

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

BAB IV ANALISA DAN PERANCANGAN

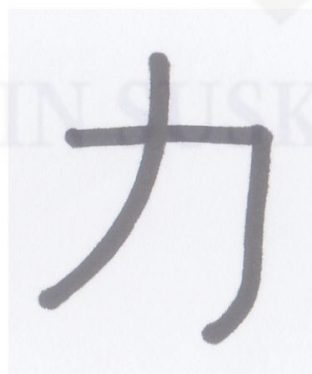
Tahapan penting dalam pembuatan suatu sistem atau aplikasi adalah tahapan analisa dan perancangan. Pada tahapan ini dilakukan proses kajian yang membahas, memahami dan menspesifikasi secara rinci mengenai pokok permasalahan dalam penelitian agar memperoleh pemahaman yang tepat dalam mengenai masalah, data, proses, hasil dan hal yang berkaitan dalam penelitian.

4.1. Analisa Metode

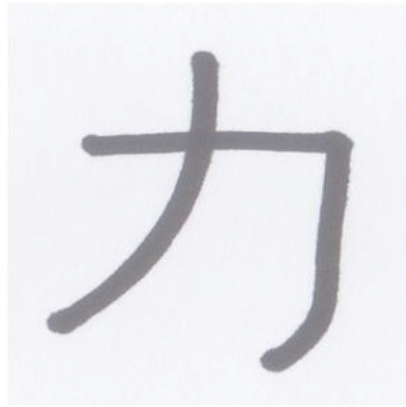
Proses analisa merupakan tahapan- tahapan dalam pengenalan karakter yang dilakukan. Dalam analisa ini, tahapan- tahapan tersebut adalah *pre-processing*, ekstraksi ciri dan tahapan terakhir adalah klasifikasi. Penjelasan pada tiga tahapan ini adalah sebagai berikut :

4.1.1. Pre- Processing

Tahapan *pre- processing* terdapat beberapa langkah yang harus dilakukan. Namun sebelum melakukan tahapan ini, data terlebih dahulu di *resize* dengan ukuran 300 x 300 piksel. Setelah ukuran citra menjadi 300x 300 piksel, maka lakukan tahapan *pre-processing*. Tahap awal yang dilakukan adalah biner. Citra awal sebelum di-*resize* dapat dilihat pada Gambar 4.1 Huruf Ka Asli dan Gambar 4.2 adalah hasil gambar yang telah di *resize* menjadi ukuran 300x 300 piksel.



Gambar 4.1 Huruf Ka Asli



Gambar 4.2 Huruf Ka setelah di-resize

Tahapan biner yang digunakan menggunakan *threshol* sebesar 0.7 dengan menggunakan Persamaan (2.1) maka nilai biner yang didapat terdapat pada Tabel 4.1 Nilai Biner :

Tabel 4.1 Nilai Biner

	1	2	3	4	5	...	35	36	37	38	39	40	...	299	300
1	1	1	1	1	1	...	1	1	1	1	1	1	...	1	1
2	1	1	1	1	1	...	1	1	1	1	1	1	...	1	1
3	1	1	1	1	1	...	1	1	1	1	1	1	...	1	1
4	1	1	1	1	1	...	1	1	1	1	1	1	...	1	1
5	1	1	1	1	1	...	1	1	1	1	1	1	...	1	1
...	1	1
145	1	1	1	1	1	...	1	1	1	1	1	1	...	1	1
146	1	1	1	1	1	...	0	0	0	1	1	1	...	1	1
147	1	1	1	1	1	...	0	0	0	0	0	0	...	1	1
148	1	1	1	1	1	...	0	0	0	0	0	0	...	1	1
149	1	1	1	1	1	...	0	0	0	0	0	0	...	1	1
150	1	1	1	1	1	...	0	0	0	0	0	0	...	1	1
...
299	1	1	1	1	1	...	1	1	1	1	1	1	...	1	1
300	1	1	1	1	1	...	1	1	1	1	1	1	...	1	1

Berikut adalah hasil biner dari citra Huruf Ka. Jika dilihat dalam sebuah gambar, maka hasil Huruf Ka yang di dapat terlihat pada Gambar 4.3 Hasil Biner



Gambar 4.3 Hasil Biner

Setelah hasil biner didapatkan, maka langkah *pre-procesing* selanjutnya adalah deteksi tepi dengan Operator Roberts. Proses yang dilakukan pada Operator Roberts dapat dihitung dengan menggunakan *kernel* konvolusi pada Persamaan (2.3). Perhitungan manual Operator Roberts dengan menggunakan hasil biner sebelumnya adalah sebagai berikut :

1. Konvolusi citra biner dengan *kernel* Roberts horizontal (R_x)

Perhitungan yang dilakukan adalah hasil biner dari Gambar 4.2. Hasil Biner dengan hasil yang diambil sebagai contoh yaitu pada baris ke 145-150 dan kolom ke 35-39. Perhitungannya dapat dilihat pada Tabel 4.2 Hasil Biner Kolom 35-39 dengan menggunakan Kernel Konvolusi R_x yang terdapat pada Persamaan (2.3), maka perhitungan yang didapatkan adalah sebagai berikut:

Tabel 4.2 Hasil Biner Kolom 35-39

	145	146	147	148	149	150
35	1	1	1	1	1	1
36	0	0	0	1	1	1
37	0	0	0	0	0	0
38	0	0	0	0	0	0
39	0	0	0	0	0	0

Pada Operator Roberts menggunakan cara konvolusi yang berarti cara perhitungannya dengan cara pergeseran kolom sebelah kanan sesuai dengan kernel konvolusi yang digunakan. Maka hasil Konvolusi Roberts horizontal pada baris ke 145-150 dan kolom ke 35-39 adalah :

Konvolusi Roberts horizontal adalah sebagai berikut :

$$(R_{x145,35}) = (x_{145,35} * R_{x1,1}) + (x_{146,35} * R_{x2,1}) + (x_{145,36} * R_{x1,2}) + (x_{146,36} * R_{x2,2})$$

$$(R_{x145,35}) = (1*1) + (1*0) + (0*0) + (0*-1) = 1$$

$$(R_{x146,35}) = (x_{146,35} * R_{x1,1}) + (x_{147,35} * R_{x2,1}) + (x_{146,36} * R_{x1,2}) + (x_{147,36} * R_{x2,2})$$

$$(R_{x146,35}) = (1*1) + (1*0) + (0*0) + (0*-1) = 1$$

Setelah dilakukan konvolusi hingga selesai, maka hasil Roberts horizontal dapat dilihat pada Tabel 4.3 Hasil Roberts Horizontal berikut ini :

Tabel 4.3 Hasil Roberts Horizontal

	145	146	147	148	149
35	1	1	0	0	0
36	0	0	0	1	1
37	0	0	0	0	0
38	0	0	0	0	0
39	0	0	0	0	0

2. Konvolusi citra biner dengan *kernel* Roberts vertikal (R_y)

Konvolusi yang dilakukan sama dengan Roberts horizontal namun *kernel* yang di gunakan berbeda. Maka perhitungannya dapat dilihat pada Tabel 4.4 Hasil Biner Kolom 35-39 dengan menggunakan Kernel Konvolusi R_y yang terdapat pada Persamaan (2.3), maka perhitungan yang didapatkan adalah sebagai berikut berikut :

Tabel 4.4 Hasil Biner Kolom 35-39

	145	146	147	148	149	150
35	1	1	1	1	1	1
36	0	0	0	1	1	1
37	0	0	0	0	0	0
38	0	0	0	0	0	0
39	0	0	0	0	0	0

Pada Operator Roberts menggunakan cara konvolusi yang berarti cara perhitungannya dengan cara pergeseran kolom sebelah kanan sesuai dengan kernel konvolusi yang digunakan. Maka hasil Konvolusi Roberts vertikal pada baris ke 145-150 dan kolom ke 35-39 adalah :

Konvolusi Roberts vertikal adalah sebagai berikut :

$$(R_{y145,35}) = (y_{145,35} * R_{y1,1}) + (y_{146,35} * R_{y2,1}) + (y_{145,36} * R_{y1,2}) + (y_{146,36} * R_{y2,2})$$

$$(R_{y145,35}) = (1 * 0) + (1 * 1) + (0 * -1) + (0 * 0) = 1$$

$$(R_{y146,35}) = (y_{146,35} * R_{y1,1}) + (y_{147,35} * R_{y2,1}) + (y_{146,36} * R_{y1,2}) + (y_{147,36} * R_{y2,2})$$

$$(R_{y146,35}) = (1 * 0) + (1 * 1) + (0 * -1) + (0 * 0) = 1$$

Setelah dilakukan konvolusi hingga selesai, maka hasil Roberts vertikal dapat dilihat pada Tabel 4.5 Hasil Roberts Vertikal berikut ini :

Tabel 4.5 Hasil Roberts Vertikal

	145	146	147	148	149
35	1	1	1	0	0
36	0	0	1	1	1
37	0	0	0	0	0
38	0	0	0	0	0
39	0	0	0	0	0

3. Lakukan persamaan Roberts pada Persamaan (2.2)

$$G_{145,35} = \sqrt{R_{x145,35}^2 + R_{y145,35}^2}$$

$$G_{145,35} = \sqrt{1^2 + 1^2} = 1.414$$

Jadi hasil pada perhitungan manual Roberts dapat dilihat pada Tabel 4.6 Hasil Perhitungan Roberts sebagai berikut:

Tabel 4.6 Hasil Perhitungan Roberts

	145	146	147	148	149
35	1.414	1.414	1	0	0
36	0	0	1	1.414	1.414
37	0	0	0	0	0
38	0	0	0	0	0
39	0	0	0	0	0

Pada hasil Roberts diatas memiliki tiga varian angka yaitu 1.414, 1 dan 0. Untuk menyederhanakan angka agar menghasilkan nilai biner, maka pada penelitian ini menggunakan kode didalam matlab yaitu *uint8*. Fungsi dari *uint8* adalah utuk

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

mengubah string menjadi bilangan biner. Dan hasil secara keseluruhan Roberts dapat dilihat pada Gambar 4.4 Hasil Gambar Roberts sebagai berikut:



Gambar 4.4 Hasil Gambar Roberts

Setelah hasil Roberts didapatkan, langkah selanjutnya adalah dilasi. Dilasi berfungsi untuk menebalkan sebuah objek yang digunakan dan citra asli akan mengalami pelebaran dengan mengikuti bentuk objeknya. Cara yang dilakukan pada dilasi menggunakan *matrix* yang disebut struktur elemen. Struktur elemen pada dilasi dapat dilihat pada Tabel 4.7 Struktur Elemen Dilasi :

Tabel 4.7 Struktur Elemen Dilasi

	-1	0	1
-1	1	1	1
0	1	1	1
1	1	1	1

Selanjutnya lakukan proses dilasi nilai citra pada *matrix* yang dihasilkan Roberts oleh *matrix* elemen menggunakan Persamaan (2.4). Perhitungan manual dilasi dengan menggunakan hasil Roberts pada Tabel 4.6 Hasil Perhitungan Roberts dan Struktur Elemen dilasi pada Tabel 4.7 Struktur Elemen Dilasi dapat dilihat sebagai berikut :

$$\begin{aligned}
 A \oplus B = & \{(145,35)+(-1,-1), (145,35)+(0,-1), (145,35)+(1,-1), (145,35)+ \\
 & (-1,0), (145,35)+(0,0), (145,35)+(1,0), (145,35)+(-1,1), \\
 & (145,35)+(0,1), (145,35)+(1,1), (146,35)+(-1,-1), (146,35)+(0,-1),
 \end{aligned}$$

$$\begin{aligned}
 & (146,35)+(1,-1), (146,35)+(-1,0), (146,35)+(0,0), (146,35)+(1,0), \\
 & (146,35)+(-1,1), (146,35)+(0,1), (146,35)+(1,1), \dots, (149,36)+(-1,-1), \\
 & (149,36)+(0,-1), (149,36)+(1,-1), (149,36)+(-1,0), \\
 & (149,36)+(0,0), (149,36)+(1,0), (149,36)+(-1,1), (149,36)+(0,1), \\
 & (149,36)+(1,1) \} \\
 = & \{(144,34), (145,34), (146,34), (147,34), (144,35), (145,35), (146,35), \\
 & (147,35), (144,36), (145,36), (146,36), (147,36), \dots, (148,35), (149,35) \\
 & , (150,35), (148,36), (149,36), (150,36), (148,37), (149,37), (150,37) \}
 \end{aligned}$$

Maka hasil pada perhitungan manual Dilasi dapat dilihat pada Tabel 4.8 Hasil Perhitungan Dilasi sebagai berikut:

Tabel 4.8 Hasil Perhitungan Dilasi

	144	145	146	147	148	149	150
34	255	255	255	255	255	0	0
35	255	255	255	255	255	255	255
36	255	255	255	255	255	255	255
37	0	0	255	255	255	255	255
38	0	0	0	0	0	255	255

Hasil dari dilasi dapat dilihat pada Gambar 4.5 Hasil Gambar Dilasi berikut ini :



Gambar 4.5 Hasil Gambar Dilasi

4.1.2. Ekstraksi Ciri Chain Code

Chain code dimulai jika proses *pre-processing* telah selesai dilakukan. Hasil yang telah didapat di dilasi akan digunakan pada proses ekstraksi ciri dengan menggunakan *chain code*. Chain code memiliki nilai arah mata angin yang digunakan pada proses *chain code*.

Langkah awal pada *chain code* adalah mencari posisi awal piksel dengan mencari kolom dan baris paling kecil dengan menelusuri nilai piksel 255. Selanjutnya telusuri piksel ketetanggaan. Jika piksel ketetanggaan adalah non objek, maka lakukan *new direction* dan jika piksel ketetanggaan adalah objek, maka masukkan *direction* ke dalam array dan perbarui piksel batas objek sebelumnya sebagai piksel batas objek yang akan dikunjungi selanjutnya. Kemudian lakukan langkah ini hingga penelusuran berhenti pada *starting pixel* yang dikunjungi telah dua kali di kunjungi.

Pada penelitian ini, hasil yang didapatkan pada ekstraksi ciri *chain code* kemudian dinormalisasi dengan panjang vektor sebesar 500. Tahap awal dalam proses normalisasi adalah hitung jumlah frekuensi dari setiap angka yang ada. Maka hasil frekuensi yang di dapat dalam *chain code* di atas adalah terdapat pada Tabel. 4.9 Hasil *Chain Code* berikut:

Tabel 4.9 Hasil Chain Code

	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₁₁₄₉	X ₁₁₅₀
Arah	5	4	5	5	6	5	4	4
Frekuensi	1	1	0	2	1	1	0	8

Setelah jumlah frekuensi sudah diketahui, maka langkan selanjutnya adalah hilangkan angka yang memiliki frekuensi satu. Maka hasil yang diperoleh dapat dilihat pada Tabel 4.10 Hasil Eliminasi *Chain Code* berikut :

Tabel 4.10 Hasil Eliminasi Chain Code

	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₁₅₄	X ₁₅₅
Arah	5	6	6	6	6	6	4	4
Frekuensi	2	12	10	6	3	5	2	8

Kemudian langkah yang dilakukan adalah akumulasikan frekuensi yang memiliki angka yang sama. Maka hasil yang didapatkan adalah pada Tabel 4.11 Hasil Akumulatif *Chain Code* berikut :

Tabel 4.11 Hasil Akumulatif Chain Code

	X ₁	X ₂	X ₃	X ₄	X ₅	X ₄₀	X ₄₁
Arah	5	6	4	5	6	3	4
Frekuensi	2	45	67	2	8	4	11

Setelah didapatkan frekuensi dari sebuah *chain code*, langkah selanjutnya yang dilakukan adalah melakukan persamaan normalisasi *chain code* yang memiliki panjang vektor yang sama. Dalam penelitian ini, panjang vektor yang diinginkan adalah sebesar 500 vektor. Maka lakukan Persamaan (2.6) untuk proses normalisasi. Penjelasan persamaannya adalah F dinyatakan sebagai frekuensi, i dinyatakan sebagai nilai awal frekuensi dan N adalah panjang vektor yang diinginkan. Maka hasil yang diperoleh menggunakan persamaan di atas dapat di contohkan dalam awal nilai yaitu 5 yang jumlah frekuensinya adalah 2 dan akhir nilai yaitu 4 dengan jumlah frekuensi adalah 11.

$$F_i^n = \frac{F_{x_1}}{\sum F_{x_1}, F_{x_2}, \dots, F_{x_{41}}} \times N$$

$$F = \frac{2}{908} \times 500 = 1,1$$

$$F_i^n = \frac{F_{x_{41}}}{\sum F_{x_1}, F_{x_2}, \dots, F_{x_{41}}} \times N$$

$$F = \frac{11}{908} \times 500 = 6$$

Jika dilihat dalam keseluruhannya, maka dapat dilihat pada Tabel 4.12 Hasil Proses Normalisasi adalah sebagai berikut ini :

Tabel 4.12 Hasil Proses Normalisasi

	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₄₀	X ₄₁
Arah	5	6	4	5	6	7	3	4
Hasil	1,1	24,78	36,89	1,1	4,41	2,2	2,203	6,057269

Selanjutnya lakukan pembulatan ke atas untuk mendapatkan angka yang bulat. Misalkan jika hasil yang di dapat adalah 1,12 maka pembulatan menjadi 1 dan jika hasil yang di dapat adalah 0,5 maka pembulatan menjadi 1 dan seterusnya. Dengan demikian diperoleh angka- angka yang menjadi frekuensi akhir dari

sebuah perhitunga *chain code* dan dapat dilihat pada Tabel 4.13 Hasil Nilai *Chain Code* Akhir berikut ini :

Tabel 4.13 Hasil Nilai *Chain Code* Akhir

Arah	5	6	4	5	6	7	3	4
Hasil	1,1	24,78	36,89	1,1	4,41	2,2	2,203	6,057269
Pembulatan	1	25	37	1	4	2	2	6

Hasil yang didapat adalah arah 5 terdapat sebanyak 1, arah 6 sebanyak 25, arah 4 sebanyak 37, ... dan arah 4 sebanyak 6. Maka hasil akhirnya dari normalisasi *chain code* adalah :

5 6
 6 6 6 6 6 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
 4
 5 6 6 6 6 6 6 7 7 ... 4 4 4 4 4 4

Hasil akhir dari normalisasi *chain code* sepanjang 500 vektor ini adalah hasil yang akan dimasukkan menjadi dataset untuk inputan dalam klasifikasi JST dengan menggunakan metode LVQ.

4.1.3. Klasifikasi *Learning Vector Quantization* (LVQ)

Learning Vector Quantization (LVQ) adalah salah satu metode dari Jaringan Saraf Tiruan (JST) dalam mengklasifikasi data. Dalam penelitian ini, LVQ digunakan untuk mengenali sebuah karakter huruf yang ada. Klasifikasi LVQ dilakukan dalam dua tahap yaitu pelatihan dan pengujian.

1. Pelatihan LVQ

Proses pelatihan LVQ dapat dilihat berdasarkan *flowchart* yang terdapat pada Gambar 3.4 *Flowchart* Pelatihan LVQ pada BAB III. Contoh perhitungan manual LVQ pada klasifikasi karakter pola Huruf Jepang Katakana menggunakan data latih dan data uji sebanyak 80% untuk data latih dan 20% untuk data uji. Maka jumlah data latih yang digunakan sebanyak 264 data dengan panjang vektor yang digunakan sebanyak 500 baris sedangkan data uji yang digunakan sebanyak 66 data dengan panjang vektor sebanyak 500 baris

1. Bobot Awal (W) , Maks. Epoch, *Learning Rate* (α) , Minimal *Learning Rate* dan Data Latih

Bobot awal dipilih secara random dan masing- masing bobot mewakili setiap kelas yang ada. Pada penelitian ini bobot awal yang akan di ambil adalah data ke-1 sampai data ke- 33. Bobot awal pada penelitian Pengenalan Pola Karakter Huruf Jepang Katakana ini dapat dilihat pada Tabel 4.14 Inisialisasi Bobot Awal (W) :

Tabel 4.14 Inisialisasi Bobot Awal (W)

Data ke-	Hasil Ekstraksi Ciri													Target
	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	...	X ₄₉₉	X ₅₀₀	
1	4	4	4	4	4	4	4	4	4	4	...	4	4	1
2	5	5	6	6	6	6	6	6	5	5	...	4	4	2
3	5	5	6	6	6	6	6	6	6	6	...	4	4	3
4	4	4	4	4	4	4	4	4	4	4	...	4	4	4
5	5	6	6	6	6	6	6	6	6	6	...	4	0	5
6	5	6	6	6	6	6	6	6	6	6	...	4	0	6
7	4	4	4	5	5	6	6	6	6	6	...	4	4	7
8	5	5	6	6	6	6	6	6	5	5	...	4	4	8
9	4	5	5	6	6	6	6	6	6	6	...	0	0	9
10	4	4	4	4	4	4	4	4	4	4	...	4	4	10
...
32	4	4	4	4	4	4	4	4	4	4	...	4	0	32
33	4	4	4	4	4	4	4	4	4	4	...	4	4	33

Setelah ditentukan bobot awal, selanjutnya tentukan maks. epoch, *learning rate* (α) dan minimal *learning rate*. Berikut adalah angkat yang digunakan dalam penelitian ini :

Maksimal Epoch : 1000

Learning rate (α) : 0,05

Minimal *learning rate* : 0,000001

Selanjutnya data yang akan dilatih adalah sisa dari data yang sudah menjadi bobot awal tadi. Pembagian untuk data latih dan data uji pada penelitian ini adalah

80%:20%, maka data yang akan di latih adalah data ke- 34 sampai data ke- 396. Data yang akan dilatih dapat dilihat pada Tabel 4.15 Nilai Data Latih (x) berikut ini :

Tabel 4.15 Nilai Data Latih (x)

Data ke-	Hasil Ekstraksi Ciri												Target	
	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	...	X ₄₉₉		X ₅₀₀
34	4	5	4	4	4	4	4	4	4	4	...	4	0	1
35	4	4	5	5	5	5	5	5	5	5	...	4	4	2
36	5	6	6	6	6	6	6	6	6	6	...	4	4	3
37	4	4	4	4	4	4	4	4	4	4	...	0	0	4
38	5	6	6	6	6	6	7	6	6	6	...	0	0	5
39	5	6	6	6	6	6	6	6	6	6	...	4	0	6
40	4	5	6	6	6	6	6	6	6	6	...	0	0	7
41	4	5	5	6	6	6	6	6	6	6	...	0	0	8
42	4	5	6	6	6	6	6	6	6	6	...	0	0	9
43	4	4	4	4	4	4	4	4	4	4	...	4	0	10
...
395	4	4	4	4	4	4	4	4	4	4	...	4	4	32
396	4	4	4	4	4	4	4	4	4	4	...	4	4	33

2. Perhitungan

Perhitungan data latih dilakukan dengan menggunakan persamaan rumus yaitu Persamaan (2.9). Untuk inialisasi bobot awal, maksimal epoch, *learning rate* dan minimal *learning rate* telah dijelaskan maka langkah berikutnya untuk pelatihan LVQ adalah sebagai berikut :

Epoch 1

Data ke- 34 = [4 5 4 4 4 4 4 4 4 4 4 ... 4 0]

Target (T) = 1

Bobot(W) 1 = [4 4 4 4 4 4 4 4 4 4 4 ... 4 4]

Langkah selanjutnya adalah tentukan perhitungan jarak *Euclidean Distance* dengan menggunakan persamaan rumus pada Persamaan (2.9). Langkah perhitungan jarak dapat dilihat pada perhitungan dibawah ini :

$$J_1 = \sqrt{(4 - 4)^2 + (5 - 4)^2 + (4 - 4)^2 + \dots + (0 - 4)^2}$$

$$= \sqrt{0 + 1 + 0 + \dots + 16}$$

$$= 1470$$

$$J_2 = \sqrt{(4 - 4)^2 + (4 - 4)^2 + (5 - 4)^2 + \dots + (4 - 4)^2}$$

$$= \sqrt{0 + 0 + 1 + \dots + 0}$$

$$= 5147$$

Lakukan perhitungan jarak *Euclidean* hingga bobot ke- 33. Sehingga hasil yang diperoleh dapat dilihat pada Tabel 4.16 Nilai Jarak Bobot Data ke- 34 adalah sebagai berikut :

Tabel 4.16 Nilai Jarak Bobot Data ke- 34

Jarak ke-	Nilai Jarak	Jarak ke-	Nilai Jarak	Jarak ke-	Nilai Jarak
1	1470	12	3259	23	3927
2	5147	13	4496	24	3628
3	4038	14	5369	25	4416
4	3029	15	4775	26	4533
5	4767	16	4960	27	5920
6	2881	17	6006	28	5162
7	5042	18	2383	29	5253
8	5467	19	5251	30	2253
9	6413	20	5601	31	5273
10	6884	21	2994	32	3166
11	6183	22	5139	33	4954

Setelah ditemukan nilai jarak dari setiap bobot, maka langkah selanjutnya tentukan J_{\min} terdapat dipoposi yang mana. Terlihat pada tabel nilai jarak paling minimum adalah 1470 yang berada pada bobot ke-1 .

$$J_{\min} = J_1 = 1$$

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Periksa apakah J_{\min} sama dengan Target yang telah ada. Dan terlihat bahwa $J_{\min} = T$. Maka lakukan perubahan bobot dengan menggunakan persamaan rumus pada Persamaan (2.10). Penjelasan perhitungan perubahan bobot dapat dijelaskan di bawah ini :

$$W_{11}(\text{baru}) = W_{11}(\text{lama}) + \alpha (X - W_{11}(\text{lama}))$$

$$= 4 + 0,05 (4-4)$$

$$= 4$$

$$W_{12}(\text{baru}) = W_{12}(\text{lama}) + \alpha (X - W_{12}(\text{lama}))$$

$$= 4 + 0,05 (5-4)$$

$$= 4,05$$

$$W_{13}(\text{baru}) = W_{13}(\text{lama}) + \alpha (X - W_{13}(\text{lama}))$$

$$= 4 + 0,05 (4-4)$$

$$= 4$$

Lakukan perubahan bobot dengan melakukan hal yang sama pada setiap bobot lama. Perubahan bobot dilakukan pada setiap data yang ada. Maka perubahan bobot dilakukan hingga data ke 264. Jika proses telah selesai hingga data ke- 264, maka untuk *epoch* pertama telah selesai. Untuk melanjutkan ke *epoch* berikutnya, lakukan pengurangan *learning rate* terlebih dahulu. Persamaan rumus pengurangan *learning rate* adalah :

$$\alpha = \alpha - (0,1 \times \alpha)$$

Maka untuk pengurangan *learning rate* pada penelitian ini adalah sebagai berikut:

$$\alpha = 0,05 - (0,1 \times 0,05)$$

$$= 0,045$$

Proses akan berhenti jika *epoch* sudah mencapai batas maksimum *epoch* atau *learning rate* telah mencapai nilai minimum *learning rate* . Setelah proses telah berhenti, maka didapatkan bobot terakhir yang akan digunakan pada pengujian

LVQ. Bobot terakhir dapat dilihat pada tabel 4.17 Bobot Terakhir Hasil Pelatihan LVQ berikut ini :

Tabel 4.17 Bobot Terakhir Hasil Pelatihan LVQ

Data ke-	Bobot Terakhir								Target
1	3,792396	3,360904	3,638752	3,575758	3,506308	...	-0,6664	-1,53731	1
2	4,570988	4,769965	5,45429	5,528793	6,625201	...	4,185244	4,168957	2
3	5,43322	5,670577	6,732349	6,7446	7,037879	...	0,853396	0,635711	3
4	4,09538	4,027198	3,65134	3,41061	3,291196	...	4,768634	5,267315	4
5	4,945185	5,963323	6,084092	6,28492	6,545743	...	3,428957	2,146706	5
6	5,755211	6,215715	6,444963	6,421869	6,69067	...	5,258768	2,616498	6
7	4,782012	4,40201	5,545024	6,180765	6,638588	...	-0,74709	-2,03844	7
8	4,484011	5,15896	5,61171	6,421499	6,383774	...	-0,85774	-0,77456	8
9	5,533386	6,254532	6,903918	6,903918	6,471569	...	-1,00071	-0,11424	9
10	3,891034	4,620526	3,342991	3,380346	3,821182	...	3,209702	3,01831	10
...
32	4,296761	5,287885	6,256633	6,164129	7,232307	...	0,928417	-1,17015	32
33	3,889766	3,78985	3,559387	3,783956	3,533426	...	4,087281	3,375061	33

2. Pengujian LVQ

Hasil dari nilai bobot terakhir yang sudah ada, akan menjadi acuan pada proses pengujian. Pada pengujian, memiliki data sebanyak 99 data. Hasil citra pada data uji dapat dilihat pada Tabel 4.18 Nilai Uji Citra Huruf Jepang Katakana:

Tabel 4.18 Nilai Uji Citra Huruf Jepang Katakana

Data ke-	Hasil Ekstraksi Ciri												Target	
	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀	...	X ₄₉₉		X ₅₀₀
1	4	4	4	4	4	4	4	4	4	4	...	4	4	1
2	5	5	6	6	5	5	5	5	5	5	...	4	4	2
3	5	6	6	6	6	6	6	6	6	6	...	0	0	3
4	5	5	4	4	4	4	4	4	4	4	...	4	4	4
5	5	5	6	6	6	6	6	6	6	6	...	0	0	5
6	4	5	6	6	6	6	6	6	6	6	...	4	4	6
7	4	6	6	6	6	6	6	6	6	6	...	4	4	7

8	4	4	6	6	6	6	6	6	6	6	...	0	0	8
9	5	6	6	6	6	6	6	6	6	6	...	0	0	9
10	4	5	4	4	5	4	4	4	4	4	...	4	4	10
...
98	4	4	4	4	4	4	4	4	4	4	...	4	0	32
99	4	4	4	4	4	4	4	4	4	4	...	3	3	33

Pengujian data uji dilakukan dengan menentukan jarak *Euclidean*-nya. Untuk melakukan persamaan *Euclidean Distance*, dapat dilakukan dengan persamaan rumus pada Persamaan (2.9) dan bobot yang digunakan adalah bobot terakhir hasil akhir dari pelatihan LVQ. Perhitungannya adalah sebagai berikut ini:

$$\text{Data uji} = [4 \ 4 \ 4 \ 4 \ 4 \ 4 \ 4 \ 4 \ 4 \ 4 \ 4 \ \dots \ 4 \ 4]$$

$$\text{Bobot (W)} = [3,792396 \ 3,360904 \ 3,638752 \ 3,575758 \ 3,506308 \ \dots \ -1,53731]$$

$$J1 = \sqrt{(4 - 3,792396)^2 + (4 - 3,360904)^2 + \dots + (4 - (-1,53731))^2}$$

$$= \sqrt{0,043099 + 0,408444 + \dots + 30,66176}$$

$$= 6153,473$$

$$J2 = \sqrt{(4 - 4,570988)^2 + (4 - 4,769965)^2 + \dots + (4 - 4,168957)^2}$$

$$= \sqrt{0,326027 + 0,592846 + \dots + 0,028546}$$

$$= 6833,26$$

Lakukan persamaan diatas hingga bobot ke- 33. Sehingga hasil dari persamaan di atas untuk 33 bobot pada data ke- 1 dapat dilihat pada Tabel 4.19 Nilai Jarak Uji Data ke- 1 adalah sebagai berikut :

Tabel 4.19 Nilai Jarak Uji Data ke-1

Jarak ke-	Nilai Jarak	Jarak ke-	Nilai Jarak	Jarak ke-	Nilai Jarak
1	6153,473	12	5670,975	23	4126,124
2	6833,260	13	4513,205	24	4427,506
3	6272,907	14	6656,504	25	5298,270

4	3448,477	15	4715,757	26	3448,476
5	5385,449	16	3457,201	27	4178,470
6	5973,298	17	5770,681	28	3603,734
7	4022,467	18	7130,549	29	5915,771
8	5668,886	19	4573,554	30	5399,569
9	4709,336	20	5423,964	31	4801,148
10	7427,817	21	6384,131	32	6877,890
11	5599,842	22	7649,677	33	4258,410

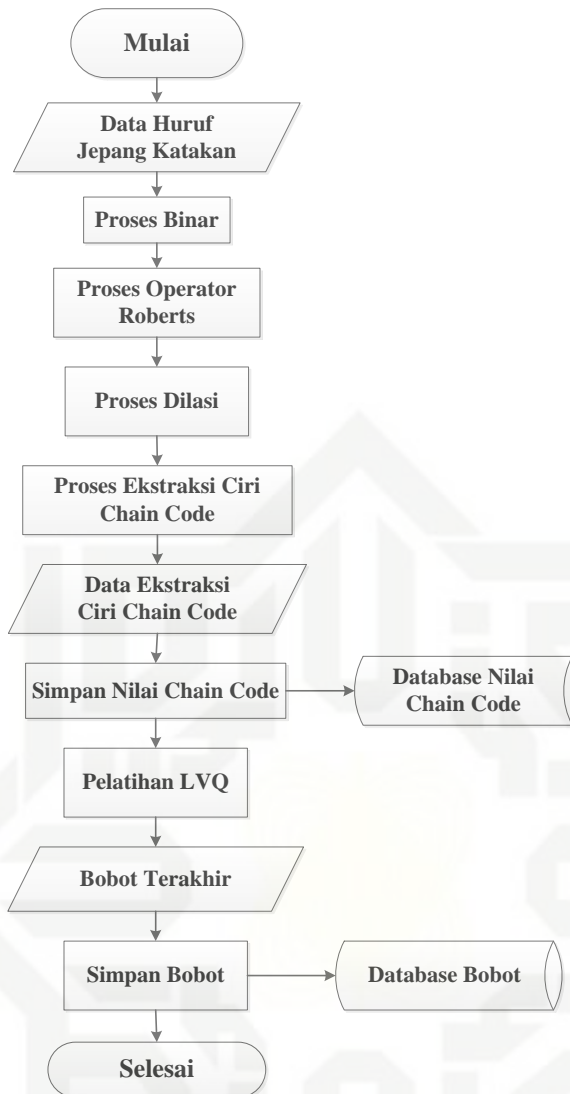
Lakukan pencarian jarak minimum. Pada Tabel 4.18 dapat dilihat hasil dari jarak minimum adalah 3448,476 yang terdapat pada jarak ke- 26. Dapat diambil kesimpulan bahwa data yang telah di uji berada pada kelas ke- 26 dan pengujian tidak sesuai dengan hasil yang didapatkan karena data yang diuji adalah data ke-1 dengan jarak ke-1 namun setelah diuji, kelas yang dikenalnya adalah jarak ke-26.

4.2. Analisa Aplikasi

Alur proses aplikasi dalam Pengenalan Pola karakter huruf Jepang menggunakan *Chain Code* dan LVQ dapat digambarkan dengan *flowchart*. *Flowchart* alur sistem terbagi menjadi dua. Pertama adalah *flowchart* pada pelatihan dan *flowchart* kedua pada pengujian. *Flowchart* pada pelatihan dapat dilihat pada Gambar 4.6 *Flowchart* Proses Pelatihan Pengenalan Huruf Jepang Katakana .

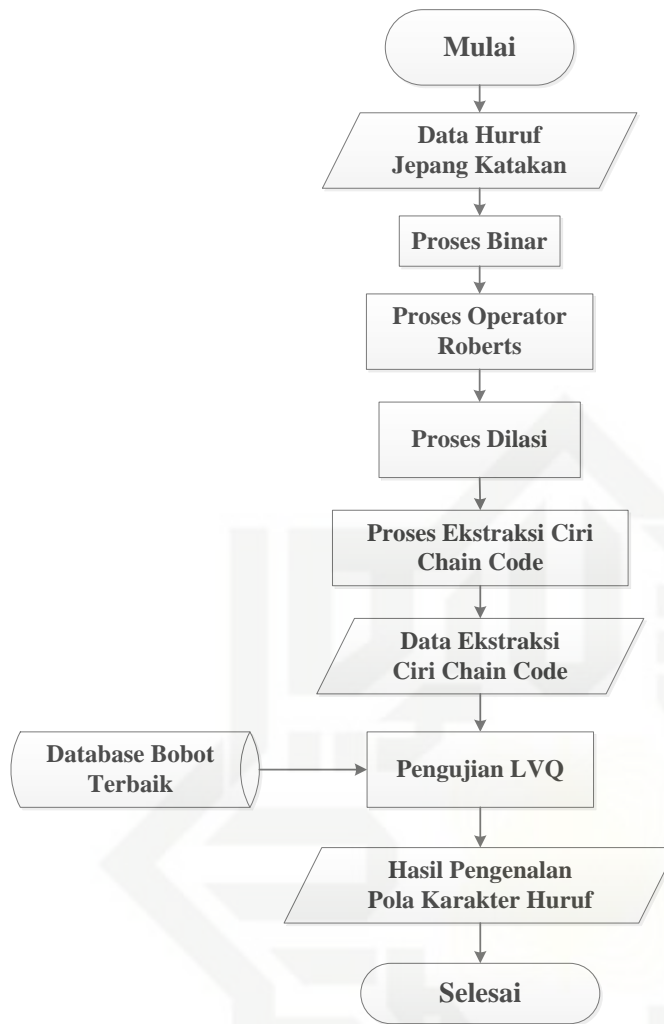
Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Gambar 4.6 Flowchart Proses Pelatihan Pengenalan Huruf Jepang Katakana

Data huruf Jepang Katakana yang sudah di scan terlebih dahulu dilakukan tahap *pre-processing*. Tahapan ini terdiri dari mengkonversikan data dari biner ke operator Roberts kemudian hasil dari operator Roberts di tebakkan deteksi tepinya dengan menggunakan dilasi. Setelah tahap *pre-processing* telah selesai lanjut ke tahap berikutnya yaitu ekstraksi ciri dengan menggunakan metode *Chain Code*. Hasil dari *chain code* akan disimpan untuk ke tahapan selanjutnya. Hasil *chain code* digunakan untuk pelatihan LVQ. Pada LVQ terdapat bobot terbaik yang nantinya digunakan untuk pengujian pada LVQ. Bobot terbaik tersebut di simpan ke dalam database penyimpanan. Selanjutnya *flowchart* pada pengujian dapat dilihat pada Gambar 4.7 *Flowchart Proses Pengujian Pengenalan Huruf Jepang Katakana*



Gambar 4.7 Flowchart Proses Pengujian Pengenalan Huruf Jepang Katakana

Alur proses Pengujian Pengenalan Pola Karakter Huruf Jepang Katakana terdapat persamaan namun perbedaannya terletak pada pelatihan LVQ. Pada pelatihan, nilai dari *chain code* di simpan terlebih dahulu baru di latih sementara pada pengujian, nilai dari *chain code* langsung digunakan pada pengujian LVQ. Setelah itu sistem akan mengenali huruf yang telah di uji tersebut.

4.3. Perancangan

Perancangan adalah tampilan grafis aplikasi yang nantinya akan digunakan dalam mengimplementasikan pengenalan pola karakter Huruf Jepang Katakana. Perancangan yang dilakukan terdiri dari beberapa rancangan. Diantaranya adalah sebagai berikut ini :

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

1.

Perancangan Halaman Awal

Perancangan halaman awal yang di tampilkan terdapat beberapa menu yang ada, diantaranya adalah Pelatihan, Pengujian, Tutorial dan Keluar. Perancangan Halaman Awal ini dapat dilihat pada Gambar 4.8 Perancangan Halaman Awal

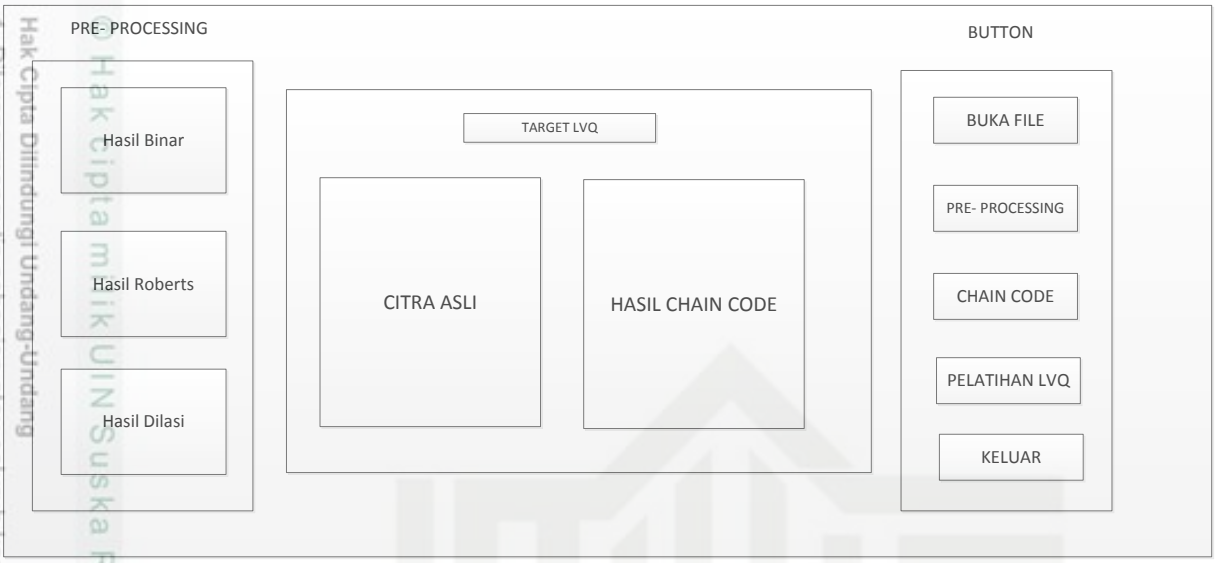


Gambar 4.8 Perancangan Halaman Awal

2.

Perancangan Halaman Pelatihan

Halaman Pelatihan adalah halaman yang dijadikan untuk melatih data yang ada. Proses yang dilakukan di pelatihan ini adalah berawal dari proses *pre-processing*, ekstraksi ciri *chain code* dan pelatihan menggunakan LVQ. Perancangan Halaman Pelatihan ini dapat dilihat pada Gambar 4.9 Perancangan Halaman Pelatihan



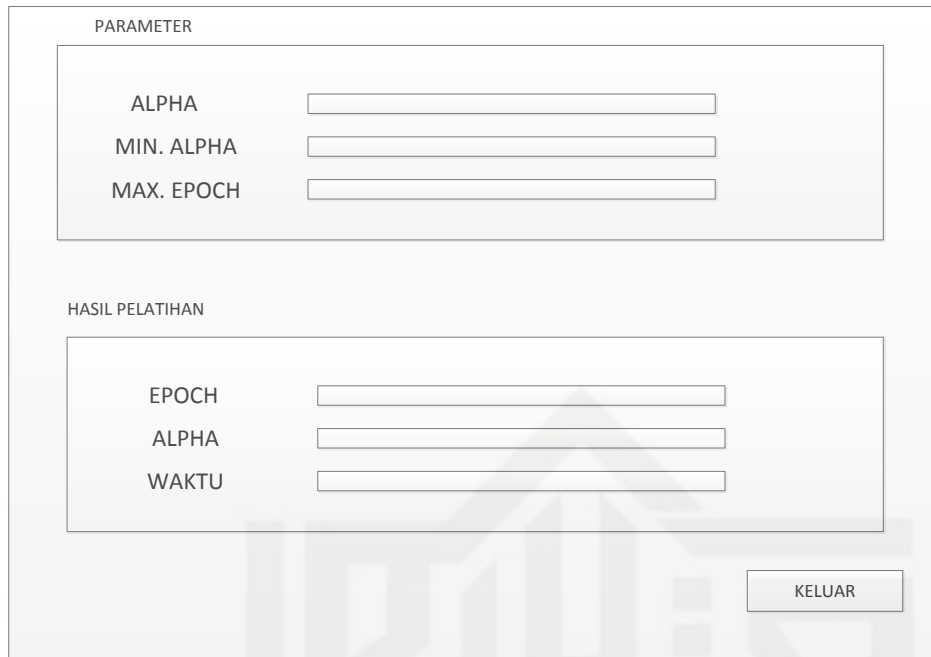
Gambar 4.9 Perancangan Halaman Pelatihan

Halaman Pelatihan ini terdapat beberapa *button* yang di gunakan. Diantaranya adalah *Buka File*, *Pre- Processing*, *Chain Code*, *Pelatihan LVQ* dan *Keluar*. Pada *Buka File* di klik, maka akan muncul *open file* yang menandakan agar memilih data yang akan di proses untuk tahap selanjutnya. Kemudian pada *button pre-processing* di klik, maka sistem akan menjalankan proses *pre-processing* di mulai dari *resize* data dan tahap akhir adalah dilasi. Hasil yang ada akan di tampilkan pada kotak Hasil Biar, Hasil Roberts dan Hasil Dilasi. Kemudian jika di klik *button chain code* maka sistem akan menjalankan proses tersebut dan hasil *chain code* akan di tampilkan berupa grafik yang akan di letakkan pada kotak Hasil Chain Code. Tahap akhir yaitu pelatihan LVQ. Pada pelatihan LVQ ini, data telah dibagi menjadi data uji dan data latih. Jika *button* pelatihan LVQ di klik, maka sistem akan melakukan proses pelatihan dan sistem akan menampilkan halaman Pelatihan LVQ yang dapat dilihat pada Gambar 4.10 Perancangan Halaman Pelatihan LVQ

Hak Cipta Diindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.



Gambar 4.10 Perancangan Halaman Pelatihan LVQ

3. Perancangan Halaman Pengujian

Perancangan Halaman Pengujian adalah halaman yang dijadikan untuk menguji data yang akan di uji. Pada halaman ini huruf yang akan di uji akan di kenali apakah huruf ini dikenali sebagai dirinya atau tidak di kenali. Perancangan Halaman Pengujian dapat dilihat pada Gambar 4.11 Perancangan Halaman Pengujian

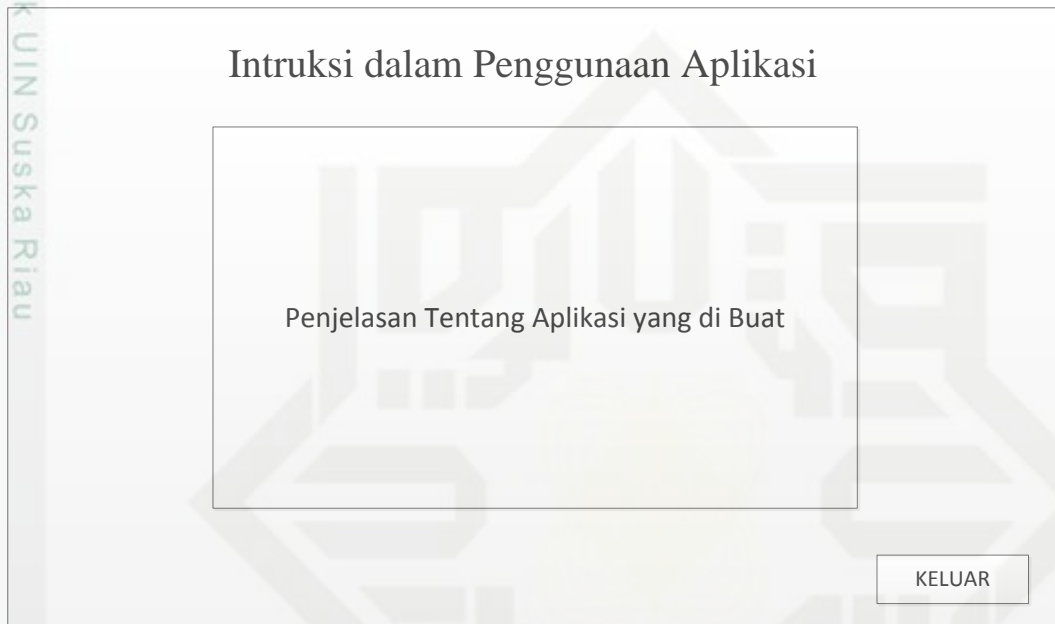


Gambar 4.11 Perancangan Halaman Pengujian

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

4. Perancangan Halaman Tutorial
Tampilan pada halaman tutorial adalah menjelaskan tentang aplikasi yang dibuat. Perancangan Halaman Tutorial dapat dilihat pada Gambar 4.12 Perancangan Halaman Tutotrial



Gambar 4.12 Perancangan Halaman Tutorial