BAB 3

ANALISIS DAN KEBUTUHAN ALGORITMA

3.1 Analisis Masalah

Jaringan saraf tiruan hopfield merupakan salah satu Algoritma *Machine Learning* yang dapat mengklasifikasikan suatu objek citra berdasarkan pelatihan yang diberikan. Proses klasifikasi akan dilakukan setelah citra di ekstrasi terlebih dahulu untuk mendapatkan ciri utama dari citra tersebut. Metode yang dipakai sebagai proses ekstraksi adalah metode kookurensi. Kookurensi adalah salah satu metode untuk memperoleh ciri-ciri citra tekstur dengan menghitung probabilitas hubungan ketetanggan antara dua piksel pada jarak dan arah tertentu. Dengan 5 ciri tekstur atau parameter yang didapat dari metode kookurensi yakni entropi, energi, kontras, homogenitas dan korelasi sudah dapat mengenali tekstur citra hasil ekstraksi ciri tersebut kemudian akan digunakan sebagai nilai masukan untuk menentukan hasil klasifikasi citra berdasarkan teksturnya.

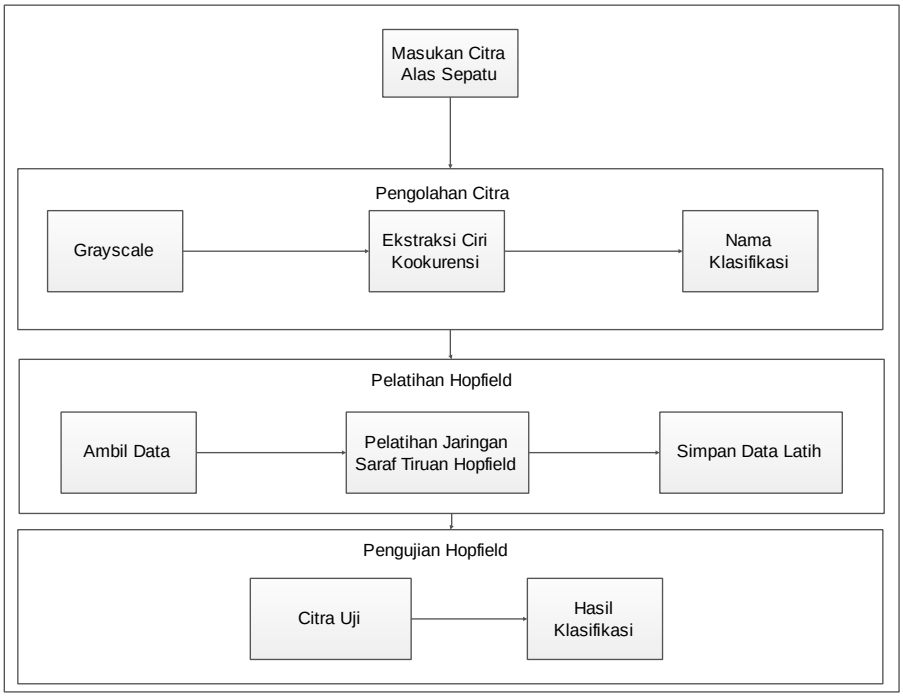
Citra tekstur alas sepatu dapat di klasifikasikan berdasarkan informasi yang terdapat pada citra alas sepatu tersebut. Karena untuk pihak-pihak tertentu citra tekstur alas sepatu bisa digunakan sebagai salah satu bukti forensik yang berharga, biasanya tertinggal pada suatu tempat kejahatan selain sidik jari, darah, dan rambut. Dengan mengenali tekstur dari sebuah citra benda adalah salah satu cara yang dapat digunakan untuk mengenali, mengidentifikasi, dan mengklasifikasikan informasi dari sebuah citra benda tersebut.

Dalam penelitian sebelumnya proses pengklasifikasian berdasarkan tekstur telah dilakukan dengan menggunakan jaringan saraf tiruan perambatan balik sebagai metode pengklasifikasian dan kookurensi sebagai ekstraksi ciri dan dalam penelitian tersebut menyarankan agar melakukan penelitian pengklasifikasian citra berdasarkan tekstur dengan menggunakan jaringan saraf tiruan hopfiled sebagai metode pengklasifikasiannya. Maka perlu dilakukan implementasi jaringan saraf tiruan hopfield dan metode kookurensi untuk mengklasifikasikan citra, untuk

mengukur tingkat keakuratan pengklasifikasian citra berdasarkan teksturnya dengan menggunakan 5 ciri kookurensi.

3.2 Analisis Proses

Pada analisis proses ini akan menjelaskan sub proses yang dimiliki oleh analisis sistem yang akan dibangun. Dapat dilihat pada Gambar 3.3.



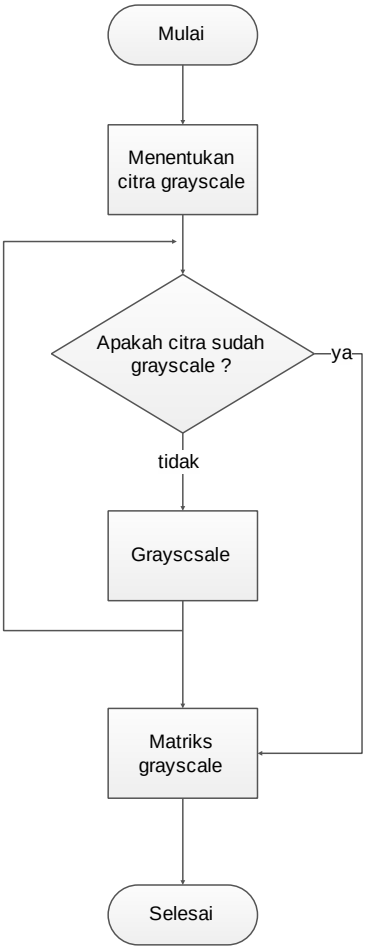
Gambar 3. 1. Alur Analisis Proses

3.2.1 Analisis Pengolahan Citra

Dalam penerapan metode kookurensi untuk ekstraksi tekstur pada citra digital dalam pengolahan citra akan diproses menjadi 3 langkah yaitu *grayscale*, ekstraksi ciri kookurensi dan nama klasifikasi. Citra masukan akan diubah menjadi keabu-abuan dengan cara *grayscale*, kemudian untuk mendapatkan ciri utama yang terdapat dalam citra digital digunakan ektraksi ciri kookurensi, dan selanjutnya menentukan nama klasifikasi

3.2.1.1 *Grayscale*

Grayscale merupakan proses untuk mengubah warna menjadi keabu-abuan. Dengan mengubah nilai RGB setiap piksel gambar menjadi satu nilai yang sama sehingga setiap piksel memiliki nilai yang sama untuk ketiga unsur warna serta didapatkan nilai matriks *grayscale*. Berikut alur dari proses *grayscale* :

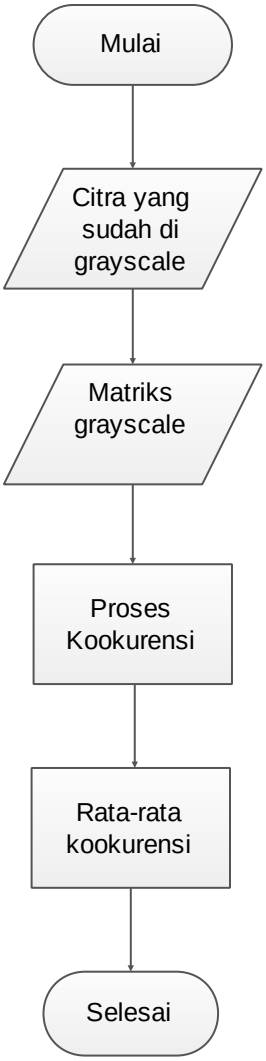


Gambar 3. 2. Alur Proses *Grayscale*

3.2.1.2 Ekstraksi Ciri Kookurensi

Ekstraksi ciri kookurensi adalah proses untuk mendaptkan ciri utama yang terdapat pada citra, citra yang telah di *grayscale* akan menghasilkan matriks *grayscale*, matriks tersebutlah yang akan digunakan pada tahap ini. Tahap ini akan menghitung 5 nilai statistik dari kookurensi yaitu kontras,

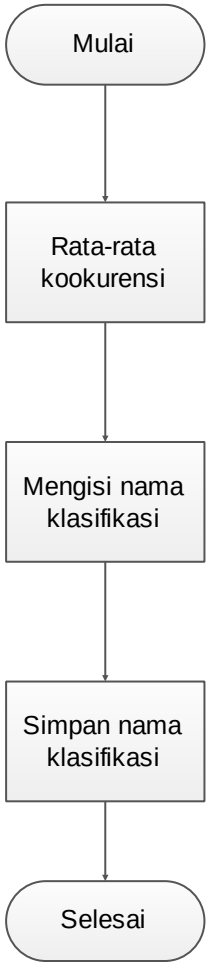
energi, entropi, homogenitas dan korelasi dengan sudut simetri $0^0, 45^0, 90^0$ dan 135^0 . Setelah didapatkan semua nilai dari sudut akan dirata-ratakan. Berikut alur proses dari ekstraksi ciri kookurensi :



Gambar 3. 3. Alur Proses Ekstraksi Ciri Kookurensi

3.2.1.3 Nama Klasifikasi

Nama klasifikasi digunakan untuk menentukan nama klasifikasi yang telah didapatkan nilai rata-rata statistik dari ekkstraksi ciri kookurensi. Berikut alur proses dari nama klasifikasi :



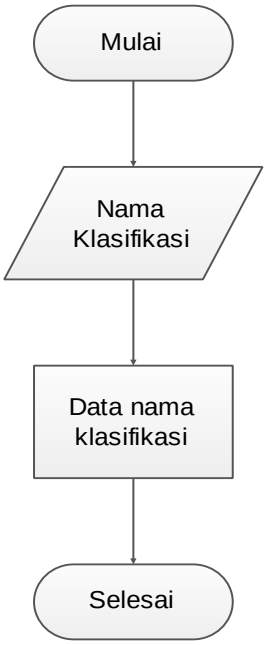
Gambar 3. 4. Alur Proses Nama Klasifikasi

3.2.2 Pelatihan Hopfield

Pelatihan hopfield dilakukan untuk mendapatkan data latih, proses ini dilakukan menjadi 3 tahap yaitu ambil data, latih data dan simpan data. Tahapan-tahapan tersebut akan dijelaskan sebagai berikut :

3.2.2.1 Ambil Data

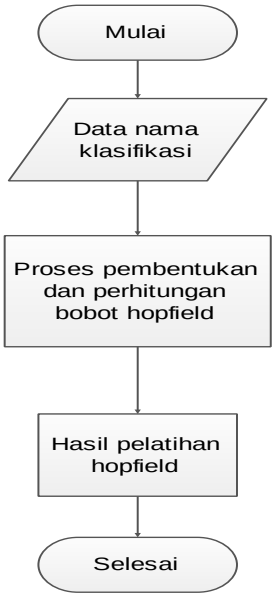
Ambil data berfungsi sebagai pengambilan data yang telah disimpan sebelumnya. Berikut alur proses dari ambil data :



Gambar 3. 5. Alur Proses Ambil Data

3.2.2.2 Pelatihan Jaringan Saraf Tiruan Hopfield

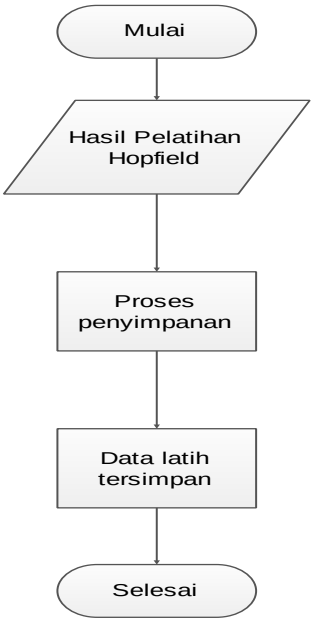
Pelatihan ini berfungsi sebagai pembelajaran jaringan hasil dari proses ini akan digunakan dalam proses selanjutnya . Berikut adalah alur proses dari pelatihan jaringan saraf tiruan hopfield :



Gambar 3. 6. Alur Proses Pelatihan Jaringan Saraf Tiruan

3.2.2.3 Simpan Data Latih

Simpan data latih berguna untuk menyimpan arsitektur beserta hasil proses pelatihan ke dalam suatu file sehingga dapat digunakan dikemudian hari. Berikut alur dari simpan data latih :



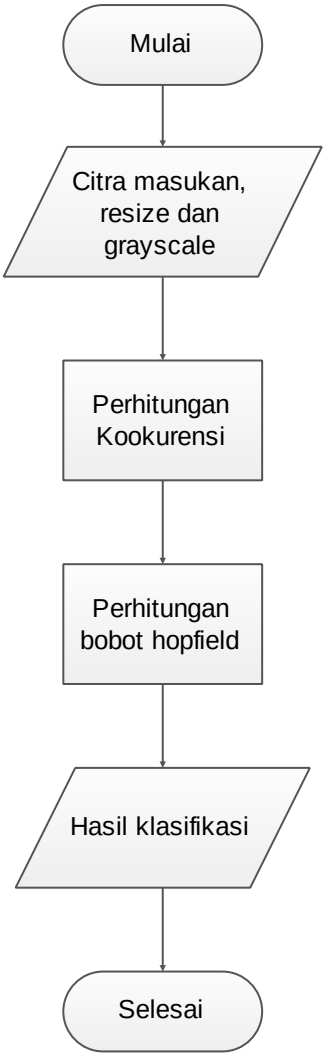
Gambar 3. 7. Alur Proses Simpan Data Latih

3.2.3 Pengujian Hopfield

Pengujian adalah tahapan terakhir pada proses klasifikasi citra berdasarkan tekstur, pada proses ini citra yang dimasukan dapat dihasilkan klasifikasinya untuk lebih jelasnya akan dijelasknn sebagai berikut :

3.2.3.1 Hasil Klasifikasi

Pada proses ini akan menunjukan hasil klasifikasi dari data yang telah diltihnya. Berikut alur proses hasil klasifikasi :



Gambar 3. 8. Alur Proses Hasil Klasifikasi

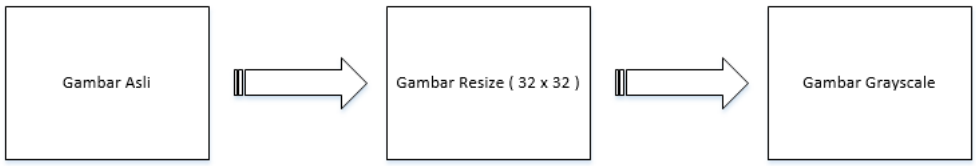
3.3 Analisis Data

Analisis data terdiri dari analisis data masukan dan analisis data keluaran. Sampel data berupa citra alas sepatu yang telah dipersiapkan sebelumnya.

3.3.1 Analisis Data Masukan

Data masukan berupa citra alas sepatu. Kemudian citra akan di resize menjadi ukuran 32 x 32 piksel dengan *true color* yakni tiap piksel terdapat 3 nilai yaitu RGB, kemudian citra akan di konversi menjadi *grayscale* hal ini dilakukan untuk menyederhanakan dan memudahkan proses selanjutnya, dengan tujuan menjadikan tiap piksel memiliki satu nilai yaitu nilai keabuan *(gray value)*. Setelah citra di konversi menjadi *grayscale* akan didapatkan matriks *grayscale* tersebut, matriks ini akan digunakan sebagai masukan proses ekstraksi citra kookurensi.

Pada proses ekstraksi ini matriks *grayscale* digunakan sebagai pembentuk matriks kookurensi, kemudian dari matriks kookurensi tersebut dapat ditentukan nilai probabiltasnya. Langkah selanjutnya adalah ekstraksi ciri setelah ekstraksi ciri ini didapatkan dan akan dijadikan sebagai masukan untuk klasifikasi.



Gambar 3. 9. Tahapan Proses Pengolahan Citra

3.3.2 Analisis Data Keluaran

Data keluaran adalah data menghasilkan nama klasifikasi dari citra, data ini didapatkan dari serangkaian proses mulai dari pengolahan citra, pelatihan dan pengujian. Di pengolahan citra menggunakan metode kookurensi yang menghasilkan ekstraksi ciri dengan bantuan *grayscale* untuk menjadikan citra memiliki satu nilai tunggal di dalam setiap pikselnya, kemudian di

pelatihan menggunakan jaringan saraf tiruan hopfield dan di pengujian menggunkan proses pengolahan citra dan pelatihan sehingga dihasilkan hasil klasifikasi.

3.4 Analisi Metode / Algoritma

Analisis algoritma yang akan diteliti pada klasifikasi citra berdasarkan tekstur ini adalah untuk menganalisis cara kerja jaringan saraf tiruan hopfield dalam mengklasifikasi citra berdasarkan tekstuk, sebelum dapat mengklafisikasi citra masukan harus melalui beberapa tahap, tahapan pengolahan citra dan pelatihan setelah itu baru pengujian pada tahapan pengujian inilah citra dapat diklasifikasikan.

3.4.1 Analisis Tahapan Pengolahan Citra

Tahapan pengolahan citra dilakukan adalah untuk mendapatkan ekstraksi ciri yang ada pada citra tersebut, berikut adalah proses untuk mendapatkan ekstraksi ciri :

3.4.1.1 Proses Grayscaling

Grayscaling adalah proses mengubah citra RGB menjadi citra keabuan, yang bertujuan untuk menyderhanakan niliai piksel yang sebelunmnya memilik 3 nilai pada setiap piksel menjadi satu nilai piksel.

Rumus yang digunakan untuk proses grayscsale adalah

Grayscale = R*0,3 + G*0,59 + B*0,11

Dimana : R = nilai red * 31 / 255

G = nilai green * 31 / 255

B = nilai blue * 31 / 255

Contoh perhitungan *grayscale* :

Untuk dapat melakukan perhitungan *grayscale* dibutuhkan citra masukan, berikut adalah citra masukan data 1 yang akan di uji :

Data masukan 1 :



Gambar 3. 10. Citra Alas Sepatu Futsal

Perhitungan grayscsale piksel (0,0) :

R = 255 * 31/ 255 = 31

G = 255 * 31/ 255 = 31

B = 255 * 31/ 255 = 31

Gray = 31*0,3 + 31*0,59 + 31*0,11

Gray = 9,3 + 18,29 + 3,41

Gray = 31

Dengan menggunakan rumus yang sama terhadap semua piksel maka akan didapatkan tabel matriks grayscsale dilampirkan pada lampiran B-5.

3.4.1.2 Ekstraksi Ciri Kookurensi

Tahapan-tahapan dalam mencari ekstraksi ciri kookurensi adalah sebagai berikut :

1. Membentuk Framework

   Dalam penelitian ini dibuat citra dengan ukuran 32 x 32 piksel dan maka akan dibentuk pula framework dengan ukuran 32 x 32 karena memiliki 32 aras keabuan. Maka akan dibentuk framework kookurensi yang dilampirkan lampiran B-6.

2. Membentuk Matriks Kookurensi

   Matriks kookurensi yang akan dibuat adalah hubungan spasial untuk d=1 dan sudut 0^0 . matriks kookurensi dibuat dengan mengisikan jumlah hubungan spasial yang ada matriks *grayscale*. Maka didapatkanlah matriks kookurensi dilampirkan pada lampiran B-7.

3. Mentranpose Matriks

   Matriks kookurensi yang sudah didapatkan kemudian di tranpose, tujuan dari tahapan ini adalah untuk mendapatkan sudut simetrisnya yaitu sudut 180^0. matriks yang terbentuk adalah sebagai berikut : Maka didapatkanlah tranpose matriks kookurensi dilampirkan pada lampiran B-8.

4. Menjumlahkan Matriks Kookurensi Dan Matriks Transpose

   Tahapan ini adalah untuk menjadikannya simetris antara sudut 0^0 dan sudut 180^0 . Maka didapatkanlah matriks simetris dilampirkan pada lampiran B-9.

5. Normalisasi

   Normalisasi didapatkan dengan cara menjumlah semua matriks simetris, kemudian dijadikan pembagi untuk semua piksel yang ada pada matriks simetris. Maka didapatkan matriks kookurensi dilampirkan pada lampiran B-10.

6. Perhitungan fitur tekstur

   Setelah didapatkan matriks kookurensi sampai mendapatkan hasil normalisasinya, tahapan selanjutnya adalah menghitung tiap fitur tekstur, yaitu : kontras, homogenitas, energi dan entropi. Perhitungan fitur tekstur adalah sebagai berikut :

   a. Kontras

      Rumus $\sum_{i,j=0}^{n-1} Pi,j(i-j)^2$ (2.6)

Dimana : i dan j adalah sifat keabuan dari resolusi 2 piksel yang berdekatan

P(i,j) adalah probabilitas kolom (i,j)

Syarat : ketika i = j, maka kontras diisi 0

$P(0,0).(0-0)^2 = 0 \cdot 0{,}00222 = 0$

$P(0,1).(0-1)^2 = 0{,}0055 \cdot 1 = 0{,}0055$

Dengan langkah yang sama untuk semua piksel maka terbentuklah matiks kontras dilampirkan pada lampiran B-10.

Maka :

$\sum_{i,j=0}^{n-1} Pi,j(i-j)^2 = 14{,}072$

Nilai kontras adalah 14,072

b. Homogenitas

Rumus $\sum_{i,j=0}^{n-1} \frac{P(i,j)}{1+|i-j|}$ (2.7)

Dimana : i dan j adalah sifat keabuan dari resolusi 2 piksel yang berdekatan

P(i,j) adalah probabilitas kolom (i,j)

$P(0,0)/[1+(0-0)^2] = 0{,}0222 / 1 = 0{,}0222$

$P(0,1)/[1+(0-1)^2] = 0{,}0055 / 2 = 0{,}0028$

Dengan langkah yang sama untuk semua piksel maka terbentuklah matiks homogenitas dilampirkan pada lampiran B-11.

Maka :

$\sum_{i,j=0}^{n-1} \frac{P(i,j)}{1+|i-j|} = 0{,}059$

Nilai homogenitas adalah 0,059

c. Energi

Rumus $\sum_{i,j=0}^{n-1} P^2(i,j)$ (2.4)

Dimana : i dan j adalah sifat keabuan dari resolusi 2 piksel yang berdekatan

P(i,j) adalah probabilitas kolom (i,j)

$P(0,0)^2 = 0{,}0222^2 = 0{,}00049$

$P(0,1)^2 = 0.0055^2 = 0{,}00037$

Dengan langkah yang sama untuk semua piksel maka terbentuklah matiks energi dilampirkan pada lampiran B-12.

Maka :

$\sum_{i,j=0}^{n-1} P^2(i,j) = 0{,}064$

Nilai energi adalah 0,064

d. Entropi

Rumus $\sum_{i,j=0}^{n-1} \mathrm{P}(i,j)(-\log P(i,j))(2.5)$

Dimana : i dan j adalah sifat keabuan dari resolusi 2 piksel yang berdekatan

P(i,j) adalah probabilitas kolom (i,j)

P((0,0)(-logP(0,0))) = (0.-log(0,0222)=0

P((0,1)(-logP(0,1))) = (0,0192.-log(0,0055) = 0,067

Dengan langkah yang sama untuk semua piksel maka terbentuklah matiks entropi dilampirkan pada lampiran B-13.

Maka :

$\sum_{i,j=0}^{n-1} \mathrm{P}(i,j)(-\log P(i,j)) = 0{,}793$

Nilai entropi adalah 0,793

Setelah dihitung semua fitur kemudian dijumlah tiap matriks dari fitur tersebut, maka didapatkan hasil sebagai berikut :

Tabel 3. 1. Nilai Fitur Sudut Simetris 0^0

Fitur	Sudut 0^0
Kontras	14,072
Homogen	0,059
Entropi	0,793
Energi	0,064
Korelasi	0,028

Dengan menggunakan langkah yang sama untuk sudut simetris 45^0, 90^0 dan 135^0 maka akan didapatkan nilai sebagai berikut :

Tabel 3. 2. Nilai Fitur Sudut Simetris 0^0, 45^0, 90^0 Dan 135^0

(Data Masukan 1)

Fitur	Sudut 0^0	Sudut 45^0	Sudut 90^0	Sudut 135^0
Kontras	14,072	25,468	25,025	25,468
Homogen	0,059	0,034	0,036	0,034
Entropi	0,793	0,772	0,783	0,772
Energi	0,064	0,056	0,046	0,056
Korelasi	0,028	0,025	0,026	0,025

Data masukan 2 :



Gambar 3. 11. Citra Alas Sepatu Badminton

Dengan menggunakan langkah-langkah yang sama seperti data masukan 1. Data masukan 2 pun akan di proses mulai dari *grayscale*, perhitungan matriks kookurensi, transpose, normalisasi dan ekstraksi fitur. Maka akan didapatkan hasil sebagai berikut :

Tabel 3. 3. Nilai Fitur Sudut Simetris 0^0, 45^0, 90^0 Dan 135^0

(Data Masukan 2)

Fitur	Sudut 0	Sudut 45	Sudut 90	Sudut 135
Kontras	2,598	1,646	2,273	1,646
Homogen	0,277	0,372	0,272	0,372
Entropi	1,494	1,511	1,420	1,511
Energi	0,138	0,191	0,131	0,191
Korelasi	0,031	0,031	0,031	0,031

3.4.1.3 Nama Klasifikasi

Setelah didapatkan semua fitur dari sudut 0^0, 45^0, 90^0 Dan 135^0 . Tahapan selanjutnya adalah memberikan nama klasifikasi, sebelum nama klasifikasi diberikan nilai-nilai dari semua sudut di rata-ratakan terlebih dahulu agar mendapatkan satu nilai tunggal dari setiap fitur yang telah didapatkan. Berikut nama klasifikasi dengan nilai dari fitur yang telah di rata-ratakan :

Tabel 3. 4. Nama Klasifikasi Dan Rata-Rata Fitur

Nama Klasifikasi	Kontras	Homogenitas	Entropi	Energi	Korelasi
Sepatu Futsal	22,508	0,041	0,78	0,055	0,026
Sepatu Badminton	2,04	0,323	1,484	0,163	0,031

3.4.2 Analisis Tahapan Pelatihan

Tahapan selanjutnya adalah pelatihan dengan menggunakan jaringan saraf tiruan hopfiled, pelatihan ini berfungsi untuk membangkitkan nilai aktivasi yang akan digunakan untuk pengklasifikasian. Langkah – langkah pada tahap ini adalah ambil data, latih data, dan simpan data. Untuk lebih jelasnya adalah sebagai berikut:

3.4.2.1 Ambil Data

Ambil data ini adalah berfungsi sebagai data masukan yang akan dilatih dengan menggunakan jaringan saraf tiruan hopfield. Data yang dipakai untuk pelatihan adalah nama klasfikasi dan rata-rata fitur, data ini di ambil dari hasil proses pengolahan citra sebelumnya. Data-datanya sebagai berikut :

Tabel 3. 5. Nama Klasifikasi Dan Rata-Rata Fitur

Nama Klasifikasi	Kontras	Homogenitas	Entropi	Energi	Korelasi
Sepatu Futsal	22,508	0,041	0,78	0,055	0,026
Sepatu Badminton	2,04	0,323	1,484	0,163	0,031

3.4.2.2 Latih Data

Latih data akan dilakukan setelah data masukan telah tersedia, kemudian akan dicari nilai aktivasinya, nilai aktivasi tersebut didapatkan dengan cara membentuk nilai bobot dan nilai bobot tersebut akan dikalikan dengan data nama klasifikasi. langkah-langkah latih data sebagai berikut :

a. Membentuk bobot latih

Bobot latih yang dipakai adalah dengan cara acak, angka yang digunakan adalah dari 0 sampai 1. Dan untuk setiap data nama klasifikasi maka akan dibentuk 1 bobot latih, dalam kasus ini ada 2 data nama klasifikasi maka akan dibentuk 2 bobot latih. Bobot latih yang dibentuk sebagai berikut :

Tabel 3. 6. Bobot Latih 1

No	w1	w2	w3	w4	w5
1	0	0,382	0,777	0,657	0,404
2	0,616	0	0,754	0,913	0,594
3	0,734	0,523	0	0,143	0,819
4	0,741	0,232	0,453	0	0,067
5	0,2	0,725	0,354	0,681	0

Tabel 3. 7. Bobot Latih 2

No	w1	w2	w3	w4	w5
1	0	0,739	0,322	0,68	0,552
2	0,729	0	0,299	0,331	0,343
3	0,078	0,113	0	0,836	0,568
4	0,331	0,068	0,67	0	0,162
5	0,686	0,457	0,92	0,795	0

b. Mengalikan data nama klasifikasi dengan bobot

Tahap ini dilakukan untuk mendapatkan nilai yang akan dimasukan untuk nilai aktivasi. Data hasil perkalian adalah sebagai berikut :

Perkalian data futsal dengan bobot 1 :

$$[22{,}508\ 0{,}041\ 0{,}78\ 0{,}055\ 0{,}026]\ X \begin{bmatrix} 0\ 0{,}382\ 0{,}777\ 0{,}657\ 0,404 \\ 0{,}616\ 0\ 0{,}754\ 0{,}913\ 0{,}594 \\ 0{,}734\ 0{,}523\ 0\ 0{,}143\ 0{,}819 \\ 0,714\ 0{,}232\ 0{,}453\ 0\ 0{,}067 \\ 0{,}2\ 0{,}725\ 0{,}354\ 0{,}681\ 0 \end{bmatrix}$$

Tabel 3. 8. Hasil Perkalian Data Futsal Dengan Bobot 1

Ciri	Data x W
Kontras	0,644
Homogenitas	9,038
Entropi	17,554
Energi	14,955
Korelasi	9,76

Perkalian data badminton dengan bobot 2 :

$$[2{,}04\ 0{,}323\ 1{,}484\ 0{,}163\ 0{,}031]\ \text{X}\ \begin{bmatrix} 0\ 0{,}739\ 0{,}322\ 0{,}68\ 0{,}552 \\ 0{,}729\ 0\ 0{,}299\ 0{,}331\ 0{,}343 \\ 0{,}078\ 0{,}113\ 0\ 0{,}836\ 0{,}568 \\ 0{,}331\ 0{,}068\ 0{,}67\ 0\ 0{,}162 \\ 0{,}686\ 0{,}457\ 0{,}92\ 0{,}795\ 0 \end{bmatrix}$$

Tabel 3. 9. Hasil Perkalian Data Badminton Dengan Bobot 2

Ciri	Nilai	Nilai x W
Kontras	2,04	0,426
Homogenitas	0,323	1,701
Entropi	1,484	0,892
Energi	0,163	2,76
Korelasi	0,031	2,106

c. Nilai Aktivasi

Nilai aktivasi digunakan untuk mengenali nama klasifikasi pada saat pengujian. Nilai akivasi yang digunakan adalah X = Y, nilai X adalah nilai hasil dari perkalian bobot (Nilai X W) dan nilai Y adalah nilai aktivasi. Didapatkan data sebagai berikut :

Tabel 3. 10. Nilai Aktivasi Data 1 (Sepatu Futsal)

No	X	Y
1	0,644	0,644
2	9,038	9,038
3	17,554	17,554
4	14,955	14,955
5	9,76	9,76

Tabel 3. 11. Nilai Aktivasi Data 2 (Sepatu Badminton)

No	X	Y
1	0,426	0,426
2	1,701	1,701
3	0,892	0,892
4	2,76	2,76
5	2,106	2,106

3.4.2.3 Simpan Data

Simpan data digunakan untuk menyimpan semua data-data yang telah dilatih pada proses pelatihan, yang nanti digunakan untuk proses pengklasifikasian citra selanjutnya.

3.4.3 Analisis Tahapan Pengujian

Tahapan pengujian adalah tahapan yang terakhir yaitu untuk menguji ketika gambar atau citra dimasukan untuk dikenali hasil klasifikasinya baik citra yang sudah dilatih ataupun citra yang belum dilatih. Langkah-langkah pada tahap pengujian adalah sebagai berikut :

3.4.3.1 Citra Uji

Citra uji ini adalah masukan citra yang akan akan diuji dan akan diklasifikasi. Sebagai contoh akan dimasukan citra uji sebagai berikut :



Gambar 3. 12. Citra Uji Yang Telah Dilatih



Gambar 3. 13. Citra Uji Yang Belum Dilatih

3.4.3.2 Hasil Klasifikasi

Hasil klasifikasi ini akan memproses citra uji dengan cara yang sama dari proses pengolahan citra dan pelatihan hopfield hasil yang akan didapatkan adalah nilai ekstraksi kookurensi dan data latih yang nantinya akan dibandingkan dengan citra uji. Langkah-langkah hasil klasifikasi adalah sebagai berikut :

a. Menghitung nilai ekstraksi ciri

   Untuk citra yang telah diuji menghasilkan nilai ekstraksi ciri

Tabel 3. 12. Nilai Ekstraksi Citra Yang Telah Diuji

Kontras	Homogenitas	Entropi	Energi	Korelasi
22,508	0,041	0,78	0,055	0,026

Untuk citra yang belum diuji menghasilkan nilai ekstrasi ciri

Tabel 3. 13. Nilai Ekstraksi Citra Yang Belum Diuji

Kontras	Homogenitas	Entropi	Energi	Korelasi
59,25	0,611	0,78	1,25	0,029

b. Mengalikan dengan bobot data yang telah dilatih

Untuk citra yang telah diuji menghasilkan nilai aktivasi

Tabel 3. 14 Citra X Bobot 1

	Nilai	w1	w2	w3	w4	w5	Nilai x W
Kontras	22,508	0	0,382	0,777	0,657	0,404	0,644
Homogenitas	0,041	0,616	0	0,754	0,913	0,594	9,038
Entropi	0,78	0,734	0,523	0	0,143	0,819	17,554
Energi	0,055	0,741	0,232	0,453	0	0,067	14,955
Korelasi	0,026	0,2	0,725	0,354	0,681	0	9,76

Tabel 3. 15. Citra X Bobot 2

	Nilai	w1	w2	w3	w4	w5	Nilai x W
Kontras	22,51	0	0,74	0,32	0,68	0,55	0,12677
Homogenitas	0,041	0,73	0	0,3	0,33	0,34	16,737174
Entropi	0,78	0,08	0,11	0	0,84	0,57	7,320605
Energi	0,055	0,33	0,07	0,67	0	0,16	16,737174
Korelasi	0,026	0,69	0,46	0,92	0,8	0	12,890429

c. Membandingkan dengan data latih

Tahap ini akan membandingkan dengan latih untuk didapatkan data terdekat.

Rumus $\sqrt{(a-c)^2+(b-d)^2}$ (2.20)

Citra uji dengan perkalian bobot 1 menghasilkan hasil aktivasi

0,644 ; 9,038 ; 17,554 ; 14,955 ; 9,76

Pembandingan dengan aktivasi 1 :

$$= \sqrt{\begin{matrix}(0{,}644 - 0{,}644)^2 + (9{,}038 - 9{,}038)^2(17{,}554\ - 17{,}554\ )^2 \\ +(14{,}955 - 14{,}955)^2(9{,}76 - 9{,}76)^2\end{matrix}}$$

$= \sqrt{0^2 + 0^2 + 0^2 + 0^2 + 0^2}$

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

= 0

Citra uji dengan perkalian bobot 2 maenghasilkan hasil aktivasi 0,128 ; 16,738 ; 7,32 ; 16,738 ; 12,89 ;
Pembandingan dengan aktivasi 2 :

$$= \sqrt{\begin{matrix}(0{,}426 - 0{,}128)^2 + (1{,}701 - 16{,}738)^2(0{,}892\ - 7{,}32\ )^2 \\ +(2{,}76 - 16{,}738)^2(2{,}106 - 12{,}89)^2\end{matrix}}$$

$= \sqrt{0{,}426^2 + -15.037^2 + -6{,}428^2 + -13{,}738^2 + -10{,}13^2}$

$= \sqrt{0{,}18 + 266.11 + 41{,}32 + 188{,}73 + 102{,}67}$

$= \sqrt{599.01}$

= 24, 48

Citra uji yang dimasukan memiliki jarak terdekat dengan aktivasi ke 1, maka citra uji tersebut akan di klasifikasikan sebagai sepatu futsal.

3.5 Analisis Kebutuhan Perangkat Lunak

Kebutuhan Perangkat lunak dalam Implementasi Metode Kookurensi Dan Jaringan Saraf Tiruan Hopfiled Untuk Klasifikasi Sepatu Berdasarkan Tekstur Alas Sepatu ini meliputi beberapa kebutuhan, diantranya :

3.5.1 Analisis Kebutuhan Non Fungsional

Kebutuhan non fungsional yang dibutuhkan untuk implementasi kedalam sistem mencakup dua hal yakni kebutuhan perangkat keras dan perangkat lunak.

a. Kebutuhan Perangkat Keras

   Spesifikasi perangkat keras yang digunakan dalam pembuatan sistem adalah:

1. Processor Intel Core i5 1.7 Ghz
2. Ram 4gb
3. Harddisk berkapasitas 500 gb
4. VGA Nvidia GeForce berkapasitas 2gb

b. Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan dalam pembuatan sistem adalah :

1. Sistem operasi window 8.1
2. Visual studio 2013

3.5.2 Analisis Kebutuhan Fungsional

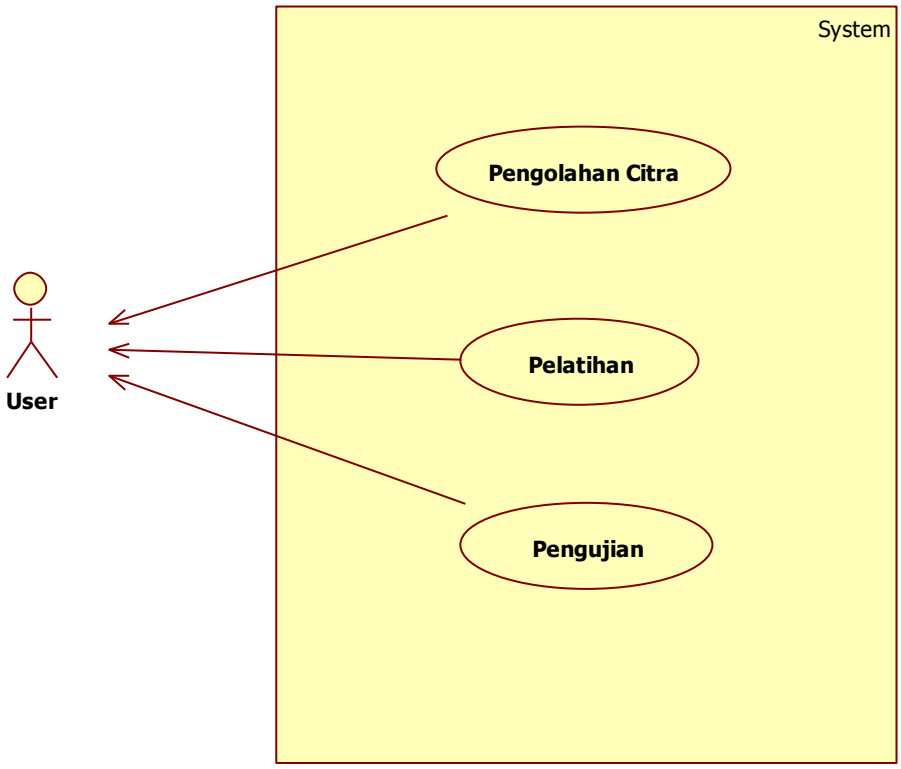
Kebutuhan fungsional adalah tindakan mengidentifikasi proses yang akan berjalan didalam sistem, yang akan dikembangkan dan menjelaskan kebutuhan yang diperlukan. Sehingga sistem dapat berjalan seperti yang diharapkan. Analisis fungsional dimodelkan dengan UML (*Unified Modeling Language*). Tahapan pemodelan dalam analisis tersebut antara lain mengidentifikasi *actor*, diagram *use case*, scenario *use case*, activity diagram, diagram sequence, diagram class dan diagram package.

3.5.2.1 Identifikasi Aktor

Aktor yang teridentifikasi dan terlibat dalam sistem hanya ada satu yakni *user*.

3.5.2.2 Diagram *Use Case*

Use Case adalah interaksi antara sistem dan aktor yang terlibat, didalamnya termasuk pertukaran pesan dan tindakan yang dilakukan oleh sistem.



Gambar 3. 14. Diagram *Use Case*

3.5.2.3 Skenario Diagram

Skenario diagram menjelaskan skenario dari setiap proses yang terdapat pada *use case* diagram. Skenario diagram pada aplikasi ini dijelaskan pada tabel-tabel berikut ini :

Tabel 3. 16. Skenario Diagram Proses Pengolahan Citra

Identifikasi	
Nomor	1
Nama	Pengolahan citra
Tujuan	Menambah dataset untuk citra alas sepatu
Deskripsi	Proses menambah data-data tentang alas sepatu
Aktor	User

Skenario Utama	
Kondisi Awal	Reaksi Sistem
1. User berada pada tampilan tanbah dataset	
2. User memilih tombol input gambar	3. Sistem menampilkan form untuk dapat menampilkan gambar
4. User memilih tombol *grayscale*	5. Sistem menampilkan citra *grayscale* dan matriks
6. User memilih ekstrasi ciri	7. Sistem menampilkan hasil perhitungan ekstraksi ciri
8. User mengisi inputan nama kelas	9. Sistem menampilkan inputan yang dimasukan user
10. User memilih tombol simpan	11. Sistem menyimpan inputan user dan data ekstraksi

Tabel 3. 17. Skenario Diagram Proses Pelatihan

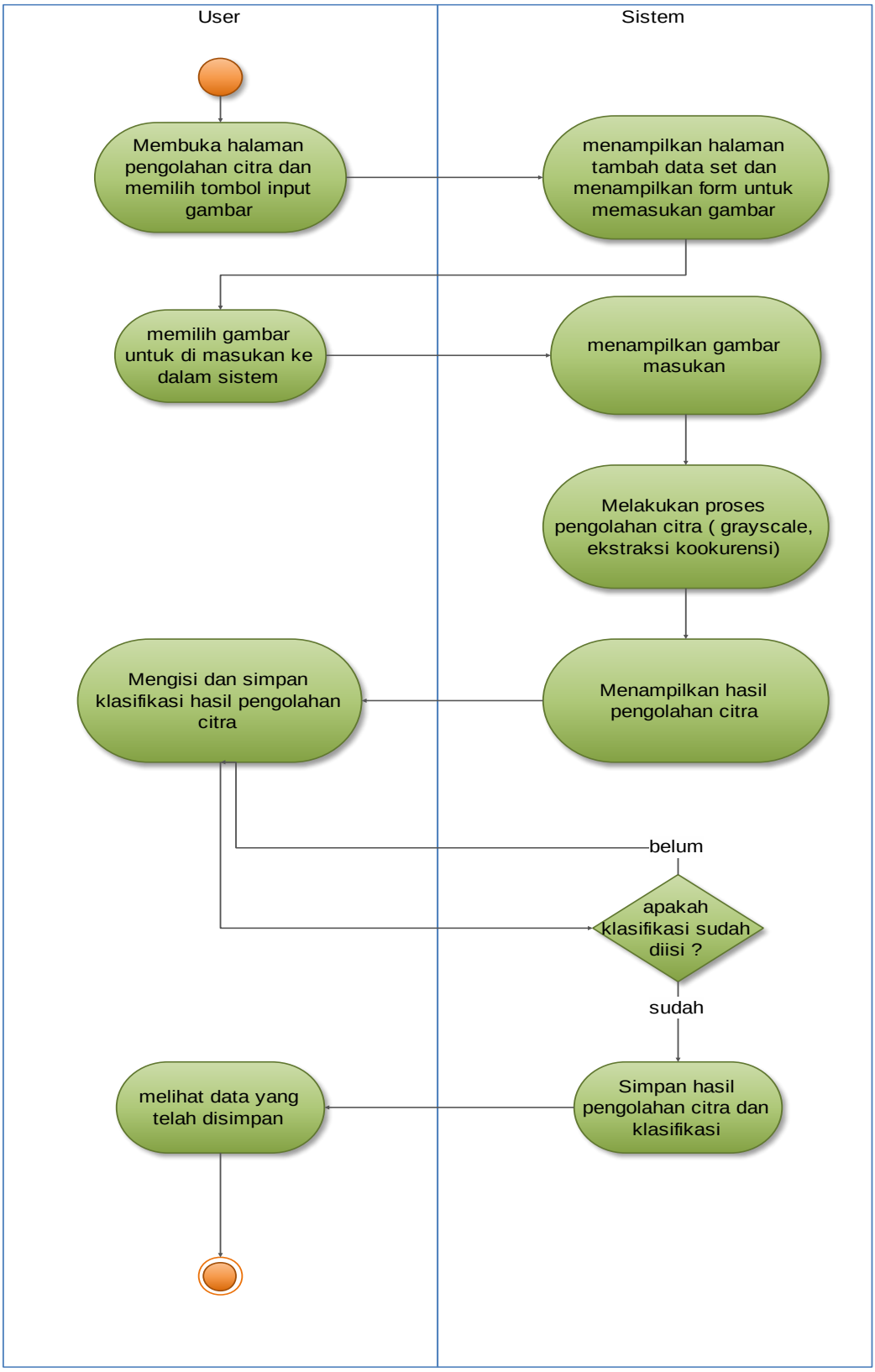
Identifikasi	
Nomor	2
Nama	Pelatihan
Tujuan	Melatih dataset yang ada untuk dapat mengklasifikasikan
Deskripsi	Proses melatih dataset yang ada
Aktor	User
Skenario Utama	
Kondisi Awal	Reaksi Sistem
1. User memilih tombol ambil data	2. Sistem akan menampilkan dataset yang telah disimpan sebelumnya
3. User memilih tombol simpan	4. Sistem akan menyimpan dasaset yang lama akan menambahkan dengan yang baru
5. User memilih tombol latih data	6. Sistem akan melatih data yang ada

Tabel 3. 18. Skenario Diagram Proses Pengujian

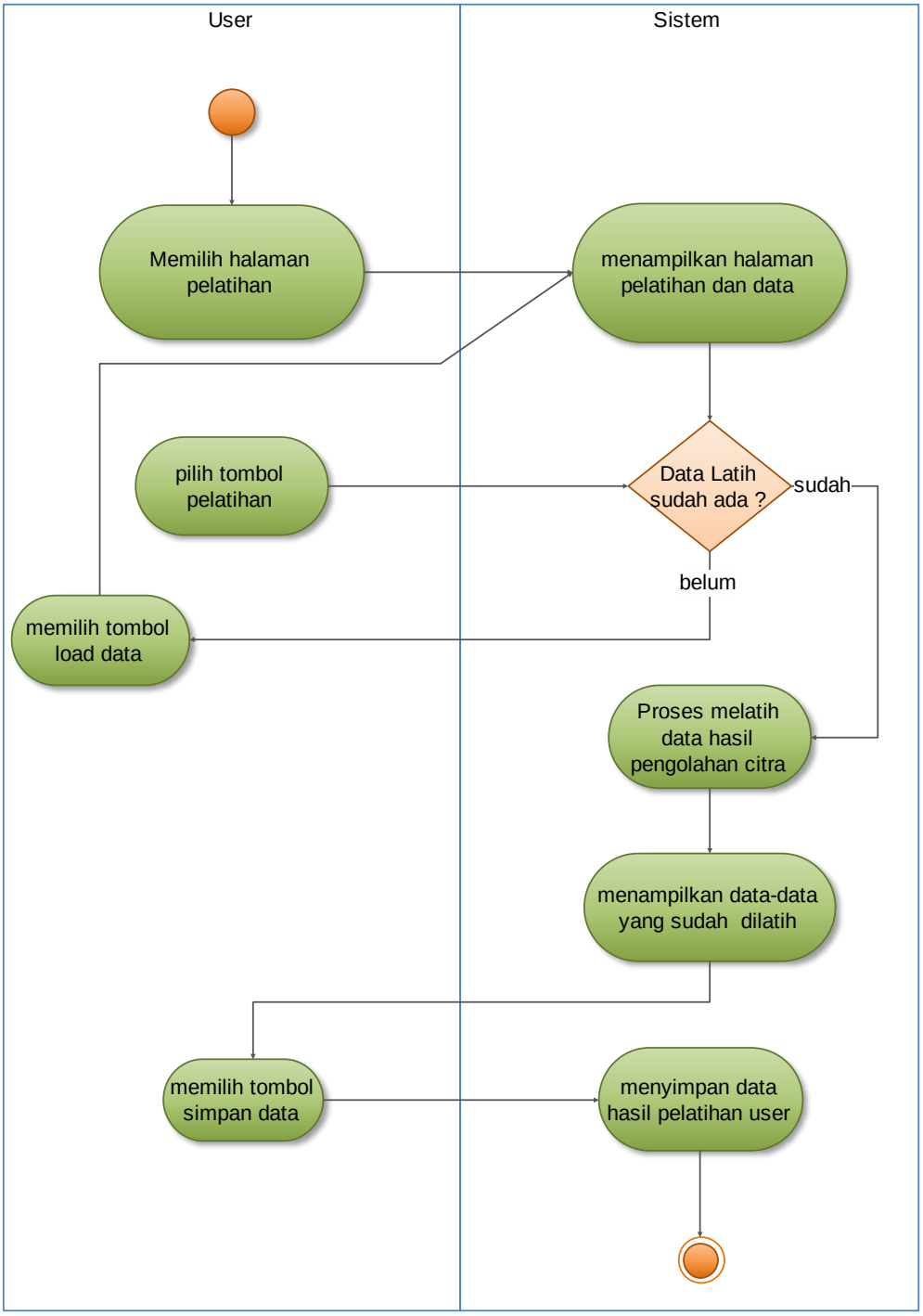
Identifikasi	
Nomor	3
Nama	Pengujian
Tujuan	Menguji dataset yang telah disimpan untuk dapat mengklasifikasikan citra inputan
Deskripsi	Proses menguji sistem untuk dapat mengklasifikasikan citra inputan
Aktor	User
Skenario Utama	
Kondisi Awal	Reaksi Sistem
1. User memilih input gambar	2. Sistem menampilkan form untuk dapat menampilkan gambar
3. User memilih tombol hasil klasifiaksi	4. Sistem akan menampilkan hasil pengolahan citra dan hopfield serta manmpilkan hasil dari klasifikasi
5. User memilih tombol latih data	6. Sistem akan melatih data yang ada

3.5.2.4 *Activity* Diagram

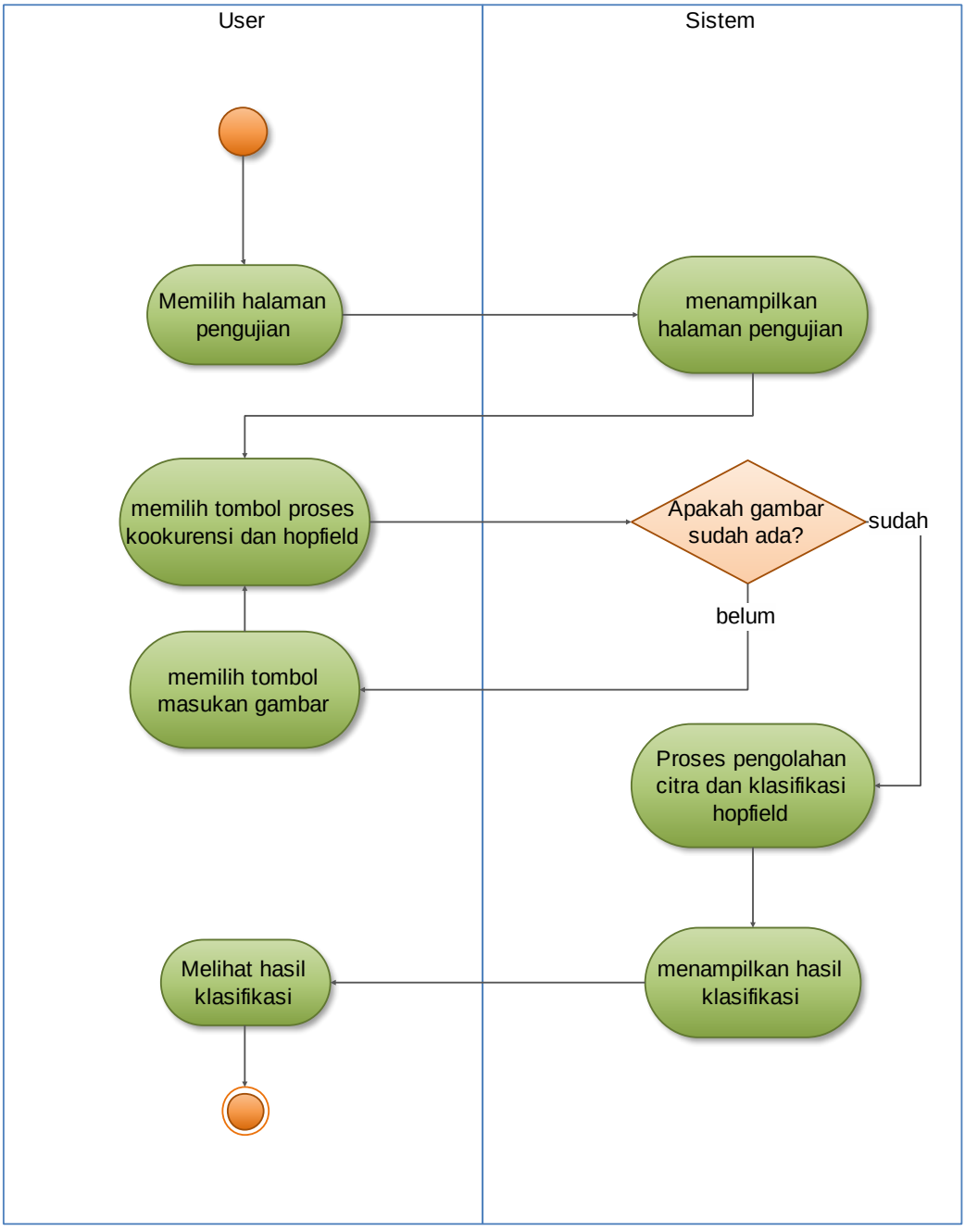
Activity diagram menjelaskan alur aktivitas dari setiap *use case*. Activity diagram dapat dilihat pada gambar dibawah ini.



Gambar 3. 15. Activity Diagram Pengolahan Citra



Gambar 3. 16. Activity Diagram Pelatihan

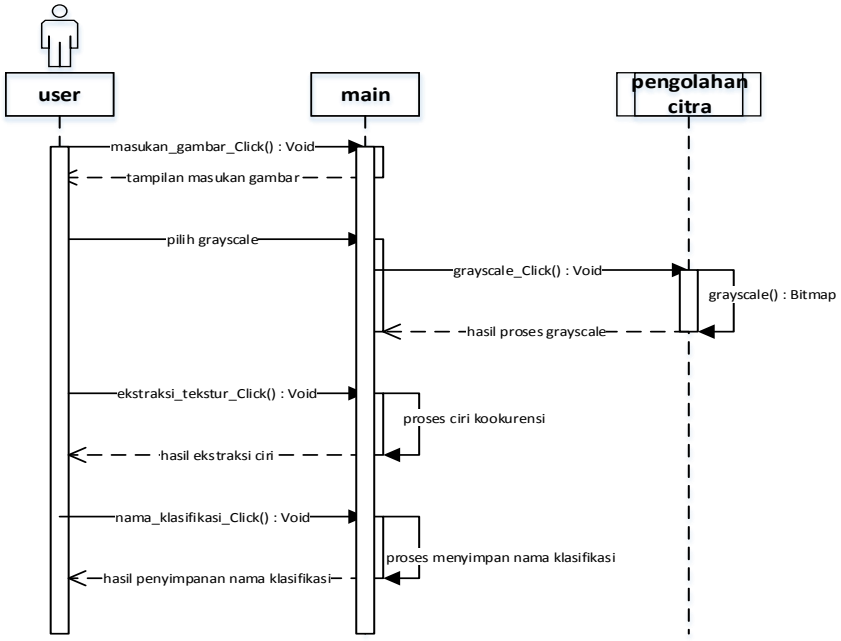


Gambar 3. 17. Activity Diagram Pengujian

3.5.2.5 *Sequence* Diagram

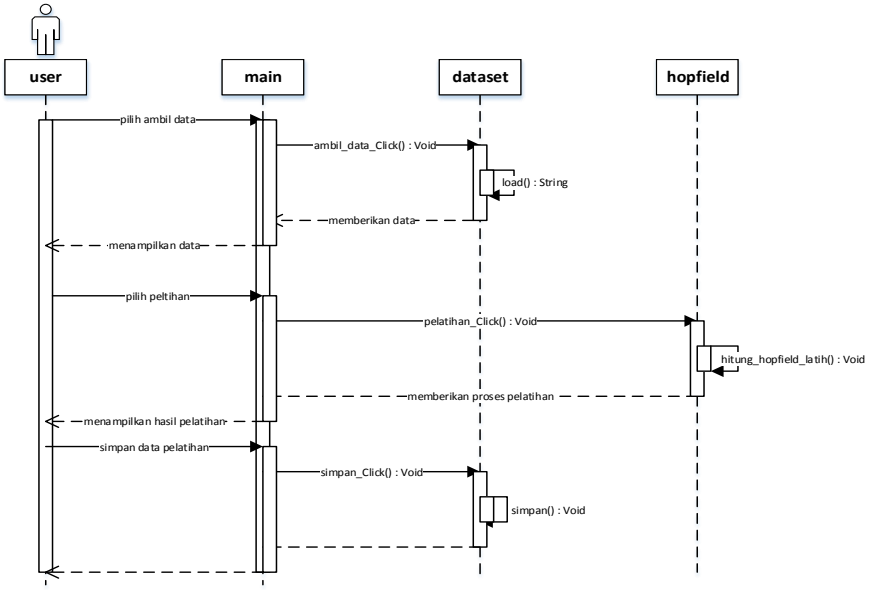
Sequence diagram menjelaskan alur yang akan berjalan dalam sistem ini. Sequence diagram dapat dilihat pada gambar dibawah ini :

1. Diagram sequence pada proses pengolahan citra



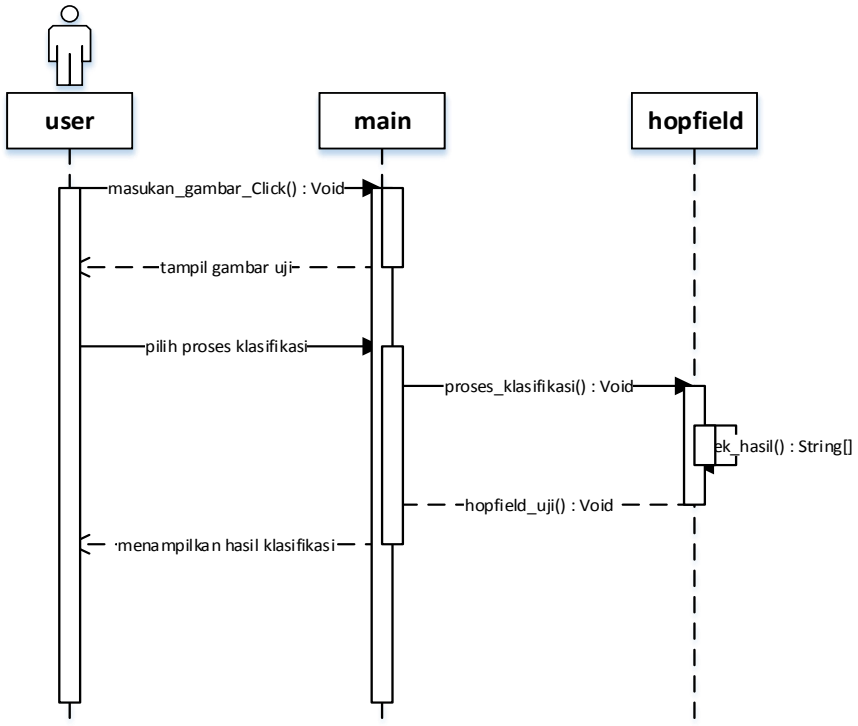
Gambar 3. 18. Sequence Diagram Pengolahan Citra

2. Diagram sequence proses pelatihan



Gambar 3. 19. Sequence Diagram Pelatihan

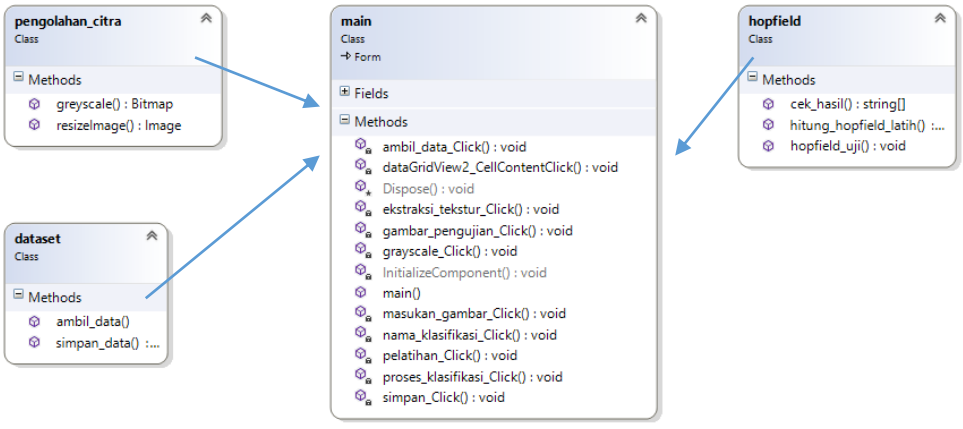
3. Diagram sequence proses Pengujian



Gambar 3. 20. Sequence Diagram Pengujian

3.5.2.6 *Class* Diagram

Class diagram menjelaskan analisi class yang akan digunakan, terutama pada class entity. Class diagram dapat dilihat pada gambar dibawah :



Gambar 3. 21. Class Diagram

3.6 Perancangan Sistem

Perancangan sistem merupakan penggambaran dan perencanaan dari beberapa elemen sistem yang terpisah ke dalam suatu kesatuan model yang utuh. Tahapan ini meliputi perancangan metode, perancangan antarmuka dan jaringan semantik.

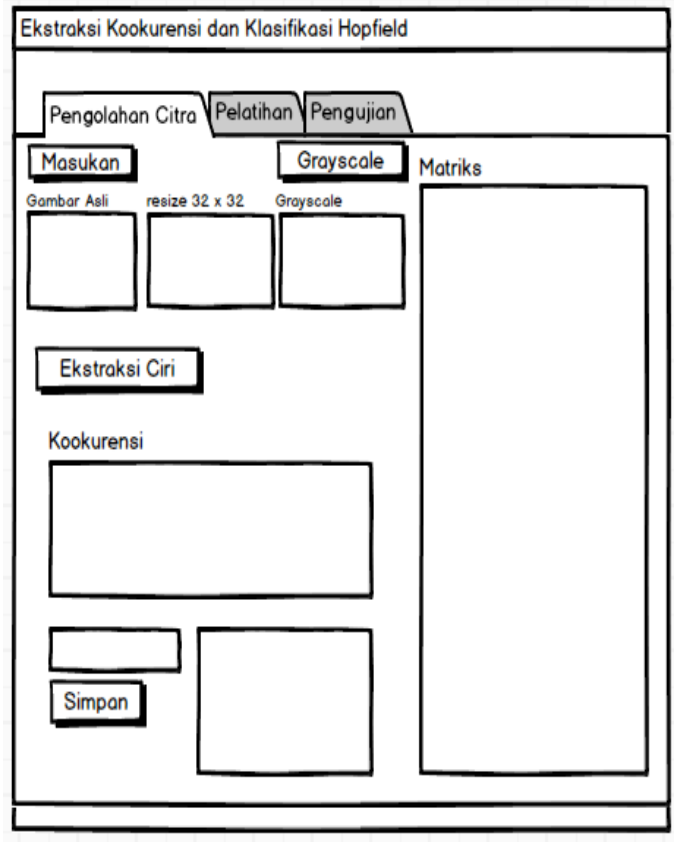
3.6.1 Perancangan Antarmuka

Perancangan Antarmuka mendeskripsikan rencana tampilan dari setiap tampilan yang akan digunakan. Perancangan terdiri dari perancangan form dan pesan.

3.6.1.1 Perancangan *Form*

1. Perancangan antarmuka Tambah Dataset (F01)

No. F01

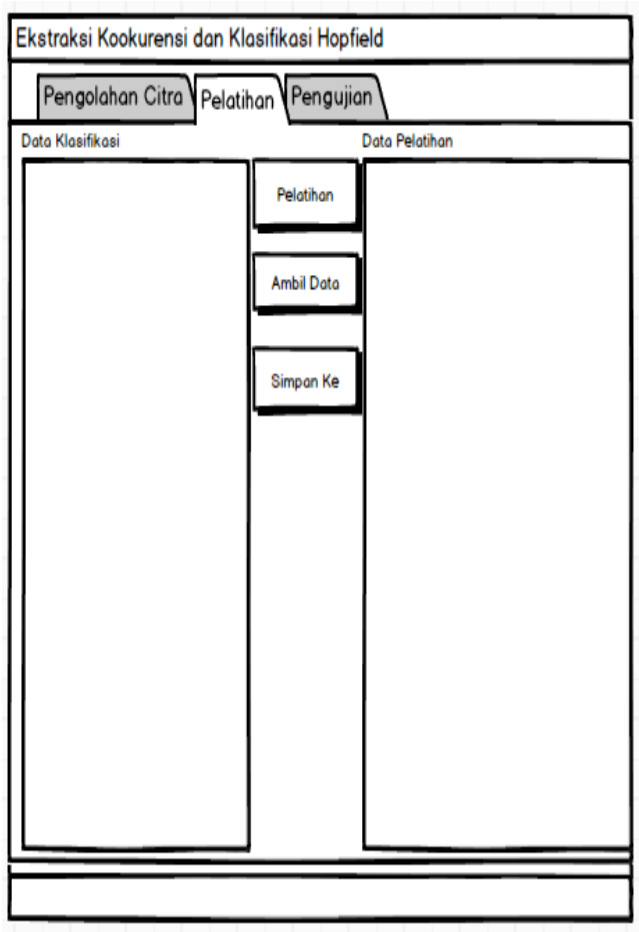


Ukuran 1090 x 591, warna disesuaikan

* Klik tombol **Masukan Gambar** untuk memasukan gambar
* Klik tombol **Proses** untuk mengubah gambar menjadi keabu-abuan dan membuat matriks pada kolom matriks
* Klik tombol **Ekstraksi Ciri** untuk menampilkan hasil ekstraksi yang dikerjakan oleh metode kookurensi, hasil akan ditampilkan pada kolom kookurensi
* Klik tombol **Simpan** untuk memasukan jenis klasifikasi dan menyimpannya kedalam dataset

Gambar 3. 22. Rancangan Form Pengolahan Citra

2. Perancangan antarmuka pelatihan F02

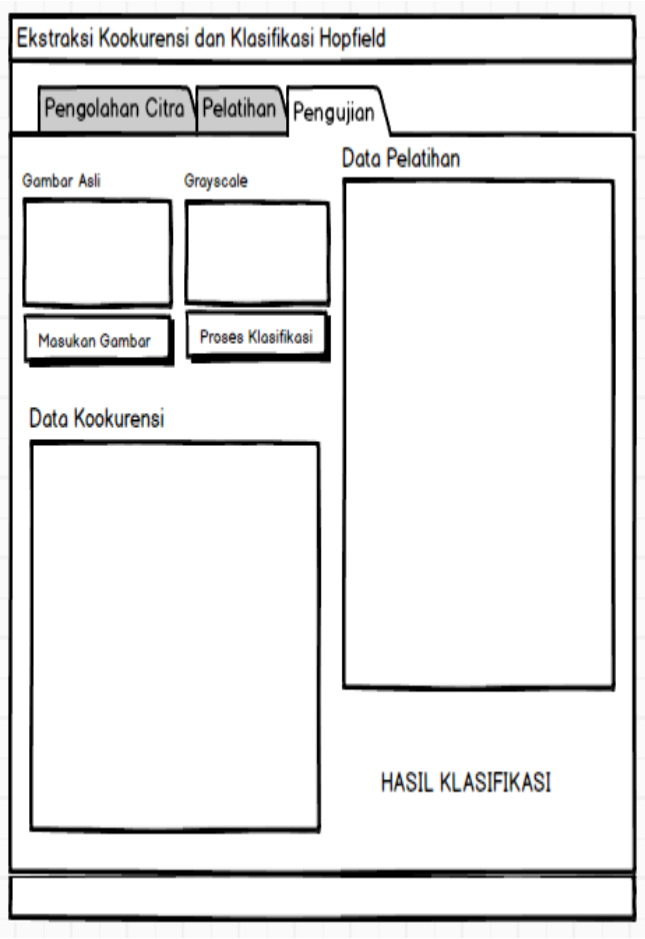
<table>
<tr>
<td>No. F02

</td>
<td>Klik tombol Load untuk menampilkan data yang berasal dari dataset yang sudah disimpan sebelumnya

Klik tombol Pelatihan untuk melatih data yang akan di uji

Klik tombol Simpan untuk menyimpan data yang telah uji</td>
</tr>
<tr>
<td colspan="2">Ukuran 1090 x 591, warna disesuaikan</td>
</tr>
</table>

Gambar 3. 23. Rancangan Form Pelatihan

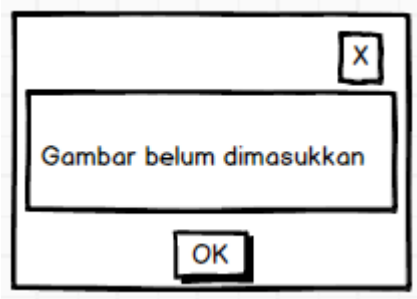
3. Perancangan antarmuka pengujian F03

No. F03	
	Klik tombol **Masukan Gambar** untuk menampilkan gambar Klik tombol **Proses** untuk memproses seluruh data sampai dihasilkan hasil klasifikasi
Ukuran 1090 x 591, warna disesuaikan	

Gambar 3. 24. Rancangan Form Pengujian

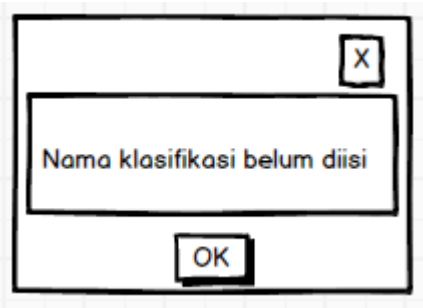
3.6.1.2 Perancangan Pesan

1. Perancangan pesan tombol *grayscale* ketika gambar belum dimasukkan (M01)



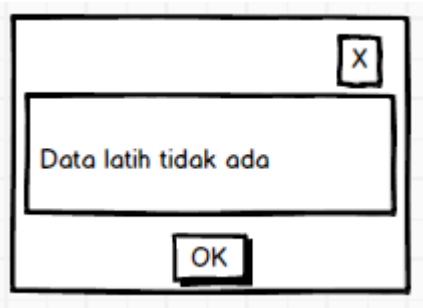
Gambar 3. 25. Perancangan Pesan M01

2. Perancangan pesan tombol tambah data ketika nama klasifikasi belum dimasukkan (M02)



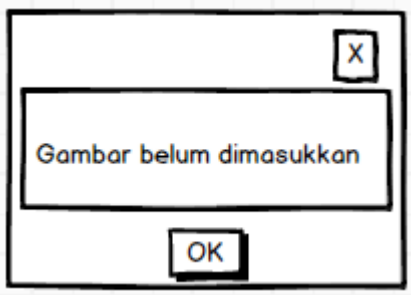
Gambar 3. 26. Perancangan Pesan M02

3. Perancangan pesan tombol pelatihan ketika data latih tidak ada (M03)



Gambar 3. 27. Perancangan Pesan M03

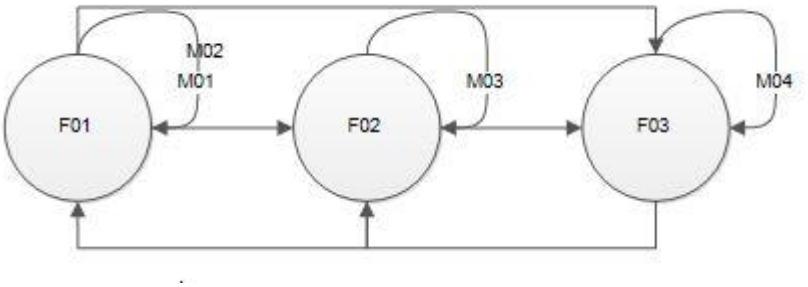
4. Perancangan pesan tombol proses klasifikasi ketika gambar belum dimasukkan (M04)



Gambar 3. 28. Perancangan Pesan M04

3.6.2 Jaringan Semantik

Jaringan semantik merupakan gambaran pengetahuan grafis yang menunjukan hubungan antar berbagai objek. Jaringan semantik terdiri dari lingkaran-lingkaran yang menunjukan objek dan informasi tentang objek-objek tertentu, untuk lebih jelasnya dapat dilihat pada Gambar



Gambar 3. 29. Jaringan Semantik