur yang benar.

4.4.2 Peralatan yang digunakan

A. Perangkat Keras

Peralatan yang digunakan untuk menguji *diagonal distance feature extraction* adalah sebuah *personal computer* yang memiliki spesifikasi seperti berikut:

- Processor : Intel, Core i3 @2.10Ghz (4CPUs)
- Memory : 4 GB
- *Operating System* : Windows 7 Ultimate – 64 Bit

B. Perangkat Lunak

Perangkat lunak pendukung dalam menguji *longest run feature extraction* adalah sebagai berikut:

- *Project* tugas akhir penulis digunakan untuk menghasilkan fitur citra.
- Matlab digunakan untuk membuat citra menjadi biner.
- Microsoft excel 2010 digunakan untuk menyimpan dan menghitung nilai biner dari matlab, serta digunakan juga untuk mencari nilai fitur *diagonal distance* secara manual.

4.4.3 Pengujian Ekstraksi *Diagonal Distance Feature*

A. Prosedur Pengujian

Prosedur pengujian adalah seperti langkah-langkah berikut ini:

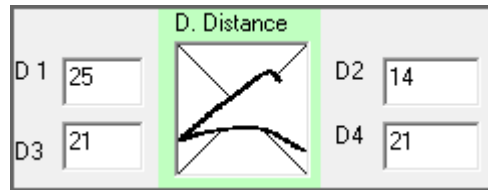
1. Buka *project* tugas akhir*.vbp
2. Masukkan (*load*) sampel citra yang akan di uji, seperti gambar 4.11



Gambar 4.11 Sampel uji *Diagonal Distance*

3. Klik entry

B. Hasil Pengujian



Gambar 4.12 Hasil uji *diagonal distance* pada aplikasi

Seluruh hasil ekstraksi dari tiap diagonal ditampilkan pada gambar 4.12. D1=Diagonal 1, D2=Diagonal 2, D3=Diagonal 3 dan D4=Diagonal 4. Selanjutnya, dengan menggunakan sampel uji yang sama akan dilakukan pengujian secara manual menggunakan matlab dan microsoft excel.

4.4.4 Pengujian Ekstraksi Diagonal Distance Feature Secara Manual

A. Prosedur Pengujian

Prosedur pengujian adalah seperti langkah-langkah berikut ini:

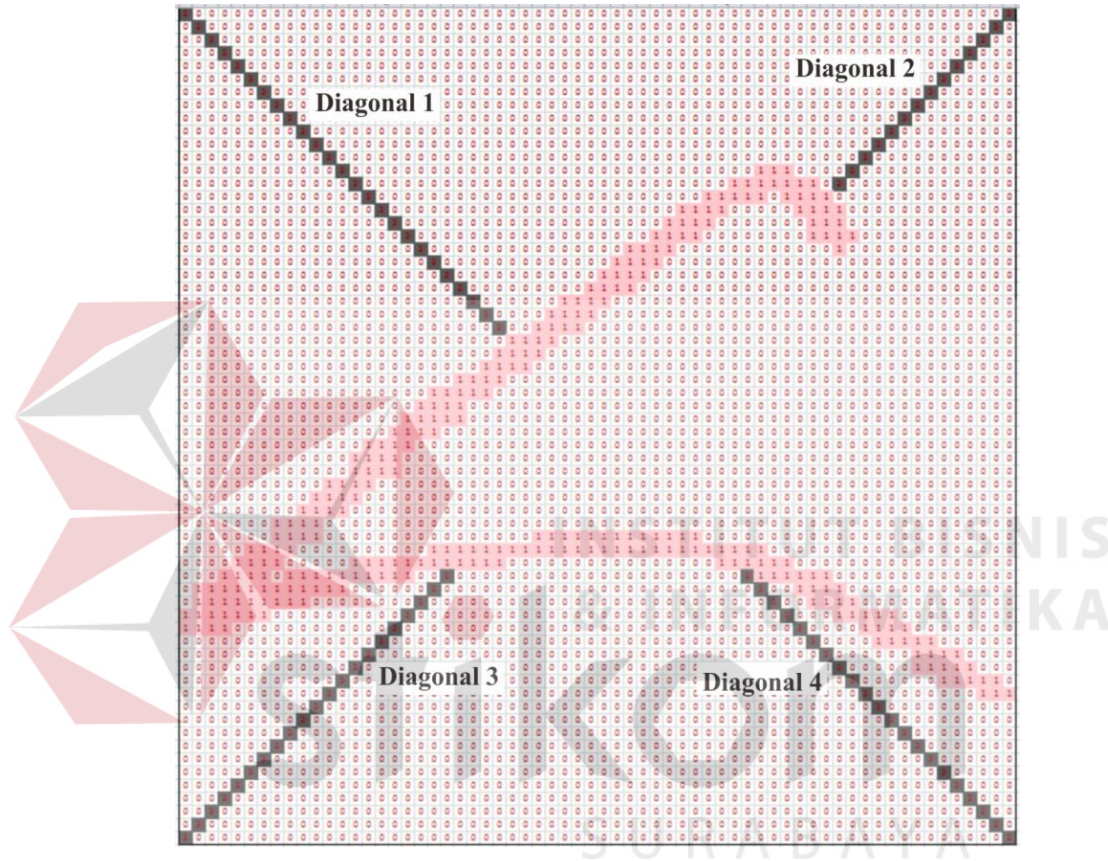
1. Buka matlab
2. Buat projek baru (pilih *file* kemudian *new project*)
3. Tuliskan *syntax* dibawah ini:

```
clc
I=imread('D:\1 Tugas Akhir\Coding TA\angka sampel\Angka6.bmp');
level = graythresh(I);
BW = im2bw(I,level);
imshow(BW);
```

4. Angka6.bmp diatas adalah citra yang sama, yang digunakan untuk pengujian melalui *project* tugas akhir penulis
5. *Syntax* diatas akan mengubah sampel uji menjadi bentuk biner, kemudian ditampilkan pada *array editor*.

6. Copy data biner pada array editor ke microsoft excel.
7. Lakukan ekstraksi secara manual untuk mencari nilai ekstraksi dari tiap area.

B. Hasil Pengujian



Gambar 4.13 Pengujian Ekstraksi Diagonal Secara manual

Gambar 4.13 adalah tampilan saat nilai biner dari *array editor* matlab diduplikat ke microsoft excel. Gunakan fungsi *sum* pada microsoft excel untuk mempermudah menghitung nilai fitur. Hasil uji coba ditunjukkan pada tabel dibawah ini:

Tabel 4.2 Hasil uji fitur *diagonal distance* secara manual

Diagonal 1	Diagonal 2	Diagonal 3	Diagonal 4
25	14	21	21

Tabel 4.2 menunjukkan bahwa hasil ekstraksi menggunakan *project* tugas akhir dan hasil ekstraksi secara manual menunjukkan hasil yang sama. Sehingga dapat diambil kesimpulan bahwa proses ekstraksi *diagonal distance* telah menghasilkan fitur dengan benar.

4.5 Pengujian Pengenalan MLP

Pada pengujian ini memiliki beberapa model pengujian untuk mendapatkan presentase keberhasilan pengenalan, dimana pengujian MLP ditentukan berdasarkan jumlah fitur ekstraksi yang digunakan sebagai *input* MLP.

4.5.1 Tujuan

Untuk mengetahui seberapa banyak pola angka yang dapat dikenali MLP dari sampel *training* maupun sampel *testing* yang masing-masing berjumlah 25 set pola Angka Jawa. Setiap set berisi Angka Jawa 0 sampai 9.

4.5.2 Peralatan yang digunakan

A. Perangkat Keras

Peralatan yang digunakan untuk menguji pengenalan MLP adalah sebuah *personal computer (Notebook)* yang memiliki spesifikasi seperti berikut:

- Processor : Intel, Core i3 @2.10Ghz (4CPUs)
- Memory : 4 GB
- *Operating System* : Windows 7 Ultimate – 64 Bit

B. Perangkat Lunak

Perangkat lunak pendukung dalam menguji metode pengenalan MLP adalah sebagai berikut:

- *Project* tugas akhir penulis digunakan untuk membuat dan mengetahui tingkat keberhasilan pengenalan MLP.
- Microsoft excel 2010 digunakan untuk menghitung keberhasilan serta kegagalan pengenalan MLP.

4.5.3 Prosedur Pengujian

1. Jalankan *project* tugas akhir*.vbp
2. Setelah terbuka klik tombol *run sample* untuk pengujian pengenalan MLP pada sampel *training*
3. Setelah proses *run sample* selesai, klik *run test* untuk pengujian pengenalan MLP pada sampel *testing*
4. Setelah proses *run sample* dan *run test* selesai aplikasi akan menyimpan hasil pengenalan pada direktori aplikasi dengan nama file "*training.csv*" dan "*testing.csv*"

4.5.4 Hasil Pengujian

Hasil pengujian pengenalan MLP dilakukan dalam beberapa tahap berdasarkan jumlah fitur yang digunakan untuk *input* MLP sebagai berikut.

A. Pengujian Pengenalan MLP Menggunakan *Input* 4 Fitur *Diagonal Distance*

Sampel untuk pengujian berjumlah 50 set. Setiap set berisi 10 Sampel yang terdiri dari satu digit Angka Jawa 0 - 9. Kemudian dari 50 set sampel

tersebut 25 set digunakan untuk *learning* MLP dan 25 set lainnya untuk *testing*. Setiap sampel Angka Jawa diekstraksi fiturnya menggunakan *diagonal distance* dan *longest run*. Pada tiap Angka Jawa yang diekstraksi menghasilkan 40 nilai fitur yang terdiri dari, 36 fitur dari ekstraksi *longest run* dan 4 fitur dari ekstraksi *diagonal distance*. 40 nilai fitur tersebut kemudian disimpan dalam sebuah file fitur*.dat. Pada pengujian ini bobot MLP yang digunakan adalah bobot hasil *learning* MLP yang hanya menggunakan 4 fitur *diagonal distance* sebagai inputnya.

1. Hasil Pengenalan MLP Menggunakan Input 4 Fitur Pada Sampel

Training

Sampel *training* adalah sampel yang nilai fiturnya digunakan untuk *learning* MLP. Selanjutnya 4 Nilai fitur *diagonal distance* dari tiap sampel *training* yang telah digunakan untuk *learning* MLP di inputkan kembali ke MLP untuk dilakukan pengenalan.

Tabel 4.3 Keberhasilan pengenalan MLP dengan input 4 fitur pada sampel *training*

Angka	Jumlah Data (buah)	Data Yang Berhasil Dikenali (buah)	Kesalahan Pengenalan (buah)	Presentase Keberhasilan (%)
0	25	25	0	100%
1	25	0	25	0%
2	25	0	25	0%
3	25	0	25	0%
4	25	0	25	0%
5	25	0	25	0%
6	25	0	25	0%
7	25	0	25	0%
8	25	0	25	0%
9	25	0	25	0%
Jumlah Persentase Total				10%

Presentase keberhasilan dihitung dengan cara data yang berhasil dikenali dibagi dengan jumlah data kemudian dikalikan 100%. Pada tabel 4.3 diatas menunjukkan tingkat keberhasilan pengenalan mencapai 10%. keberhasilan pengenalan paling besar yaitu pada pola Angka Jawa “0” dengan tingkat keberhasilan pengenalan 100%. selain angka “0” kegagalan pengenalan mencapai 100%. Untuk mengetahui kesalan-kesalahan pengenalan pada tiap angka ditunjukkan pada tabel4.4.

Tabel 4.4 Kesalahan pengenalan MLP dengan *input* 4 fitur pada sampel *training*

Angka	Dikenali Sebagai Angka										Presentase Kesalahan (%)
	0	1	2	3	4	5	6	7	8	9	
0											0%
1	23				2						100%
2	25										100%
3	23				2						100%
4	25										100%
5	25										100%
6	25										100%
7	25										100%
8	25										100%
9	25										100%
Rata-Rata Error											%90

Presentase kesalahan dihitung dengan cara jumlah kesalahan dibagi dengan jumlah data(25) kemudian dikalikan 100%. Pada tabel 4.4 diatas menunjukkan selain angka “0” semuanya salah dikenali sebagai angka lain sebesar 100% . Angka ”1” dan ”3” salah dikenali sebagai angka “0” sebanyak 23 kali dan salah dikenali sebagai angka “4” sebanyak 2 kali.

2. Hasil Pengenalan Menggunakan *Input* 4 Fitur MLP Pada Sampel *Testing*

Sampel *testing* adalah sampel yang nilai fiturnya tidak digunakan untuk *learning* MLP. 4 Nilai fitur *diagonal distance* dari tiap sampel *testing* di inputkan ke MLP untuk dilakukan pengenalan.

Tabel 4.5 Keberhasilan pengenalan MLP dengan *input* 4 fitur pada sampel *testing*

Angka	Jumlah Data (buah)	Data Yang Berhasil Dikenali (buah)	Kesalahan Pengenalan (buah)	Presentase Keberhasilan (%)
0	25	25	0	100%
1	25	0	25	0%
2	25	0	25	0%
3	25	0	25	0%
4	25	0	25	0%
5	25	0	25	0%
6	25	0	25	0%
7	25	0	25	0%
8	25	0	25	0%
9	25	0	25	0%
Jumlah Persentase Total				10%

Sama seperti pada sampel *training* keberhasilan pengenalan pada sampel *testing* hanya mencapai 10%. keberhasilan pengenalan paling besar yaitu pada pola Angka Jawa “0” dengan tingkat keberhasilan pengenalan 100%. selain angka “0” kegagalan pengenalan mencapai 100%. Untuk mengetahui kesalahan-kesalahan pengenalan pada tiap angka ditunjukkan pada tabel 4.6.

Tabel 4.6 Kesalahan pengenalan MLP dengan *input* 4 fitur pada sampel *testing*

Angka	Dikenali Sebagai Angka									Presentase Kesalahan (%)	
	0	1	2	3	4	5	6	7	8		9
0											0%
1	22			2	1						100%
2	24				1						100%
3	23				2						100%
4	25										100%
5	24				1						100%
6	25										100%
7	23				2						100%
8	25										100%
9	25										100%
Rata-Rata Error											%90

Presentase kesalahan dihitung dengan cara jumlah kesalahan dibagi dengan jumlah data(25) kemudian dikalikan 100%. Pada tabel 4.6 diatas menunjukkan selain angka "0" semuanya salah dikenali sebagai angka lain sebesar 100% . kesalahan pengenalan paling bervariasi yaitu angka "1" yang dikenali sebagai angka "0" sebanyak 22 kali, salah dikenali sebagai angka 3 sebanyak 2 kali dan salah dikenali sebagai angka "4" sebanyak 1 kali.

B. Pengujian Pengenalan MLP Menggunakan *Input* 36 Fitur *Longest Run*

Sampel untuk pengujian berjumlah 50 set. Setiap set berisi 10 Sampel yang terdiri dari satu digit Angka Jawa 0 - 9. Kemudian dari 50 set sampel tersebut 25 set digunakan untuk *learning* MLP dan 25 set lainnya untuk *testing*. Setiap sampel Angka Jawa diekstraksi fiturnya menggunakan *diagonal distance* dan *longest run*. Pada tiap Angka Jawa yang diekstraksi menghasilkan 40 nilai fitur

yang terdiri dari, 36 fitur dari ekstraksi *longest run* dan 4 fitur dari ekstraksi *diagonal distance*. 40 nilai fitur dari tiap sampel tersebut kemudian disimpan dalam sebuah file fitur*.dat. Pada pengujian ini bobot MLP yang digunakan adalah bobot hasil *learning* MLP yang hanya menggunakan 36 fitur *longest run* sebagai inputnya.

1. Hasil Pengenalan MLP Menggunakan *Input* 36 Fitur Pada Sampel *Training*

Sampel *training* adalah sampel yang nilai fiturnya digunakan untuk *learning* MLP. Selanjutnya 36 Nilai fitur *longest run* dari tiap sampel *training* yang telah digunakan untuk *learning* MLP di inputkan kembali ke MLP untuk dilakukan pengenalan.

Tabel 4.7 Keberhasilan pengenalan MLP dengan *input* 36 fitur pada sampel *training*

Angka	Jumlah Data (buah)	Data Yang Berhasil Dikenali (buah)	Kesalahan Pengenalan (buah)	Presentase Keberhasilan (%)
0	25	25	0	100%
1	25	25	0	100%
2	25	25	0	100%
3	25	25	0	100%
4	25	25	0	100%
5	25	25	0	100%
6	25	25	0	100%
7	25	25	0	100%
8	25	25	0	100%
9	25	25	0	100%
Jumlah Persentase Total				100%

Tabel 4.7 menunjukkan tingkat keberhasilan pengenalan mencapai 100% pada seluruh angka. tidak ada kesalahan satupun pada pengenalan Angka Jawa pada sampel *training*. Agar dapat diketahui keberhasilan pada sampel *testing*, selanjutnya dipaparkan juga pengujian pada sampel *testing*.

2. Hasil Pengenalan MLP Menggunakan *Input* 36 Fitur Pada Sampel *Testing*

Sampel *testing* adalah sampel yang nilai fiturnya tidak digunakan untuk *learning* MLP. 36 Nilai fitur *longest run* dari tiap sampel *testing* di inputkan ke MLP untuk dilakukan pengenalan.

Tabel 4.8 Keberhasilan pengenalan MLP dengan *input* 36 fitur pada sampel *testing*

Angka	Jumlah Data (buah)	Data Yang Berhasil Dikenali (buah)	Kesalahan Pengenalan (buah)	Presentase Keberhasilan (%)
0	25	25	0	100%
1	25	25	0	100%
2	25	22	3	88%
3	25	21	4	84%
4	25	25	0	100%
5	25	24	1	96%
6	25	24	1	96%
7	25	23	2	92%
8	25	23	2	92 %
9	25	24	1	96 %
Jumlah Persentase Total				94,4%

Tabel 4.8 menunjukkan tingkat keberhasilan pengenalan mencapai 94,4%. keberhasilan pengenalan paling besar terjadi pada pola Angka Jawa “0”,”1”, dan ”4” dengan tingkat keberhasilan pengenalan 100%. Untuk mengetahui kesalahan-kesalahan pengenalan ditunjukkan pada tabel 4.9.

Tabel 4.9 Kesalahan pengenalan MLP dengan *input* 36 fitur pada sampel *testing*

Angka	Dikenali Sebagai Angka									Presentase Kesalahan (%)	
	0	1	2	3	4	5	6	7	8		9
0											0%
1											0%
2	2			1							12%
3	1	2	1								16%
4											0%
5			1								4%
6	1										4%
7	1									1	8%
8								1		1	8%
9								1			4%
Rata-Rata Error											5,6%

Tabel 4.9 menunjukkan kesalahan pengenalan paling besar terjadi pada angka “3” yang salah dikenali sebagai angka lain sebanyak 4 kali. Kesalahan pengenalan terkecil terjadi pada angka “0” dan “1” karena tidak terjadi kesalahan pengenalan sama sekali.

C. Pengujian Pengenalan MLP Menggunakan *Input* 40 Fitur

Pada pengujian ini, *input* MLP menggunakan 40 fitur yang dihasilkan oleh 2 metode ekstraksi. 36 fitur berasal dari ekstraksi *longest run* dan 4 fitur dari ekstraksi *diagonal distance*.

1. Hasil Pengenalan MLP Menggunakan *Input* 40 Fitur Pada Sampel *Training*

Sampel untuk pengujian berjumlah 50 set. Setiap set berisi 10 Sampel yang terdiri dari satu digit Angka Jawa 0 - 9. Kemudian dari 50 set sampel tersebut 25 set digunakan untuk *learning* MLP dan 25 set lainnya untuk *testing*. Setiap sampel Angka Jawa diekstraksi fiturnya menggunakan *diagonal distance*

dan *longest run*. Pada tiap Angka Jawa yang diekstraksi menghasilkan 40 nilai fitur yang terdiri dari, 36 fitur dari ekstraksi *longest run* dan 4 fitur dari ekstraksi *diagonal distance*. 40 nilai fitur dari tiap sampel tersebut kemudian disimpan dalam sebuah file fitur*.dat. Pada pengujian ini bobot MLP yang digunakan adalah bobot hasil *learning* MLP yang menggunakan 40 fitur tersebut sebagai inputnya.

Tabel 4.10 Keberhasilan pengenalan MLP dengan *input* 40 fitur pada sampel *training*

Angka	Jumlah Data (buah)	Data Yang Berhasil Dikenali (buah)	Kesalahan Pengenalan (buah)	Presentase Keberhasilan (%)
0	25	23	2	92 %
1	25	1	24	4 %
2	25	25	0	100 %
3	25	25	0	100 %
4	25	24	1	96 %
5	25	21	4	84 %
6	25	16	9	64 %
7	25	22	3	88 %
8	25	0	25	0 %
9	25	10	15	40 %
Jumlah Persentase Total				66,8 %

Tabel 4.10 menunjukkan tingkat keberhasilan pengenalan MLP dengan *input* 40 fitur mencapai 66,8%. Keberhasilan pengenalan tertinggi pada angka “2” dan “3” dengan keberhasilan pengenalan mencapai 100% dan keberhasilan pengenalan paling rendah terjadi pada angka “8” dengan keberhasilan pengenalan 0%. Untuk mengetahui kesalan-kesalahan pengenalan pada tiap angka ditunjukkan pada tabel 4.11.

Tabel 4.11 Kesalahan pengenalan MLP dengan *input* 40 fitur pada sampel *training*

Angka	Dikenali Sebagai Angka									Presentase Kesalahan (%)	
	0	1	2	3	4	5	6	7	8		9
0					2						8 %
1	8		1					15			96 %
2											0 %
3											0 %
4			1								4 %
5	3		1								16 %
6			8	1							36 %
7	2									1	12 %
8	12							13			100 %
9	4							11			60 %
Rata-Rata Error											33,2 %

Tabel 4.11 Menunjukkan kesalahan pengenalan terkecil terjadi pada angka “2” dan “3” yang semua sampelnya berhasil dikenali dengan benar. Kesalahan pengenalan terbanyak terjadi pada angka “8” yang sama sekali tidak berhasil dikenali dengan benar, melainkan salah dikenali sebagai angka “0” sebanyak 12 kali dan dikenali sebagai angka “7” sebanyak 13 kali.

2. Hasil Pengenalan MLP Menggunakan *Input* 40 Fitur Pada Sampel *Testing*

Sampel *testing* adalah sampel yang nilai fiturnya tidak digunakan untuk *learning* MLP. 36 Nilai fitur *longest run* dari tiap sampel *testing* di inputkan ke MLP untuk dilakukan pengenalan.

Tabel 4.12 Keberhasilan pengenalan MLP dengan *input* 40 fitur pada sampel *testing*

Angka	Jumlah Data (buah)	Data Yang Berhasil Dikenali (buah)	Kesalahan Pengenalan (buah)	Presentase Keberhasilan (%)
0	25	21	4	84 %
1	25	5	20	20 %
2	25	22	3	88 %
3	25	20	5	80 %
4	25	24	1	96 %
5	25	18	7	72 %
6	25	18	7	72 %
7	25	18	7	72 %
8	25	0	25	0 %
9	25	11	14	44 %
Jumlah Persentase Total				62,8 %

Tabel 4.12 menunjukkan tingkat keberhasilan pengenalan MLP dengan *input* 40 fitur mencapai 62,8%. Keberhasilan pengenalan paling besar terjadi pada pola Angka Jawa “4” dengan tingkat keberhasilan pengenalan mencapai 96% dan keberhasilan pengenalan paling kecil terjadi pada pola Angka Jawa “8” dengan tingkat keberhasilan pengenalan 0%. Untuk mengetahui kesalahan-kesalahan pengenalan pada sampel *testing* ditunjukkan pada tabel 4.13

Tabel 4.13 Kesalahan pengenalan MLP dengan *input* 40 fitur pada sampel *testing*

Angka	Dikenali Sebagai Angka									Presentase Kesalahan (%)	
	0	1	2	3	4	5	6	7	8		9
0					4						16 %
1	9							11			80 %
2	2			1							12 %
3			1					4			20 %
4	1										4 %
5	4		3								28 %
6			5		2						28 %
7	4			2						1	28 %
8	11							14			100 %
9	1							13			56 %
Rata-Rata Error											37,2 %

Tabel 4.13 menunjukkan kesalahan pengenalan paling besar terjadi pada angka “8” yang sama sekali tidak berhasil dikenali dengan benar, melainkan salah dikenali sebagai angka “0” sebanyak 11 kali dan salah dikenali sebagai angka 7 sebanyak 14 kali.

Dari penelitian yang telah dilakukan menunjukkan bahwa *input* MLP menggunakan ekstraksi *longest run* menghasilkan pengenalan yang lebih baik dibandingkan dengan ekstraksi *diagonal distance*. Presentase keberhasilan pengenalan MLP dengan input ekstraksi *longest run* (36 fitur) mencapai 100% pada sampel *training* sedangkan keberhasilan pengenalan MLP dengan input ekstraksi *diagonal distance* (4 fitur) hanya mencapai 10%. Presentase keberhasilan pengenalan MLP pada sampel *testing* dengan input ekstraksi *longest run* (36 fitur) mencapai 94,4% sedangkan keberhasilan pengenalan MLP dengan input ekstraksi *diagonal distance* (4 fitur) hanya mencapai 10%.

Kecilnya pengenalan MLP dengan *input* 40 fitur disebabkan karena hasil ekstraksi *diagonal distance* (4 fitur) yang buruk. Presentase keberhasilan

pengenalan MLP dengan input ekstraksi 40 fitur atau hasil 2 metode ekstraksi yaitu *longest run* dan *diagonal distance*, mencapai 66,8% pada sampel *training* dan 62,8% pada sampel *testing*. Sebagai perbandingan keberhasilan pengenalan Angka Jawa yang hanya menggunakan *input* ekstraksi *longest run* jauh lebih bagus yaitu mencapai 100% pada sampel *training* dan 94,4% pada sampel *testing*. Sehingga ketika MLP hanya menggunakan *input* dari ekstraksi *longest run* (36 fitur) saja akan menghasilkan pengenalan yang lebih optimal.

