

**SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN BIBIT UNGGUL
SAPI BALI BERDASARKAN WARNA KULIT MENGGUNAKAN
METODE K-NEAREST NEIGHBOR
(Studi Kasus Balai Pembibitan Ternak Unggul Sapi Bali di Bali)**

SKRIPSI

Untuk memenuhi persyaratan mencapai gelar Sarjana Komputer



Disusun Oleh :

Indra Ekaristio Putra

NIM. 105090600111043

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS BRAWIJAYA
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER**

MALANG

2014

LEMBAR PERSETUJUAN

**SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN BIBIT UNGGUL
SAPI BALI BERDASARKAN WARNA KULIT MENGGUNAKAN
METODE K-NEAREST NEIGHBOR**

(Studi Kasus Balai Pembibitan Ternak Unggul Sapi Bali di Bali)

**SKRIPSI
KONSENTRASI KOMPUTASI CERDAS DAN VISUALISASI**

**Untuk memenuhi sebagian persyaratan untuk
mencapai gelar Sarjana Komputer**



**Disusun Oleh:
INDRA EKARISTIO PUTRA
105090600111043**

**Telah diperiksa dan disetujui oleh
Dosen Pembimbing**

Pembimbing I

Pembimbing II

Arief Andy Soebroto, S.T., M.Kom.

NIP. 19720425 199903 1 002

Ahmad Afif Supianto, S.Si., M.Kom.

NIK. 820623 16 1 1 0425

LEMBAR PENGESAHAN

**SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN BIBIT UNGGUL
SAPI BALI BERDASARKAN WARNA KULIT MENGGUNAKAN
METODE K-NEAREST NEIGHBOR**

(Studi Kasus Balai Pembibitan Ternak Unggul Sapi Bali di Bali)

SKRIPSI

LABORATORIUM KOMPUTASI CERDAS DAN VISUALISASI

Untuk memenuhi sebagian persyaratan untuk mencapai gelar Sarjana Komputer

Disusun oleh:

Indra Ekaristio Putra
105090600111043

Setelah dipertahankan di depan Majelis Penguji
pada tanggal 3 Desember 2014
dan dinyatakan memenuhi syarat untuk memperoleh
gelar sarjana dalam bidang Ilmu Komputer

Penguji I,

Penguji II,

Wayan Firdaus Mahmudy, S.Si., M.T., Ph.D
NIP. 19720919 199702 1 001

Budi Darma Setiawan, S.Kom., M.Cs.
NIK. 841015 06 1 1 0090

Penguji III,

Ir. Heru Nurwasito, M.Kom.
NIP. 19650402 199002 1 001

Mengetahui,
Ketua Program Studi Informatika / Ilmu Komputer

Drs. Mardji, MT.
NIP. 19670801 199203 1 001

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70.

Malang, 2014

Mahasiswa,

Indra Ekaristio Putra
NIM. 105090600111043

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Tuhan yang Maha Esa, yang telah melimpahkan berkat dan anugerah-Nya kepada penulis sehingga penulis dapat menyelesaikan skripsi ini yang berjudul **“SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN BIBIT UNGGUL SAPI BALI BERDASARKAN WARNA KULIT MENGGUNAKAN METODE K-NEAREST NEIGHBOR”**

Pada penyusunan Skripsi ini tidak semata-mata hasil kerja penulis sendiri, melainkan juga berkat dan bimbingan dan dorongan dari pihak-pihak yang telah membantu, baik secara materi maupun non materi. Maka dari itu penulis ingin mengucapkan banyak terima kasih yang tak terhingga serta penghargaan yang setinggi-tingginya kepada orang-orang yang telah membantu penulis secara langsung maupun tidak langsung kepada yang terhormat:

1. Bapak Arief Andy Soebroto, S.T., M.Kom dan Ahmad Afif Supianto, S.Si., M.Kom. selaku dosen pembimbing skripsi yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
2. Bapak Ir. Sutrisno, M.T, Bapak Ir. Heru Nurwasito, M.Kom, Bapak Himawat Aryadita, S.T, M.Sc, dan Bapak Eddy Santoso, S.Kom selaku Ketua, Wakil Ketua 1, Wakil Ketua 2 dan Wakil Ketua 3 Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
3. Orang tua Penulis atas semua bantuan bimbingan dan kesabarannya dalam proses pengerjaan skripsi.
4. Sahabat Penulis serta teman-teman Ilmu Komputer angkatan 2010 yang memotivasi dan saling menyemangati.
5. Segenap Bapak dan Ibu dosen yang telah mendidik dan mengajarkan ilmunya kepada Penulis selama menempuh pendidikan di Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
6. Segenap staff dan karyawan di Program Studi Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya yang telah banyak membantu Penulis dalam pelaksanaan penyusunan skripsi ini.

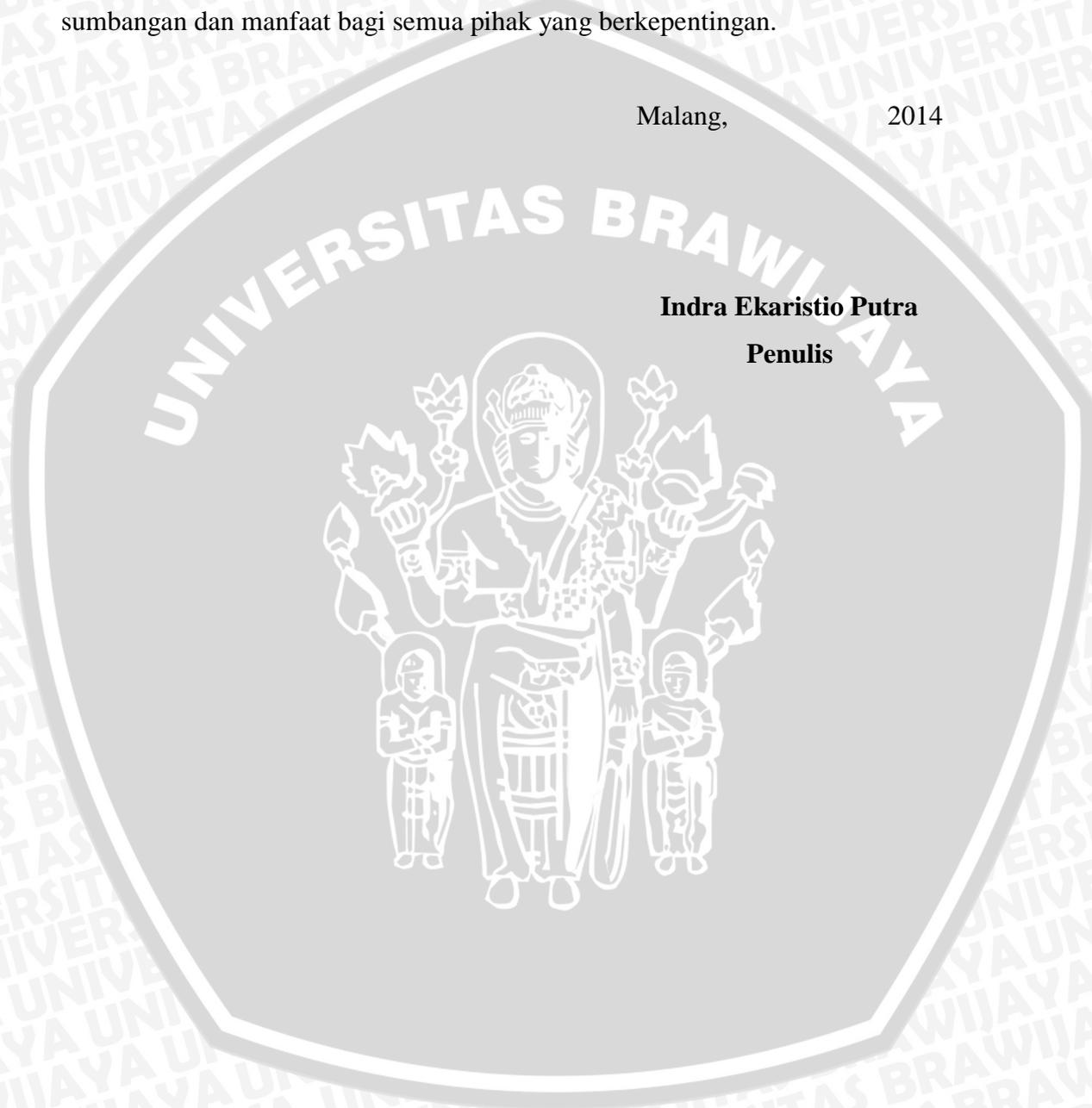
Penulis menyadari bahwa skripsi ini tentunya tidak terlepas dari berbagai kekurangan dan kesalahan. Oleh karena itu, segala kritik dan saran yang bersifat membangun sangat Penulis harapkan dari berbagai pihak demi penyempurnaan penulisan skripsi ini. Penulis berharap agar skripsi ini dapat memberikan sumbangan dan manfaat bagi semua pihak yang berkepentingan.

Malang,

2014

Indra Ekaristio Putra

Penulis



ABSTRAK

Sapi bali merupakan sapi asli Indonesia yang terus dikembangkan dan dijaga kualitasnya. Penilaian terhadap kualitas sapi bali dapat dilihat dari berbagai aspek, salah satunya adalah aspek warna. Kemampuan mata manusia mempunyai keterbatasan dalam melakukan klasifikasi citra warna dari kulit sapi bali. Suatu sistem pendukung keputusan yang mampu mengklasifikasi fitur warna kulit sapi bali dapat membantu mengatasi masalah keterbatasan tersebut. Tujuan dari penelitian ini adalah membuat sistem aplikasi yang dapat membantu peternak yang ada di BPTU Sapi Bali untuk mengklasifikasi citra warna kulit sapi bali. Penelitian ini menggunakan metode K-Nearest Neighbor dalam proses untuk mengklasifikasi fitur warna kulit sapi bali. Fitur yang diekstraksi adalah nilai rata-rata dan nilai standar deviasi warna dasar red, green, blue (RGB). Penelitian ini melakukan pengujian metode untuk memperoleh nilai K terbaik, ukuran citra terbaik, dan jumlah data latih terbaik yang akan digunakan. Sapi bali jantan menggunakan nilai $K=3$, ukuran citra = 128×128 piksel, dan jumlah data latih=45. Sedangkan sapi bali betina menggunakan nilai $K=6$, ukuran citra = 64×64 piksel, dan jumlah data latih=30. Hasil pengujian akurasi sistem untuk sapi bali jantan adalah 100%, sedangkan hasil pengujian akurasi sistem untuk sapi bali betina adalah 66,67%.

Kata Kunci: KNN, RGB, Standar Deviasi, Jarak Euclidean, Bobot, SPK, Sapi Bali

ABSTRACT

Bali cattle is an Indonesian native cattle were constantly bred and maintained quality. Assessment of the quality of Bali cattle can be seen from various aspects, one of is the aspect color. The ability of the human eye has limitations in the classification of skin color image of Bali cattle. A decision support system that is able to classify skin color feature of Bali cattle can help overcome the limitations issue. The purpose of this research is to create a system application that can help farmers in BPTU Sapi Bali to classify skin color image of Bali cattle. This research uses the K-Nearest Neighbor method in the process of classifying skin color feature of Bali cattle. The extracted features are the average value and standard deviation value of the basic colors red, green, blue (RGB). This research tested a method to obtain the best value of K, the best image size, and the amount of training data best that will be used. Male Bali cattle using a value of $K = 3$, image size = 128×128 pixel, and the amount of training data = 45. While the female Bali cattle using a value of $K = 6$, image size = 64×64 pixel, and the amount of training data = 30. The results of testing the accuracy of the system for male Bali cattle is 100%, while the results of testing the accuracy of the system for female Bali cattle is 66.67%.

Keyword : *KNN, RGB, Standart Deviation, Euclidean Distance, Weighted Voting, DSS, Bali Cattle*

DAFTAR ISI

LEMBAR PERSETUJUAN.....	i
LEMBAR PENGESAHAN	ii
PERNYATAAN ORISINALITAS SKRIPSI	iii
KATA PENGANTAR	iv
ABSTRAK.....	vi
<i>ABSTRACT</i>	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xvi
DAFTAR PERSAMAAN.....	xviii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan.....	3
1.5. Manfaat.....	3
1.6. Sistematika Pembahasan	4
BAB II TINJAUAN PUSTAKA.....	6
2.1. Kajian Pustaka.....	6
2.2. Sapi.....	10
2.2.1. Sapi Bali	10
2.2.2. Karakteristik Sapi Bali	11
2.2.3. Keunggulan Sapi Bali	12
2.3. Pengolahan Citra Digital	12

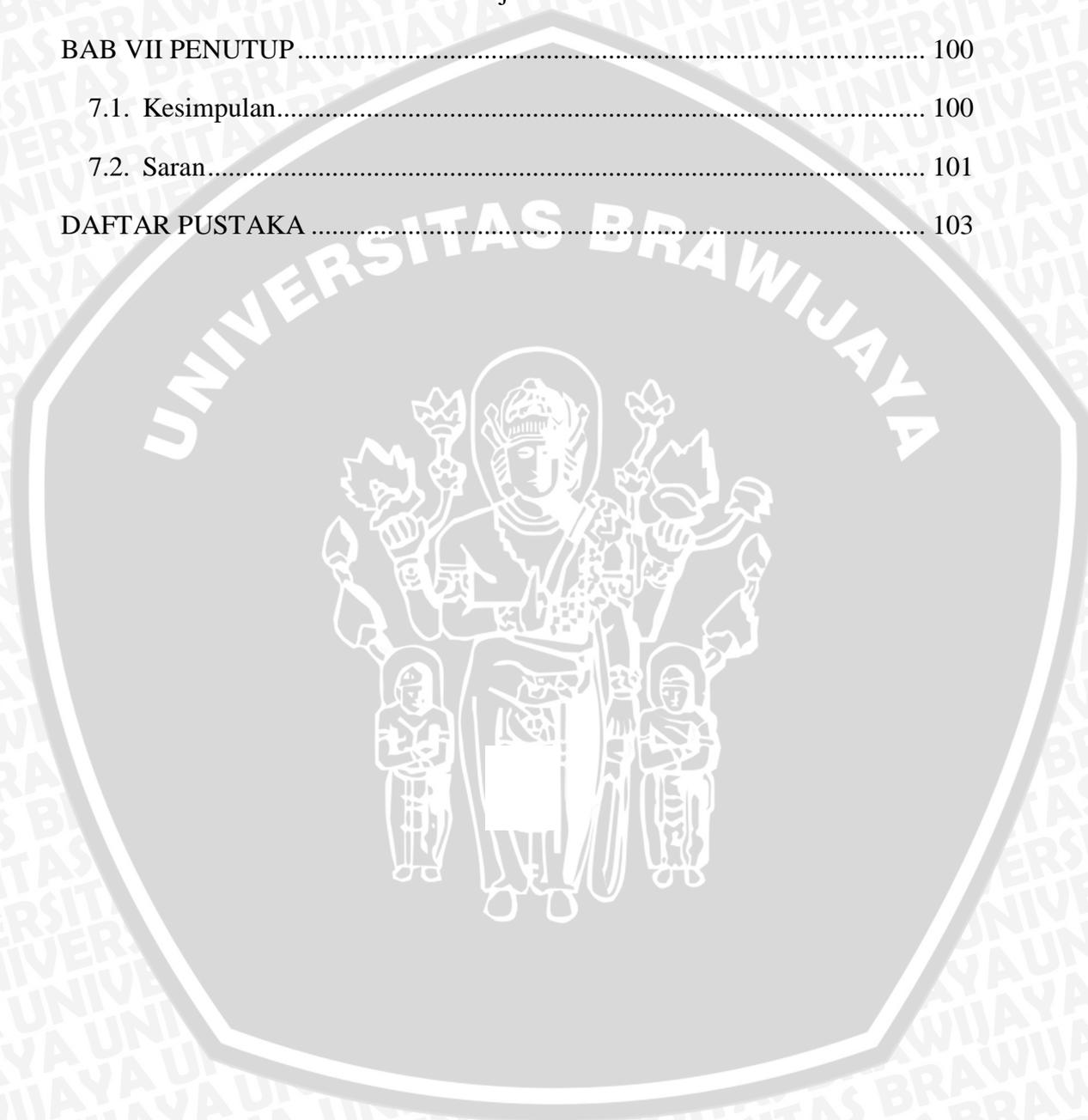


2.4. Sistem Pendukung Keputusan.....	14
2.5. Klasifikasi.....	15
2.6. Nearest Neighbor Classifier	16
2.6.1. Algoritma Nearest Neighbor	17
2.6.2. Klasifikasi dengan Nearest Neighbor.....	18
2.7. Parameter Statistik.....	19
2.8. Perhitungan Akurasi	21
2.9. JAVA.....	21
2.10. Database.....	25
2.10.1. Sistem Manajemen Database	25
2.10.2. MySQL	26
BAB III METODOLOGI.....	28
3.1. Studi Literatur	28
3.2. Pengumpulan Data	29
3.3. Analisa Kebutuhan	29
3.4. Perancangan Sistem.....	30
3.4.1. Blok Diagram Sistem	30
3.4.2. Arsitektur Implementasi Sistem Pendukung Keputusan.....	31
3.4.3. Arsitektur Program.....	31
3.5. Implementasi	32
3.6. Pengujian dan Analisis	33
3.7. Pengambilan Kesimpulan dan Saran.....	34
BAB IV PERANCANGAN	35
4.1. Analisa Kebutuhan Perangkat Lunak	36
4.1.1. Identifikasi Pengguna.....	36
4.1.2. Daftar Kebutuhan Sistem	36

4.2. Perancangan SPK	37
4.2.1. Perancangan Subsistem Manajemen Berbasis Pengetahuan.....	38
4.2.2. Perancangan Subsistem Manajemen Data	40
4.2.2.1. Data Flow Diagram (DFD)	40
4.2.2.2. Entity Relation Diagram (ERD).....	44
4.2.2.3. Physical Diagram	47
4.2.3. Perancangan Subsistem Manajemen Model.....	52
4.2.3.1. Pemodelan Algoritma KNN Weighted	52
4.2.3.2. Pemodelan Olah Data Sapi.....	63
4.2.3.3. Pemodelan Olah Data User	66
4.2.4. Perancangan Subsistem Antarmuka.....	69
4.2.4.1. Antarmuka Login	69
4.2.4.2. Antarmuka Admin.....	70
4.2.4.3. Antarmuka Olah Data User.....	70
4.2.4.4. Antarmuka Olah Data Sapi	71
4.2.4.5. Antarmuka User	72
BAB V IMPLEMENTASI.....	74
5.1. Spesifikasi Sistem	74
5.1.1. Spesifikasi Perangkat Keras.....	75
5.1.2. Spesifikasi Perangkat Lunak.....	75
5.2. Implementasi Algoritma KNN Weighted.....	75
5.2.1. Algoritma Ekstraksi Fitur.....	75
5.2.2. Algoritma Perhitungan Jarak Euclidean.....	77
5.2.3. Algoritma Pengurutan Nilai Euclidean	78
5.2.4. Algoritma Perhitungan <i>Weighted Voting</i>	79
5.3. Implementasi Antarmuka	80
5.3.1. Halaman Login.....	81
5.3.2. Halaman Admin	81

5.3.3. Halaman Tambah Anggota	82
5.3.4. Halaman Ubah Anggota.....	83
5.3.5. Halaman Tambah Sapi	83
5.3.6. Halaman Ubah Sapi.....	84
5.3.7. Halaman Crop Sapi	84
5.3.8. Halaman User.....	85
BAB VI PENGUJIAN DAN ANALISIS	89
6.1. Pengujian Parameter.....	90
6.1.1. Pengujian Nilai K.....	90
6.1.1.1. Skenario Uji Coba 1	90
6.1.1.1.1. Tujuan	90
6.1.1.1.2. Prosedur.....	90
6.1.1.1.3. Hasil	91
6.1.1.2. Analisis Skenario Uji Coba 1	92
6.1.2. Pengujian Ukuran Citra.....	92
6.1.2.1. Skenario Uji Coba 2	92
6.1.2.1.1. Tujuan	93
6.1.2.1.2. Prosedur.....	93
6.1.2.1.3. Hasil	93
6.1.2.2. Analisis Skenario Uji Coba 2	94
6.1.3. Pengujian Jumlah Data Latih	95
6.1.3.1. Skenario Uji Coba 3	95
6.1.3.1.1. Tujuan	95
6.1.3.1.2. Prosedur.....	95
6.1.3.1.3. Hasil	96
6.1.3.2. Analisis Skenario Uji Coba 3	97
6.2. Pengujian Akurasi Antara Pakar dan Sistem.....	97
6.2.1. Skenario Uji Coba 4	97

6.2.1.1. Tujuan	97
6.2.1.2. Prosedur.....	98
6.2.1.3. Hasil	98
6.2.2. Analisis Hasil Skenario Uji Coba 4	99
BAB VII PENUTUP	100
7.1. Kesimpulan.....	100
7.2. Saran.....	101
DAFTAR PUSTAKA	103



DAFTAR GAMBAR

Gambar 2.1 Sapi Bali (a) Jantan dan (b) Betina.....	11
Gambar 2.2 Sistem koordinat citra digital	12
Gambar 2.3 Campuran warna merah, hijau, dan biru menghasilkan warna putih	13
Gambar 2.4 Skema warna kubik RGB	13
Gambar 2.5 Skematik DSS	15
Gambar 2.6 Proses Pekerjaan Klasifikasi	15
Gambar 2.7 KNN dengan nilai K yang besar	17
Gambar 2.8 Skema pengkompilasian hingga pengeksekusian kode Java.....	23
Gambar 2.9 Arsitektur DBMS	26
Gambar 3.1 Diagram Blok Metodologi Penelitian	28
Gambar 3.2 Arsitektur Blok Diagram Perancangan Aplikasi.....	30
Gambar 3.3 Arsitektur Implementasi SPK Pemilihan Bibit Unggul Sapi Bali.....	31
Gambar 3.4 Blok Diagram Program	32
Gambar 4.1 Pohon Perancangan.....	35
Gambar 4.2 Arsitektur SPK Pemilihan Sapi Bali	38
Gambar 4.3 Diagram Konteks.....	40
Gambar 4.4 DFD Level 1	41
Gambar 4.5 DFD Level 2 Proses Olah Data <i>User</i>	42
Gambar 4.6 DFD Level 2 Proses Olah Data Sapi.....	43
Gambar 4.7 DFD Level 2 Proses Pelatihan Data Training	44
Gambar 4.8 ERD SPK Pemilihan Bibit Unggul Sapi Bali.....	46
Gambar 4.9 <i>Physical Diagram Admin</i> Pemilihan Bibit Unggul Sapi Bali	47
Gambar 4.10 Rancangan Algoritma KNN Weighted	53
Gambar 4.11 Pseudocode Algoritma Perhitungan Jarak Euclidean.....	53

Gambar 4.12 Diagram Alir Ekstraksi Fitur	55
Gambar 4.13 Pseudocode Ekstraksi Fitur	56
Gambar 4.14 Diagram Alir Menghitung Jarak Euclidean	56
Gambar 4.15 Pseudocode Algoritma Perhitungan Jarak Euclidean.....	57
Gambar 4.16 Diagram Alir Proses Pengurutan Nilai Jarak Euclidean	59
Gambar 4.17 Pseudocode Algoritma Mengurutkan Nilai Jarak Euclidean	59
Gambar 4.18 Diagram Alir Perhitungan Weighted Voting	61
Gambar 4.19 Pseudocode Algoritma Menghitung Nilai Weighted Voting	62
Gambar 4.20 Diagram Alir Proses Tambah Data Sapi	63
Gambar 4.21 Pseudocode Algoritma Proses Tambah Data Sapi	64
Gambar 4.22 Diagram Alir Proses Ubah Data Sapi.....	65
Gambar 4.23 Pseudocode Algoritma Proses Ubah Data Sapi.....	65
Gambar 4.24 Diagram Alir Proses Hapus Data Sapi	66
Gambar 4.25 Pseudocode Algoritma Proses Hapus Data Sapi	66
Gambar 4.26 Diagram Alir Proses Tambah Data <i>User</i>	67
Gambar 4.27 Pseudocode Algoritma Proses Tambah Data <i>User</i>	67
Gambar 4.28 Diagram Alir Proses Ubah Data <i>User</i>	68
Gambar 4.29 Pseudocode Algoritma Proses Ubah Data <i>User</i>	68
Gambar 4.30 Diagram Alir Proses Hapus Data <i>User</i>	69
Gambar 4.31 Pseudocode Algoritma Proses Hapus Data <i>User</i>	69
Gambar 4.32 Antarmuka Login	70
Gambar 4.33 Antarmuka Admin	70
Gambar 4.34 Antarmuka Olah Data <i>User</i>	71
Gambar 4.35 Antarmuka Olah Data Sapi	71
Gambar 4.36 Antarmuka Crop Gambar Sapi Bali	72
Gambar 4.37 Antarmuka <i>User</i>	72

Gambar 5.1 Pohon Implementasi	74
Gambar 5.2 Antarmuka Halaman Login	81
Gambar 5.3 Antarmuka Halaman Admin Tab Data Anggota	81
Gambar 5.4 Antarmuka Halaman Admin Tab Data Sapi	82
Gambar 5.5 Antarmuka Halaman Tambah Anggota	82
Gambar 5.6 Antarmuka Halaman Ubah Anggota	83
Gambar 5.7 Antarmuka Halaman Tambah Sapi	83
Gambar 5.8 Antarmuka Halaman Ubah Sapi	84
Gambar 5.9 Antarmuka Halaman <i>Crop</i> Sapi	85
Gambar 5.10 Antarmuka Halaman User Tab Data Uji	85
Gambar 5.11 Antarmuka Halaman User Tab Pengujian	86
Gambar 5.12 Antarmuka Halaman User Tab Klasifikasi	87
Gambar 5.13 Antarmuka Halaman User Tab Skenario Rata-Rata Sapi Bali Jantan	87
Gambar 5.14 Antarmuka Halaman User Tab Skenario Rata-Rata Sapi Bali Betina	88
Gambar 6.2 Grafik Perbandingan Akurasi Pengujian Nilai K	92
Gambar 6.3 Grafik Perbandingan Akurasi Ukuran Pixel	94
Gambar 6.4 Posisi <i>Cropping</i>	95
Gambar 6.4 Grafik Perbandingan Hasil Pengujian Jumlah Data Latih Antara Sapi Bali Jantan dan Sapi Bali Betina	97

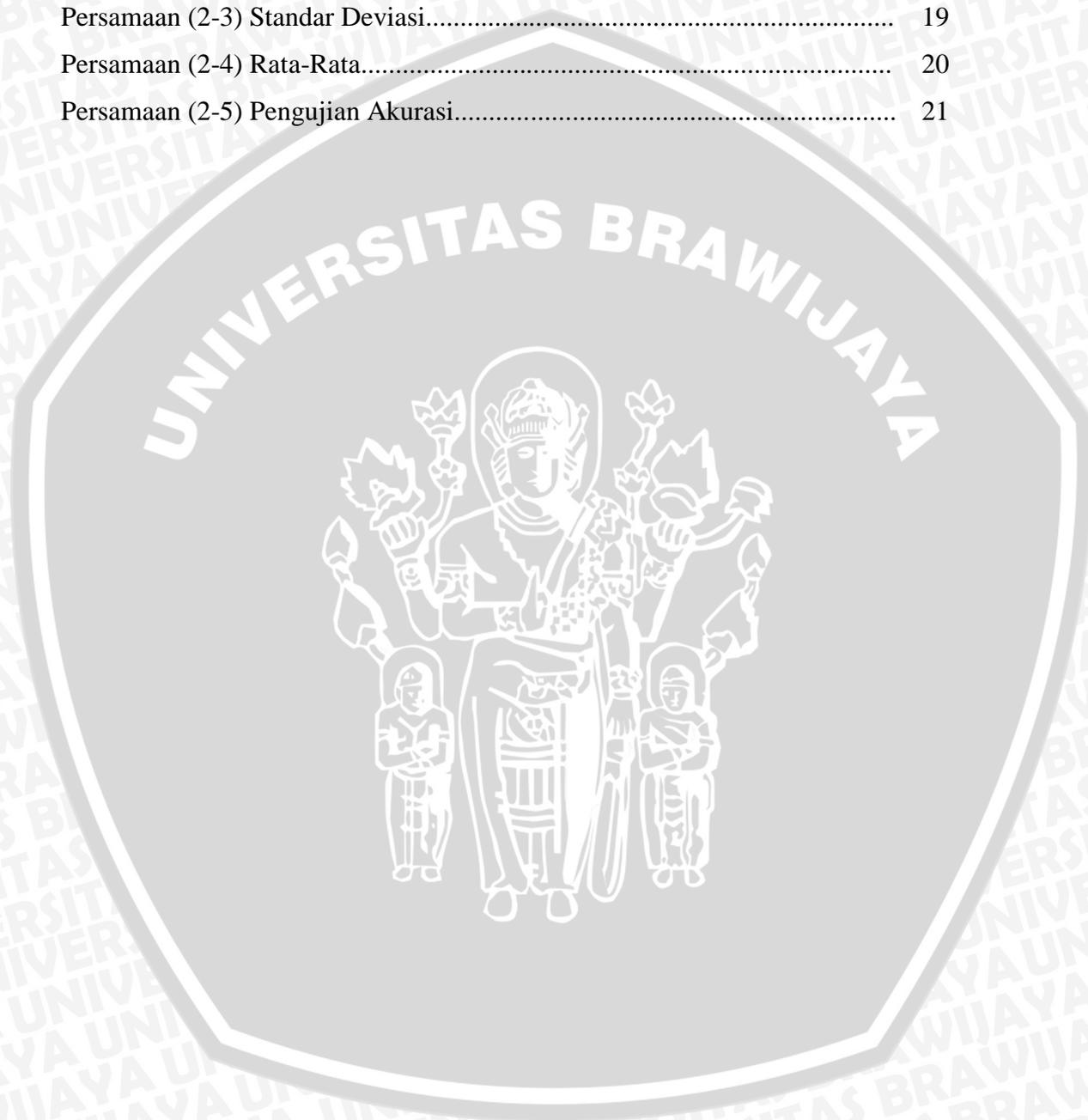
DAFTAR TABEL

Tabel 2.1 Kajian Pustaka.....	7
Tabel 2.2 Bagan kedudukan sapi Bali dalam Subfamili Bovinae.....	10
Tabel 2.3 Matriks Konfusi untuk Klasifikasi Dua Kelas	16
Tabel 2.4 Contoh Nilai RGB Potongan Gambar 5 × 5 pixel.....	20
Tabel 2.4 Java pada berbagai system operasi.....	22
Tabel 3.1 Tabel Model Data.....	31
Tabel 4.1 Identifikasi Pengguna.....	36
Tabel 4.2 Kebutuhan Fungsional Sistem	37
Tabel 4.3 Kriteria Fitur Warna Kulit Sapi Bali.....	39
Tabel 4.4 Struktur tabel user.....	48
Tabel 4.5 Struktur tabel data Sapi Bali	48
Tabel 4.6 Struktur tabel crop Sapi Bali.....	48
Tabel 4.7 Struktur tabel crop Sapi Bali 2.....	49
Tabel 4.8 Struktur tabel crop Sapi Bali 3.....	49
Tabel 4.9 Struktur tabel user.....	50
Tabel 4.10 Struktur tabel uji crop Sapi Bali.....	50
Tabel 4.11 Struktur tabel uji crop Sapi Bali 2.....	51
Tabel 4.12 Struktur tabel uji crop Sapi Bali 3.....	51
Tabel 4.13 Tabel Data Latih Sapi Bali Jantan.....	54
Tabel 4.14 Tabel Data Uji.....	54
Tabel 4.15 Hasil Perhitungan Euclidean Distance	57
Tabel 4.16 Hasil Pengurutan Nilai Jarak Euclidean.....	60
Tabel 4.17 Data Latih Yang Terdekat.....	62
Tabel 5.1 Spesifikasi Perangkat Keras.....	75

Tabel 5.2 Spesifikasi Perangkat Lunak	75
Tabel 5.6 Implementasi Algoritma Ekstraksi Fitur	76
Tabel 5.3 Implementasi Algoritma Perhitungan Jarak Euclidean.....	77
Tabel 5.4 Implementasi Algoritma Pengurutan Nilai Euclidean	78
Tabel 5.5 Implementasi Algoritma Perhitungan Weighted Voting	79
Tabel 6.1 Akurasi Pengujian Nilai K Sapi Bali Jantan	91
Tabel 6.2 Akurasi Pengujian Nilai K Sapi Bali Betina	91
Tabel 6.3 Akurasi Pengujian Ukuran Pixel Terhadap Nilai K Terbaik Sapi Bali Jantan.....	93
Tabel 6.4 Akurasi Pengujian Ukuran Pixel Terhadap Nilai K Terbaik Sapi Bali Betina	93
Tabel 6.5 Akurasi Pengujian Jumlah Data Latih Terhadap Nilai K dan Ukuran Pixel Terbaik Sapi Bali Jantan	96
Tabel 6.6 Hasil Akurasi Pengujian Jumlah Data Latih Terhadap Nilai K Dan Ukuran Pixel Terbaik Sapi Bali Betina.....	96
Tabel 6.7 Hasil Pengujian Akurasi Antara Pakar dan Sistem untuk Sapi Bali Jantan.....	98
Tabel 6.8 Hasil Pengujian Akurasi Antara Pakar dan Sistem untuk Sapi Bali Betina	98

DAFTAR PERSAMAAN

Persamaan (2-1) Jarak Euclidean.....	17
Persamaan (2-2) Weighted Voting.....	18
Persamaan (2-3) Standar Deviasi.....	19
Persamaan (2-4) Rata-Rata.....	20
Persamaan (2-5) Pengujian Akurasi.....	21



BAB I PENDAHULUAN

1.1. Latar Belakang

Peternakan di Indonesia terus melakukan upaya dalam pengembangbiakan ternaknya. Pengupayaan tersebut didasari pada semakin lemahnya produktifitas dan perkembangan ternak sapi di Indonesia [12:74-75]. Salah satu ternak yang sekarang diupayakan adalah sapi Bali [6:28]. Pembudidayaan sapi Bali di Balai Pembibitan Ternak Unggul (BPTU) Bali terus dilakukan untuk meningkatkan mutu genetik bibit ternak sapi Bali. Penilaian terhadap kualitas mutu sapi dapat dilihat dari berbagai aspek, salah satunya dari aspek warna. Penilaian tersebut sangat sulit dilakukan secara langsung oleh peternak mengingat terbatasnya kemampuan mata dalam mengklasifikasikan kualitas sapi Bali. Keterbatasan kemampuan mata untuk melakukan klasifikasi terhadap citra warna sapi Bali dapat dibantu dengan aplikasi sistem pendukung keputusan (SPK). SPK tersebut melakukan klasifikasi dengan menggunakan fitur dari citra sapi Bali yaitu, nilai warna red, green, dan blue (RGB). Hasil dari proses klasifikasi pola warna tersebut akan dibagi menjadi tiga kelas, Baik (Ternak Unggul), Sedang, dan Buruk (Penyimpangan).

Ada tiga penelitian yang membahas mengenai pola warna pada citra digital dan metode yang digunakan. Penelitian yang pertama oleh Ratri Andaruresmi adalah “Identifikasi Tipe Wilayah Berbasis Pengolahan Citra Penginderaan Jarak Jauh”. Penelitian ini menggunakan metode KNN dalam proses pengidentifikasiannya dan menyimpulkan bahwa akurasi penggunaan metode jarak euclidean lebih baik daripada metode jarak cityblock. Penelitian yang kedua oleh Retno Nugroho Whidhiasih adalah “Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan KNN dan LDA”. Penelitian ini melakukan perbandingan penggunaan metode KNN dan LDA serta perbandingan penggunaan variabel R-G dan R-G-B. Kesimpulan yang dihasilkan adalah metode KNN dan LDA sama baiknya, akan tetapi lebih baik menggunakan variabel R-G-B daripada R-G. Penelitian yang ketiga oleh Mutihera Safitri adalah “Deteksi Hutan Mangrove di Pantai Utara Jakarta Menggunakan Citra dari *Google Earth* dengan

Metode Curvelet”. Penelitian ini menggunakan metode KNN untuk klasifikasinya dan menggunakan metode curvelet untuk ekstraksi ciri. Proses ekstraksi ciri dibantu dengan metode statistik dengan menggunakan nilai statistik mean dan standar deviasi, serta skewness. Kesimpulan dari penelitian ini adalah untuk meningkatkan performansi sistem dapat digunakan nilai statistik mean dan standar deviasi. Dari ketiga penelitian tersebut, penulis menyimpulkan bahwa metode yang akan digunakan dalam proses pengklasifikasian penelitian ini adalah metode KNN. Penentuan metode dari algoritma KNN adalah menggunakan perhitungan jarak Euclidean dan menggunakan variabel *mean* dan standar deviasi komponen warna RGB dari citra warna kulit sapi Bali.

Berdasarkan permasalahan yang telah dijelaskan diatas dan melihat beberapa penelitian sebelumnya, maka judul yang diusulkan dalam penelitian ini adalah “Sistem Pendukung Keputusan Pemilihan Bibit Unggul Sapi Bali Berdasarkan Warna Kulit Menggunakan Metode K-Nearest Neighbor”.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan di atas, maka dirumuskan masalah sebagai berikut:

1. Bagaimana merancang sistem pendukung keputusan untuk klasifikasi sapi Bali berdasarkan fitur warna kulit menggunakan metode *K-Nearest Neighbor*.
2. Bagaimana implementasi metode *K-Nearest Neighbor* untuk pendukung keputusan pengklasifikasian sapi Bali berdasarkan fitur warna kulit.
3. Bagaimana menguji tingkat akurasi dari implementasi metode *K-Nearest Neighbor*.

1.3. Batasan Masalah

Berdasarkan latar belakang dan rumusan masalah yang telah dikemukakan, penelitian ini mempunyai batasan-batasan masalah sebagai berikut:

1. Data input yang digunakan berbentuk gambar (citra) yang diambil menggunakan kamera *Single Lens Reflex* (SLR) merk Canon.
2. Data input berformat (*.jpg) dengan 3 ukuran yaitu 32×32 pixels, 64×64 pixels, 128×128 pixels.

3. Pencahayaan dari pengambilan gambar konstan.
4. Pengambilan gambar sapi dilakukan di habitat sapi Bali, tepatnya di kawasan milik Balai Pembibitan Ternak Unggul Sapi Bali (BPTU Sapi Bali) yang berada di Jembrana, Bali.
5. Warna yang menjadi fokus utama dalam analisis adalah warna tubuh dari bagian samping sapi Bali.
6. Umur pada sapi Bali yang akan dijadikan objek penelitian yaitu untuk Sapi Bali Jantan dan Sapi Bali Betina berumur 2 tahun ke atas.
7. Pengolahan data untuk pendukung keputusan kualitas sapi Bali menggunakan metode *k-nearest neighbor (KNN)* dengan warna kulit sapi sebagai fitur utama.
8. Parameter yang digunakan dibatasi pada nilai indeks warna *mean* dan standar deviasi komponen warna RGB.
9. Pada pengujian ini nilai *k* yang digunakan *random*.
10. Aplikasi yang dihasilkan berbasis *desktop* dibangun menggunakan bahasa pemrograman JAVA.
11. Database menggunakan MySQL.
12. Sasaran pada penelitian ini ditunjukkan pada hewan ternak.
13. Hasil klasifikasi atau output yang diberikan ada tiga yaitu baik, sedang, atau buruk.

1.4. Tujuan

Penelitian ini bertujuan untuk membangun suatu sistem pendukung keputusan pengklasifikasian kualitas sapi Bali berdasarkan fitur warna kulit dengan menggunakan metode *K-Nearest Neighbor*, sekaligus untuk menguji sistem ini.

1.5. Manfaat

Penulisan tugas akhir ini diharapkan mempunyai manfaat yang baik. Adapun manfaat yang diharapkan adalah sebagai berikut:

1. Memberikan hasil yang optimal dan akurat dalam klasifikasi sapi Bali.
2. Memberikan kontribusi dalam penerapan sistem pendukung keputusan untuk klasifikasi sapi Bali.

3. Membantu para peternak agar lebih mudah dalam proses pengklasifikasi hewan ternak untuk jumlah yang banyak.

1.6. Sistematika Pembahasan

Sistematika penulisan penelitian ditunjukkan untuk memberikan gambaran dan uraian dari penyusunan tugas akhir secara garis besar yang meliputi beberapa bab, sebagai berikut:

BAB I : PENDAHULUAN

Menguraikan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika pembahasan

BAB II : TINJAUAN PUSTAKA

Menguraikan tentang dasar teori dan referensi yang menunjang dalam penelitian ini. Teori yang terdapat pada bab ini antara lain mengenai sapi Bali, pengolahan citra digital, metode klasifikasi menggunakan *K-Nearest Neighbor*.

BAB III : METODOLOGI PENELITIAN

Menguraikan tentang langkah-langkah yang dilakukan dalam menyelesaikan penelitian skripsi yang terdiri dari studi literatur, analisa kebutuhan, metode pengambilan data, perancangan, implementasi, pengujian dan analisis serta pengambilan kesimpulan.

BAB IV : PERANCANGAN

Membahas mengenai analisis kebutuhan dan perancangan sistem klasifikasi kualitas sapi Bali dengan metode *K-Nearest Neighbor*

BAB V : IMPLEMENTASI

Membahas mengenai segala hal yang berkaitan dengan implementasi sistem perangkat lunak yang digunakan untuk klasifikasi sapi Bali. Meliputi implementasi *sourcecode* dan antar muka sistem.

BAB VI : PENGUJIAN DAN ANALISIS

Memuat penjelasan proses pengujian dan hasil pengujian dari sistem pendukung keputusan klasifikasi kualitas sapi Bali, menggunakan metode *K-Nearest Neighbor*, serta analisis dari pengujian tersebut.

BAB VII : PENUTUP

Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian perangkat lunak yang dikembangkan dalam skripsi ini serta saran-saran untuk pengembangan lebih lanjut.



BAB II

TINJAUAN PUSTAKA

Bab ini berisi kajian pustaka dan pembahasan mengenai teori dasar yang berhubungan dengan sistem pendukung keputusan pemilihan bibit unggul sapi bali berdasarkan fitur warna kulit dengan menggunakan metode *K-Nearest Neighbor*. Kajian pustaka membahas penelitian sebelumnya dan metode pada kasus yang berbeda. Dasar teori membahas teori penunjang yang berkaitan dengan penelitian diantaranya sapi Bali, sistem pendukung keputusan, pengolahan citra digital, data mining, klasifikasi, *K-Nearest Neighbor*, JAVA, dan database.

2.1. Kajian Pustaka

Kajian pustaka pada penelitian ini akan membahas penelitian sebelumnya oleh Ratri Andaruresmi, Retno Nugroho Whidhiasih, dan Mutihera Safitri. Ketiga penelitian tersebut akan membahas mengenai metode dan fitur yang digunakan. Metode dan fitur yang dibahas pada penelitian sebelumnya akan digunakan oleh peneliti sebagai acuan. Ketiga penelitian yang akan di bahas pada kajian pustaka ini dapat dilihat pada tabel 2.1.

Penelitian pertama adalah penelitian yang dilakukan oleh Ratri Andaruresmi. Penelitian yang berjudul *Identifikasi Tipe Wilayah Berbasis Pengolahan Citra Penginderaan Jarak Jauh*. Tujuan dari penelitian ini adalah untuk membuat suatu perangkat lunak yang mampu mengidentifikasi tipe wilayah melalui pengolahan citra dengan menggunakan KNN. Kriteria dalam penelitian ini adalah citra RGB penginderaan jarak jauh *google earth*. Pada pemrosesan awal, citra RGB diubah menjadi citra grayscale, kemudian untuk mengurangi derau citra difilter dengan filter median, dan untuk masukan pada proses ekstrasi ciri citra perlu dideteksi tepi. Ekstrasi ciri dilakukan menggunakan fungsi Gabor *wavelet* yang digunakan untuk mengekstrasi ciri dari citra hasil deteksi tepi. Hasil ekstrasi ciri dianalisa untuk pemilihan vektor ciri dengan akurasi yang tinggi. Proses pengidentifikasian menggunakan algoritma KNN dan didapatkan bahwa metode jarak *euclidean* lebih baik dibanding metode jarak *cityblock* [1:121-126].

Tabel 2.1 Kajian Pustaka

No	Judul	Objek	Metode	Hasil
1	Identifikasi Tipe Wilayah Berbasis Pengolahan Citra Penginderaan Jarak Jauh [1:121-126]	Input berupa citra berwarna penginderaan jarak jauh <i>google earth</i> dengan ukuran 640×480 piksel. Parameter yang pertama diambil adalah nilai RGB.	Metode <i>K-Nearest Neighbor</i> . 1. Citra masukan yang merupakan citra berwarna (RGB) diubah ke <i>grayscale</i> 2. Untuk mengurangi derau citra difilter dengan menggunakan filter median. 3. Menentukan deteksi tepi citra dengan metode Canny. 4. Ekstraksi ciri dari hasil deteksi tepi citra dengan 2D Gabor Wavelet dan melakukan pemilihan vektor ciri 5. Mengidentifikasi citra dengan metode <i>K-Nearest Neighbor</i> .	Pengidentifikasian tipe wilayah menggunakan KNN dengan menghitung jarak <i>Euclidean</i> menghasilkan tingkat akurasi sebesar 86,67%. Sedangkan dengan menggunakan jarak <i>cityblock</i> menghasilkan tingkat akurasi sebesar 80%.
2	Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan KNN Dan LDA [15:29-35]	Input berupa citra buah belimbing dengan parameter R-G dan R-G-B.	Metode KNN dan LDA 1. Ekstraksi ciri dengan menggunakan Analisis Komponen Utama (PCA) 2. Melakukan <i>3 fold cross validation</i> . 3. Mengklasifikasikan dengan metode KNN (RG dan RGB), metode LDA linier dan mahalanobis (RGB).	Akurasi pengklasifikasian buah belimbing dengan metode KNN 2 variabel R-G menghasilkan akurasi 80%. Metode KNN dengan tiga variabel R-G-B dan metode LDA linier maupun Mahalanobis menghasilkan akurasi 91%.
3	Deteksi Hutan Mangrove Di Pantai Utara Jakarta Menggunakan Citra Dari <i>Google Earth</i> Dengan Metode Curvelet [11:1-9]	Input berupa citra dari <i>google earth</i> dengan ukuran 1280×779 piksel.	Metode curvelet (ekstraksi ciri) dan KNN (klasifikasi) 1. Akuisisi Citra dari <i>google earth</i> 2. Melakukan koreksi radiometrik untuk menghilangkan gangguan tutupan atmosfer dan kabut. 3. Melakukan <i>crop</i> citra untuk citra latih dan uji 4. Melakukan preprocessing untuk mengubah warna RGB menjadi HSV. 5. Melakukan ekstraksi ciri menggunakan metode curvelet. Karena citra <i>crop</i> berukuran kecil, maka	Hasil perhitungan luas seluruhnya memiliki akurasi tertinggi sebesar 85,13% dengan jenis jarak <i>Euclidean</i> , nilai $K=1$, dan jumlah ciri tiga buah (mean, standar deviasi, skewness). Hasil perhitungan ketepatan gambar seluruhnya memiliki akurasi tertinggi sebesar 81,80% dengan jenis jarak <i>cityblock</i> , nilai $K=1$,

			<p>digunakan metode statistik untuk mendapatkan ciri tekstur, yaitu menggunakan mean dan standar deviasi.</p> <p>6. Mengklasifikasikan daerah pantai utara jakarta menjadi 4 kelas menggunakan metode KNN.</p>	<p>dan jumlah ciri dua buah (mean dan standar deviasi).</p>
4	<p>Sistem Pendukung Keputusan Pemilihan Bibit Unggul Sapi Bali Berdasarkan Warna Kulit Menggunakan Metode K-Nearest Neighbor</p>	<p>Input berupa citra kulit sapi Bali dengan ukuran 200×200 piksel. Parameter yang digunakan adalah komponen warna RGB.</p>	<p>Metode <i>K-Nearest Neighbor</i>. Langkah-langkah:</p> <ol style="list-style-type: none"> 1. Memasukkan data training berupa citra sapi Bali yang sudah diklasifikasikan berdasarkan umur dan <i>gender</i> oleh admin. 2. Memasukkan data uji berupa citra sapi Bali oleh pengguna. 3. Mengklasifikasi menggunakan algoritma KNN 	<p>Pengklasifikasian sapi Bali yang diinputkan sebagai data uji termasuk sapi dengan kualitas baik, sedang, atau buruk. Akurasi pengklasifikasian sapi Bali dengan metode KNN.</p>

Sumber:[1:121-126][15:29-35][11:1-9]

Penelitian yang kedua dilakukan oleh Retno Nugroho Whidhiasih dengan judul *Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan KNN dan LDA*. Tujuan dari penelitian ini adalah untuk memprediksi tingkat kemanisan buah belimbing melalui citra buah belimbing. Pada penelitian ini, parameter yang digunakan adalah komponen warna Red-Green dan Red-Green-Blue dari citra buah belimbing. Melalui komponen warna tersebut, buah belimbing akan dikelompokkan menjadi tiga, yaitu manis, sedang dan asam. Pada tahap pertama adalah pra proses dimana pada tahap ini dilakukan reduksi citra/ekstraksi ciri dengan menggunakan PCA. Penelitian ini menggunakan *3 fold cross validation* dimana data akan dibagi menjadi 3 bagian. Dua bagian akan dijadikan data latih dan satu bagian akan menjadi data uji, hal tersebut dilakukan secara bergiliran oleh masing-masing bagian. Data yang diujikan dibagi menjadi dua jenis data yaitu data citra R-G dan data citra R-G-B untuk metode KNN. Sedangkan untuk metode LDA Linear dan LDA Mahalanobis menggunakan data citra R-G-B. Hasil yang didapat adalah akurasi menggunakan data citra R-G hanya 80,80% sedangkan akurasi dengan data citra R-G-B adalah 91% untuk KNN, LDA Linear dan LDA Mahalanobis [15:29-35].

Penelitian yang ketiga dilakukan oleh Mutihera Safitri dengan judul *Deteksi Hutan Mangrove di Pantai Utara Jakarta Menggunakan Citra dari Google Earth dengan Metode Curvelet*. Tujuan dari penelitian ini adalah untuk mengimplementasikan sebuah sistem simulasi yang dapat mengidentifikasi hutan mangrove dan menghitung luas hutan mangrove di pantai utara Jakarta. Pada penelitian ini input yang digunakan adalah citra dari *Google Earth* dengan ukuran 1280×779 piksel. Dalam citra tersebut terdapat gangguan atmosfer dan kabut. Sehingga dilakukan koreksi radiometrik untuk menghilangkan gangguan tersebut. Setelah itu dilakukan crop citra untuk data latih dan uji sebesar 10×10 piksel. Kemudian mengubah citra RGB menjadi HSV untuk mempermudah proses selanjutnya. Pada proses ekstraksi ciri dengan metode curvelet digunakan metode statistik untuk mendapatkan ciri tekstur, yaitu mean, standar deviasi, dan skewness. Hal tersebut dilakukan karena citra yang digunakan berukuran kecil. Hasil pengujiannya adalah perhitungan luas seluruhnya memiliki akurasi tertinggi sebesar 85,13% dengan jenis jarak Euclidean, nilai $K=1$, dan jumlah ciri tiga buah

(mean, standar deviasi, skewness). Dan hasil perhitungan ketepatan gambar seluruhnya memiliki akurasi tertinggi sebesar 81,80% dengan jenis jarak cityblock, nilai K=1, dan jumlah ciri dua buah (mean dan standar deviasi) [11:1-9].

Berdasarkan pada tiga penelitian yang telah dikemukakan, peneliti mengambil judul *Sistem Pendukung Keputusan Pemilihan Bibit Unggul Sapi Bali Berdasarkan Warna Kulit Menggunakan Metode K-Nearest Neighbor*. Penelitian ini nantinya diharapkan dapat membantu dalam proses pengklasifikasian kualitas sapi Bali. Sapi Bali akan dikelompokkan menjadi tiga yaitu baik, sedang, dan buruk. Parameter atau kriteria yang digunakan adalah *mean* dan standar deviasi komponen warna RGB. Sedangkan untuk pengklasifikasiannya menggunakan algoritma KNN dengan metode perhitungan jarak *euclidean*.

2.2. Sapi

Di berbagai belahan dunia ada berbagai macam jenis sapi. Mulai dari jenis sapi yang berasal dari ras murni sampai jenis sapi yang berasal dari persilangan. Berdasarkan *breed* (ras) murninya, sapi dapat digolongkan ke dalam 3 (tiga) bangsa sapi, yakni bangsa sapi Eropa (*Bos Taurus*), bangsa sapi India (*Bos Indicus*), dan bangsa sapi Bali (*Bos Sondaicus*) [6:17].

2.2.1. Sapi Bali

Sapi Bali merupakan ras murni karena keturunan asli banteng liar yang telah mengalami proses *domestikasi* (penjinakkan) [2:1]. Pada tabel 2.2 dapat dilihat bahwa sapi Bali mempunyai nama latin *Bibos javanicus*.

Tabel 2.2 Bagan kedudukan sapi Bali dalam Subfamili Bovinae

Subfamili	Genus	Subgenus	Spesies Liar	Sudah Dijinakkan
Bovinae	Bos	Bos	Sudah lenyap	Bangsa sapi <i>Bos taurus</i> Bangsa sapi <i>Bos indicus</i>
		Bibos	<i>Bos (Bibos) banteng</i> (banteng)	Bangsa sapi bali <i>Bos (Bibos) javanicus</i>
			<i>Bos (Bibos) gaurus</i> (gaur)	<i>Bos (Bibos) frontalis</i> (mithan)
			<i>Bos (Bibos) sauveli</i> (kouprey)	
Pöepagus	<i>Bos (pöepagus) mutus</i> (yak liar)	<i>Bos (pöepagus) grunniens</i> (penjinakan yak)		
Bovinae	Bison	<i>Bison bison</i> (bison amerika)		
		<i>Bison bonanas</i> (bison eropa)		
Bovinae		<i>Bubalus arnee</i>		Bubalus bubalus (semua jenis kerbau sungai dan lumpur)

	Bubalus	<i>Bubalus depressiconis</i> (anoa)		<i>Bubalus depressiconis depressiconis</i> (anoa dataran rendah)
				<i>Bubalus depressiconis quarlesi</i> (anoa pegunungan)
				<i>Bubalus mindorensis</i> (tamarao)
	Syncerus	Syncerus caffer (<i>kerbau afrika</i>)	<i>Syncerus caffer caffer</i> (<i>kerbau tanjung harapan</i>)	<i>Syncerus caffer nanus</i> (<i>kerbau kongo</i>)

Sumber: [2:3]

2.2.2. Karakteristik Sapi Bali

Sapi Bali sebagai salah satu bangsa Sapi memiliki ciri-ciri tersendiri yang berbeda dengan bangsa sapi lainnya. Sapi Bali jantan dewasa bibit unggul memiliki warna bulu badan hitam dan memiliki tanduk agak di bagian luar dari kepala. Sapi Bali betina dewasa bibit unggul memiliki warna bulu badan merah bata dan tanduk agak di bagian dalam dari kepala. Kedua sapi Bali jantan dan betina memiliki warna kaki putih dan memiliki “telau”, yakni bulu putih pada bagian pantatnya dan terdapat “garis” (bulu) hitam di sepanjang punggungnya. Meskipun keturunan dari banteng tapi sapi bali tidak memiliki punuk seperti banteng. Sapi bali mempunyai kemiripan pada bentuk badan yang besar dan bentuk dada yang cenderung gagah. Sewaktu lahir, baik sapi Bali jantan maupun betina memiliki bulu berwarna merah bata. Setelah dewasa, warna bulu sapi Bali jantan berubah menjadi hitam karena adanya pengaruh hormon *testosteron* (hormon kelamin jantan). Jika sapi Bali jantan dikebiri, warna bulunya yang hitam akan berubah menjadi merah. Sapi Bali juga sering mengalami penyimpangan dari ciri-ciri khususnya. Sapi Bali jantan maupun betina akan berwarna coklat pucat jika mengalami penyimpangan [6:18-19].



(a)



(b)

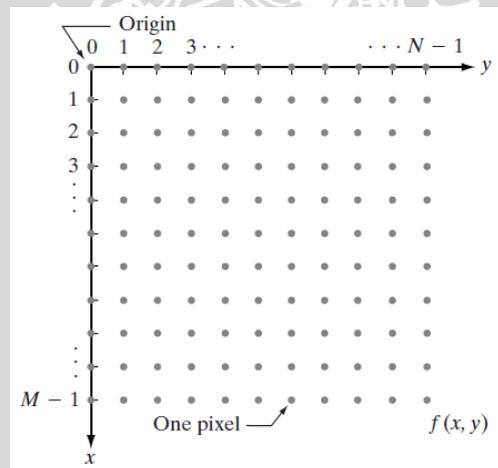
Gambar 2.1 Sapi Bali (a) Jantan dan (b) Betina
Sumber: Observasi

2.2.3. Keunggulan Sapi Bali

Menjadi keturunan langsung dari banteng membuat sapi Bali mempunyai keunggulan dibanding dengan bangsa sapi lainnya. Keunggulan tersebut antara lain seperti kemampuan adaptasi yang baik di daerah yang buruk sekalipun. Sapi Bali memiliki tingkat kesuburan yang tinggi sehingga dapat dilihat jumlah sapi Bali sangat banyak di daerah Bali bahkan di daerah lain. Sapi Bali juga memiliki persentasi produksi karkas yang tinggi di banding jenis-jenis sapi tropis lainnya. Berdasarkan hal tersebut, sapi Bali memiliki potensi untuk dikembangkan, baik sebagai ternak potong atau ternak kerja [6:26-28].

2.3. Pengolahan Citra Digital

Citra merupakan suatu fungsi kontinu dari intensitas cahaya dalam bidang dua dimensi, dengan (x,y) menyatakan koordinat citra dan nilai f pada koordinat (x,y) menyatakan tingkat kecerahan atau derajat keabuan [5:50]. Bentuk sistem koordinat sebuah citra digital dapat dilihat pada gambar 2.2.

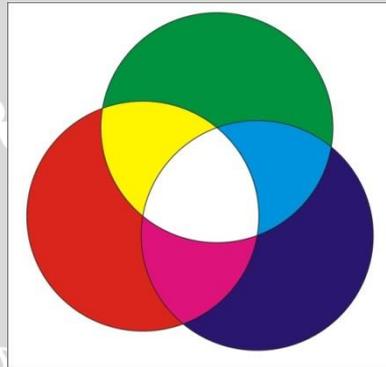


Gambar 2.2 Sistem koordinat citra digital

Sumber: [5:55]

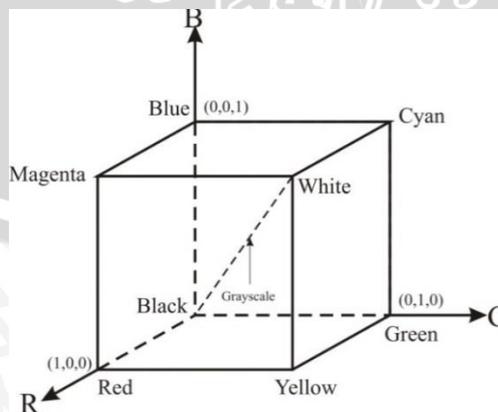
Salah satu karakteristik penting dari citra adalah warna. Dalam hal ini manusia bisa membedakan ribuan warna dan tingkat intensitasnya dibandingkan dengan puluhan tingkat keabuan. Manusia mempunyai 5-7 juta sel yang dibagi menjadi tiga kategori sensor (merah, hijau, dan biru). 65% sel kerucut sensitif pada warna merah, 33% hijau, dan 2% biru. Pada karakteristik mata manusia warna terlihat seperti kombinasi warna variabel yang disebut warna primer, yaitu merah (*red*), hijau (*green*), dan biru (*blue*) [5:284].

Warna primer dapat digunakan untuk menghasilkan warna sekunder. Warna sekunder itu antara lain, warna magenta perpaduan antara warna merah dan warna biru. Kemudian warna cyan, perpaduan warna hijau dan biru. Serta warna kuning, perpaduan antara warna merah dan hijau. Mencampur 3 warna primer atau 1 warna sekunder dan 1 warna primer yang berseberangan akan menghasilkan warna putih. Perpaduan warna tersebut dapat dilihat pada gambar 2.3.



Gambar 2.3 Campuran warna merah, hijau, dan biru menghasilkan warna putih
Sumber: [5:286]

Model warna adalah sebuah sistem koordinat yang bisa memetakan semua warna dalam sistem tersebut dengan sebuah titik. Salah satu model warna adalah model warna RGB. Dalam model ini tiap warna ditunjukkan dengan kombinasi tiga warna primer. Ketiga warna primer tersebut membentuk sistem koordinat cartesian tiga dimensi [5:290]. Skema warna kubik RGB dapat dilihat pada gambar 2.4. Titik sepanjang diagonal utama mempunyai nilai-nilai keabuan, dari hitam di titik pusat koordinat sampai putih di titik (1,1,1).



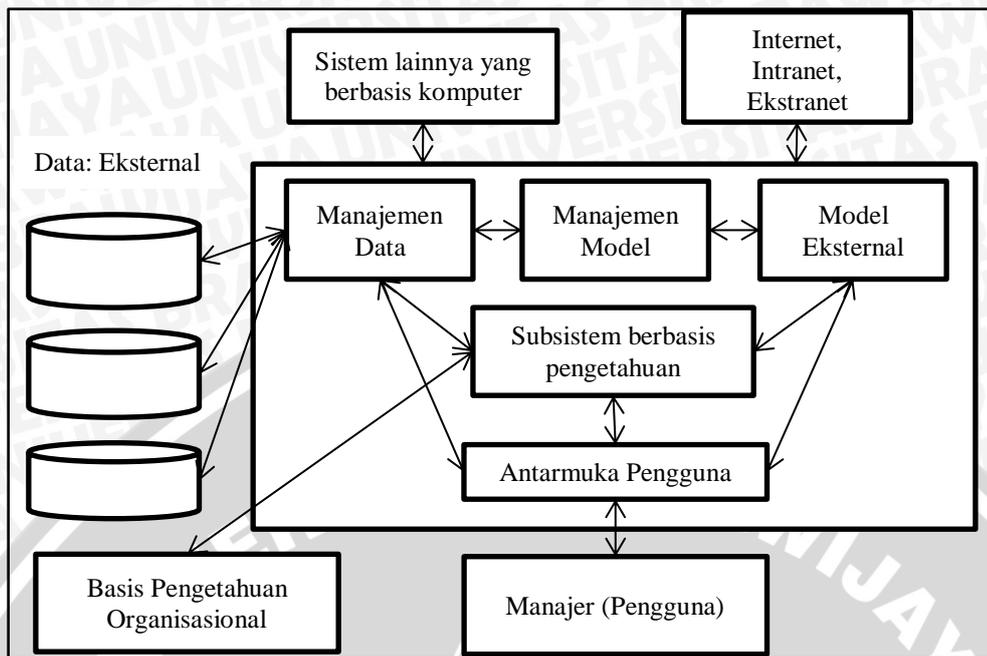
Gambar 2.4 Skema warna kubik RGB
Sumber: [5:290]

2.4. Sistem Pendukung Keputusan

Sistem Pendukung Keputusan atau sering disebut *Decision Support System* (DSS) merupakan sebuah alat bantu yang menggunakan informasi berbasis komputer. DSS ini digunakan manajer untuk memecahkan masalah semi struktur, dimana manajer dan komputer harus bekerja sama sebagai tim pemecah masalah. DSS mempunyai beberapa komponen untuk optimasi proses membuat keputusan. Komponen tersebut antara lain [14:143-145]:

- a. Subsistem manajemen data yang digunakan untuk memajemen data yang relevan dan dikelola oleh perangkat lunak *Database Manajemen Sistem* (DBMS).
- b. Subsistem manajemen model yang sering disebut sistem manajemen basis model (MBMS). Komponen ini dapat dikoneksikan ke penyimpanan eksternal yang ada pada model dan dapat diimplementasikan pada sistem pengembangan web.
- c. Subsistem antarmuka pengguna merupakan subsistem yang digunakan pengguna berkomunikasi dengan DSS. Kontribusi unik dari DSS berasal dari interaksi yang intensif antara komputer dan pembuat keputusan.
- d. Subsistem manajemen berbasis-pengetahuan merupakan komponen opsional namun dapat memberikan banyak manfaat bagi tiga komponen utama diatas. Komponen ini diinterkoneksi dengan respitori pengetahuan dari perusahaan yang sering disebut basis pengetahuan organisasional.

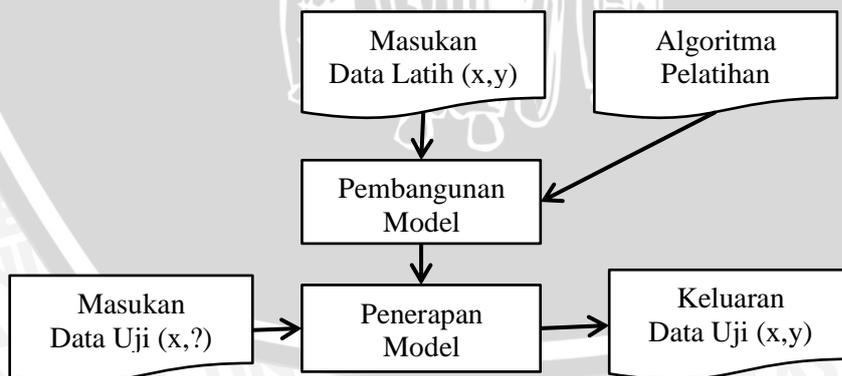
Komponen-komponen tersebut dapat digambarkan sebagai skematik DSS seperti pada gambar 2.5 berikut.



Gambar 2.5 Skematik DSS
Sumber: [13:144]

2.5. Klasifikasi

Klasifikasi dapat didefinisikan sebagai pekerjaan yang melakukan pelatihan/pembelajaran.. Proses pelatihan tersebut akan menghasilkan suatu model yang kemudian disimpan sebagai memori. Dalam klasifikasi ada suatu model yang menerima masukan, kemudian mampu melakukan pemikiran terhadap masukan. Pada akhirnya memberi jawaban sebagai keluaran hasil pemikiran tersebut.



Gambar 2.6 Proses Pekerjaan Klasifikasi
Sumber: [10:46]

Pada gambar 2.6 ditunjukkan mengenai kerangka kerja klasifikasi. Terdapat sejumlah data latih (x,y) untuk digunakan sebagai data pembangunan model. Model tersebut kemudian dipakai untuk memprediksi kelas dari data uji

($x, ?$) sehingga diketahui kelas y yang sesungguhnya. Model yang sudah dibangun pada saat pelatihan kemudian dapat digunakan untuk memprediksi tabel kelas data baru yang belum diketahui. Dalam pembangunan model selama proses pelatihan tersebut diperlukan suatu algoritma yang disebut algoritma pelatihan (learning algorithm). Ada banyak algoritma pelatihan yang sudah dikembangkan oleh para peneliti, seperti *Artificial Neural Network*, *Support Vector Machine*, *K-Nearest Neighbor*, dan sebagainya.

Sebuah sistem yang melakukan klasifikasi diharapkan dapat melakukan klasifikasi semua set data dengan benar, tetapi tidak dapat dipungkiri bahwa kinerja suatu sistem tidak bisa 100% benar. Sehingga sebuah sistem klasifikasi juga harus diukur kinerjanya. Pengukuran kinerja klasifikasi dilakukan dengan matriks konfusi (*confusion matrix*) [10:47].

Tabel 2.3 Matriks Konfusi untuk Klasifikasi Dua Kelas

f_{ij}		Kelas hasil prediksi (j)	
		Kelas = 1	Kelas = 0
Kelas asli (i)	Kelas = 1	f_{11}	f_{10}
	Kelas = 0	f_{01}	f_{00}

Sumber: [10:47]

Dari tabel 2.3, dapat dilihat matriks konfusi yang melakukan klasifikasi masalah biner (dua kelas), yaitu kelas 0 dan 1. Setiap sel f_{ij} dalam matriks menyatakan jumlah data dari kelas i yang hasil prediksinya masuk ke kelas j . Berdasarkan isi matriks konfusi, kita dapat mengetahui jumlah data dari masing-masing kelas yang diprediksi secara benar ($f_{11} + f_{00}$) dan data yang diklasifikasi secara salah ($f_{10} + f_{01}$). Kuantitas matriks konfusi dapat diringkas menjadi dua nilai, yaitu akurasi dan laju error [10:47].

2.6. Nearest Neighbor Classifier

Algoritma *Nearest Neighbor* (kadang disebut *K-Nearest Neighbor*/K-NN). Algoritma ini merupakan algoritma yang melakukan klasifikasi berdasarkan kedekatan lokasi (jarak) suatu data dengan data yang lain. Prinsip sederhana yang diadopsi oleh algoritma K-NN adalah “Jika suatu hewan berjalan seperti bebek, bersuara *kwek-kwek* seperti bebek, dan penampilannya seperti bebek, hewan itu mungkin bebek” [10:48].

2.6.1. Algoritma Nearest Neighbor

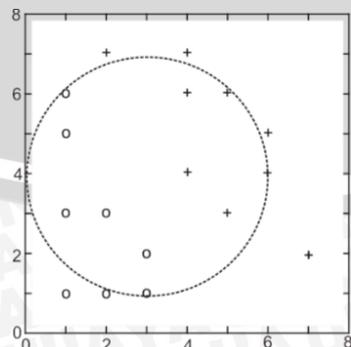
Algoritma *Nearest Neighbor* sangat bergantung pada proses pembelajarannya. Algoritma ini melakukan pendekatan dengan menemukan semua data pelatihan yang secara relatif sama dengan atribut dari data uji. Hal tersebut digunakan untuk menentukan nama kelas dari data uji. *K-Nearest Neighbor classifier* umumnya didasarkan pada jarak *Euclidean* antara data uji dan data pelatihan yang telah ditentukan. Persamaan jarak *euclidean* ditunjukkan dalam persamaan (2-1).

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2} \dots\dots\dots (2-1)$$

dimana:

- $d(x_i, x_j)$ = jarak *euclidean* antara vektor x_i dan vektor x_j
- $a_r(x_i)$ = fitur ke r dari vektor x_i
- $a_r(x_j)$ = fitur ke r dari vektor x_j
- n = jumlah fitur pada vektor x_i dan x_j

Diberikan suatu data uji, kemudian kita akan menghitung kedekatan dengan titik yang ada dalam kumpulan data latih, dengan menggunakan ukuran kedekatan. Titik data diklasifikasikan berdasarkan nama kelas dari tetangganya. Jika k terlalu kecil, maka *nearest neighbor classifier* mungkin akan rentan pada kesalahan kecocokan karena *noise* terdapat dalam data pelatihan. Disisi lain, jika k terlalu besar, *nearest neighbor classifier* mungkin akan salah mengklasifikasikan data tes kerana bisa mengakibatkan distorsi data yang besar seperti pada gambar 2.7.



Gambar 2.7 KNN dengan nilai K yang besar
Sumber: [10:50]

Dilihat dari permasalahan di atas akar masalahnya adalah pemilihan nilai K yang tepat. Cara voting mayoritas masih mempunyai kelemahan jika terdapat jumlah kelas yang sama. Untuk mengatasi itu ditambahkan penggunaan bobot untuk menghitung kandidat yang sebaiknya diambil oleh data uji dari K -tetangga terdekat. Penggunaan bobot tersebut sering disebut dengan *weighted voting*. Persamaan *weighted voting* ditunjukkan dalam persamaan (2-2).

$$w_i = \frac{1}{d(x', x_i)^2} \dots \dots \dots (2-2)$$

Dimana:

w_i = *weighted voting* kelas i

$d(x', x_i)$ = jarak *euclidean* antara vektor x' dan vektor x_i

Kelas dengan vote terbanyak akan dipilih sebagai kelas dari data uji.

2.6.2. Klasifikasi dengan Nearest Neighbor

Nearest Neighbor merupakan teknik klasifikasi yang sederhana, tetapi mempunyai hasil kerja yang bagus. Meskipun begitu, K -NN juga mempunyai kelebihan dan kekurangan. Beberapa karakteristik K -NN adalah sebagai berikut [10:58]:

1. K -NN merupakan algoritma yang menggunakan seluruh data latih untuk melakukan proses klasifikasi (*complete storage*). Hal ini mengakibatkan proses prediksi yang sangat lama untuk data dalam jumlah yang sangat besar. Pendekatan lain adalah dengan menggunakan mean data dari setiap kelas, kemudian menghitung jarak terdekat data uji ke mean data setiap kelas. Tapi hasilnya akan kurang memuaskan karena model akan hanya membentuk *hyperplane* linear tepat di tengah-tengah di antara 2 kelas yang memisahkan 2 kelas (untuk kasus 2 kelas).
2. Algoritma K -NN tidak membedakan setiap fitur dengan suatu bobot. Tidak seperti pada *Artificial Neural Network* (ANN) yang berusaha menekan fitur yang tidak mempunyai kontribusi terhadap klasifikasi menjadi 0 pada bagian bobot. K -NN tidak memiliki bobot untuk masing-masing fitur.



3. Karena K-NN masuk kategori *lazy learner* yang menyimpan sebagian atau semua data dan hampir tidak ada proses pelatihan. K-NN cepat dalam proses pelatihan, tetapi lambat saat proses prediksi.
4. Hal yang rumit adalah menentukan nilai K yang paling sesuai.
5. Karena K-NN pada prinsipnya memilih tetangga terdekat, parameter jarak juga penting untuk dipertimbangkan sesuai dengan kasus datanya. Euclidean sangat cocok untuk menggunakan jarak terdekat antara dua data, tetapi Manhattan sangat teguh (*robust*) untuk mendeteksi *outlier* dalam data.

2.7. Parameter Statistik

Parameter yang digunakan dalam perhitungan analisis frekuensi penelitian ini meliputi parameter standar deviasi (*S*). Untuk memahami standar deviasi diperlukan sebuah himpunan data. Himpunan data tersebut dapat diperoleh dari sampel sebuah populasi.

Standar deviasi dari sebuah himpunan data adalah ukuran seberapa tersebar nilai data-data tersebut. Apabila penyebaran data sangat besar terhadap nilai rata-rata, maka nilai standar deviasi akan besar. Akan tetapi jika penyebaran data sangat kecil terhadap nilai rata-rata, maka nilai standar deviasi akan kecil [13:87-88]. Persamaan standar deviasi dapat ditunjukkan dalam persamaan (2-3).

$$S = \sqrt{\frac{\sum_{i=1}^N (X_{ij} - \bar{X})^2}{N}} \dots\dots\dots (2-3)$$

dimana :

- S* = standar deviasi komponen warna RGB
- X_{ij}* = nilai pengukuran dari nilai RGB baris ke *i* dan kolom ke *j*
- \bar{X} = nilai rata-rata RGB
- N* = jumlah data RGB

Parameter \bar{X} yang merupakan persamaan rata-rata ditunjukkan dalam persamaan (2-4).



$$\bar{X} = \frac{\sum_{i=1}^N X_{ij}}{N} \dots\dots\dots (2-4)$$

dimana :

\bar{X} = nilai rata-rata RGB

X_{ij} = nilai pengukuran dari nilai RGB baris ke i dan kolom ke j

N = jumlah data RGB

Untuk lebih memahami tentang menghitung standar deviasi dari RGB maka akan diberikan suatu contoh kasus. Data warna diambil dari data latih SB-1 dengan ukuran pixel 32. Perhitungan contoh kasus menggunakan 5×5 pixel dari 32 pixel data latih SB-1. Komponen warna tiap pixelnya dapat dilihat pada tabel 2.4.

Tabel 2.4 Contoh Nilai RGB Potongan Gambar 5 × 5 pixel

X_{ij}	1	2	3	4	5
1	R = 146 G = 147 B = 152	R = 140 G = 141 B = 146	R = 137 G = 138 B = 143	R = 139 G = 140 B = 145	R = 140 G = 141 B = 146
2	R = 137 G = 138 B = 143	R = 134 G = 135 B = 140	R = 136 G = 137 B = 142	R = 140 G = 141 B = 146	R = 142 G = 143 B = 148
3	R = 145 G = 146 B = 151	R = 141 G = 142 B = 147	R = 136 G = 137 B = 142	R = 141 G = 142 B = 147	R = 138 G = 139 B = 144
4	R = 149 G = 150 B = 155	R = 141 G = 142 B = 147	R = 136 G = 137 B = 142	R = 137 G = 138 B = 143	R = 136 G = 137 B = 142
5	R = 140 G = 141 B = 146	R = 130 G = 131 B = 136	R = 126 G = 127 B = 132	R = 134 G = 135 B = 140	R = 142 G = 143 B = 148

Sumber: [Lampiran 4]

Langkah awal yang dilakukan adalah mencari nilai rata-rata RGB. Contoh perhitungan yang akan dilakukan adalah menghitung rata-rata dari warna merah (R). Persamaan yang digunakan adalah persamaan (2-4).

$$\begin{aligned} \bar{R} &= \frac{X_{11} + X_{12} + X_{13} + X_{14} + X_{15} + X_{21} + X_{22} + \dots + X_{55}}{25} \\ &= \frac{146 + 140 + 137 + 139 + 140 + 137 + 134 + \dots + 142}{25} \\ &= \frac{3463}{25} \\ &= 138,52 \end{aligned}$$



Cara perhitungan tersebut juga dilakukan dalam mencari nilai rata-rata dari hijau (\bar{G}) dan biru (\bar{B}). Langkah selanjutnya setelah mendapat nilai rata-rata RGB adalah menghitung nilai standar deviasi RGB menggunakan persamaan (2-3).

$$\begin{aligned}
 S_R &= \sqrt{\frac{(X_{11}-\bar{R})^2+(X_{12}-\bar{R})^2+(X_{13}-\bar{R})^2+\dots+(X_{55}-\bar{R})^2}{25}} \\
 &= \sqrt{\frac{(146-138,52)^2+(140-138,52)^2+(137-138,52)^2+\dots+(142-138,52)^2}{25}} \\
 &= \sqrt{\frac{562,24}{25}} \\
 &= \sqrt{22,4896} \\
 &= 4,74
 \end{aligned}$$

Perhitungan tersebut dilakukan juga dalam pencarian standar deviasi *green* (S_G) dan standar deviasi *blue* (S_B).

2.8. Perhitungan Akurasi

Perhitungan akurasi dilakukan untuk mengetahui tingkat akurasi dari hasil klasifikasi, dengan cara menghitung jumlah record uji yang kelasnya diprediksi secara tepat. Persamaan pengujian akurasi ditunjukkan dalam persamaan (2-5).

$$\text{Akurasi} = \frac{\sum \text{Data uji benar diklasifikasi}}{\sum \text{Total data uji}} \times 100\% \dots \dots \dots (2-5)$$

Jumlah prediksi benar adalah jumlah record data uji yang diprediksi kelasnya menggunakan metode klasifikasi dan hasilnya sama dengan kelas sebenarnya. Sedangkan jumlah total prediksi adalah jumlah keseluruhan record yang diprediksi kelasnya (seluruh data uji). Metode klasifikasi berusaha untuk mencari model yang memiliki tingkat akurasi yang tinggi ketika model tersebut diterapkan pada data uji [7:319].

2.9. JAVA

Java adalah bahasa pemrograman yang serbaguna. Hal tersebut dikarenakan bahasa Java dapat digunakan untuk membuat program seperti halnya Pascal atau C++. Java juga mendukung sumber daya internet yang saat ini populer, yaitu *World Wide Web* dan juga mendukung aplikasi LAN maupun WAN [3:3].

Berbicara mengenai Java berarti ada dua bagian yang dibicarakan, yakni Java sebagai bahasa pemrograman dan Java sebagai platform. Sebagai bahasa pemrograman, Java dikenal sebagai bahasa pemrograman tinggi dengan beberapa fitur utama. Fitur pertama, Java dirancang untuk mudah dipelajari terutama untuk programmer yang telah mengenal C/C++. Fitur kedua, Java sudah mempunyai paradigma pemrograman masa depan yaitu paradigma pemrograman berorientasi object. Fitur ketiga, Java dirancang sebagai bahasa pemrograman yang handal dan aman. Aplikasi yang dibangun dengan bahasa Java sangat handal dengan manajemen memori yang bagus [3:3-4].

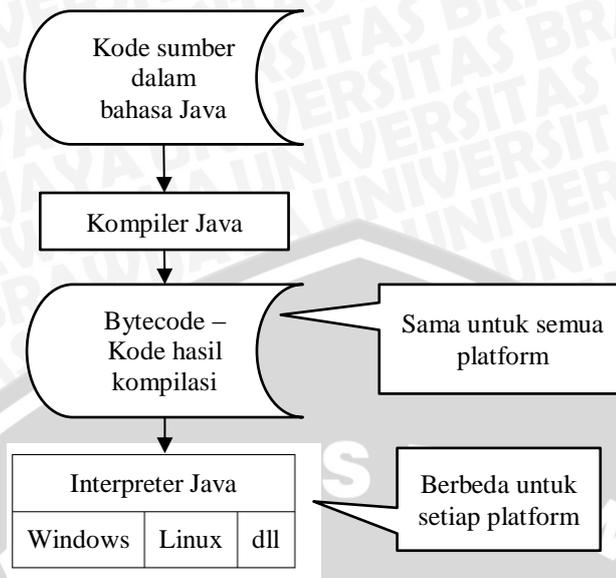
Fitur keempat, Java dirancang untuk dapat dijalankan di semua platform. Java compiler yang digunakan untuk mengkompilasi kode program Java dirancang untuk menghasilkan kode netral terhadap semua arsitektur *hardware* yang disebut sebagai *Java bytecode*. JVM (*Java Virtual Machine*) menjadi bagian dari teknologi Java yang menyediakan media untuk dijelankannya aplikasi Java (*Java bytecode*). Fitur kelima, Java dirancang untuk menghasilkan aplikasi-aplikasi dengan performansi yang terbaik [3:4].

Tabel 2.4 Java pada berbagai system operasi

Sistem Operasi	Vendor
Linux	Banyak Perusahaan
MacOS	Apple
Solaris	Sun Microsystem
Keluarga Windows	Microsoft Corporation
HP-UX	Hewlett Packard
Netware	Novell
Digital Unix	Digital Equipment Corporation
AIX	IBM
OS/2	IBM
DG/UX	Data General Corporation
IRIX	Silicon Graphics

Sumber:[8:3]

Ketidakbergantungan terhadap platform sering dinyatakan dengan istilah portabilitas. Portabilitas Java tidak hanya sebatas pada program sumber (*source code*), melainkan juga pada tingkat kode biner yang disebut *bytecode*. Langkah-langkah pengkompilasian hingga pengekseskuan kode Java dapat dilihat pada gambar 2.8.



Gambar 2.8 Skema pengkompilasian hingga pengeksekusian kode Java
Sumber:[9:4]

Kode yang disebut *bytecode* dapat dijalankan pada berbagai system operasi karena kode ini berbeda dengan kode mesin. Kode mesin sangat bergantung pada platform, sedangkan *bytecode* dapat dimengerti oleh semua platform yang dilengkapi interpreter Java. Java *bytecode* bisa dianggap sebagai kode-kode mesin dari JVM, yang kemudian JVM akan menginterpretasikan kode-kode tersebut ke kode mesin dari arsitektur yang bersangkutan [3:4-5].

Java API menyediakan fitur-fitur yang menarik untuk pengembangan aplikasi. Fitur-fitur yang ditawarkan antara lain sebagai berikut [16:17]:

a. Applet

Java applet merupakan program Java yang berjalan di atas browser. Berguna untuk pengembangan aplikasi-aplikasi berbasis web.

b. Java Networking

Java Networking merupakan sekumpulan API yang menyediakan fungsi-fungsi untuk aplikasi-aplikasi jaringan.

c. Java Database Connectivity (JDBC)

JDBC menyediakan sekumpulan API yang dapat digunakan untuk mengakses database.

d. Java Security

Java security menyediakan sekumpulan API untuk mengatur *security* dari aplikasi Java baik secara *high-level* ataupun *low-level*, seperti *public/private key management* dan *certificates*.

e. Java Swing

Java Swing menyediakan sekumpulan API untuk membangun aplikasi-aplikasi GUI (*Graphical User Interface*).

f. Java RMI

Java RMI menyediakan sekumpulan API untuk membangun aplikasi-aplikasi java yang mirip dengan model RPC (*Remote Procedure Call*). Jadi, object-object Java bias di-*call* secara *remote* pada jaringan computer.

g. Java 2D/3D

Java 2D/3D menyediakan sekumpulan API untuk membangun grafik-grafik 2D/3D yang menari dan juga akses ke printer.

h. Java Server Pages

Perkembangan dari Java Servlet yang digunakan untuk menggantikan aplikasi-aplikasi CGI, JSP (*Java Server Pages*) yang mirip ASP dan PHP merupakan alternative terbaik untuk solusi aplikasi internet.

i. JNI (Java Native Interface)

JNI menyediakan sekumpulan API yang digunakan untuk mengakses fungsi-fungsi pada library yang dibuat dengan bahasa pemrograman lain.

j. Java Sound

Java Sound menyediakan sekumpulan API untuk manipulasi *sound*.

k. Java IDL + CORBA

Java IDL (*Interface Definition Language*) menyediakan dukungan Java untuk implementasi CORBA (*Common Object Request Broker*) yang merupakan model distributed-object untuk solusi aplikasi besar di dunia *networking*.

l. Java Card

Java Card utamanya digunakan untuk aplikasi-aplikasi pada *smart card*, yang sederhana wujudnya seperti SIM Card pada *handphone*.

m. JTAPI (Java Teephony API)

Java Telephony API menyediakan sekumpulan API untuk memanfaatkan devices-devices telephony, sehingga cocok untuk aplikasi-aplikasi CTI (*Computer Telephony Integration*).

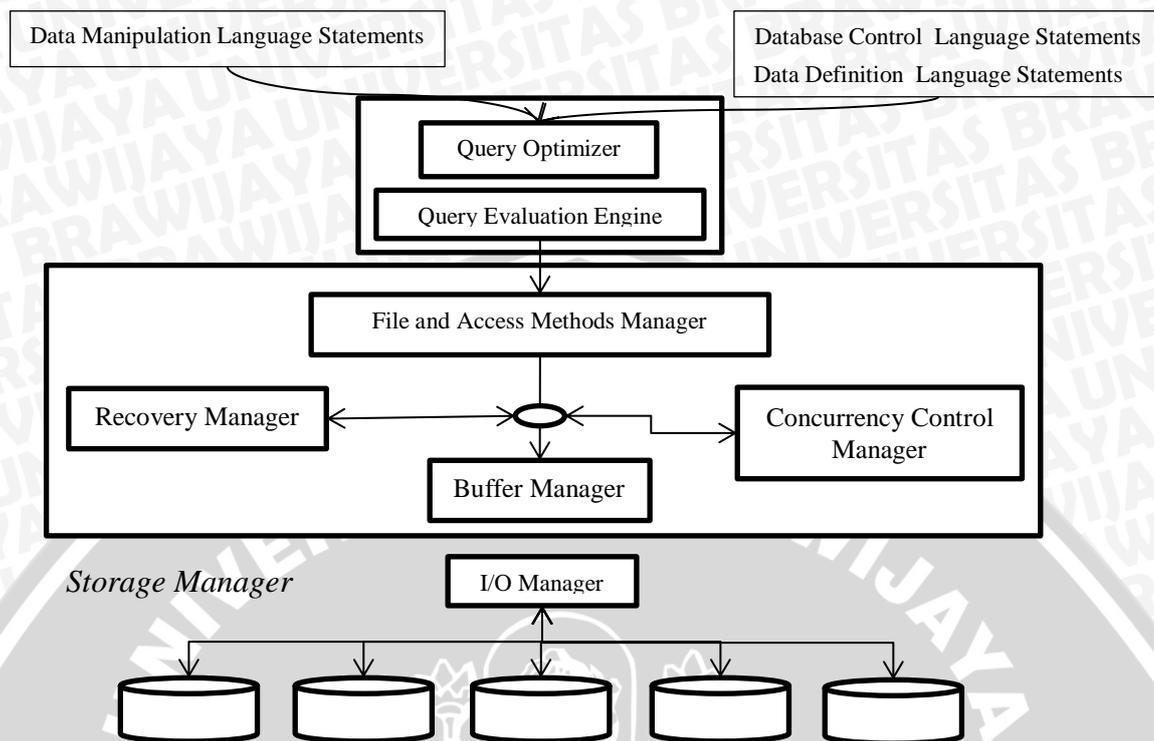
2.10. Database

Database adalah sebuah tempat penyimpanan yang besar dimana terdapat kumpulan data di dalamnya. Dalam merancang database hal yang perlu diperhatikan adalah efisiensi. Banyaknya data yang sama mengurangi efisiensi database dan dapat memakan banyak memori. Aplikasi database mengatur data yang bersifat homogen dalam jumlah banyak. Data tersebut akan dibagi dan didistribusikan ke pengguna dan diproses [4:3]. Untuk mengontrol data dalam *database* dibutuhkan suatu sistem yang biasa disebut Sistem Manajemen Database atau *Database Management System* (DBMS).

2.10.1. Sistem Manajemen Database

Sistem Manajemen Database merupakan suatu sistem aplikasi yang memungkinkan pengguna untuk melakukan implementasi dan mengolah database. Daur hidup dari aplikasi database ini sangat kompleks mengingat tidak hanya melakukan analisis, desain, pengembangan, dan pengujian, tapi juga manajemen berkelanjutan dan pembaharuan untuk proses yang lebih baik. Ada bahasa untuk melakukan pengolahan aplikasi database ini, yaitu *Data Definition Language* (DDL), *Data Manipulation Language* (DML), dan *Database Control Language* (DCL) [4:4].

Pernyataan pada DDL, DML, dan DCL di proses dan dioptimasi pada *query optimizer*. Kemudian *query evaluation engine* meneruskannya ke *storage manager* untuk diproses dan hasilnya akan dikembalikan pada tampilan pengguna dan program. Proses diatas digambarkan sebagai arsitektur DBMS seperti pada gambar 2.9.



Gambar 2.9 Arsitektur DBMS
Sumber: [4:5]

Antarmuka dari *storage manager* adalah *file and access method manager*. Antarmuka ini berinteraksi dengan *storage* melalui *buffer* yang ada di memori utama dari server. *Buffer manager* membawa data ke memori utama dan memanggil *input-output manager*. Interaksi dari *file and access method manager* dengan *buffer manager* dikontrol dan dimonitor oleh *recovery manager* dan *concurrency control manager*.

2.10.2. MySQL

Keandalan dari DBMS dilihat dari cara kerja *optimizer*-nya dalam melakukan proses perintah-perintah SQL. MySQL merupakan solusi dari permasalahan tersebut. MySQL dapat dikatakan lebih unggul dibanding database server lainnya dalam query data. Hal tersebut dikarenakan kecepatan dalam proses query-nya sangat cepat dan MySQL merupakan *Relation Database Management System* (RDBMS) yang bisa didapat secara gratis [9:1-2].

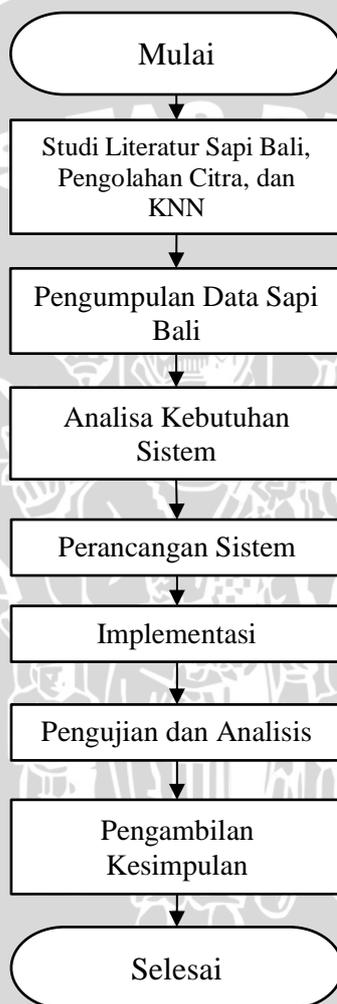
Keunggulan MySQL bisa dilihat melalui beberapa keistimewaan yang dimilikinya. Keistimewaan yang dimiliki MySQL antara lain [9:3]:

- a. MySQL dapat berjalan stabil diberbagai sistem operasi.

- b. MySQL didistribusikan secara gratis di bawah lisensi *General Public License* (GPL).
- c. MySQL dapat digunakan oleh beberapa user dalam waktu bersamaan (multiuser).
- d. MySQL mempunyai kecepatan yang menakjubkan dalam menangani query sederhana.
- e. MySQL memiliki tipe kolom yang sangat kompleks.
- f. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELCET dan WHERE dalam query.
- g. MySQL memiliki beberapa lapisan sekuritas untuk tingkat keamanannya.
- h. MySQL mampu menagani database dalam skala besar.
- i. MySQL dapat melakukan koneksi dengan client menggunakan protokol TCP/IP, Unix, dan NT.
- j. MySQL dapat mendeteksi pesan kesalahan pada client dengan menggunakan lebih dari dua puluh bahasa.
- k. MySQL mempunyai interface dengan dukungan API.
- l. MySQL dilengkapi berbagai tool yang dapat digunakan untuk administrasi database.
- m. MySQL memiliki struktur tabel yang lebih fleksibel dibanding PostgreSQL ataupun Oracle.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan mengenai langkah-langkah yang akan ditempuh dalam penelitian, yaitu studi literatur, pengumpulan data, analisa kebutuhan, perancangan, implementasi, pengujian dan analisis serta pengambilan kesimpulan.



Gambar 3.1 Diagram Blok Metodologi Penelitian
Sumber: [Perancangan]

3.1. Studi Literatur

Studi literatur merupakan data sekunder yang telah ada sebelumnya seperti dari sebuah literatur atau buku. Studi literatur ini nantinya akan berfungsi untuk mempelajari teori yang akan digunakan dalam penulisan ini. Penyusunan studi literatur adalah sebagai berikut:

1. Sapi Bali
2. Citra Digital
3. Sistem Pendukung Keputusan
4. Klasifikasi
5. Metode *K-Nearest Neighbor*
6. Pemrograman Java
7. Sistem Manajemen Basis Data

3.2. Pengumpulan Data

Pengumpulan data dilakukan secara langsung di Balai Pembibitan Ternak Unggul yang terletak di kabupaten Jembrana, Bali. Pengumpulan data diambil dengan cara observasi. Objek dari penelitian ini adalah warna kulit dari sapi Bali yang diambil gambarnya menggunakan kamera SLR.

3.3. Analisa Kebutuhan

Analisa kebutuhan bertujuan untuk mendapatkan semua kebutuhan yang diperlukan dalam pengimplementasian Implementasi Pendukung Keputusan. Analisis kebutuhan disesuaikan dengan lokasi dan variabel penelitian, menentukan kebutuhan data yang akan digunakan, dan mempersiapkan alat dan bahan penelitian. Kebutuhan yang digunakan dalam pembuatan penelitian ini meliputi:

1. Kebutuhan Hardware, meliputi:
 - a. Processor Intel(R) Core(TM) i5 CPU M 480 2,67 GHz
 - b. Memory 2 GB
 - c. VGA ATI Radeon HD 6470M
2. Kebutuhan Software, meliputi:
 - a. Sistem operasi windows 7 Profesional
 - b. NetBeans IDE 7.4
 - c. *Java Development Kit (JDK 7)*
 - d. *Snipping Tool*
 - e. Basisdata MySQL

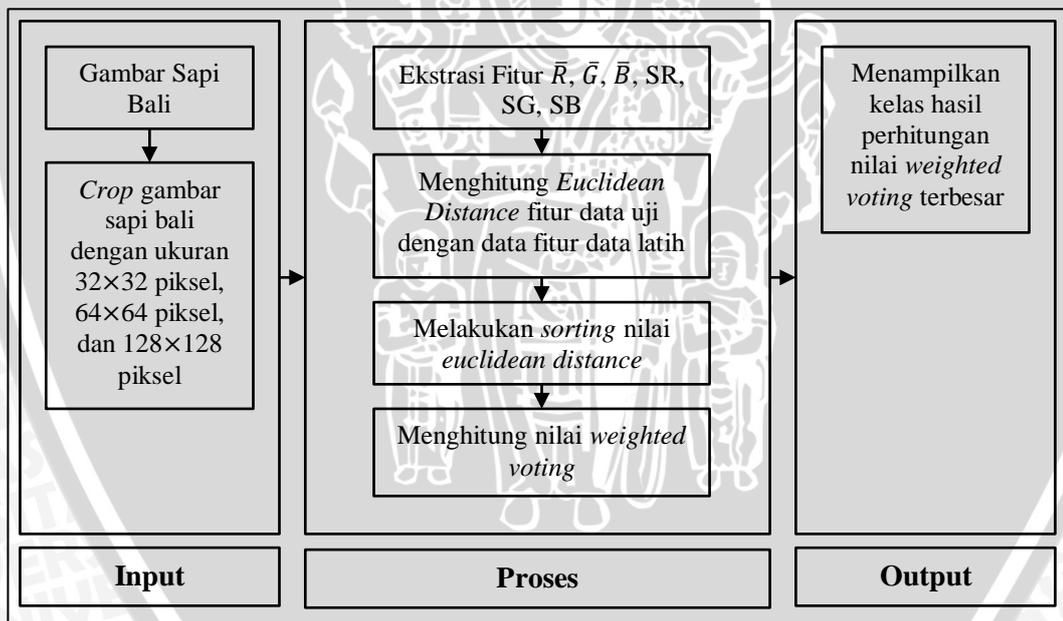
3. Data yang dibutuhkan meliputi:
 - a. Gambar sapi Bali yang berada di BPTU Jembrana, Bali.

3.4. Perancangan Sistem

Perancangan sistem dibangun berdasarkan hasil pengambilan data dan analisis kebutuhan yang dilakukan. Perancangan sistem menjelaskan deskripsi umum sistem, desain dari model *K-Nearest Neighbor* sebagai metode klasifikasi kualitas sapi Bali dan arsitekturnya dalam pengklasifikasian sapi Bali.

3.4.1. Blok Diagram Sistem

Data inputan yang digunakan adalah gambar sapi bali yang berada di BPTU Sapi Bali. Sedangkan hasil keluaran (output) dari sistem ini adalah data hasil dari pengklasifikasian kualitas sapi bali berdasarkan *weighted voting* terbesar. Arsitektur blok diagram perancangan aplikasi sistem pendukung keputusan ini ditunjukkan dalam gambar 3.2.



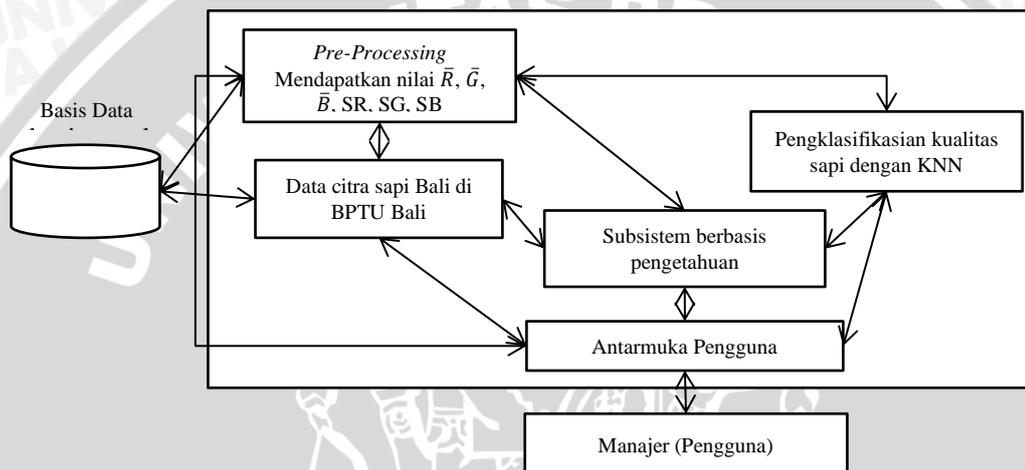
Gambar 3.2 Arsitektur Blok Diagram Perancangan Aplikasi
Sumber: [Perancangan]

Pada gambar 3.2 dijelaskan bagaimana cara kerja sistem. Data inputannya adalah gambar sapi bali. Gambar sapi bali tersebut kemudian diekstraksi fiturnya untuk mendapatkan nilai \bar{R} , \bar{G} , \bar{B} , SR, SG, SB. Dilakukan perhitungan *euclidean distance* terhadap fitur data uji dengan fitur data latih. Hasil perhitungan *euclidean distance* kemudian diurutkan mulai dari yang terkecil kemudian dilakukan

perhitungan *weighted voting* tiap kelas. Hasil dari sistem ini adalah kelas yang mempunyai nilai *weighted voting* terbesar.

3.4.2. Arsitektur Implementasi Sistem Pendukung Keputusan

Model Perancangan Sistem digunakan sebagai inti dari proses pengklasifikasian kualitas sapi Bali. Tahap yang ada dalam sistem ini adalah *data preprocessing* yaitu data gambar sapi Bali yang kemudian diambil nilai *mean* dan standar deviasi RGB. Arsitektur implementasi sistem ditunjukkan oleh gambar 3.3 berikut ini:



Gambar 3.3 Arsitektur Implementasi SPK Pemilihan Bibit Unggul Sapi Bali
Sumber: [Perancangan]

Permodelan data yang digunakan dirancang berdasarkan hasil pengambilan data yang telah dilakukan. Data akan dikelompokkan berdasarkan kelompok usia dan jenis kelamin. Pada tabel 3.1 ditunjukkan model data yang digunakan pada sistem.

Tabel 3.1 Tabel Model Data

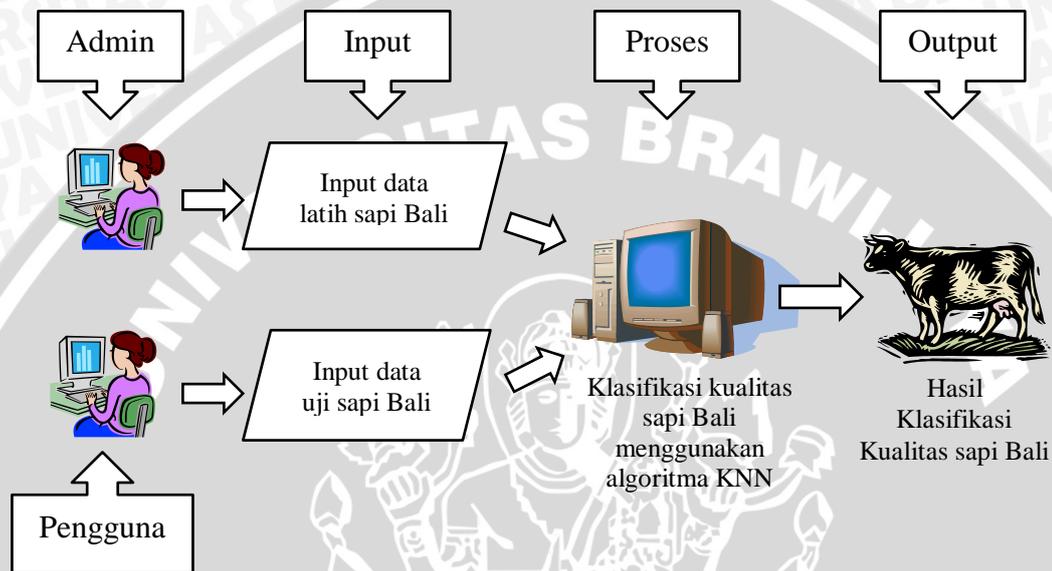
Atribut	Representasi
Mean Red	Angka
Mean Green	Angka
Mean Blue	Angka
Standar Deviasi Red	Angka
Standar Deviasi Green	Angka
Standar Deviasi Blue	Angka

Sumber: [Perancangan]

3.4.3. Arsitektur Program

Sistem memiliki 2 pokok, yaitu bagian admin dan bagian pengguna. Admin dapat melakukan penginputan data training yang akan disimpan dalam

database. Sedangkan pengguna dapat melakukan proses pengujian kepada sistem, dalam hal ini melakukan proses klasifikasi kualitas sapi Bali. Terlebih dahulu pengguna memberikan input berupa gambar sapi Bali. Setelah proses input selesai, sistem akan memproses pengklasifikasian. Kemudian sistem akan memberikan output berupa hasil klasifikasi berdasarkan input yang diberikan pengguna. Gambaran umum ditunjukkan pada gambar 3.3.



Gambar 3.4 Blok Diagram Program
Sumber: [Perancangan]

3.5. Implementasi

Implementasi program klasifikasi kualitas sapi Bali berdasarkan fitur warna kulit menggunakan metode *K-Nearest Neighbor* ini dilakukan dengan mengacu pada perancangan sistem. Pembuatan perangkat lunak dilakukan menggunakan bahasa pemrograman Java. Implementasi program ini meliputi :

1. Pembuatan *user interface* berupa aplikasi dekstop.
2. Penginputan data training oleh admin dan pembacaan data training oleh sistem untuk pembelajaran.
3. Melakukan proses klasifikasi berdasarkan data training yang diberikan.
4. Melakukan proses penentuan kualitas sapi Bali dengan menggunakan algoritma *K-Nearest Neighbor*.
5. Menghasilkan *output* berupa kualitas sapi Bali.

3.6. Pengujian dan Analisis

Pada tahap ini dilakukan pengujian metode *K-Nearest Neighbor* dan sistem yang dibangun yaitu Sistem Pendukung Keputusan Pemilihan Bibit Unggul Sapi Bali Berdasarkan Warna Kulit Menggunakan Metode K-Nearest Neighbor. Berdasarkan batasan masalah yang sudah dikemukakan, maka dataset yang digunakan untuk penelitian ini adalah 90 data gambar sapi bali untuk 2 jenis kelamin yaitu 45 data untuk jantan dan 45 data untuk betina. Tiap jenis kelamin dibagi lagi ke dalam 3 kelas yaitu baik, sedang, dan buruk. Terdapat 15 data di tiap kelasnya. Data yang dipilih untuk data latih dalam penelitian ini adalah 10 data gambar sapi bali yang pertama di tiap kelas. Untuk tiap satu gambar sapi bali dijadikan 3 data latih sehingga keseluruhan data latih yang digunakan ada 90 data latih tiap jenis kelamin. Pengujian yang akan dilakukan antara lain :

1. Pengujian terhadap nilai K. Pengujian menggunakan seluruh data latih terhadap nilai K 3 sampai 30 dengan rentang nilai 3 (contoh: 3, 6, 9, 12, 15, 18, 21, 24, 27, 30). Nilai K terbaik didapatkan dengan cara mencari nilai rata-rata akurasi terbesar tiap K di semua ukuran piksel.
2. Pengujian terhadap ukuran pixel. Pengujian ini dilakukan terhadap ketiga ukuran citra, yaitu 128×128 piksel, 64×64 piksel, dan 32×32 piksel. Ukuran citra terbaik didapatkan dengan cara mencari nilai akurasi dari ketiga ukuran citra pada nilai K terbaik yang didapat pada pengujian sebelumnya.
3. Pengujian terhadap jumlah data latih. Pengujian ini dilakukan terhadap jumlah data latih masing-masing 10, 15, 20, 25, dan 30 tiap kelas. Sehingga jumlah data latih yang digunakan adalah 30, 45, 60, 75 dan 90 tiap jenis kelamin. Pengujian ini menggunakan nilai K terbaik dan ukuran citra terbaik yang didapat dari pengujian sebelumnya. Jumlah data latih terbaik didapatkan dengan cara mencari nilai akurasi terbesar pada kelima jumlah data latih yang dipilih.
4. Pengujian akurasi antara pakar dan sistem. Pengujian ini menggunakan nilai K, ukuran citra, dan jumlah data latih terbaik yang didapat dari pengujian sebelumnya. Klasifikasi dari pakar akan dibandingkan dengan hasil klasifikasi dari sistem, sehingga didapatkan tingkat akurasi yang didapat oleh sistem.

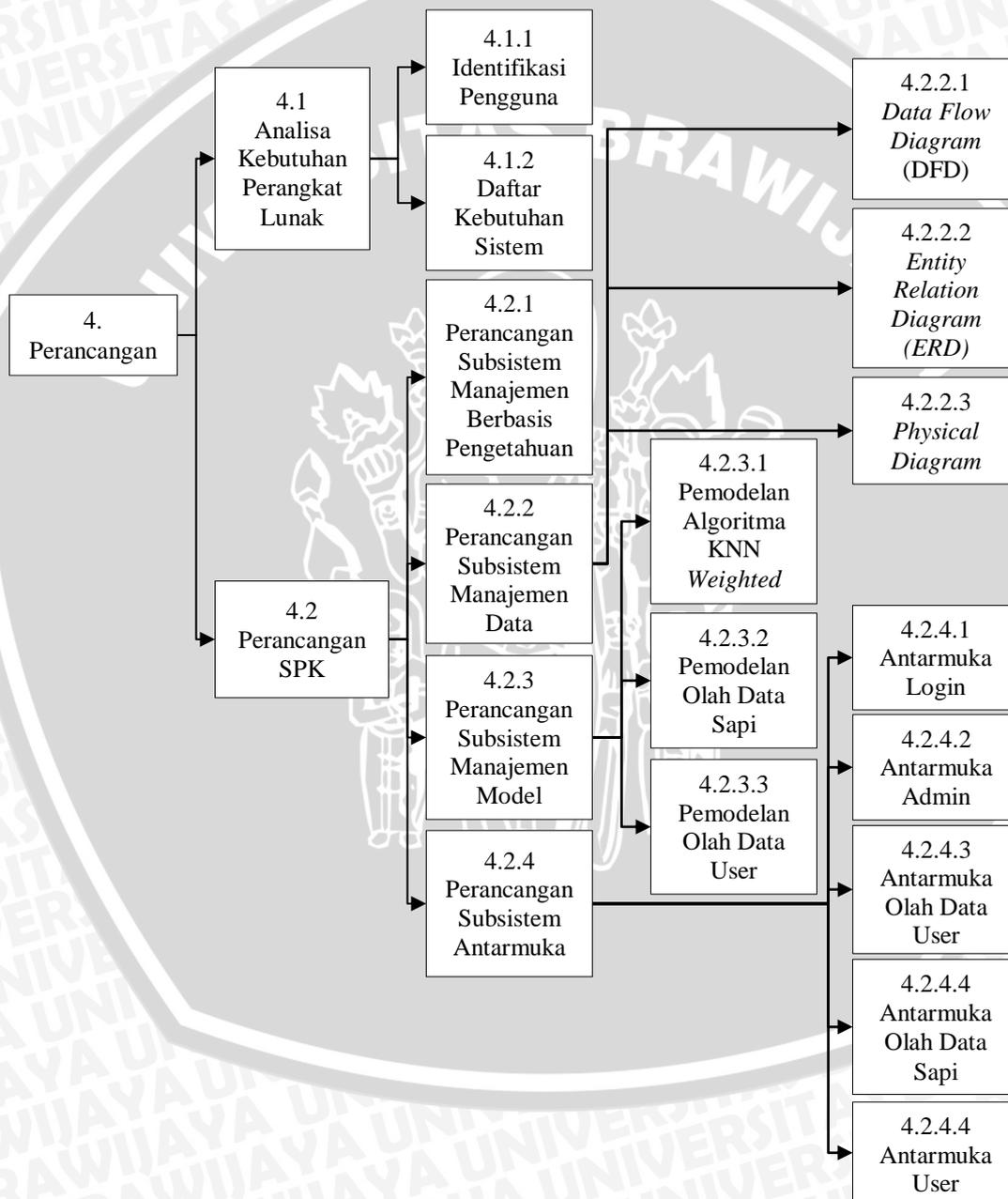
3.7. Pengambilan Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan setelah perancangan, implementasi dan pengujian sistem selesai dilakukan. Kesimpulan didapat dari hasil pengujian dan analisis sistem yang sudah dibangun. Tahap terakhir adalah penulisan saran yang dimaksudkan untuk memperbaiki kesalahan-kesalahan yang terjadi. Saran juga dimaksudkan untuk memberikan pertimbangan atas pengembangan aplikasi selanjutnya.



BAB IV PERANCANGAN

Bab ini menjelaskan mengenai perancangan sistem yang akan dibangun seperti analisa kebutuhan perangkat lunak dan perancangan SPK. Pada gambar 4.1 menunjukkan tentang gambaran umum pohon perancangan.



Gambar 4.1 Pohon Perancangan
Sumber : [Perancangan]



4.1. Analisa Kebutuhan Perangkat Lunak

Sistem pemilihan bibit unggul Sapi Bali merupakan upaya untuk meningkatkan kualitas dari bibit unggul Sapi Bali. Permasalahannya terletak pada metode pemilihan yang peternak lakukan masih sering terjadi kesalahan. Hal tersebut dikarenakan peternak melakukan pemilihan dengan cara melihat secara langsung pola warna kulit sapi Bali. Sistem ini berusaha mengatasi permasalahan tersebut dengan membangun sebuah aplikasi berbasis dekstop yang bisa digunakan oleh peternak di BPTU Sapi Bali dalam menentukan kualitas sapi Bali. Pada analisa kebutuhan ini diawali dengan mengidentifikasi aktor yang terlibat dalam sistem.

4.1.1. Identifikasi Pengguna

Tahap ini bertujuan untuk melakukan identifikasi terhadap pengguna yang akan berinteraksi dengan sistem. Pada penelitian ini terdapat dua pengguna yaitu petugas dan peternak. Petugas selaku *admin* akan melakukan tambah, ubah, dan hapus untuk data Sapi Bali dan data *user* sesuai dengan kebutuhan. Peternak selaku *user* dapat melakukan pengujian skenario uji coba dan bisa mendapatkan tingkat akurasi. Tabel 4.1 menampilkan identifikasi dari kedua pengguna tersebut.

Tabel 4.1 Identifikasi Pengguna

Pengguna	Deskripsi Pengguna	Keterangan
Petugas BPTU Sapi Bali (PTG)	Petugas BPTU Sapi Bali merupakan pengguna sistem yang bertugas untuk mengolah data <i>user</i> dan data Sapi Bali.	<i>Admin</i>
Peternak (PTR)	Peternak merupakan pengguna yang menggunakan sistem pendukung keputusan untuk melakukan skenario uji coba pada data Sapi Bali untuk diketahui tingkat akurasi.	<i>User</i>

Sumber : [Perancangan]

4.1.2. Daftar Kebutuhan Sistem

Daftar kebutuhan sistem ini akan menguraikan tentang kebutuhan fungsional sistem dan pengguna yang terlibat di dalamnya. Daftar kebutuhan sistem ditunjukkan pada Tabel 4.2.

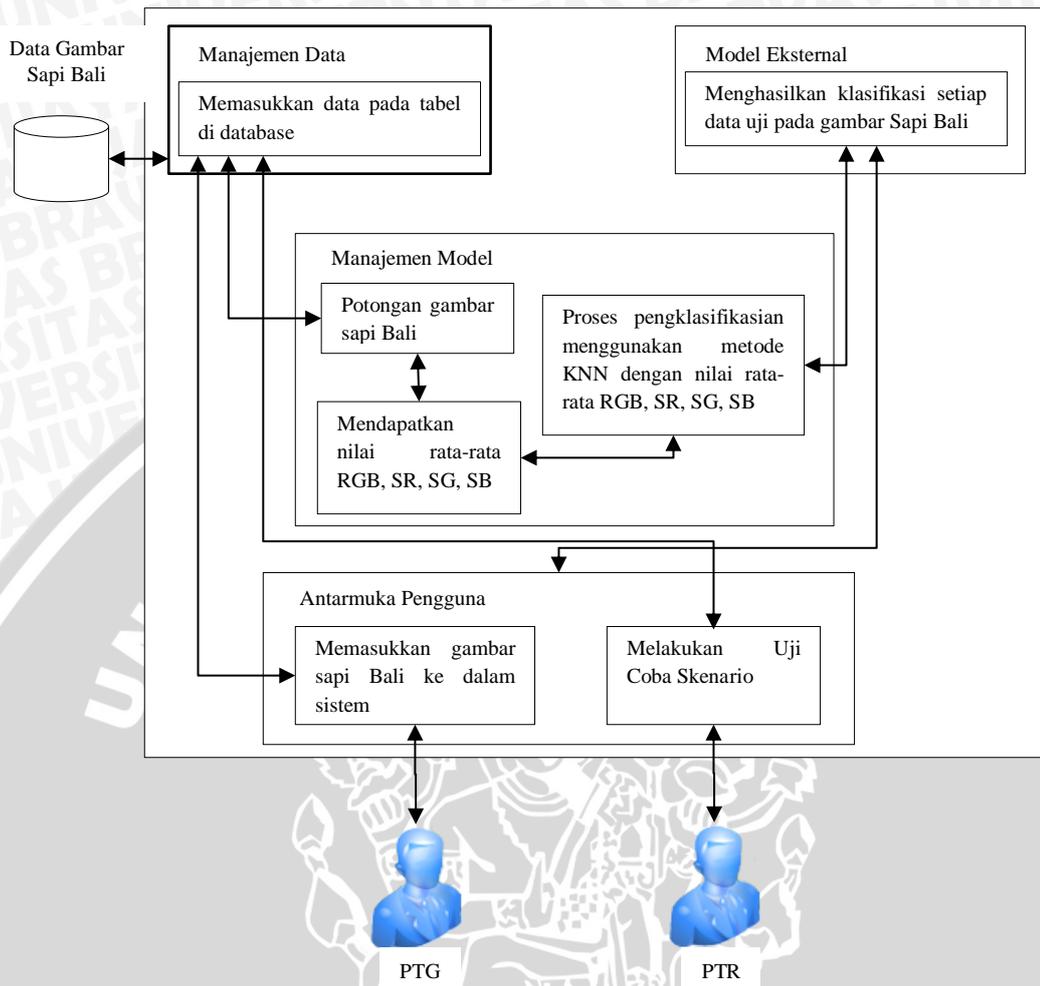
Tabel 4.2 Kebutuhan Fungsional Sistem

Pengguna	Kebutuhan	Nama Use Case
PTG PTR	Sistem menyediakan antarmuka untuk melakukan <i>Login</i>	<i>Login</i>
PTG	Sistem menyediakan antarmuka untuk olah data Sapi Bali	Olah Data Sapi Bali
PTG	Sistem menyediakan menu untuk menambah data Sapi Bali	Tambah Data Sapi Bali
PTG	Sistem menyediakan menu untuk mengubah data Sapi Bali	Ubah Data Sapi Bali
PTG	Sistem menyediakan menu untuk menghapus data Sapi Bali	Hapus Data Sapi Bali
PTG	Sistem menyediakan menu untuk mengcropp Gambar Sapi Bali	<i>Cropp</i> Gambar Sapi Bali
PTG	Sistem menyediakan antarmuka untuk olah data <i>user</i>	Olah Data <i>User</i>
PTG	Sistem menyediakan antarmuka untuk menambah <i>user</i> baru	Tambah Data <i>User</i>
PTG	Sistem menyediakan antarmuka untuk mengubah data <i>user</i>	Ubah Data <i>User</i>
PTG	Sistem menyediakan antarmuka untuk menghapus data <i>user</i>	Hapus Data <i>User</i>
PTR	Sistem menyediakan menu untuk skenario pelatihan data	Skenario Pelatihan Data
PTR	Sistem menyediakan menu untuk menguji data <i>testing</i>	Tambah Data Uji
PTR	Sistem menyediakan menu untuk <i>crop</i>	Crop Data Uji
PTR	Sistem menyediakan menu untuk menampilkan hasil klasifikasi	Hasil Klasifikasi
PTR	Sistem menyediakan antarmuka untuk menampilkan hasil akurasi	Hasil Akurasi
PTG PTR	Sistem menyediakan antarmuka untuk melakukan <i>Logout</i>	<i>Logout</i>

Sumber : [Perancangan]

4.2. Perancangan SPK

Perancangan SPK membahas mengenai subsistem manajemen berbasis pengetahuan, subsistem manajemen data, subsistem manajemen model, dan subsistem antarmuka. Dalam pengambilan keputusan, sistem akan menerapkan metode KNN untuk pengambilan keputusan proses klasifikasi. Basis pengetahuan yang digunakan pada sistem ini berasal dari data pemotongan gambar-gambar sapi Bali yang akan dicari nilai *mean red*, *green*, *blue* (RGB), standart deviasi *red*, *green*, *blue* (SR, SG, dan SB). Arsitektur SPK dapat dilihat pada gambar 4.2.



Gambar 4.2 Arsitektur SPK Pemilihan Sapi Bali
 Sumber: [Perancangan]

4.2.1. Perancangan Subsistem Manajemen Berbasis Pengetahuan

Subsistem berbasis pengetahuan merupakan informasi yang dipakai untuk mendukung pengambilan keputusan. Dalam hal ini berisi mengenai pengetahuan untuk pemilihan bibit unggul sapi Bali. Basis pengetahuan ini didapatkan dari observasi langsung dan wawancara peternak yang berada di BPTU Sapi Bali.

Dari hasil observasi didapatkan 15 data gambar tiap kelas (Baik, Sedang, dan Buruk) untuk jantan dan betina. Tiap kelas dibagi menjadi 10 gambar untuk data latih dan 5 gambar untuk data uji. Tiap gambar akan dibuat menjadi 3 data latih, sehingga didapatkan tiap kelas terdapat 30 data latih. Pada penelitian ini fitur yang digunakan adalah komponen warna *mean* dan standar deviasi RGB (\bar{R} , \bar{G} , \bar{B} , SR, SG, SB) untuk proses klasifikasinya. Fitur tersebut didapat dari proses cropping gambar sapi pada tiga posisi yang berbeda (leher sampai kaki depan,

perut, dan daerah kaki belakang). Citra yang digunakan ada 3 ukuran yaitu 128×128 piksel, 64×64 piksel, dan 32×32 piksel. Kriteria fitur warna kulit sapi Bali dapat dilihat pada tabel 4.3.

Tabel 4.3 Kriteria Fitur Warna Kulit Sapi Bali

Jenis Kelamin	Umur	Klasifikasi	Crop 1	Crop 2	Crop 3
Jantan	2 Tahun	Baik (SB-17)			
			R:68;G:69;B:74 SR:9;SG:10;SB:12	R:81;G:82;B:85 SR:6;SG:6;SB:7	R:93;G:93;B:93 SR:3;SG:4;SB:5
		Sedang (SB-32)			
			R:116;G:86;B:66 SR:12;SG:10;SB:8	R:101;G:71;B:48 SR:17;SG:13;SB:9	R:132;G:100;B:76 SR:11;SG:12;SB:13
		Buruk (SB-52)			
			R:172;G:133;B:103 SR:5;SG:7;SB:8	R:172;G:134;B:105 SR:6;SG:8;SB:8	R:173;G:135;B:107 SR:7;SG:6;SB:7
Betina	2 Tahun	Baik (SB-77)			
			R:159;G:108;B:69 SR:30;SG:23;SB:17	R:155;G:107;B:68 SR:24;SG:19;SB:14	R:162;G:112;B:75 SR:24;SG:19;SB:14
		Sedang (SB-88)			
			R:188;G:129;B:71 SR:21;SG:16;SB:12	R:202;G:143;B:84 SR:10;SG:9;SB:8	R:204;G:144;B:89 SR:16;SG:14;SB:13
		Buruk (SB-112)			
			R:221;G:172;B:117 SR:8;SG:8;SB:8	R:212;G:167;B:114 SR:5;SG:5;SB:5	R:199;G:153;B:104 SR:9;SG:8;SB:7

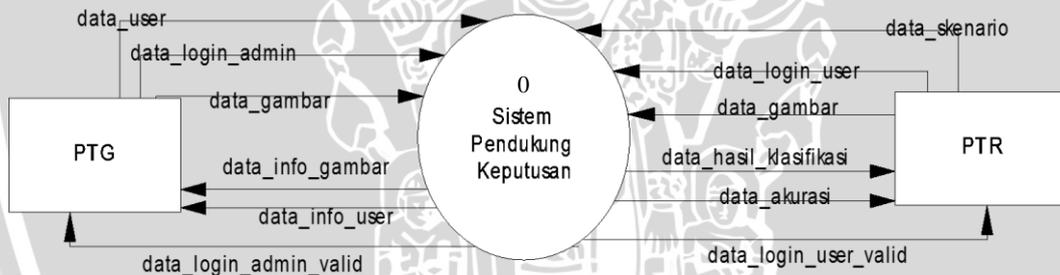
Sumber: [Observasi][Lampiran 3]

4.2.2. Perancangan Subsistem Manajemen Data

Subsistem manajemen data membahas mengenai *Data Flow Diagram* (DFD), *Entity Relation Diagram* (ERD) dan *Physical Diagram*. DFD akan membahas mengenai aliran data pada sistem, ERD akan membahas mengenai hubungan antar data yang terdapat pada database berdasarkan entitas data, dan *Physical Diagram* akan membahas mengenai struktur fisik dari database dalam sistem.

4.2.2.1. Data Flow Diagram (DFD)

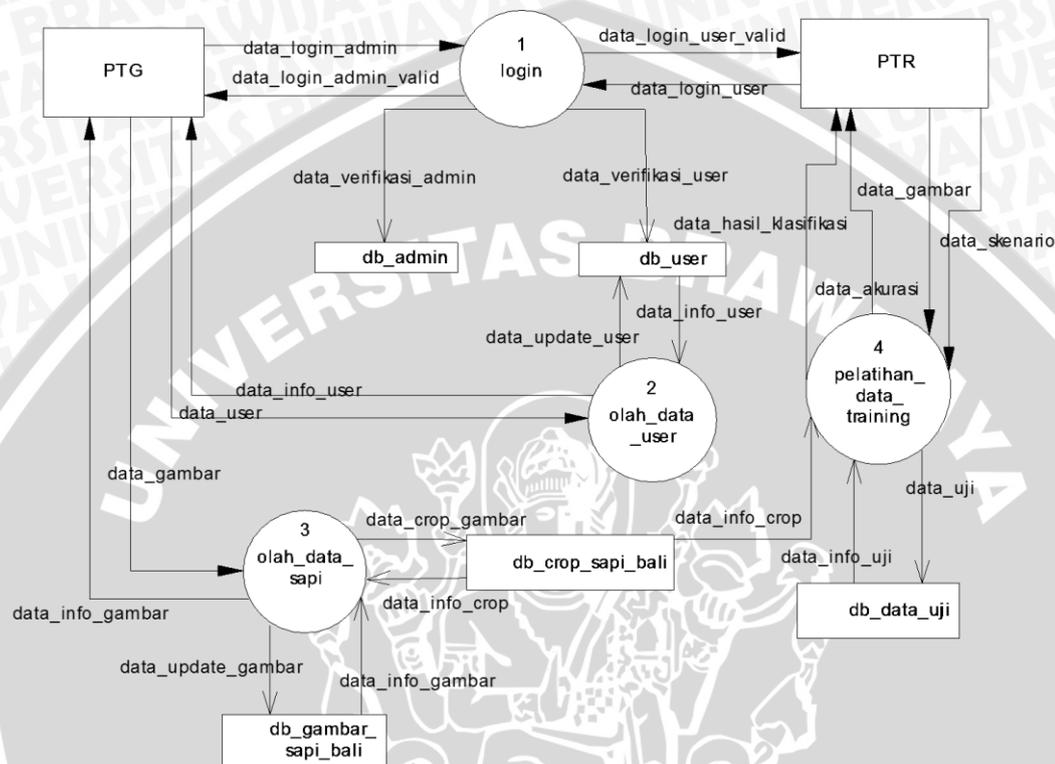
DFD akan membahas mengenai aliran data yang terdapat di dalam sistem. Tujuan dibuatnya DFD adalah untuk memberikan indikasi mengenai bagaimana data ditransformasikan saat data bergerak di dalam sistem. DFD memiliki beberapa tingkatan level. Diagram awal dari DFD yaitu diagram konteks, yang akan ditunjukkan pada gambar 4.3.



Gambar 4.3 Diagram Konteks
Sumber: [Perancangan]

Diagram konteks di atas memiliki 2 pengguna yaitu PTG dan PTR sebagai pengguna sistem. PTG melakukan proses login ke sistem dengan memasukkan *username* dan *password*. Sistem akan memberikan hak akses kepada PTG untuk mengolah data *training* sapi bali dan mengolah data *user*. Sistem juga akan memberikan informasi kepada PTG mengenai semua data latih dan data *user* yang terdapat di dalam sistem. PTR juga melakukan proses login ke sistem dengan memasukkan *username* dan *password*. Sistem akan memberikan hak akses kepada PTR untuk melakukan pelatihan data latih berupa klasifikasi sapi Bali. Sistem akan memberikan informasi berupa hasil klasifikasi sapi Bali dan tingkat akurasi prosesnya.

DFD level 1 merupakan representasi lebih detail dari diagram konteks. Proses Sistem Pendukung Keputusan yang terdapat pada diagram konteks diuraikan menjadi 4 proses utama yaitu login, olah data user, olah data sapi bali, dan pelatihan data training. DFD level 1 dapat dilihat pada gambar 4.4.



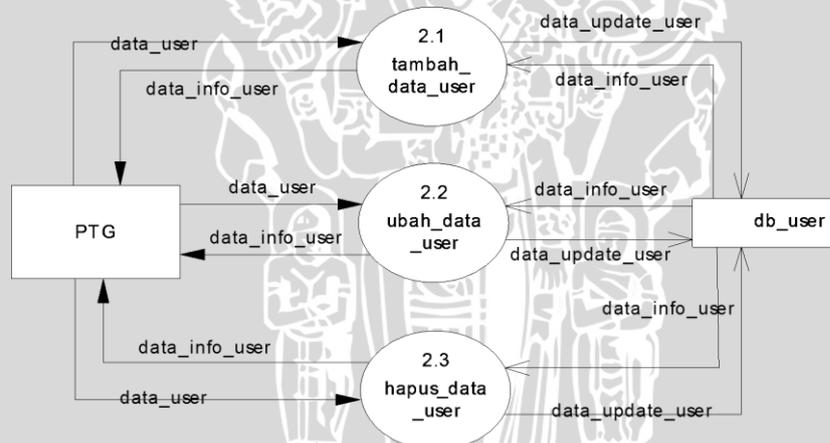
Gambar 4.4 DFD Level 1
Sumber: [Perancangan]

Pada gambar 4.4 terdapat 4 proses utama dari kedua pengguna (PTG dan PTR). Berikut adalah penjelasan dari proses yang terdapat pada DFD level 1:

1. Proses Login : Proses login dilakukan oleh kedua pengguna yaitu PTG dan PTR. Kedua pengguna ini memasukkan username dan password, kemudian sistem akan melakukan verifikasi dengan mengecek username dan password pada database. Sistem akan memberikan informasi hasil verifikasi setelah melakukan pengecekan.
2. Proses Olah Data User : Proses olah data user hanya bisa dilakukan oleh PTG. Pada proses ini PTG memasukkan data *user* sesuai yang dibutuhkan kemudian melakukan pembaharuan pada database. Sistem akan menampilkan informasi dari proses yang dilakukan PTG.

3. Proses Olah Data Sapi : Proses olah data sapi hanya bisa dilakukan oleh PTG. Pada proses ini PTG memasukkan data sapi sesuai yang dibutuhkan kemudian melakukan pembaharuan pada database. Sistem akan menampilkan informasi dari proses yang dilakukan PTG.
4. Proses Pelatihan Data Latih : Proses pelatihan data latih dilakukan oleh PTR. Pada proses ini PTR akan memasukkan data sapi yang akan diklasifikasi sebagai data uji. Kemudian sistem akan mengklasifikasi data uji tersebut dengan menggunakan metode KNN. Sistem juga akan mengukur tingkat akurasi proses klasifikasi. Sistem akan menampilkan informasi hasil klasifikasi dan tingkat akurasi dari proses tersebut.

DFD level 2 merupakan representasi dari DFD level 1. Semua proses pada DFD level 1 akan direpresentasikan pada DFD level 2. DFD level 2 yang akan dijabarkan lebih detail adalah proses olah data *user*. DFD level 2 olah data user dapat dilihat pada gambar 4.5.



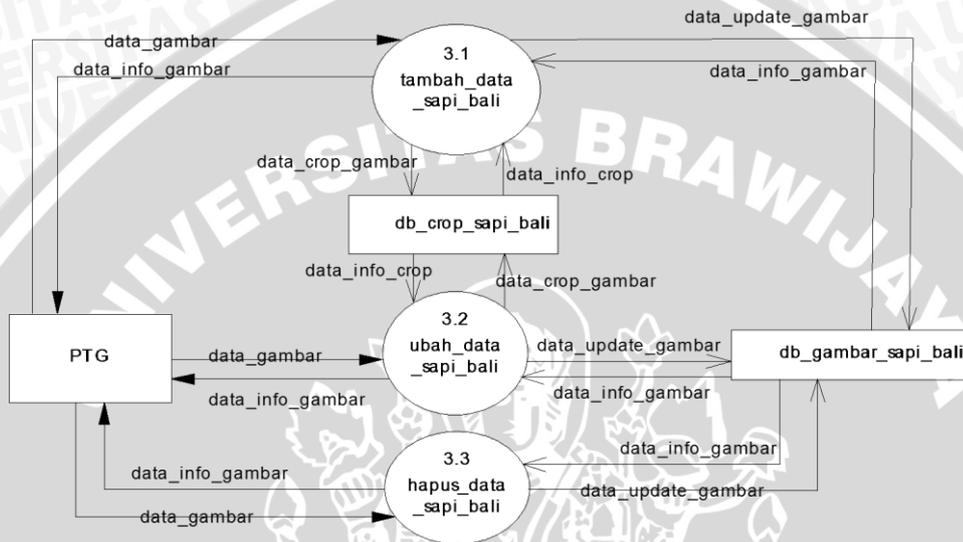
Gambar 4.5 DFD Level 2 Proses Olah Data *User*
Sumber: [Perancangan]

Berikut adalah penjelasan dari DFD level 2 dari proses olah data user:

1. Proses tambah data *user* dilakukan oleh PTG. PTG diminta untuk memasukkan data *user* kemudian sistem akan menyimpannya ke dalam database.
2. Pada proses ubah data *user*, PTG diminta untuk memilih id *user* yang akan diedit dan melakukan edit data *user*. Sistem akan melakukan pembaharuan di dalam database dan menyampaikan hasil dari pembaharuan tersebut.

3. Proses hapus data *user*, PTG diminta untuk memasukkan data id *user* yang akan dihapus. Sistem akan menghapus data yang diinginkan PTG pada database.

DFD level 2 yang akan dijelaskan berikutnya adalah proses olah data sapi. Pada DFD ini terdapat proses tambah data sapi bali, ubah data sapi bali, dan hapus data sapi bali. DFD level 2 proses olah data sapi ditunjukkan pada gambar 4.6.

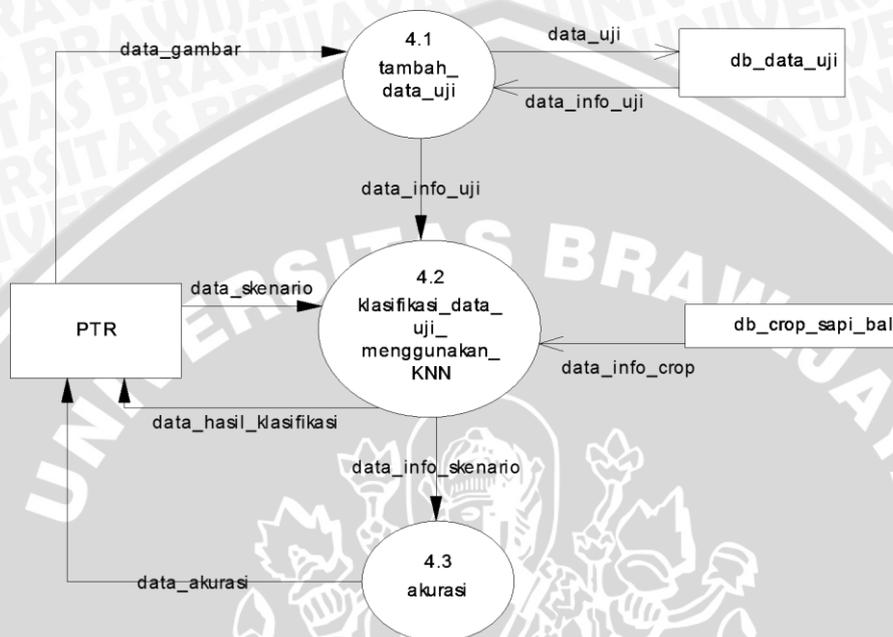


Gambar 4.6 DFD Level 2 Proses Olah Data Sapi
Sumber: [Perancangan]

Berikut adalah penjelasan dari DFD level 2 proses olah data sapi:

1. Proses tambah data sapi bali dilakukan oleh PTG. PTG diminta untuk memasukkan data sapi bali. Sistem akan mengambil gambar sapi bali kemudian meminta PTG untuk memilih bagian gambar yang akan dipotong. Setelah itu sistem akan mengirimkan data hasil pemotongan gambar dan akan memasukkan data tersebut ke dalam database.
2. Proses ubah data sapi bali. Pada proses ini sistem akan menampilkan data lama dari sapi dan PTG diminta untuk memasukkan data sapi bali yang baru. Sistem akan melakukan pembaharuan di dalam database dan akan menampilkan hasil dari pembaharuan tersebut.
3. Proses hapus data sapi. Pada proses ini PTG diminta untuk memasukkan data id sapi yang akan dihapus. Sistem akan menghapus data sapi yang diinginkan PTG pada database.

DFD level 2 yang terakhir adalah proses pelatihan data training. Pada DFD ini terdapat 3 proses yaitu tambah data uji, klasifikasi data uji menggunakan KNN, dan perhitungan akurasi. DFD level 2 proses pelatihan data training ditunjukkan pada gambar 4.7.



Gambar 4.7 DFD Level 2 Proses Pelatihan Data Training
Sumber: [Perancangan]

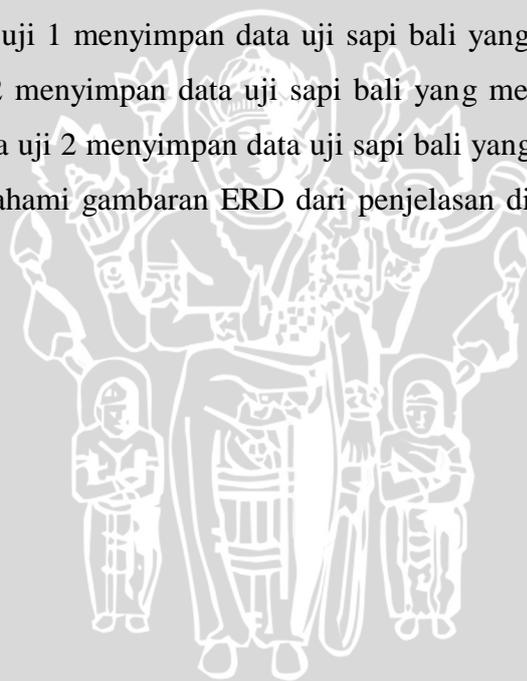
Proses tambah data uji akan dilakukan oleh PTR. PTR diminta untuk memasukkan data uji berupa gambar sapi bali yang belum diketahui kelasnya. Sistem akan mengambil data gambar sapi bali kemudian PTR diminta untuk melakukan pemotongan gambar di bagian tertentu. Sistem kemudian akan melakukan proses klasifikasi data uji tersebut dengan melakukan pelatihan terhadap data training yang berada di dalam database. Sistem akan memberikan informasi mengenai hasil dari klasifikasi dan tingkat akurasi dari klasifikasi tersebut.

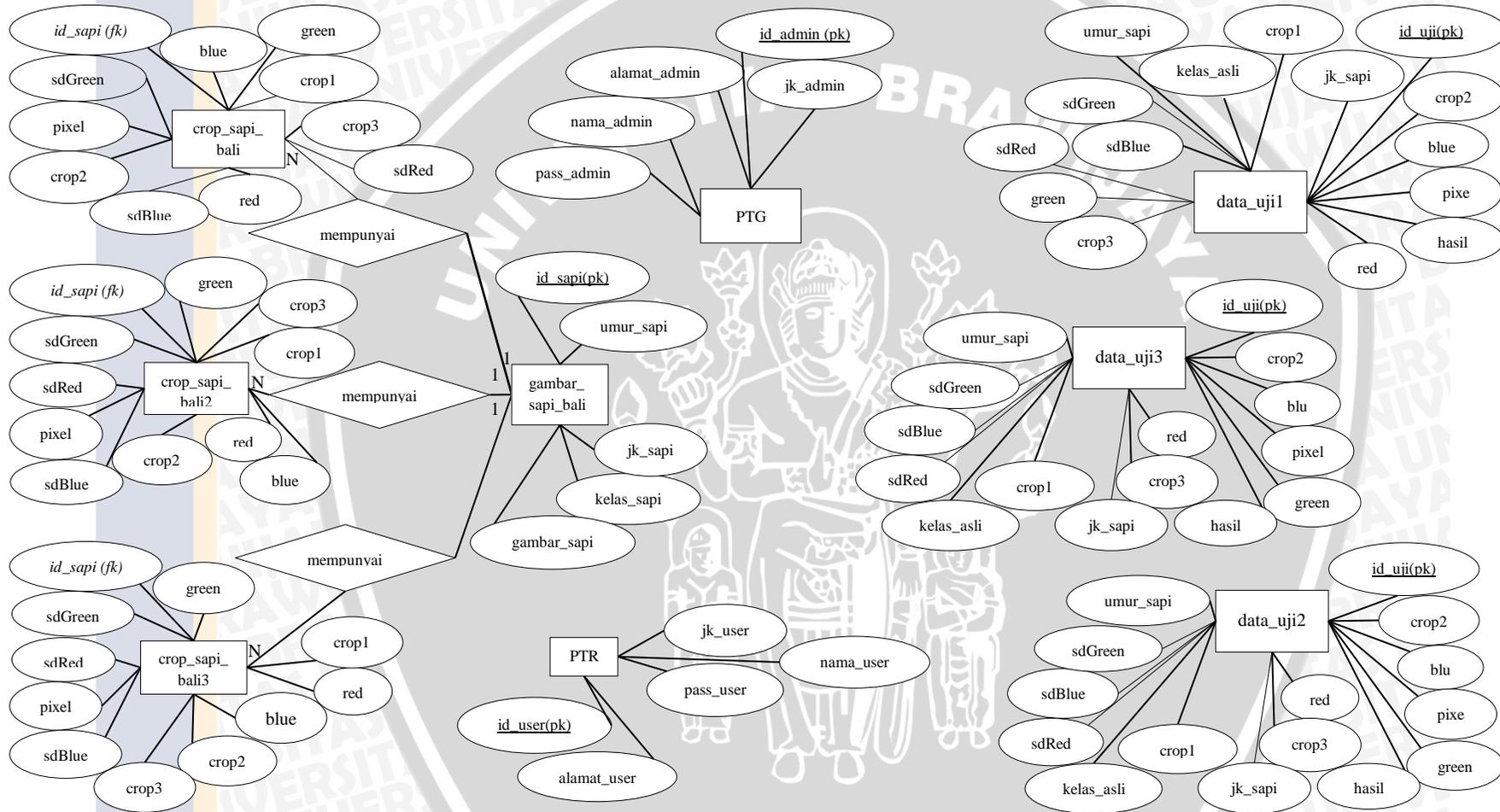
4.2.2.2. Entity Relation Diagram (ERD)

ERD merupakan sebuah diagram yang akan membahas mengenai relasi antar data yang terdapat pada database berdasarkan entitas data. Terdapat 2 entitas utama yang berlaku sebagai pengguna dalam sistem ini. Kedua entitas tersebut adalah PTG dan PTR. PTG dapat melakukan penambahan data gambar sapi bali yang akan dijadikan sebagai data latih dan juga mengubahnya jika terjadi

perubahan. Informasi yang terdapat pada entitas gambar sapi bali adalah informasi umum dari data sapi. Informasi mengenai warna dan komponen warna terdapat pada entitas crop sapi bali. Sehingga hubungan antara entitas gambar sapi bali dengan crop sapi bali adalah mempunyai. Gambar sapi bali mempunyai crop sapi bali 1, crop sapi bali 2, dan crop sapi bali 3 sebagai entitas pembantunya, dimana id sapi pada crop sapi bali merupakan foreign key dari id sapi bali pada gambar sapi bali.

Entitas PTR mempunyai hak untuk memberikan perlakuan terhadap data uji. Perlakuan yang dilakukan oleh PTR adalah memilih data gambar sapi bali yang akan akan diujikan. Dalam database terdapat 3 tabel data uji yang masing-masing merepresentasikan nilai pixel yang digunakan dalam pemotongan gambar sapi bali. Entitas data uji 1 menyimpan data uji sapi bali yang mempunyai pixel 128. Entitas data uji 2 menyimpan data uji sapi bali yang mempunyai pixel 64. Sedangkan Entitas data uji 2 menyimpan data uji sapi bali yang mempunyai pixel 32. Untuk lebih memahami gambaran ERD dari penjelasan di atas dapat dilihat pada gambar 4.8.

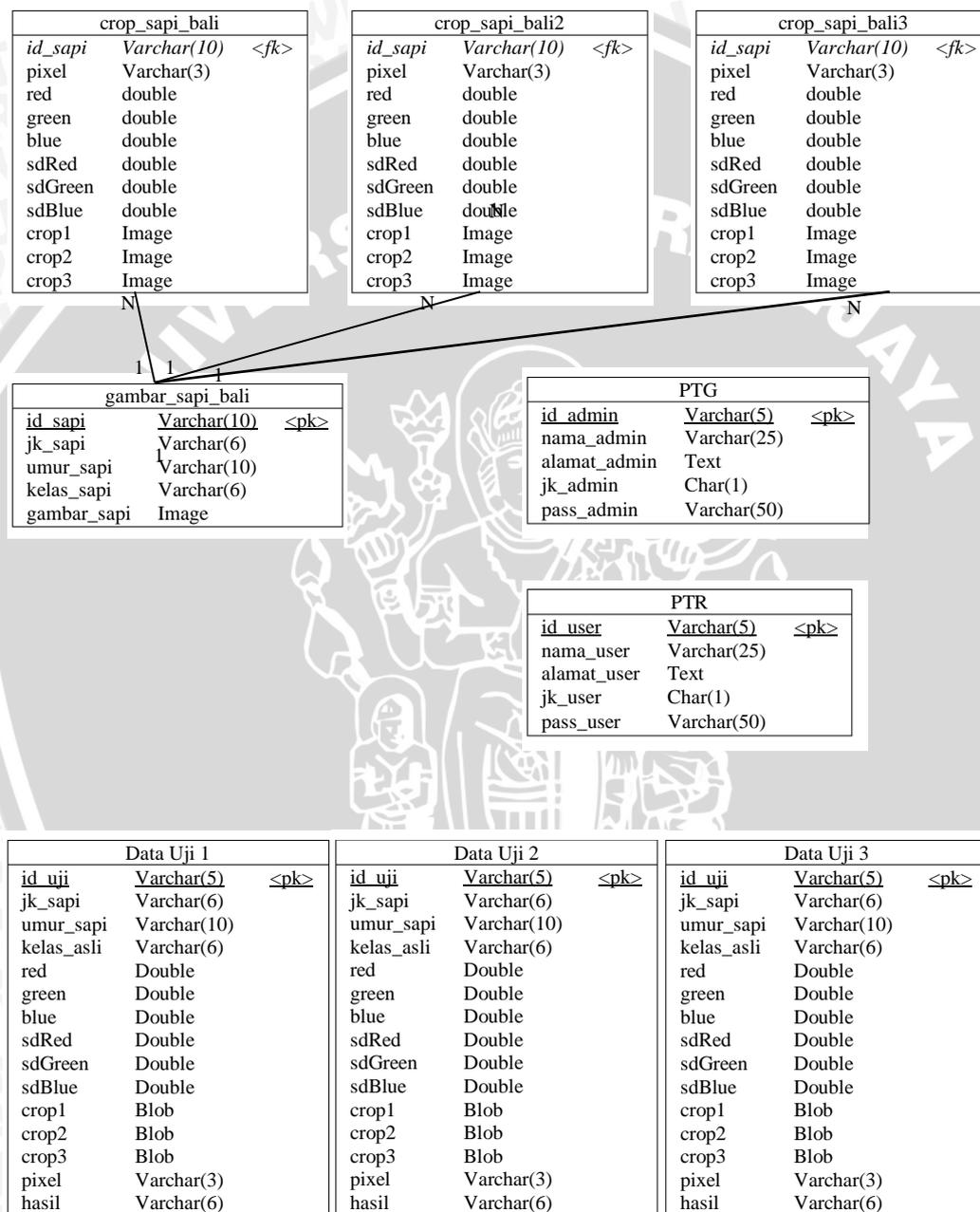




Gambar 4.8 ERD SPK Pemilihan Bibit Unggul Sapi Bali
 Sumber: [Perancangan]

4.2.2.3. Physical Diagram

Sistem pada penelitian ini menggunakan MySQL sebagai server DBMS. Physical Diagram dapat membantu pembuatan database sistem karena merupakan sebuah struktur fisik dari database. Struktur physical diagram yang dilakukan oleh PTG ditunjukkan pada gambar 4.9.



Gambar 4.9 Physical Diagram Admin Pemilihan Bibit Unggul Sapi Bali
 Sumber: [Perancangan]

Struktur dari masing-masing tabel pada *Physical Data Model* SPK pemilihan bibit unggul Sapi Bali yang dilakukan oleh PTG adalah sebagai berikut:

1. Tabel Admin

Tabel *admin* digunakan untuk menyimpan informasi seperti id *admin*, nama *admin*, alamat *admin*, jenis kelamin *admin*, dan *password admin*.

Tabel 4.4 Struktur tabel user

No	Nama Field	Tipe	Lebar	Keterangan
1	<u>id_admin</u>	Varchar	5	Primary Key
2	nama_admin	Varchar	25	-
3	alamat_admin	Text	-	-
4	jk_admin	Char	1	-
5	pass_admin	Varchar	50	-

Sumber: [Perancangan]

2. Tabel Gambar Sapi Bali

Tabel data Sapi Bali untuk menyimpan data dan informasi dari gambar Sapi Bali yang disimpan. Tabel ini menyimpan data id Sapi, JK Sapi, Umur Sapi, kelas dan gambar Sapi.

Tabel 4.5 Struktur tabel data Sapi Bali

No	Nama Field	Tipe	Lebar	Keterangan
1	<u>id_Sapi</u>	Varchar	10	Primary Key
2	jk_Sapi	Varchar	6	-
3	umur_Sapi	Varchar	10	-
4	kelas_sapi	Varchar	6	-
5	gambar_Sapi	Image	-	-

Sumber: [Perancangan]

3. Tabel Crop Sapi Bali

Tabel crop Sapi Bali untuk menyimpan data dan informasi dari hasil cropping gambar Sapi Bali yang disimpan. Tabel ini menyimpan data id Sapi, Pixel dengan nilai 128, Red, Green, Blue, Standart Deviasi (Red, Green, Bue), crop1, crop2, dan crop3.

Tabel 4.6 Struktur tabel crop Sapi Bali

No	Nama Field	Tipe	Lebar	Keterangan
1	<u>id_sapi</u>	Varchar	10	Foreign Key dari tabel Gambar Sapi Bali
2	pixel	varchar	3	-
3	red	Double	-	-
4	green	Double	-	-
5	blue	Double	-	-

6	sdRed	Double	-	-
7	sdGreen	Double	-	-
8	sdBlue	Double	-	-
9	crop1	Image	-	-
10	crop2	Image	-	-
11	crop3	Image	-	-

Sumber: [Perancangan]

4. Tabel Crop Sapi Bali 2

Tabel crop Sapi Bali 2 untuk menyimpan data dan informasi dari hasil cropping gambar Sapi Bali yang disimpan. Tabel ini menyimpan data id Sapi, Pixel dengan nilai 64, Red, Green, Blue, Standart Deviasi (Red, Green, Blue), crop1, crop2, dan crop3.

Tabel 4.7 Struktur tabel crop Sapi Bali 2

No	Nama Field	Type	Lebar	Keterangan
1	<i>id_sapi</i>	Varchar	10	Foreign Key dari tabel Gambar Sapi Bali
2	pixel	varchar	3	-
3	red	Double	-	-
4	green	Double	-	-
5	blue	Double	-	-
6	sdRed	Double	-	-
7	sdGreen	Double	-	-
8	sdBlue	Double	-	-
9	crop1	Image	-	-
10	crop2	Image	-	-
11	crop3	Image	-	-

Sumber: [Perancangan]

5. Tabel Crop Sapi Bali 3

Tabel crop Sapi Bali 3 untuk menyimpan data dan informasi dari hasil cropping gambar Sapi Bali yang disimpan. Tabel ini menyimpan data id Sapi, Pixel dengan nilai 32, Red, Green, Blue, Standart Deviasi (Red, Green, Blue), crop1, crop2, dan crop3.

Tabel 4.8 Struktur tabel crop Sapi Bali 3

No	Nama Field	Type	Lebar	Keterangan
1	<i>id_sapi</i>	Varchar	10	Foreign Key dari tabel Gambar Sapi Bali
2	pixel	varchar	3	-
3	red	Double	-	-
4	green	Double	-	-
5	blue	Double	-	-

6	sdRed	Double	-	-
7	sdGreen	Double	-	-
8	sdBlue	Double	-	-
9	crop1	Image	-	-
10	crop2	Image	-	-
11	crop3	Image	-	-

Sumber: [Perancangan]

6. Tabel User

Tabel *user* digunakan untuk menyimpan informasi yang terdapat pada *user* seperti id *user*, nama *user*, alamat *user*, jenis kelamin *user*, dan *password*.

Tabel 4.9 Struktur tabel user

No	Nama Field	Tipe	Lebar	Keterangan
1	<u>id_user</u>	Varchar	5	Primary Key
2	nama_user	Varchar	25	-
3	alamat_user	Text	-	-
4	jk_user	Char	1	-
5	pass_user	Varchar	50	-

Sumber: [Perancangan]

7. Tabel Data Uji 1

Tabel data uji 1 untuk menyimpan data dan informasi dari hasil olah data uji yang dimasukkan user. Tabel ini menyimpan data id Sapi, jenis kelamin sapi, umur sapi, kelas asli, Red, Green, Blue, Standart Deviasi Red, Standart Deviasi Green, Standart Deviasi Blue, crop1, crop2, crop3, piksel dengan nilai 128, nilai K, dan hasil.

Tabel 4.10 Struktur tabel uji crop Sapi Bali

No	Nama Field	Tipe	Lebar	Keterangan
1	<u>id_uji</u>	Varchar	10	Primary Key
2	jk_sapi	Varchar	6	-
3	umur_sapi	Varchar	10	-
4	kelas_asli	Varchar	6	-
5	red	Double	-	-
6	green	Double	-	-
7	blue	Double	-	-
8	sdRed	Double	-	-
9	sdGreen	Double	-	-
10	sdBlue	Double	-	-
11	crop1	Image	-	-
12	crop2	Image	-	-
13	crop3	Image	-	-
14	pixel	Varchar	3	-

15	hasil	Varchar	6	-
----	-------	---------	---	---

Sumber: [Perancangan]

8. Tabel Data Uji 2

Tabel data uji 2 untuk menyimpan data dan informasi dari hasil olah data uji yang dimasukkan user. Tabel ini menyimpan data id Sapi, jenis kelamin sapi, umur sapi, kelas asli, Red, Green, Blue, Standart Deviasi Red, Standart Deviasi Green, Standart Deviasi Blue, crop1, crop2, crop3, piksel dengan nilai 64, nilai K, dan hasil.

Tabel 4.11 Struktur tabel uji crop Sapi Bali 2

No	Nama Field	Tipe	Lebar	Keterangan
1	id_uji	Varchar	10	Primary Key
2	jk_sapi	Varchar	6	-
3	umur_sapi	Varchar	10	-
4	kelas_asli	Varchar	6	-
5	red	Double	-	-
6	green	Double	-	-
7	blue	Double	-	-
8	sdRed	Double	-	-
9	sdGreen	Double	-	-
10	sdBlue	Double	-	-
11	crop1	Image	-	-
12	crop2	Image	-	-
13	crop3	Image	-	-
14	pixel	Varchar	3	-
15	hasil	Varchar	6	-

Sumber: [Perancangan]

9. Tabel Data Uji 3

Tabel data uji 1 untuk menyimpan data dan informasi dari hasil olah data uji yang dimasukkan user. Tabel ini menyimpan data id Sapi, jenis kelamin sapi, umur sapi, kelas asli, Red, Green, Blue, Standart Deviasi Red, Standart Deviasi Green, Standart Deviasi Blue, crop1, crop2, crop3, piksel dengan nilai 32, nilai K, dan hasil.

Tabel 4.12 Struktur tabel uji crop Sapi Bali 3

No	Nama Field	Tipe	Lebar	Keterangan
1	id_uji	Varchar	10	Primary Key
2	jk_sapi	Varchar	6	-
3	umur_sapi	Varchar	10	-
4	kelas_asli	Varchar	6	-
5	red	Double	-	-

6	green	Double	-	-
7	blue	Double	-	-
8	sdRed	Double	-	-
9	sdGreen	Double	-	-
10	sdBlue	Double	-	-
11	crop1	Image	-	-
12	crop2	Image	-	-
13	crop3	Image	-	-
14	pixel	Varchar	3	-
15	hasil	Varchar	6	-

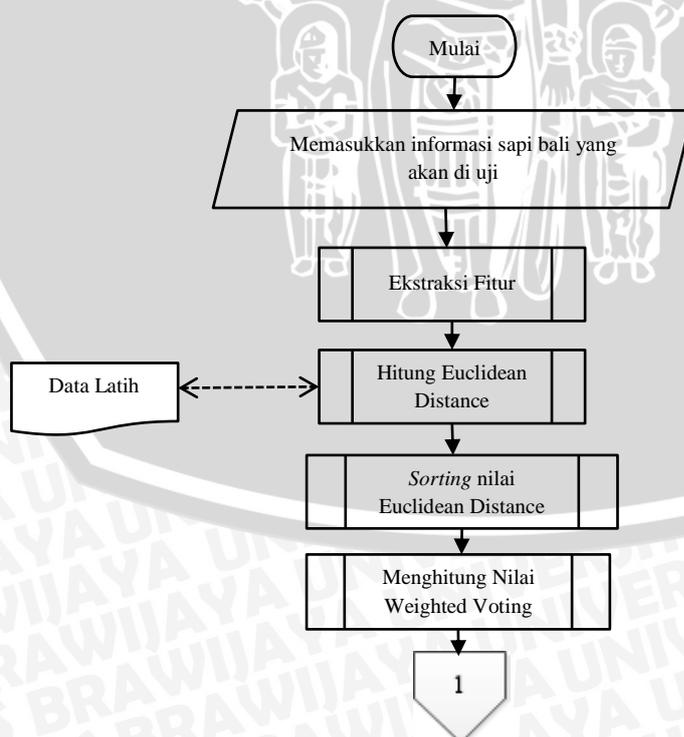
Sumber: [Perancangan]

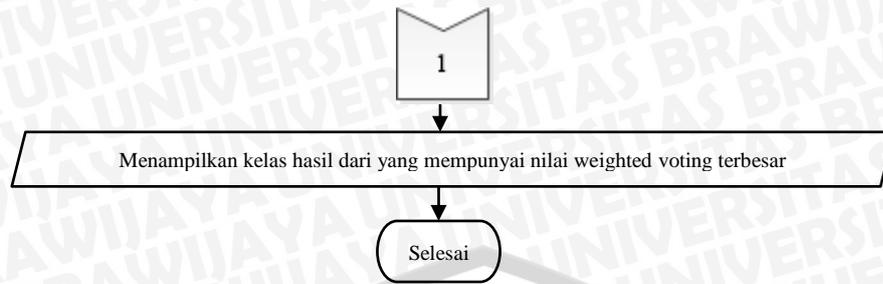
4.2.3. Perancangan Subsistem Manajemen Model

Subsistem manajemen model yang dilakukan meliputi olah data *user*, olah data sapi, dan penggunaan algoritma KNN *Weighted* untuk proses pelatihan fitur. Fitur yang digunakan adalah rata-rata RGB dan rata-rata standar deviasi RGB yang didapatkan dari proses *cropping* gambar sapi bali.

4.2.3.1. Pemodelan Algoritma KNN *Weighted*

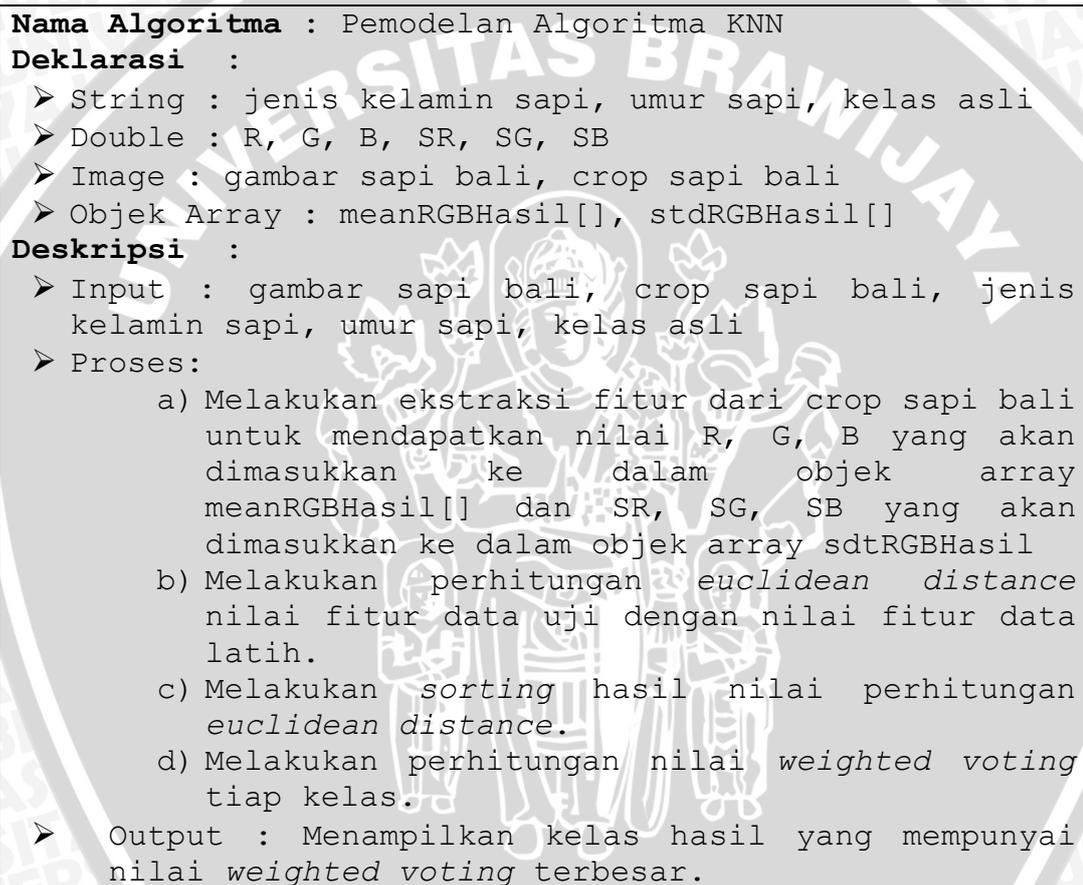
Pada pemodelan algoritma KNN ini akan dilakukan proses pencarian jarak terdekat data uji dengan data latih sebanyak *K*. Diagram alir penggunaan algoritma KNN secara umum dapat dilihat pada gambar 4.10.





Gambar 4.10 Rancangan Algoritma KNN Weighted
Sumber: [Perancangan]

Pseudocode pemodelan algoritma KNN sebagai berikut :



Gambar 4.11 Pseudocode Algoritma Perhitungan Jarak Euclidean
Sumber: [Perancangan]

Pemahaman algoritma KNN dapat dibantu dengan contoh perhitungan. Contoh perhitungan manual dengan algoritma KNN pada kasus pemilihan bibit unggul Sapi Bali diawali dengan menyiapkan data latih. Data latih pada contoh perhitungan ini sebanyak 15 data yang terbagi kedalam tiga kelas (Baik, Sedang dan Buruk) dapat dilihat pada tabel 4.13.

Tabel 4.13 Tabel Data Latih Sapi Bali Jantan

Data Citra ke i	Id Gambar	Fitur ke r						Kelas	Keterangan
		\bar{R}	\bar{G}	\bar{B}	SR	SG	SB		
1	SB-1	141,43	140,74	142,91	5,63	5,95	6,1	Baik	Diambil 5 data sapi baik dari 30 data latih sapi bali jantan baik
2	SB-3	154,57	145,9	150,01	5,66	4,45	3,84	Baik	
3	SB-5	145,55	137,31	140,6	6,05	5,45	4,99	Baik	
4	SB-7	145,86	135,4	136,92	3,99	3,54	3,09	Baik	
5	SB-9	157,31	162,46	186,7	7,81	7,17	5,86	Baik	
6	SB-23	132,68	111,17	102,31	7,78	6,16	5,55	Sedang	Diambil 5 data sapi sedang dari 30 data latih sapi bali jantan sedang
7	SB-25	150,62	126,96	112,09	8,27	6,62	5,82	Sedang	
8	SB-27	136,23	114,35	102,93	4,92	4,04	3,14	Sedang	
9	SB-30	119,11	90,45	70,73	5,47	4,94	4,98	Sedang	
10	SB-39	149,31	126,97	112,51	8	7,65	7,24	Sedang	
11	SB-41	224,94	160,26	100,29	3,58	3,61	3,68	Buruk	Diambil 5 data sapi buruk dari 30 data latih sapi bali jantan buruk
12	SB-43	229,73	165,38	110,41	4,36	3,91	4,2	Buruk	
13	SB-47	192,66	143,57	95,58	4,31	3,87	2,93	Buruk	
14	SB-49	192,86	143,41	96,9	4,49	4,48	4,21	Buruk	
15	SB-58	168,23	126,61	92,53	9,03	7,89	6,78	Buruk	

Sumber: [Perancangan][Lampiran 5]

Data latih yang terdapat pada tabel 4.13 merupakan vektor x_i dan contoh data latih yang diambil dari data sapi bali jantan. Tiap kelas sapi terdapat 5 contoh data. Sedangkan untuk data uji yang merupakan vektor x_j akan diambil juga dari data sapi jantan. Berdasarkan persamaan (2-4) dan tabel 2.4 didapatkan nilai fitur \bar{R} , \bar{G} , \bar{B} . Sedangkan untuk nilai SR, SG, dan SB didapatkan dari perhitungan menggunakan persamaan (2-3). Data uji dapat dilihat pada tabel 4.14.

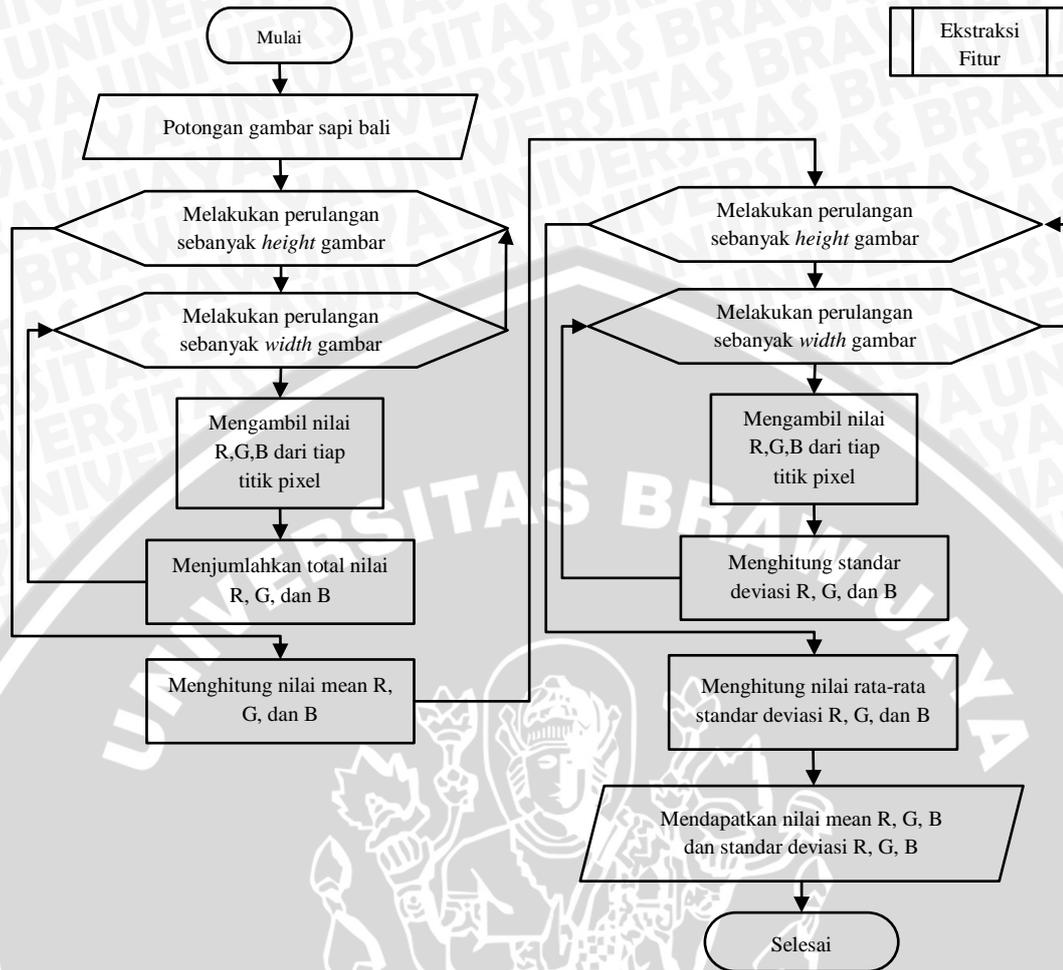
Tabel 4.14 Tabel Data Uji

Citra ke j	Fitur ke r						Kelas
	\bar{R}	\bar{G}	\bar{B}	SR	SG	SB	
1	106,53	81,09	64,99	5,45	4,74	4,36	?

Sumber: [Perancangan]

Ekstraksi Fitur

Ekstraksi fitur adalah tahapan yang dilakukan sebelum sistem melakukan proses perhitungan algoritma KNN. Hasil perhitungan pada tabel 2.4 didasari diagram alir ekstraksi fitur yang dapat dilihat pada gambar 4.12.



Gambar 4.12 Diagram Alir Ekstraksi Fitur
 Sumber: [Perancangan]

Pseudocode ekstraksi fitur adalah sebagai berikut:

Nama Algoritma : Ekstraksi Fitur
Deklarasi :
 ➤ Double array : meanRGB, rgbMeanStd
 ➤ Double : rcount, gcount, bcount, rrr, ggg, bbb
 ➤ Integer : count
 ➤ Image : gambar
Deskripsi :
 ➤ Input : gambar
 ➤ Proses:
 a) Melakukan perulangan sebanyak *height* gambar.
 b) Melakukan perulangan sebanyak *width* gambar.
 c) Mengambil nilai R,G,B tiap pixel
 d) Menghitung mean R, G,dan B
 e) Melakukan perulangan sebanyak *height* gambar.
 f) Melakukan perulangan sebanyak *width* gambar.
 g) Mengambil nilai R,G,B tiap pixel

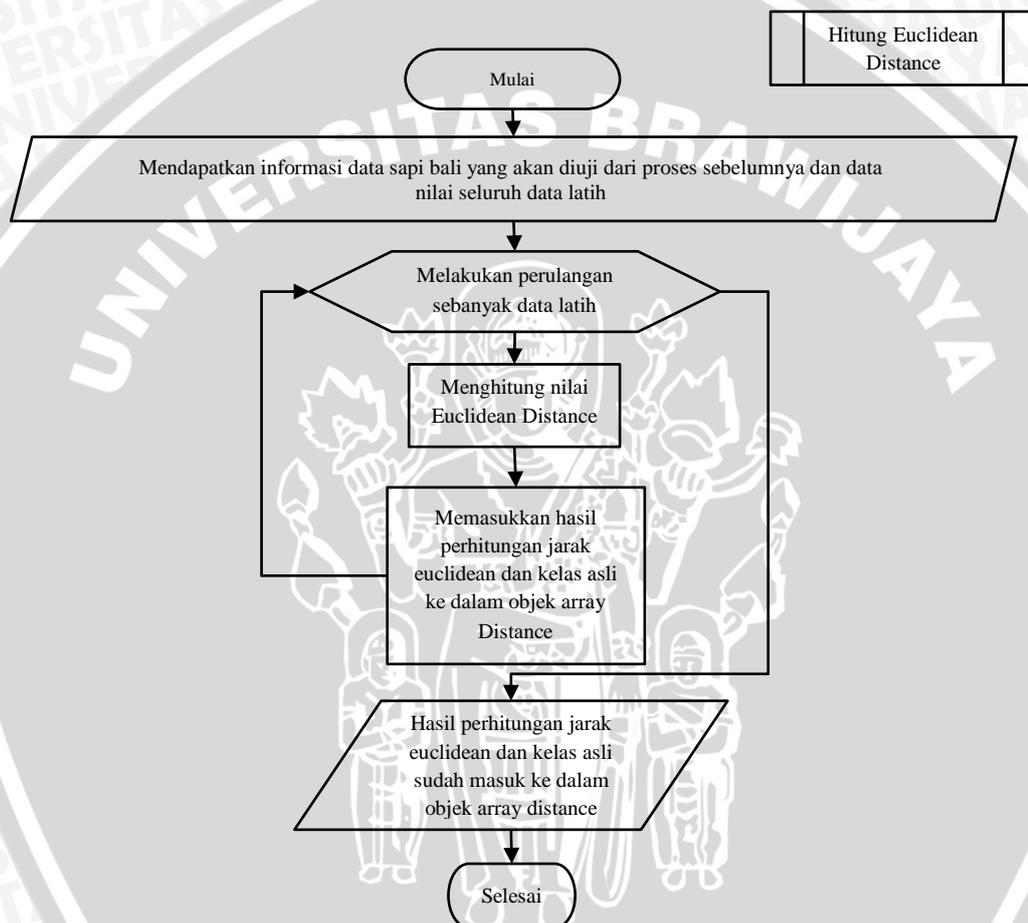
h) Menghitung standar deviasi R,G, dan B
 ➤ Output : hasil perhitungan meanRGB[] dan rgbMeanStd[]

Gambar 4.13 Pseudocode Ekstraksi Fitur

Sumber : [Perancangan]

Perhitungan Algoritma KNN

Tahap awal dari algoritma KNN adalah mencari jarak terdekat data latih dari data uji dengan menggunakan persamaan (2-1). Diagram alir untuk mencari nilai euclidean dapat dilihat pada gambar 4.14.



Gambar 4.14 Diagram Alir Menghitung Jarak Euclidean

Sumber: [Perancangan]

Pseudocode algoritma proses perhitungan jarak *euclidean* sebagai berikut :

Nama Algoritma : Menghitung Jarak Euclidean
Deklarasi :
 ➤ Double : jarak euclidean
 ➤ Integer : jumlah data latih
 ➤ String : jenis kelamin sapi, umur sapi, kelas sapi
 ➤ Objek Array: distance[[[]], meanRGBHasil[], stdRGBHasil[], dataLatih[]]
Deskripsi :

- Input : meanRGBHasil[], stdRGBHasil[], jumlah data latih
- Proses:
 - a) Melakukan perulangan sebanyak jumlah data latih.
 - b) Jarak euclidean = menghitung nilai kedekatan fitur R, G, B, SR, SG, SB yang ada pada objek array dataLatih[] dengan fitur-fitur data uji yang ada pada objek array meanRGBHasil[] dan stdRGBHasil[].
 - c) Memasukkan hasil perhitungan jarak euclidean dan kelas sapi pada objek array distance[][]
- Output : hasil perhitungan jarak euclidean dan kelas sapi sudah masuk pada objek array distance[][]

Gambar 4.15 Pseudocode Algoritma Perhitungan Jarak Euclidean
Sumber: [Perancangan]

Contoh perhitungan untuk mencari jarak antara data latih citra ke i dengan data uji citra ke j berdasarkan persamaan (2-1). Dimana data latih yang digunakan adalah citra ke 2 dan data uji yang digunakan citra ke 1. Jumlah n fitur ke r dari kedua vektor adalah 6. Diketahui a_r dari x_i , a_1 dari $x_2 = 154,57$, a_2 dari $x_2 = 145,9$, a_3 dari $x_2 = 150,01$, a_4 dari $x_2 = 5,66$, a_5 dari $x_2 = 4,45$, a_6 dari $x_2 = 3,84$. Dan diketahui juga a_r dari x_j , a_1 dari $x_1 = 106,53$, a_2 dari $x_1 = 81,09$, a_3 dari $x_1 = 64,99$, a_4 dari $x_1 = 5,45$, a_1 dari $x_2 = 4,74$, a_1 dari $x_2 = 4,36$.

$$\begin{aligned}
 d(x_2, x_1) &= \sqrt{\sum_{r=1}^6 (a_r(x_2) - a_r(x_1))^2} \\
 &= \sqrt{(154,57 - 106,53)^2 + (145,9 - 81,09)^2 + (150,01 - 64,99)^2 +} \\
 &\quad \sqrt{(5,66 - 5,45)^2 + (4,45 - 4,74)^2 + (3,84 - 4,36)^2} \\
 &= \sqrt{2307,84 + 4200,34 + 7228,4 + 0,0441 + 0,08 + 0,27} \\
 &= \sqrt{13736,97} \\
 &= 117,2048
 \end{aligned}$$

Berdasarkan contoh perhitungan jarak euclidean diatas maka didapat nilai jarak euclidean data uji terhadap data latih seperti yang ditunjukkan pada tabel 4.15. Hasil perhitungan di atas terdapat pada bagian yang diberi kotak tebal.

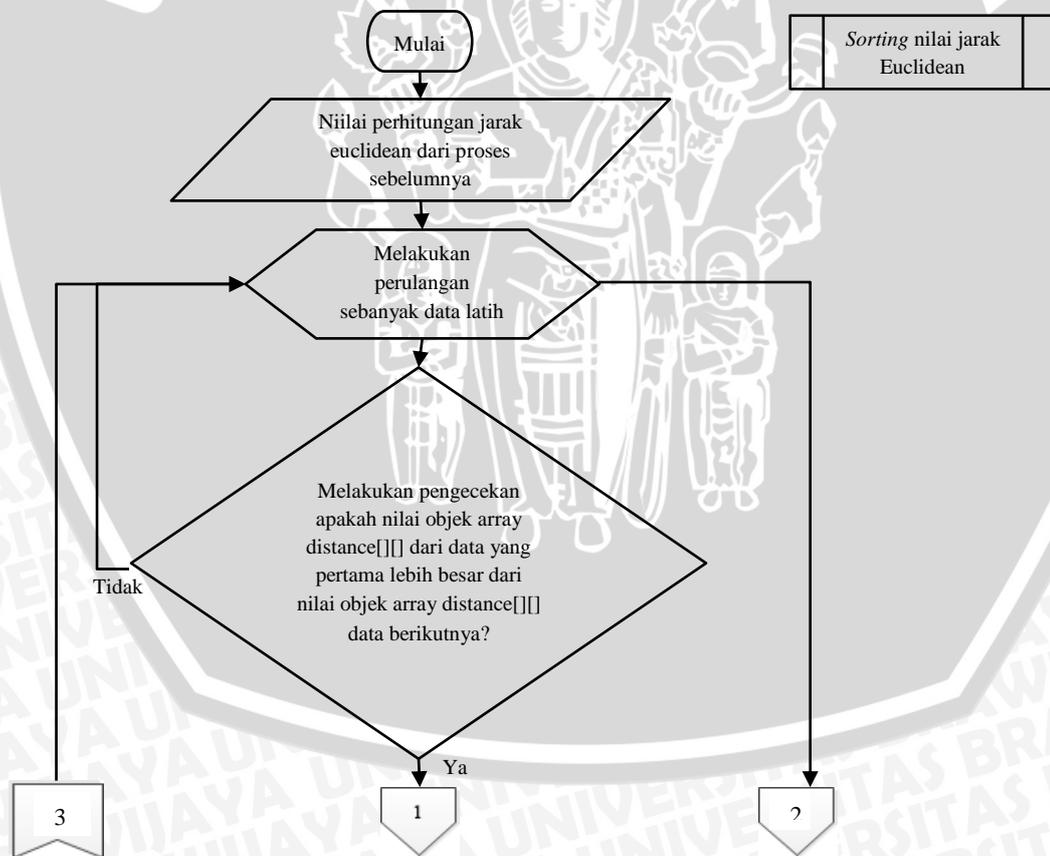
Tabel 4.15 Hasil Perhitungan Euclidean Distance

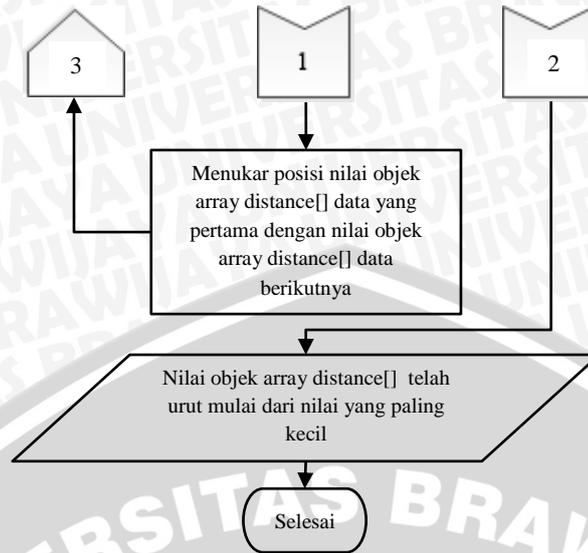
$d(x_i, x_j)$	X1	Kelas
X1	104,1738	Baik
X2	117,2048	Baik
X3	101,9872	Baik

X4	98,36429	Baik
X5	155,0056	Baik
X6	54,68339	Sedang
X7	79,24654	Sedang
X8	58,56638	Sedang
X9	16,71043	Sedang
X10	78,8449	Sedang
X11	146,7656	Buruk
X12	156,0382	Buruk
X13	110,7338	Buruk
X14	111,1571	Buruk
X15	81,64562	Buruk

Sumber: [Perancangan]

Tahap berikutnya setelah mendapat nilai jarak euclidean adalah mengurutkan kumpulan data latih. Pengurutan tersebut dimulai dari citra yang mempunyai jarak *euclidean* dengan nilai terkecil sampai terbesar. Diagram alir proses mengurutkan nilai jarak euclidean dapat dilihat pada gambar 4.16.





Gambar 4.16 Diagram Alir Proses Pengurutan Nilai Jarak Euclidean
Sumber: [Perancangan]

Pseudocode algoritma proses pengurutan nilai jarak *euclidean* sebagai berikut :

Nama Algoritma : Mengurutkan Nilai Jarak Euclidean
Deklarasi :
 ➤ Objek array : distance[]
 ➤ Integer : jumlah data latih
Deskripsi :
 ➤ Input : distance[][]
 ➤ Proses:
 a) Melakukan perulangan sebanyak jumlah data latih.
 b) Melakukan pengecekan apakah nilai objek array distance[][] dari data yang pertama lebih besar dari nilai objek array distance[][] dari data berikutnya. Jika iya maka lanjut ke langkah d. Jika tidak maka kembali ke langkah b.
 c) Melakukan pertukaran nilai objek array distance[][] data yang pertama dengan objek array distance[][] data berikutnya.
 ➤ Output : Nilai objek array distance[][] telah urut mulai dari nilai paling kecil

Gambar 4.17 Pseudocode Algoritma Mengurutkan Nilai Jarak Euclidean
Sumber: [Perancangan]

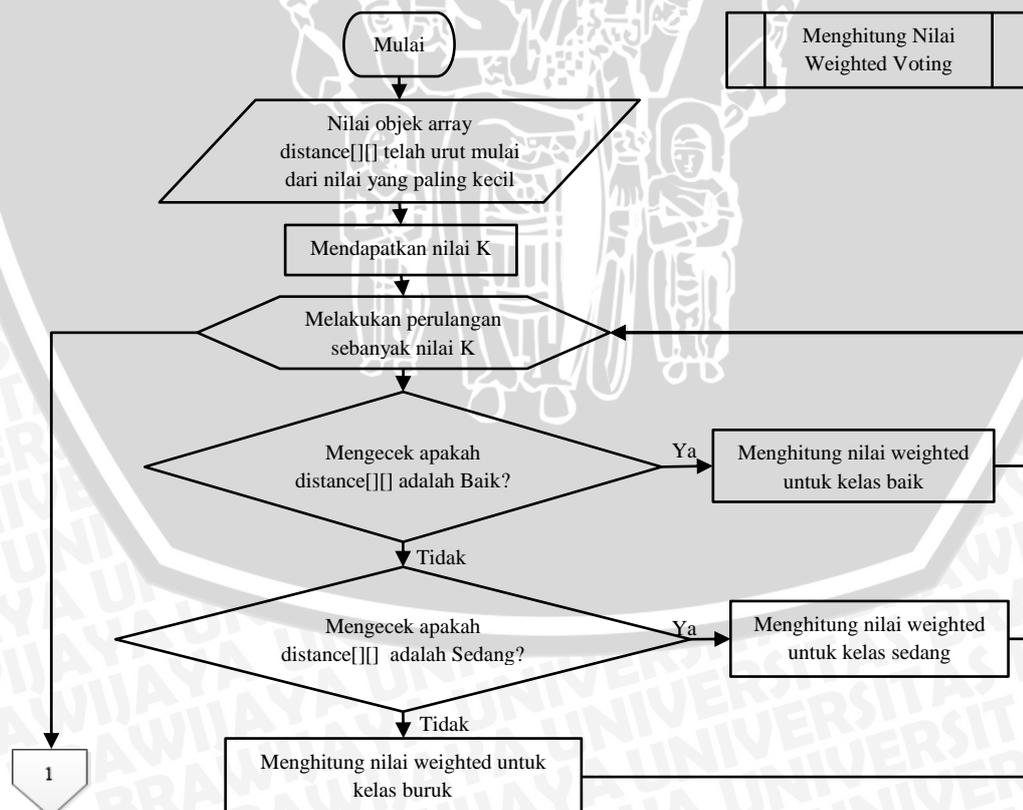
Contoh mengurutkan nilai jarak euclidean dari kumpulan data latih dimulai dari nilai terkecil sampai nilai terbesar dapat dilihat pada tabel 4.16.

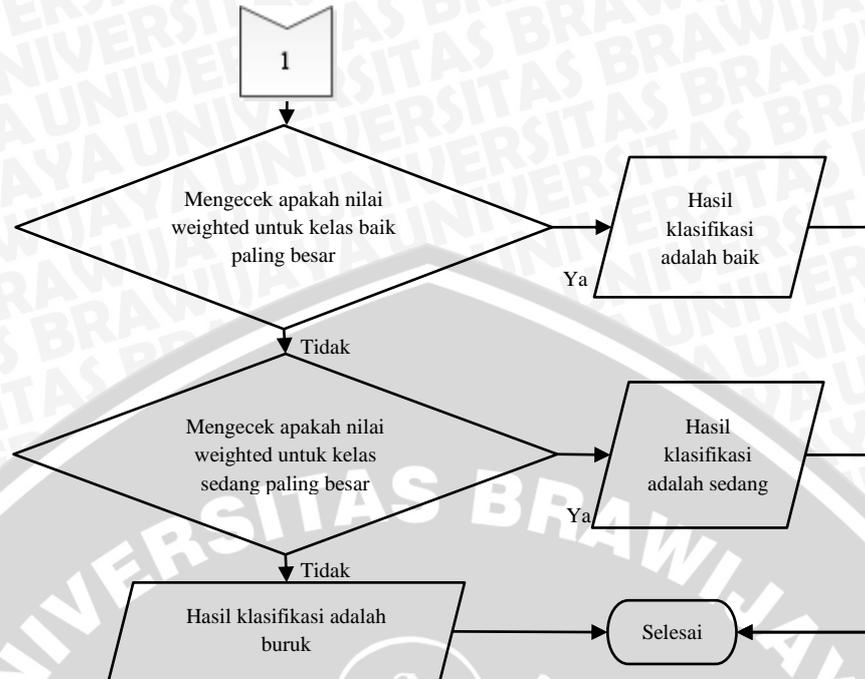
Tabel 4.16 Hasil Pengurutan Nilai Jarak Euclidean

Data Terdekat	$d(x_i, x_j)$	X1	Kelas
1	X9	16,71043	Sedang
2	X6	54,68339	Sedang
3	X8	58,56638	Sedang
4	X10	78,8449	Sedang
5	X7	79,24654	Sedang
6	X15	81,64562	Buruk
7	X4	98,36429	Baik
8	X3	101,9872	Baik
9	X1	104,1738	Baik
10	X13	110,7338	Buruk
11	X14	111,1571	Buruk
12	X2	117,2048	Baik
13	X11	146,7656	Buruk
14	X5	155,0056	Baik
15	X12	156,0382	Buruk

Sumber: [Perancangan]

Langkah terakhir adalah mendapatkan nilai K yang akan digunakan untuk proses pengklasifikasian menggunakan persamaan (2-2). Diasumsikan nilai K yang dipilih adalah 9, sehingga dipilih juga 9 data yang mempunyai jarak terdekat seperti yang ditunjukkan pada tabel 4.16. Diagram alir proses menghitung nilai *weighted voting* dapat dilihat pada gambar 4.18.





Gambar 4.18 Diagram Alir Perhitungan Weighted Voting
 Sumber: [Perancangan]

Pseudocode algoritma proses perhitungan *weighted voting* sebagai berikut:

```

Nama Algoritma : Menghitung Weighted Voting
Deklarasi :
    ➤ Double : baik, sedang, buruk
    ➤ String : kelas Hasil
Deskripsi :
    ➤ Input : nilai objek array distance[][] yang telahurut.
    ➤ Proses:
        a) Mendapatkan nilai K.
        b) Melakukan perulangan sebanyak nilai K.
        c) Mengecek apakah kelasSapi adalah "Baik". Jika ya maka lanjut ke langkah d. Jika tidak maka lanjut ke langkah e.
        d) Menghitung nilai weighted voting untuk kelas baik.
        e) Mengecek apakah kelasSapi adalah "Sedang". Jika ya maka lanjut ke langkah f. Jika tidak maka lanjut ke langkah g.
        f) Menghitung nilai weighted voting untuk kelas sedang.
        g) Menghitung nilai weighted voting untuk kelas buruk.
        h) Melakukan pengecekan apakah nilai weighted kelas baik paling besar. Jika ya maka lanjut ke langkah i. Jika tidak, lanjut ke langkah j.
        i) Memasukkan kata "Baik" ke variabel kelas Hasil.
        j) Melakukan pengecekan apakah nilai weigthed kelas sedang paling besar. Jika ya maka lanjut ke
    
```



- langkah k. Jika tidak, lanjut ke langkah m.
 k) Memasukkan kata "Sedang" ke variabel kelas Hasil.
 l) Memasukkan kata "Buruk" ke variabel kelas Hasil.
 ➤ Output : Mendapatkan dan menampilkan kelasHasil.

Gambar 4.19 Pseudocode Algoritma Menghitung Nilai Weighted Voting
 Sumber: [Perancangan]

Proses menghitung nilai *weighted voting* diawali dengan mendapatkan nilai K untuk mengambil nilai dari data latih yang telah diurutkan sebanyak K. Kemudian menghitung *weighted voting* dari tiap kelas dengan persamaan (2-2). Telah dipilih nilai K adalah 9 dan kesembilan data terdekat tersebut dapat dilihat pada tabel 4.17.

Tabel 4.17 Data Latih Yang Terdekat

$d(x_i, x_j)$	C1	Kelas
X9	16,71043	Sedang
X6	54,68339	Sedang
X8	58,56638	Sedang
X10	78,8449	Sedang
X7	79,24654	Sedang
X15	81,64562	Buruk
X4	98,36429	Baik
X3	101,9872	Baik
X1	104,1738	Baik

Sumber: [Perancangan]

Langkah selanjutnya adalah dihitung nilai *weighted voting* per kelas. Dimana diketahui bahwa dari 9 data terdekat kelas baik ada 4 data, kelas sedang ada 5 data, dan kelas buruk ada 0 data.

$$\begin{aligned}
 W_{\text{baik}} &= \frac{1}{d(x', x_i)^2} \\
 &= \frac{1}{(98,36429)^2} + \frac{1}{(101,9872)^2} + \frac{1}{(104,1738)^2} \\
 &= 0,000292
 \end{aligned}$$

$$\begin{aligned}
 W_{\text{sedang}} &= \frac{1}{d(x', x_i)^2} \\
 &= \frac{1}{(16,71043)^2} + \frac{1}{(54,68339)^2} + \frac{1}{(58,56638)^2} + \frac{1}{(78,8449)^2} + \frac{1}{(79,24654)^2} \\
 &= 0,004527228
 \end{aligned}$$

$$\begin{aligned}
 W_{\text{buruk}} &= \frac{1}{d(x', x_i)^2} \\
 &= \frac{1}{(81,64562)^2} \\
 &= 0,00015
 \end{aligned}$$

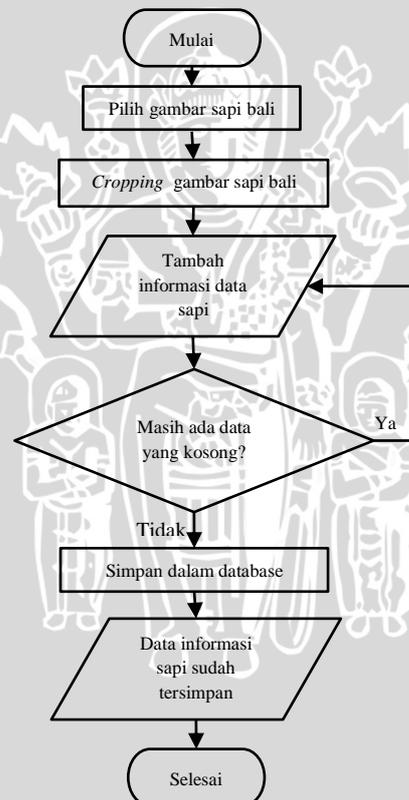
Berdasarkan hasil dari perhitungan tersebut, data uji diklasifikasikan kedalam kelas Sedang karena nilai *weighted voting* kelas sedang mempunyai nilai terbesar.

4.2.3.2. Pemodelan Olah Data Sapi

Proses yang dilakukan pada olah data sapi adalah menambah, mengubah, dan menghapus data sapi. Data sapi bali ini yang akan digunakan sebagai data latih untuk proses klasifikasi dari data uji. Olah data sapi hanya dilakukan admin.

1. Proses Tambah Data Sapi Bali

Proses ini melakukan penambahan data sapi berupa id sapi, jenis kelamin sapi, umur sapi, kelas sapi, dan *crop* sapi. Diagram alir proses tambah data sapi dapat dilihat pada gambar 4.20.



Gambar 4.20 Diagram Alir Proses Tambah Data Sapi
Sumber: [Perancangan]

Pseudocode algoritma proses tambah data sapi sebagai berikut:

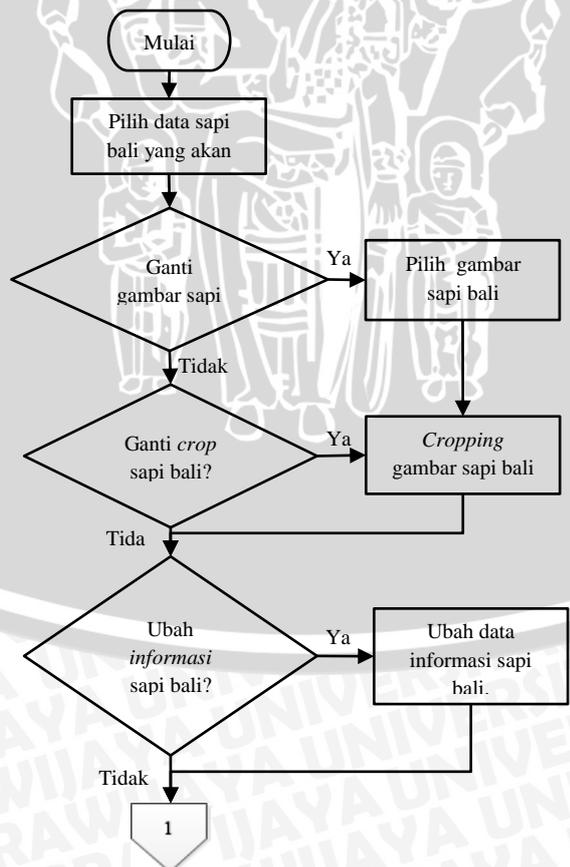
<p>Nama Algoritma : tambah data sapi</p> <p>Deklarasi :</p> <ul style="list-style-type: none"> ➤ String : id sapi, jenis kelamin sapi, umur sapi, kelas sapi

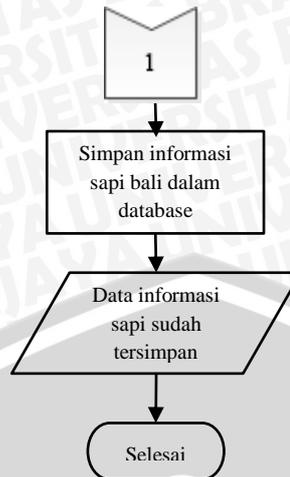
➤ Image : gambar sapi bali, crop sapi bali
Deskripsi :
 ➤ Input : id sapi, jk sapi, umur sapi, kelas sapi, gambar sapi bali, crop sapi bali
 ➤ Proses :
 a) Memilih gambar sapi bali dari direktori file.
 b) Melakukan *cropping* gambar tubuh bagian samping sapi bali.
 c) Memasukkan informasi data sapi berupa id sapi, jk sapi, umur sapi, kelas sapi.
 d) Menyimpan ke dalam database semua informasi data sapi bali.
 ➤ Output : Menampilkan pesan bahwa data sapi bali berhasil disimpan

Gambar 4.21 Pseudocode Algoritma Proses Tambah Data Sapi
 Sumber: [Perancangan]

2. Proses Ubah Data Sapi Bali

Proses ini melakukan pengubahan data sapi berupa id sapi, jenis kelamin sapi, umur sapi, kelas sapi, dan *crop* sapi. Diagram alir proses ubah data sapi dapat dilihat pada gambar 4.22.





Gambar 4.22 Diagram Alir Proses Ubah Data Sapi
Sumber: [Perancangan]

Pseudocode algoritma proses ubah data sapi sebagai berikut:

Nama Algoritma : ubah data sapi
Deklarasi :

- String : id sapi, jenis kelamin sapi, umur sapi, kelas sapi
- Image : gambar sapi bali, crop sapi bali

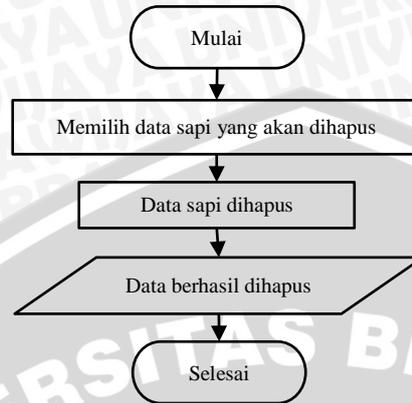
Deskripsi :

- Input : id sapi, jk sapi, umur sapi, kelas sapi, gambar sapi bali, crop sapi bali
- Proses :
 - a) Memilih data sapi bali yang akan diubah oleh admin.
 - b) Jika akan mengubah gambar sapi bali maka admin memilih gambar sapi bali dari direktori. Jika tidak maka lanjut ke proses c.
 - c) Jika akan mengubah crop sapi bali maka admin melakukan *cropping* gambar tubuh bagian samping sapi bali. Jika tidak maka lanjut ke proses d.
 - d) Jika akan mengubah data informasi sapi bali maka admin memasukkan informasi data sapi berupa id sapi, jk sapi, umur sapi, kelas sapi. Jika tidak maka lanjut ke proses e.
 - e) Menyimpan ke dalam database semua informasi data sapi bali oleh sistem.
- Output : Menampilkan pesan bahwa data informasi sudah tersimpan

Gambar 4.23 Pseudocode Algoritma Proses Ubah Data Sapi
Sumber: [Perancangan]

3. Proses Hapus Data Sapi Bali

Proses ini melakukan hapus data sapi. Diagram alir proses ubah data sapi dapat dilihat pada gambar 4.24.



Gambar 4.24 Diagram Alir Proses Hapus Data Sapi
Sumber: [Perancangan]

Pseudocode algoritma proses hapus data user sebagai berikut:

<p>Nama Algoritma : hapus data user</p> <p>Deklarasi :</p> <ul style="list-style-type: none"> ➤ String : id user, nama user, alamat user, jenis kelamin user, password user <p>Deskripsi :</p> <ul style="list-style-type: none"> ➤ Input : id user ➤ Proses : <ul style="list-style-type: none"> a) Memilih data sapi yang akan dihapus oleh admin. b) Menghapus data sapi yang dilakukan oleh sistem. ➤ Output : Menampilkan pesan bahwa data sapi berhasil dihapus

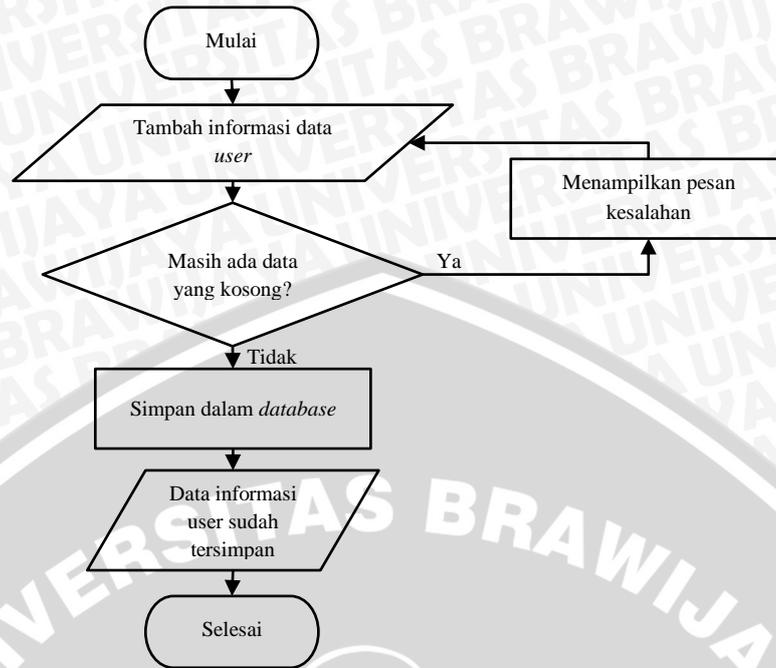
Gambar 4.25 Pseudocode Algoritma Proses Hapus Data Sapi
Sumber: [Perancangan]

4.2.3.3. Pemodelan Olah Data User

Proses yang dilakukan pada olah data *user* adalah menambah, mengubah, dan menghapus data *user*. Dalam hal ini adalah peternak yang akan berinteraksi dengan sistem pendukung keputusan pemilihan bibit unggul sapi bali. Penambahan data *user* ini hanya dilakukan oleh PTG.

1. Proses Tambah Data *User*

Proses ini melakukan penambahan data *user* berupa id *user*, nama *user*, alamat *user*, jenis kelamin *user*, dan *password user*. Diagram alir proses tambah data user dapat dilihat pada gambar 4.26.



Gambar 4.26 Diagram Alir Proses Tambah Data *User*
 Sumber: [Perancangan]

Pseudocode algoritma proses tambah data user sebagai berikut:

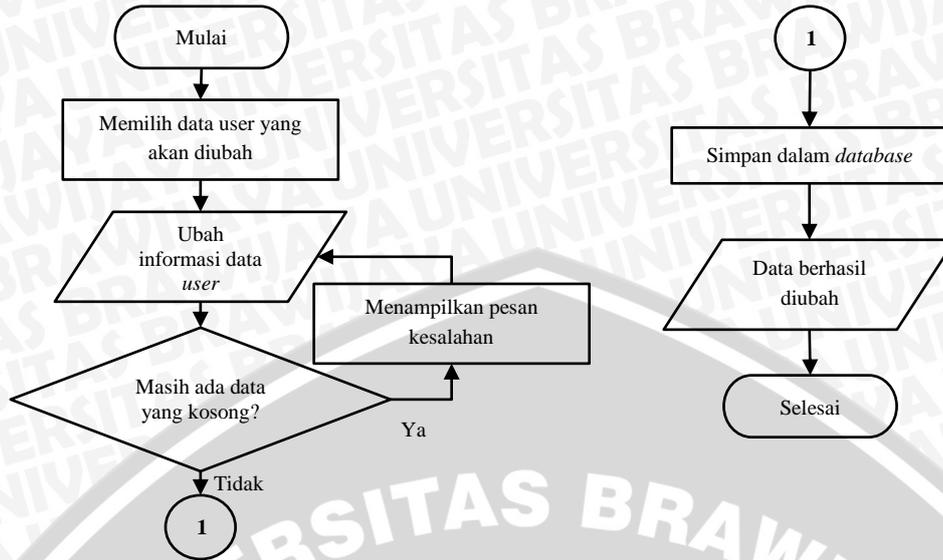
```

Nama Algoritma : tambah data user
Deklarasi :
    ➤ String : id user, nama user, alamat user, jenis kelamin user, password user
Deskripsi :
    ➤ Input : id user, nama user, alamat user, jenis kelamin user, password user
    ➤ Proses :
        a) Memasukkan informasi data user berupa id user, nama user, alamat user, jenis kelamin user, password user yang dilakukan oleh admin
        b) Melakukan pengecekan apakah ada data yang masih kosong atau tidak oleh sistem dan akan memberikan pesan kesalahan.
        c) Memasukkan informasi data user ke dalam database jika semua data sudah terisi.
    ➤ Output : Menampilkan pesan bahwa informasi data user sudah tersimpan
    
```

Gambar 4.27 Pseudocode Algoritma Proses Tambah Data *User*
 Sumber: [Perancangan]

2. Proses Ubah Data *User*

Proses ini melakukan pengubahan data *user* berupa *id user*, *nama user*, *alamat user*, *jenis kelamin user*, dan *password user*. Diagram alir proses ubah data user dapat dilihat pada gambar 4.28.



Gambar 4.28 Diagram Alir Proses Ubah Data User
Sumber: [Perancangan]

Pseudocode algoritma proses ubah data user sebagai berikut:

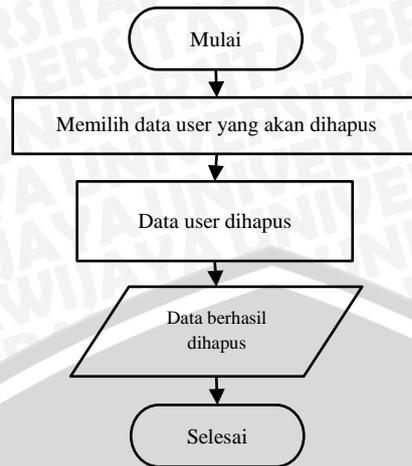
```

Nama Algoritma : ubah data user
Deklarasi :
    ➤ String : id user, nama user, alamat user, jenis kelamin user, password user
Deskripsi :
    ➤ Input : id user, nama user, alamat user, jenis kelamin user, password user
    ➤ Proses :
        a) Memilih data user yang akan diubah oleh admin.
        b) Memasukkan informasi data user berupa id user, nama user, alamat user, jenis kelamin user, password user yang dilakukan oleh admin
        c) Melakukan pengecekan apakah ada data yang masih kosong atau tidak oleh sistem dan akan memberikan pesan kesalahan.
        d) Memasukkan informasi data user ke dalam database jika semua data sudah terisi.
    ➤ Output : Menampilkan pesan bahwa data user berhasil diubah
    
```

Gambar 4.29 Pseudocode Algoritma Proses Ubah Data User
Sumber: [Perancangan]

3. Proses Hapus Data User

Proses ini melakukan hapus data *user*. Diagram alir proses ubah data user dapat dilihat pada gambar 4.30.



Gambar 4.30 Diagram Alir Proses Hapus Data User
Sumber: [Perancangan]

Pseudocode algoritma proses hapus data user sebagai berikut:

Nama Algoritma : hapus data user
Deklarasi :
 ➤ String : id user, nama user, alamat user, jenis kelamin user, password user
Deskripsi :
 ➤ Input : id user
 ➤ Proses :
 a) Memilih data user yang akan dihapus oleh admin.
 b) Menghapus data user yang dilakukan oleh sistem.
 c) Menampilkan pesan bahwa data user berhasil dihapus.
 ➤ Output : Menampilkan pesan bahwa data user berhasil dihapus

Gambar 4.31 Pseudocode Algoritma Proses Hapus Data User
Sumber: [Perancangan]

4.2.4. Perancangan Subsistem Antarmuka

Subsistem antarmuka akan membahas mengenai rancangan antarmuka dari login, admin, olah data user, olah data sapi, dan user. Antarmuka proses klasifikasi akan ada pada antarmuka user.

4.2.4.1. Antarmuka Login

Pada antarmuka login, admin (petugas BPTU) atau user (peternak) diminta untuk mengisi field username dan password untuk dapat masuk ke menu masing-masing. Antarmuka login dapat dilihat pada gambar 4.32.

Gambar 4.32 Antarmuka Login
Sumber: [Perancangan]

4.2.4.2. Antarmuka Admin

Pada halaman antarmuka admin terdapat tab menu untuk olah data user dan olah data sapi. Pada tab menu olah data *user* dan tab menu olah data sapi terdapat tombol untuk *refresh*, tambah, ubah, dan hapus data user. Untuk tab menu sapi terdapat informasi dari tiga ukuran *cropping* data sapi. Antarmuka admin dapat dilihat pada gambar 4.33.

Gambar 4.33 Antaramuka Admin
Sumber: [Perancangan]

4.2.4.3. Antarmuka Olah Data User

Pada antarmuka olah data user terdapat halaman untuk tambah data user dan ubah data user. Untuk halaman tambah dan ubah data user tidak ada perbedaan yang signifikan. Perbedaannya hanya terletak pada tombol tambah. Pada halaman ubah data user, tombol tambah akan menjadi tombol ubah. Untuk

jenis kelamin akan dibuat *radio button* untuk opsi pilihannya. Antarmuka olah data user dapat dilihat pada gambar 4.34.

Judul Halaman	
ID Anggota	<input type="text"/>
Nama Anggota	<input type="text"/>
Alamat Anggota	<input type="text"/>
Jenis Kelamin	<input type="text"/>
Password	<input type="text"/>
	<input type="button" value="Tambah"/>
	<input type="button" value="Batal"/>

Gambar 4.34 Antarmuka Olah Data User
Sumber: [Perancangan]

4.2.4.4. Antarmuka Olah Data Sapi

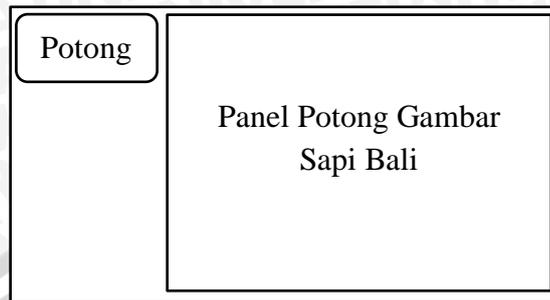
Pada antarmuka olah data sapi terdapat halaman untuk tambah data sapi dan ubah data sapi. Untuk halaman tambah dan ubah data sapi tidak ada perbedaan yang signifikan. Pada halaman ubah data sapi, tombol tambah akan menjadi tombol ubah. Antarmuka olah data sapi dapat dilihat pada gambar 4.35.

Judul Halaman	
Gambar Sapi Bali	Informasi Sapi Bali
	<input type="text" value="Komponen Warna Rata-Rata"/>
<input type="button" value="Pilih gambar"/>	<input type="button" value="Potong"/>
<input type="button" value="Tambah"/>	<input type="button" value="Batal"/>
Crop Gambar Sapi Bali	Crop Gambar Sapi Bali
Crop Gambar Sapi Bali	Crop Gambar Sapi Bali
Data Komponen Warna Crop Gambar Sapi Bali	

Gambar 4.35 Antarmuka Olah Data Sapi
Sumber: [Perancangan]

Pada bagian informasi sapi bali terdapat *field* untuk id sapi, *combo box* untuk jenis kelamin sapi, umur sapi, dan kelas sapi. Pada bagian komponen warna crop gambar sapi bali terdapat *field* untuk menampilkan nilai RGB dari ketiga

hasil crop gambar sapi bali. Sedangkan pada bagian komponen warna rata-rata terdapat field untuk menampilkan rata-rata RGB dari hasil *crop*.



Gambar 4.36 Antarmuka Crop Gambar Sapi Bali
Sumber: [Perancangan]

Jika tombol potong pada antarmuka olah data sapi ditekan maka akan mengarah pada antarmuka crop gambar sapi bali seperti pada gambar 4.36. Setelah melakukan pemotongan gambar maka akan kembali ke antarmuka olah data sapi.

4.2.4.5. Antarmuka User

Pada antarmuka user terdapat tampilan untuk melakukan proses memilih data uji dan proses klasifikasi. Antarmuka user dapat dilihat pada gambar 4.37.

Menu			
Judul Halaman			
Data Uji	Pengujian	Klasifikasi	Skenario Rata-Rata
Tabel Data Uji			

Gambar 4.37 Antarmuka User
Sumber: [Perancangan]

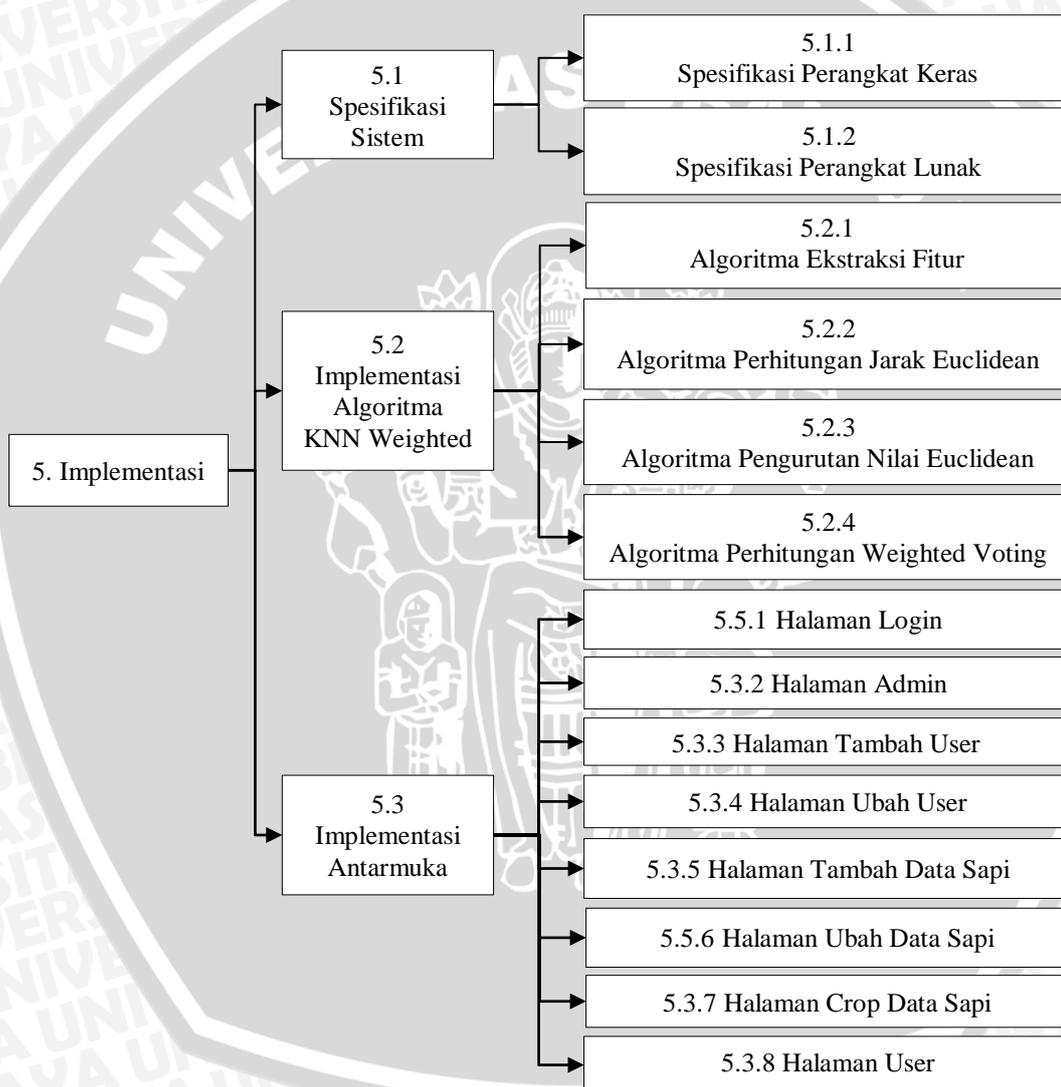
Pada tab data uji terdapat tabel untuk menampilkan seluruh data uji yang telah diujikan. Pada tab pengujian terdapat halaman untuk memasukkan informasi data sapi bali yang akan diuji, pemilihan besar data latih yang digunakan, dan pemilihan nilai K. Pada tab klasifikasi terdapat halaman untuk menampilkan

proses klasifikasi seperti proses perhitungan *euclidean distance*, *sorting* nilai euclidean, dan hasil perhitungan *weighted voting*. Pada tab skenario rata-rata terdapat halaman untuk menampilkan hasil pengujian yang telah dibahas pada bab sebelumnya.



BAB V IMPLEMENTASI

Bab ini menjelaskan mengenai spesifikasi sistem, implementasi algoritma, dan implementasi antarmuka yang dibangun untuk sistem pendukung keputusan klasifikasi kualitas sapi bali berdasarkan perancangan yang dilakukan pada bab sebelumnya. Pohon implementasi dapat dilihat pada gambar 5.1.



Gambar 5.1 Pohon Implementasi
Sumber: [Implementasi]

5.1. Spesifikasi Sistem

Implementasi merupakan tahap meletakkan sistem sehingga siap untuk dioperasikan. Spesifikasi sistem dalam penelitian ini dibutuhkan untuk membantu



proses implementasi tersebut. Spesifikasi sistem yang akan dibahas adalah spesifikasi perangkat keras dan spesifikasi perangkat lunak.

5.1.1. Spesifikasi Perangkat Keras

Perangkat keras yang digunakan untuk mengimplementasikan sistem dapat dilihat pada tabel 5.1.

Tabel 5.1 Spesifikasi Perangkat Keras

Perangkat Keras	Keterangan
Processor	Intel(R) Core(TM) i5 M 480 @ 2.67GHz
Memory RAM	2048MB
Chip Type	AMD Radeon HD 7400M Series
DAC Type	Internal DAC (400Mhz)
Approx. Total Memory	1741 MB

Sumber: [Implementasi]

5.1.2. Spesifikasi Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasi sistem dapat dilihat pada tabel 5.2.

Tabel 5.2 Spesifikasi Perangkat Lunak

Perangkat Lunak	Keterangan
Sistem Operasi	Windows 8 Pro
Bahasa Pemrograman	Java
Tools	Netbeans
Server	XAMPP 1.7.1
DBMS	MySQL

Sumber: [Implementasi]

5.2. Implementasi Algoritma KNN Weighted

Implementasi algoritma bertujuan untuk mengimplementasikan algoritma yang digunakan dalam sistem. Dalam bab ini algoritma yang akan dibahas adalah algoritma metode *KNN Weighted*.

5.2.1. Algoritma Ekstraksi Fitur

Algoritma ekstraksi fitur merupakan algoritma untuk mencari nilai *mean* dan standar deviasi RGB dari suatu gambar dengan ukuran pixel tertentu. Langkah pertama adalah mencari *mean* dari komponen warna RGB. Kemudian mencari standar deviasi dari komponen RGB tersebut. Implementasi algoritma ekstraksi fitur dapat dilihat pada tabel 5.4.

Tabel 5.6 Implementasi Algoritma Ekstraksi Fitur

```

1 public static double [] getStdRGB( BufferedImage gambar){
2     double [] MeanRGB = new double [3];
3     double [] rgbMeanStd = new double[3];
4     int count = 0;
5     double rcount = 0,rrr = 0;
6     double gcount = 0,ggg = 0;
7     double bcount = 0,bbb = 0;
8     for (int i = 0; i < gambar.getHeight(); i++) {
9         for (int j = 0; j < gambar.getWidth(); j++) {
10            int c = gambar.getRGB(j, i);
11            int r = (c & 0x00ff0000) >> 16;
12            int g = (c & 0x0000ff00) >> 8;
13            int b = (c & 0x000000ff);
14
15            rcount += r;
16            gcount += g;
17            bcount += b;
18            count++;
19        }
20    }
21
22    MeanRGB[0] = rcount / count;
23    MeanRGB[1] = gcount / count;
24    MeanRGB[2] = bcount / count;
25
26    for (int i = 0; i < gambar.getHeight(); i++) {
27        for (int j = 0; j < gambar.getWidth(); j++) {
28            int c = gambar.getRGB(j, i);
29            int r = (c & 0x00ff0000) >> 16;
30            int g = (c & 0x0000ff00) >> 8;
31            int b = (c & 0x000000ff);
32
33            rrr += Math.sqrt(Math.pow((r-MeanRGB[0]),2));
34            ggg += Math.sqrt(Math.pow((g-MeanRGB[1]),2));
35            bbb += Math.sqrt(Math.pow((b-MeanRGB[2]),2));
36        }
37    }
38
39    rgbMeanStd[0] = rrr/count;
40    rgbMeanStd[1] = ggg/count;
41    rgbMeanStd[2] = bbb/count;
42
43    return rgbMeanStd;
44 }

```

Sumber: [Implementasi]

- 1 : Terdapat parameter gambar yang berfungsi untuk mengambil gambar yang telah diinputkan oleh pengguna.
- 2-3 : Pendeklarasian double array variabel MeanRGB untuk menyimpan *mean* RGB dan rgbMeanStd untuk menyimpan standar deviasi RGB.
- 4-7 : Pemberian nilai awal untuk variabel count, rrr, ggg, bbb, rcount, gcount, dan bcount.
- 8 : Proses melakukan perulangan sebanyak ukuran *height* gambar.
- 9 : Proses melakukan perulangan sebanyak ukuran *width* gambar.
- 10-13 : Proses mengambil nilai RGB tiap pixel.
- 15-18 : Proses penjumlahan nilai-nilai RGB di semua pixel dan jumlah keseluruhan pixel.
- 22-24 : Proses perhitungan *mean* dari warna RGB.
- 26 : Proses melakukan perulangan sebanyak ukuran *height* gambar.
- 27 : Proses melakukan perulangan sebanyak ukuran *width* gambar.
- 28-31 : Proses mengambil nilai RGB tiap pixel.
- 33-35 : Proses perhitungan standar deviasi dari warna RGB.
- 39-41 : Proses menghitung rata-rata nilai standar deviasi RGB.
- 43 : Mengembalikan nilai rata-rata standar deviasi RGB.

5.2.2. Algoritma Perhitungan Jarak Euclidean

Algoritma perhitungan jarak euclidean merupakan langkah awal dalam proses klasifikasi metode KNN. Data uji yang dimasukkan akan dihitung jaraknya terhadap sejumlah data latih yang telah ditentukan. Implementasi algoritma perhitungan jarak euclidean dapat dilihat pada tabel 5.3.

Tabel 5.3 Implementasi Algoritma Perhitungan Jarak Euclidean

1	public void euclideanDistance () {
2	Distance = new Object [jumData][3];
3	for(int i=0;i<jumData;i++){
4	jarak = (Math.pow(userCropSapi.meanRGBHasil[0]- Double.parseDouble(String.valueOf(dataLatih[i][1])), 2))+ (Math.pow(userCropSapi.meanRGBHasil[1]- Double.parseDouble(String.valueOf(dataLatih[i][2])), 2))+ (Math.pow(userCropSapi.meanRGBHasil[2]- Double.parseDouble(String.valueOf(dataLatih[i][3])), 2))+ (Math.pow(userCropSapi.stdRGBHasil[0]- Double.parseDouble(String.valueOf(dataLatih[i][4])), 2))+ (Math.pow(userCropSapi.stdRGBHasil[1]-



```

Double.parseDouble(String.valueOf(dataLatih[i][5]), 2))+
(Math.pow(userCropSapi.stdRGBHasil[2]-
Double.parseDouble(String.valueOf(dataLatih[i][6]), 2));
5   Euclidean=Math.sqrt(jarak);
6   Distance[i][0]=i+1;
7   Distance[i][1]=Euclidean;
8   Distance[i][2]=String.valueOf(dataLatih[i][7]);
9   }
10  }

```

Sumber: [Implementasi]

Baris ke:

- 2 : Pendeklarasian objek Distance
- 3 : Melakukan perulangan sebanyak data latih
- 4-5 : Proses perhitungan jarak euclidean. Fitur-fitur dari data uji dibandingkan dengan fitur-fitur pada data latih.
- 6-8 : Proses memasukkan hasil perhitungan. Nilai-nilai yang dimasukkan adalah citra ke-i, nilai euclidean, dan kelas data latih.

5.2.3. Algoritma Pengurutan Nilai Euclidean

Langkah berikutnya yang dilakukan setelah nilai jarak euclidean didapatkan adalah melakukan pengurutan (*sorting*) nilai euclidean. Nilai euclidean diurutkan mulai dari nilai yang menunjukkan jarak yang terdekat, yaitu nilai yang paling kecil. Implementasi algoritma pengurutan nilai euclidean dapat dilihat pada tabel 5.4.

Tabel 5.4 Implementasi Algoritma Pengurutan Nilai Euclidean

```

1   public void sortingEuclidean(){
2       for (int i=0; i<jumData-1; i++) {
3           for (int j=i+1; j<jumData; j++) {
4               if (Double.parseDouble(String.valueOf(Distance[i][1])) >
5                   Double.parseDouble(String.valueOf(Distance[j][1]))) {
6                   int tempNilai =
7                       Integer.parseInt(String.valueOf(Distance[i][0]));
8                   double tempNilaiAkar =
9                       Double.parseDouble(String.valueOf(Distance[i][1]));
10                  Object tempKelas = Distance[i][2];
11                  Distance[i][0] = Distance[j][0];
12                  Distance[i][1] = Distance[j][1];
13                  Distance[i][2] = Distance[j][2];
14                  Distance[j][0] = tempNilai;
15                  Distance[j][1] = tempNilaiAkar;
16                  Distance[j][2] = tempKelas;

```

14	}
15	}
16	}
17	}

Sumber: [Penelitian]

Baris ke:

2-3 : Proses perulangan sebanyak data latih.

4 : Proses pengecekan apakah nilai dari data pertama lebih besar dari data kedua. Jika data pertama lebih besar maka proses yang ada di baris berikutnya akan dikerjakan.

Pertukaran Nilai

5-7 : Proses untuk memasukkan data pada objek Distance ke dalam variabel bantuan.

8-10 : Objek Distance data pertama diisi dengan nilai pada objek Distance data kedua.

11-13 : Proses mengembalikan nilai pada variabel bantuan ke dalam objek data kedua.

5.2.4. Algoritma Perhitungan *Weighted Voting*

Langkah terakhir adalah proses menghitung nilai *weighted voting*. Langkah pertama adalah mendapatkan nilai K. Kemudian mendapatkan nilai objek Distance sebanyak K untuk dilakukan proses perhitungan menurut kelasnya. Implementasi algoritma perhitungan *weighted voting* dapat dilihat pada tabel 5.5.

Tabel 5.5 Implementasi Algoritma Perhitungan *Weighted Voting*

1	public void weighted(int k) throws SQLException{
2	for(int i=0;i<k;i++){
3	if(Distance[i][2].equals("Baik")){
4	baik=baik + (1/ (Math.pow (Double.parseDouble (String.valueOf (Distance[i][1]), 2)));
5	}else if(Distance[i][2].equals("Sedang")){
6	sedang=sedang+ (1/(Math.pow (Double.parseDouble (String.valueOf (Distance[i][1]), 2)));
7	}else if(Distance[i][2].equals("Buruk")){
8	buruk=buruk+ (1/(Math.pow (Double.parseDouble (String.valueOf(Distance[i][1]), 2)));
9	}
10	}
11	if(baik>sedang&&baik>buruk){
12	hasil = "Baik";

```

13 }else if(sedang>baik&&sedang>buruk){
14     hasil = "Sedang";
15 }else if(buruk>sedang&&buruk>baik){
16     hasil = "Buruk";
17 }
18 }

```

Sumber: [Penelitian]

- 1 : Terdapat parameter k untuk mengambil nilai K yang diinputkan oleh pengguna.
- 2 : Proses perulangan sebanyak K untuk mengambil data sebanyak K dari objek Distance yang telah diurutkan.
- 3,5,7 : Proses pengecekan apakah kelas yang berada pada K terdekat adalah Baik, Sedang atau Buruk. Jika masuk ke dalam salah satu kelas tersebut maka akan dilakukan perhitungan *weighted voting* seperti pada baris 4,6, dan 8.
- 4 : Proses perhitungan *weighted voting* kelas baik
- 6 : Proses perhitungan *weighted voting* kelas sedang
- 8 : Proses perhitungan *weighted voting* kelas buruk
- 11 : Melakukan pengecekan apakah nilai *weighted voting* kelas baik paling besar. Jika paling besar, maka melakukan proses pada baris 12. Jika tidak, maka melakukan proses baris 13.
- 12 : Memasukkan nilai pada variabel hasil yaitu “Baik”
- 13 : Melakukan pengecekan apakah nilai *weighted voting* kelas sedang paling besar. Jika paling besar, maka melakukan proses pada baris 14. Jika tidak, maka melakukan proses baris 15.
- 14 : Memasukkan nilai pada variabel hasil yaitu “Sedang”
- 15 : Melakukan pengecekan apakah nilai *weighted voting* kelas buruk paling besar. Jika paling besar, maka melakukan proses pada baris 16.
- 16 : Memasukkan nilai pada variabel hasil yaitu “Buruk”

5.3. Implementasi Antarmuka

Implementasi antarmuka bertujuan untuk mendeskripsikan atau mengimplementasikan rancangan antarmuka yang dibahas pada bab sebelumnya. Antarmuka penelitian ini dibuat dalam bentuk dekstop.

5.3.1. Halaman Login

Halaman login merupakan antarmuka yang berisi *textfield* untuk mengisi username dan password dari pengguna yang akan masuk ke dalam sistem. Pengguna akan masuk ke halaman administrator jika memasukkan *username* dan *password* yang ada pada database *admin*. Pengguna juga bisa masuk ke halaman peternak jika memasukkan *username* dan *password* yang ada pada database *user*. Antarmuka halaman login dapat dilihat pada gambar 5.2.



Gambar 5.2 Antarmuka Halaman Login
Sumber: [Implementasi]

5.3.2. Halaman Admin

Halaman admin merupakan antarmuka yang akan dimasuki oleh pengguna yang memasukkan username dan password yang sesuai dengan database admin. Antarmuka administrator ini mempunyai 2 tab utama yaitu tab data anggota dan tab data sapi. Tampilan awal yang dimunculkan antarmuka administrator adalah tab data anggota seperti yang ditunjukkan pada gambar 5.3.



Gambar 5.3 Antarmuka Halaman Admin Tab Data Anggota
Sumber: [Implementasi]

Untuk tampilan tab data sapi ditunjukkan pada gambar 5.4.

ID Sapi Bali	Red	Green	Blue	SDRed	SDGreen	SDBlue	JK	Umur	Kelas
SB-151	145.318969..	108.66112..	86.3187255..	16.3617965..	13.0951691..	12.1654033..	Betina	2 tahun	Baik
SB-152	103.111938..	78.0549926..	55.7389933..	11.4503844..	8.74937371..	6.84257147..	Betina	2 tahun	Baik
SB-153	121.270751..	66.1315714..	20.5498980..	21.4009015..	16.4317888..	11.7308483..	Betina	2 tahun	Baik
SB-154	81.6469799..	55.8901163..	32.9333902..	8.46404104..	5.96379545..	4.86919491..	Betina	2 tahun	Baik
SB-155	99.7161458..	77.0263264..	52.5967610..	6.40728157..	6.96691739..	7.38183954..	Betina	2 tahun	Baik
SB-156	128.222371..	97.3303485..	79.6324259..	14.8645031..	9.21371329..	9.30546792..	Betina	2 tahun	Baik
SB-157	107.406514..	79.2848714..	64.7054443..	10.3224973..	7.19394542..	6.35972344..	Betina	2 tahun	Baik
SB-158	91.8939333..	64.4231492..	47.2633870..	9.79739379..	6.82131663..	5.48934953..	Betina	2 tahun	Baik
SB-159	96.5352783..	62.9151814..	35.8285522..	21.8483599..	15.2102054..	9.47744498..	Betina	2 tahun	Baik
SB-160	182.001200..	124.460306..	75.7848665..	15.5983762..	13.7046424..	12.4586872..	Betina	2 tahun	Baik
SB-61	156.935343..	125.257893..	111.074015..	7.70997619..	6.55148318..	6.59004696..	Betina	2 tahun	Baik
SB-62	156.868957..	119.965352..	99.7769571..	13.0028993..	10.0890092..	8.82098891..	Betina	2 tahun	Baik
SB-63	132.353047..	100.953430..	79.1319173..	25.5561559..	19.8940718..	18.3796984..	Betina	2 tahun	Baik
SB-64	110.121887..	83.3516845..	62.3471883..	12.6018888..	9.12361497..	6.16977788..	Betina	2 tahun	Baik
SB-65	153.953491..	101.331665..	61.5367024..	11.5576723..	11.3963361..	11.6672110..	Betina	2 tahun	Baik
SB-66	153.172505..	95.7507120..	49.2890014..	11.4010494..	11.8769244..	13.5912808..	Betina	2 tahun	Baik
SB-67	99.2357788..	60.7295549..	39.8101399..	18.4393991..	10.8240476..	7.34393394..	Betina	2 tahun	Baik
SB-68	82.4586791..	56.0521240..	32.8023885..	8.51713837..	5.87517358..	4.64424217..	Betina	2 tahun	Baik
SB-69	125.411092..	97.4637247..	74.8672485..	11.7710195..	9.24875766..	7.85940126..	Betina	2 tahun	Baik
SB-70	100.877408..	77.3059082..	56.1962890..	5.06295503..	4.88017981..	4.73518036..	Betina	2 tahun	Baik
SB-71	184.720643..	139.294576..	122.887268..	10.9594949..	9.92639624..	10.3688941..	Betina	2 tahun	Baik
SB-72	165.110371..	114.202026..	96.5807698..	8.75593599..	7.59137997..	7.52452128..	Betina	2 tahun	Baik
SB-73	133.075134..	99.9911092..	85.1103922..	12.1043248..	9.81060772..	8.91396255..	Betina	2 tahun	Baik
SB-74	116.374694..	85.1832682..	70.9096272..	9.22341813..	6.69800891..	6.27614446..	Betina	2 tahun	Baik
SB-75	123.504781..	87.1386718..	70.1933797..	14.6338103..	11.4272697..	10.0151341..	Betina	2 tahun	Baik
SB-76	102.038065..	70.4065102..	53.9478556..	8.19478127..	5.84810627..	5.27917784..	Betina	2 tahun	Baik
SB-77	176.893359..	146.656545..	108.893659..	18.8796959..	15.2400066..	14.6909999..	Betina	2 tahun	Baik

Gambar 5.4 Antarmuka Halaman Admin Tab Data Sapi
Sumber: [Implementasi]

Pada gambar 5.3 dan 5.4 terdapat beberapa tombol seperti tombol *refresh*, tambah, ubah, dan hapus. Tombol tambah pada tab data anggota berfungsi untuk melakukan penambahan data anggota. Tombol ubah pada tab data anggota berfungsi untuk melakukan perubahan informasi data anggota yang telah dipilih. Tombol hapus pada tab data anggota berfungsi untuk menghapus data anggota yang telah dipilih. Sedangkan tombol tambah pada tab data sapi berfungsi untuk melakukan penambahan data sapi. Tombol ubah pada tab data sapi berfungsi untuk melakukan perubahan informasi data sapi yang telah dipilih. Tombol hapus pada tab data sapi berfungsi untuk menghapus data sapi yang telah dipilih.

5.3.3. Halaman Tambah Anggota

Halaman Tambah Anggota adalah bagian dari antarmuka admin. Halaman ini akan tampil jika pengguna menekan tombol tambah yang ada pada tab data anggota. Antarmuka halaman tambah anggota ditunjukkan pada gambar 5.5.

Gambar 5.5 Antarmuka Halaman Tambah Anggota
Sumber: [Implementasi]

Textfield ID Anggota selain berfungsi untuk mengisi identitas anggota, berfungsi juga untuk mengisi username dari anggota baru yang akan ditambah. Tombol tambah berfungsi untuk melakukan penambahan informasi yang telah dimasukkan oleh pengguna ke dalam database *user*. Tombol batal berfungsi untuk menutup halaman tambah anggota dan kembali ke halaman admin.

5.3.4. Halaman Ubah Anggota

Halaman ubah anggota merupakan bagian dari antarmuka admin. Halaman ini akan tampil jika pengguna menekan tombol ubah pada tab data anggota. Antarmuka halaman ubah anggota ditunjukkan pada gambar 5.6.

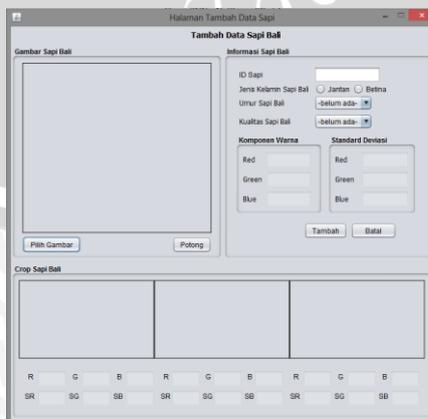


Gambar 5.6 Antarmuka Halaman Ubah Anggota
Sumber: [Implementasi]

Pada gambar 5.6 ditampilkan data anggota dengan ID user yang akan diubah informasinya. Terdapat tombol ubah yang berfungsi melakukan perubahan data pada database user sesuai dengan informasi yang telah dimasukkan oleh pengguna.

5.3.5. Halaman Tambah Sapi

Halaman Tambah Sapi adalah bagian dari antarmuka admin. Halaman ini akan tampil jika pengguna menekan tombol tambah yang ada pada tab data sapi. Antarmuka halaman tambah sapi ditunjukkan pada gambar 5.7.



Gambar 5.7 Antarmuka Halaman Tambah Sapi
Sumber: [Implementasi]

Terdapat beberapa tombol yang harus ditekan oleh pengguna secara berurutan jika akan menambah data latih sapi bali. Tombol pertama yang berfungsi untuk memilih data gambar sapi bali yang akan dimasukkan ke dalam database adalah tombol pilih gambar. Tombol kedua yang berfungsi untuk memilih bagian gambar sapi bali yang akan dipotong adalah tombol potong. Pada bagian *Crop* sapi bali dan komponen warna akan tampil nilai-nilai RGB dan SR, SG, SB dari gambar yang telah dipotong tadi. Kemudian pengguna harus memasukkan nilai pada bagian informasi sapi bali untuk bisa melakukan penambahan data dengan menekan tombol tambah.

5.3.6. Halaman Ubah Sapi

Halaman ubah sapi merupakan bagian dari antarmuka admin. Halaman ini akan tampil jika pengguna menekan tombol ubah yang ada pada tab data sapi. Antarmuka halaman ubah sapi ditunjukkan pada gambar 5.8.



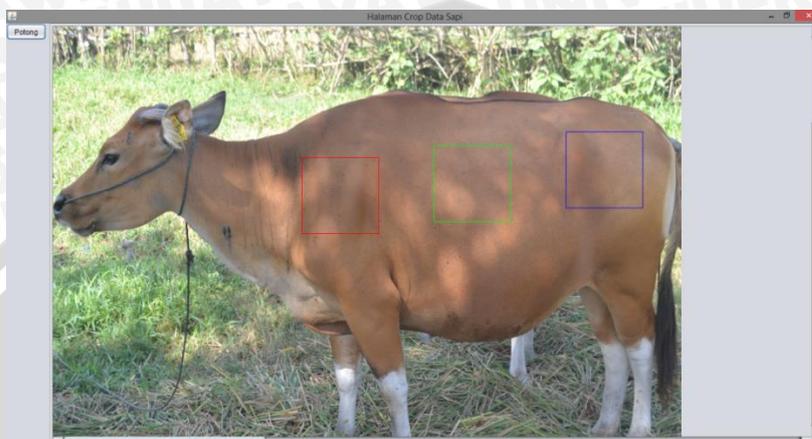
Gambar 5.8 Antarmuka Halaman Ubah Sapi
Sumber: [Implementasi]

Informasi data sapi bali telah dipilih akan ditampilkan pada antarmuka ini. Jika akan mengubah *crop* gambar sapi bali maka pengguna harus menandai *checkbox* yang ada di samping tombol potong. Pengguna juga bisa langsung mengubah informasi data sapi bali jika tidak ingin mengubah *crop* gambar sapi bali.

5.3.7. Halaman Crop Sapi

Halaman *crop* sapi merupakan antarmuka yang akan tampil jika pengguna menekan tombol potong pada antarmuka tambah sapi atau antarmuka ubah sapi.

Pada halaman crop sapi akan ditampilkan gambar sapi yang telah dipilih, dan akan ada 3 buah kotak dengan warna merah, hijau, dan biru yang mewakili daerah *crop* 1, *crop* 2, dan *crop* 3. Antarmuka halaman *crop* sapi ditunjukkan pada gambar 5.9.



Gambar 5.9 Antarmuka Halaman *Crop* Sapi
Sumber: [Implementasi]

5.3.8. Halaman User

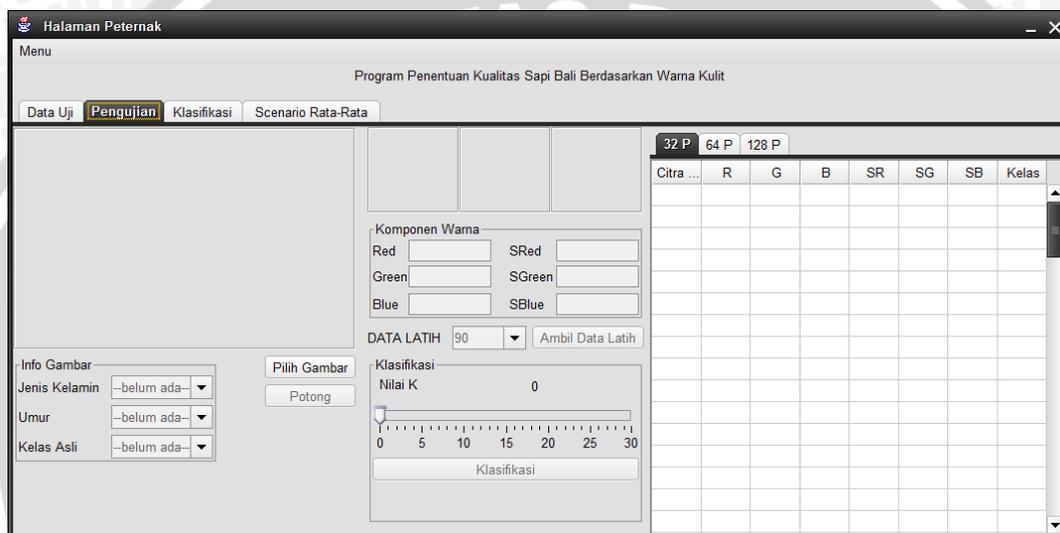
Halaman user merupakan antarmuka yang akan dimasuki pengguna yang memasukkan *username* dan *password* yang sesuai dengan database *user*. Pada antarmuka ini terdapat 4 tab yang menunjukkan proses pengujian pemilihan bibit unggul sapi bali. Tab-tab tersebut antara lain tab data uji, tab pengujian, tab klasifikasi, dan tab skenario rata-rata. Tab data uji menampilkan hasil pengujian yang telah dilakukan oleh pengguna. Dalam tab data uji terdapat 3 tab lagi yang menampilkan data uji dengan 3 ukuran pixel yang berbeda, yaitu pixel 32, pixel 64, dan pixel 128. Antarmuka halaman *user* tab data uji ditunjukkan pada gambar 5.10.

ID Uji	JK	Umur	Kelas Asli	Red	Green	Blue	SDRed	SDGreen	SDBlue	Pixel	Data Lath	Nilai K	Hasil Wei.
001	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	3	Baik
002	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	6	Baik
003	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	9	Baik
004	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	12	Baik
005	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	15	Baik
006	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	18	Baik
007	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	21	Baik
008	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	24	Baik
009	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	27	Baik
010	Jantan	2 Tahun	Baik	86.40039	88.15299	101.4690	5.934029	5.987157	6.196965	32	90	30	Baik
011	Jantan	2 Tahun	Baik	67.53938	68.22819	66.39257	2.613805	2.334108	2.226915	32	90	3	Baik
012	Jantan	2 Tahun	Baik	67.53938	68.22819	66.39257	2.613805	2.334108	2.226915	32	90	6	Sedang
013	Jantan	2 Tahun	Baik	67.53938	68.22819	66.39257	2.613805	2.334108	2.226915	32	90	9	Sedang
014	Jantan	2 Tahun	Baik	67.53938	68.22819	66.39257	2.613805	2.334108	2.226915	32	90	12	Sedang
015	Jantan	2 Tahun	Baik	67.53938	68.22819	66.39257	2.613805	2.334108	2.226915	32	90	15	Sedang
016	Jantan	2 Tahun	Baik	67.53938	68.22819	66.39257	2.613805	2.334108	2.226915	32	90	18	Sedang

Gambar 5.10 Antarmuka Halaman User Tab Data Uji
Sumber: [Implementasi]



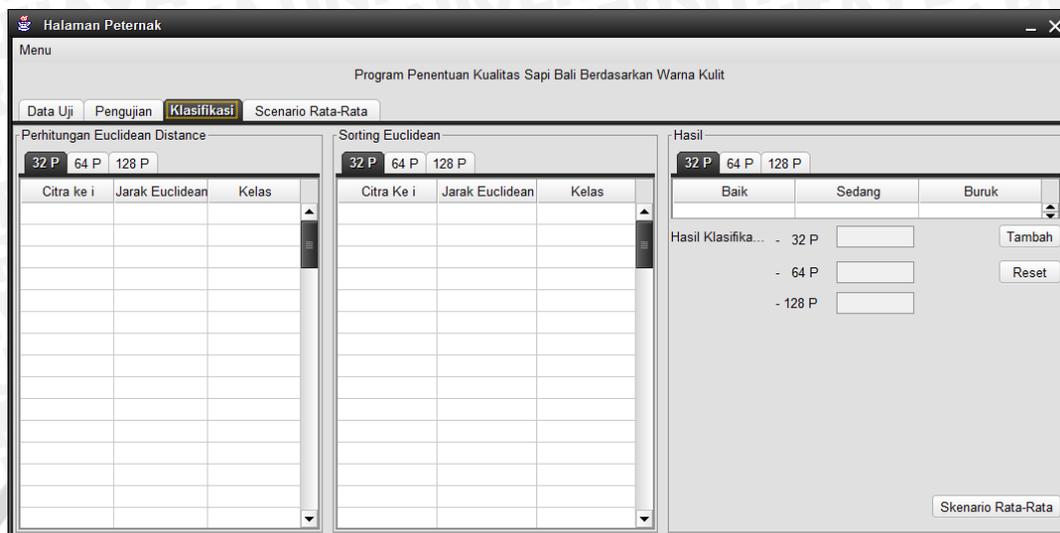
Tab selanjutnya adalah tab pengujian. Pada tab ini terdapat tombol pilih gambar untuk memilih gambar sapi bali yang akan diuji. Kemudian tombol potong untuk melakukan *crop* gambar sapi bali. Terdapat juga tombol ambil data latih untuk memilih jumlah data latih yang telah dipilih secara acak. Pada bagian kanan antarmuka terdapat tabel untuk menampilkan data latih yang digunakan untuk proses klasifikasi. Pada bagian bawah terdapat *slider* untuk menentukan nilai K dan tombol klasifikasi untuk memulai proses klasifikasi. Antarmuka halaman *user* tab pengujian ditunjukkan pada gambar 5.11.



Gambar 5.11 Antarmuka Halaman User Tab Pengujian
Sumber: [Implementasi]

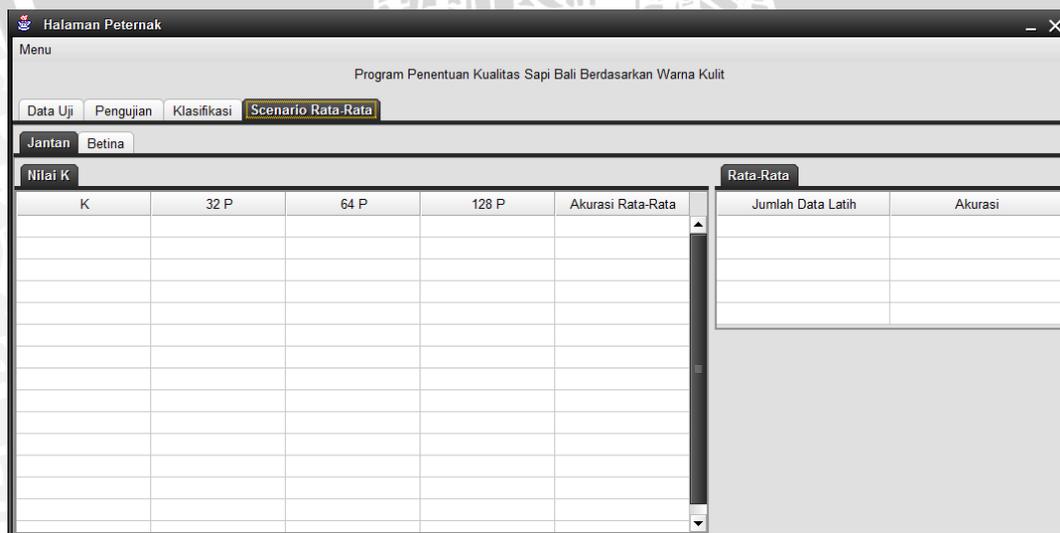
Tab selanjutnya adalah tab klasifikasi. Pada tab ini terdapat tabel jarak euclidean, tabel *sorting* euclidean, dan tabel *weighted voting*. Tabel jarak euclidean berfungsi untuk menampilkan hasil perhitungan nilai jarak data uji terhadap semua data latih yang telah terpilih pada tab sebelumnya. Tabel *sorting* euclidean berfungsi untuk menampilkan hasil pengurutan nilai euclidean mulai dari yang mempunyai nilai terkecil sampai yang mempunyai nilai terbesar. Tabel *weighted voting* berfungsi untuk menampilkan hasil perhitungan *weighted* perkelas. Tombol tambah untuk menyimpan hasil uji ke dalam database. Tombol *reset* berfungsi untuk mengosongkan semua informasi yang ada pada tab pengujian, tab klasifikasi dan tab skenario rata-rata. Tombol skenario rata-rata berfungsi untuk memproses algoritma yang digunakan dalam proses pengujian

skenario. Antarmuka Halaman User Tab Klasifikasi ditunjukkan pada gambar 5.12.



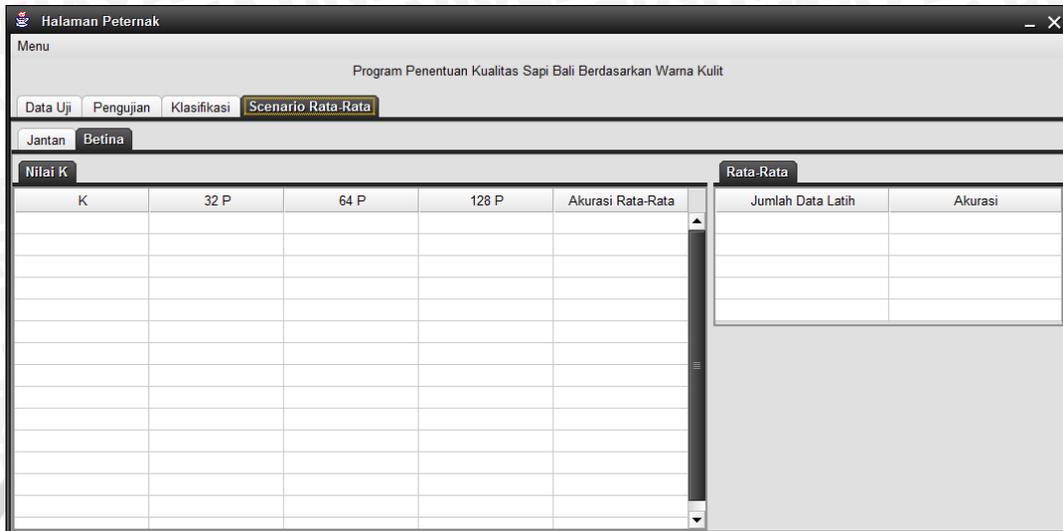
Gambar 5.12 Antarmuka Halaman User Tab Klasifikasi
Sumber: [Implementasi]

Tab selanjutnya adalah tab skenario rata-rata. Pada tab skenario terdapat subtab Jantan dan subtab Betina. Tab Jantan berfungsi untuk menampilkan akurasi hasil skenario pengujian nilai K, pixel terbaik, dan jumlah data latih sapi bali jantan. Tab Betina berfungsi untuk menampilkan akurasi hasil skenario pengujian nilai K, pixel terbaik, dan jumlah data latih sapi bali betina. Antarmuka halaman user tab skenario rata-rata sapi bali jantan ditunjukkan pada gambar 5.13.



Gambar 5.13 Antarmuka Halaman User Tab Skenario Rata-Rata Sapi Bali Jantan
Sumber: [Implementasi]

Sedangkan untuk antarmuka halaman user tab skenario rata-rata sapi bali betina ditunjukkan pada gambar 5.14.

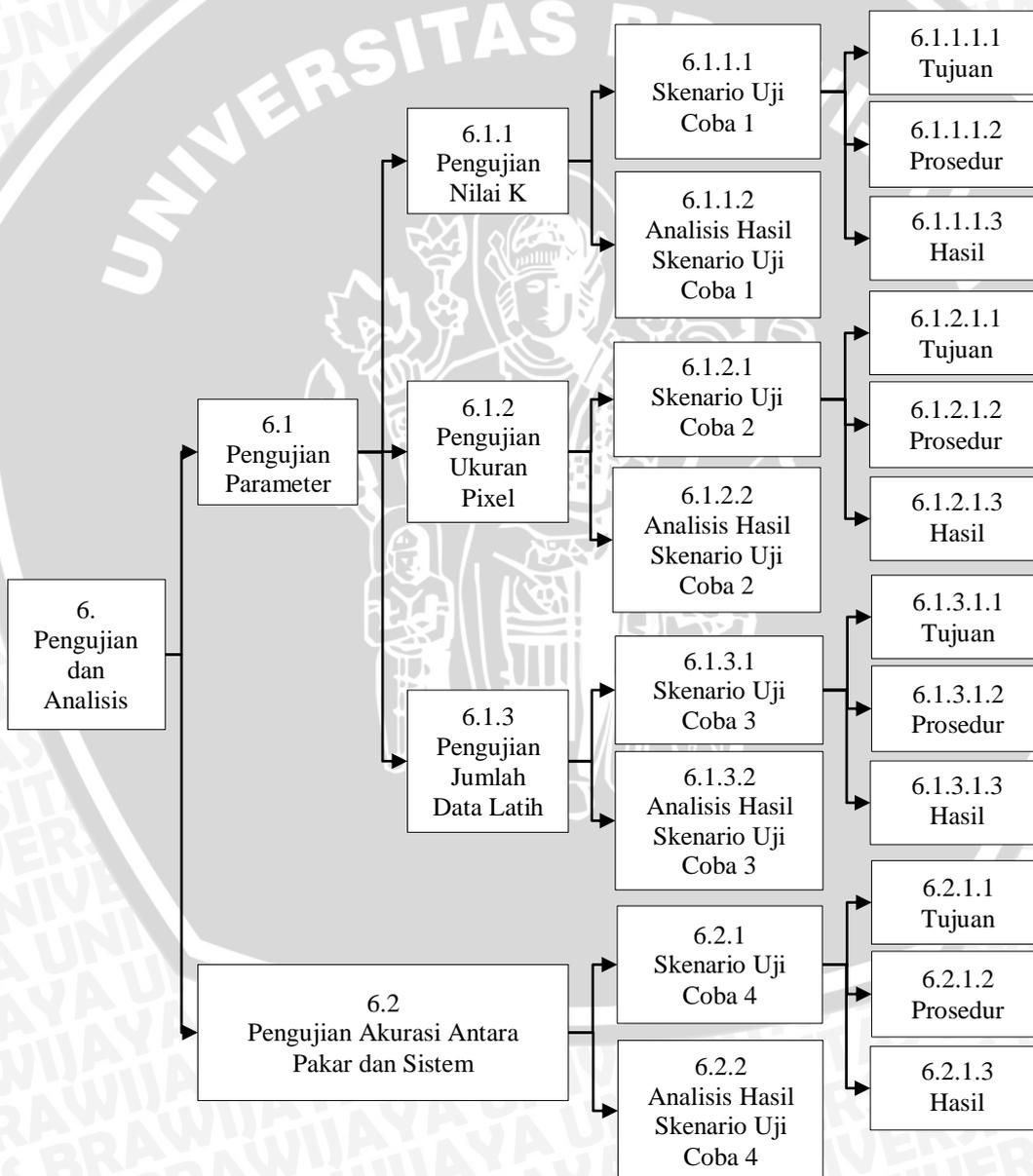


Gambar 5.14 Antarmuka Halaman User Tab Skenario Rata-Rata Sapi Bali Betina
Sumber: [Implementasi]



BAB VI PENGUJIAN DAN ANALISIS

Bab ini menjelaskan mengenai tahapan pengujian dan analisis hasil sistem pendukung keputusan pemilihan bibit unggul sapi bali berdasarkan warna kulit menggunakan metode KNN. Pengujian akan dilakukan sesuai dengan perancangan uji coba sebelumnya. Pohon pengujian dapat dilihat pada gambar 6.1.



Gambar 6.1 Pohon Pengujian
Sumber: [Pengujian]



6.1. Pengujian Parameter

Pengujian yang dilakukan pertama kali adalah pengujian parameter. Pengujian ini bertujuan untuk mencari parameter-parameter terbaik yang akan digunakan dalam pengujian sistem. Parameter-parameter tersebut meliputi nilai K, ukuran pixel, dan jumlah data latih. Data latih dan data uji yang digunakan dalam pengujian parameter ini diambil dengan menggunakan kamera SLR merk Canon. Pengaturan kamera menggunakan Mode Program (pengaturan untuk *shutter speed* dan *aperture* otomatis sedangkan pengaturan yang lain manual) dengan ISO:400 untuk semua data. Sehingga gambar yang dihasilkan sama dengan gambar yang ditangkap, tidak terlalu terang dan tidak terlalu gelap. Pengambilan gambar sambilan dilakukan 2 kali dalam satu hari saat matahari tidak terlalu terik, pada jam 8:00 – 10:00 pagi dan jam 14:00 – 16:00.

6.1.1. Pengujian Nilai K

Sub bab ini akan menjelaskan mengenai skenario uji coba pertama yang akan dilakukan yaitu menguji nilai K yang akan digunakan serta analisis hasil dari pengujian nilai K yang akan digunakan.

6.1.1.1. Skenario Uji Coba 1

Dalam skenario uji coba ini akan membahas mengenai tujuan, prosedur, dan hasil dari skenario uji coba pertama yaitu pengujian nilai K.

6.1.1.1.1. Tujuan

Skenario ini dilakukan dengan tujuan untuk mencari nilai K yang mempunyai akurasi tertinggi di semua ukuran pixel dan keseluruhan data latih untuk tiap jenis kelamin.

6.1.1.1.2. Prosedur

Sistem akan melakukan pengujian terhadap nilai K dengan rentangan nilai sebagai berikut : 3, 6, 9, 12, 15, 18, 21, 24, 27, 30. Diasumsikan nilai K yang dipilih peneliti adalah mulai dari 3. Hal tersebut karena nilai $K = 3$ memiliki kondisi minimum jika terjadi kondisi kelas yang merata maka akan diperoleh kelas baik terdapat 1, kelas sedang terdapat 1, dan kelas buruk terdapat 1. Peneliti melakukan pengujian dengan varian nilai K sebanyak 10 terhadap 30 data uji yang berasal dari dataset (5 data terakhir ditiap kelas untuk 2 jenis kelamin).

6.1.1.1.3. Hasil

Hasil pengujian nilai K untuk sapi bali jantan yang dapat dilihat pada tabel 6.1. Sedangkan untuk hasil pengujian nilai K untuk sapi bali betina dapat dilihat pada tabel 6.2.

Tabel 6.1 Akurasi Pengujian Nilai K Sapi Bali Jantan

Nilai K	Akurasi Citra 32×32 piksel (%)	Akurasi Citra 64×64 piksel (%)	Akurasi Citra 128×128 piksel (%)	Akurasi Rata-Rata (%)
3	86,67	86,67	86,67	86,67
6	80,00	86,67	86,67	84,44
9	80,00	80,00	86,67	82,23
12	80,00	80,00	86,67	82,23
15	80,00	80,00	86,67	82,23
18	80,00	80,00	86,67	82,23
21	80,00	80,00	86,67	82,23
24	80,00	80,00	86,67	82,23
27	80,00	80,00	86,67	82,23
30	80,00	80,00	86,67	82,23

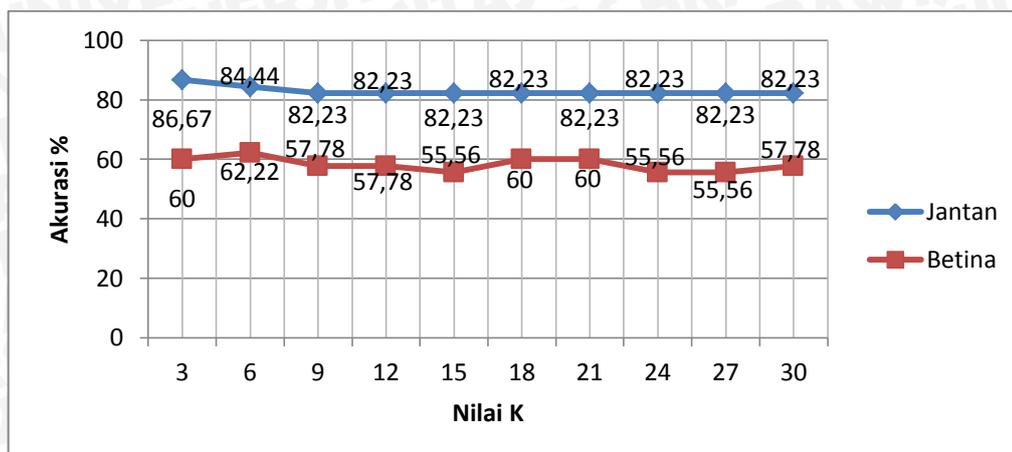
Sumber: [Pengujian][Lampiran 6]

Tabel 6.2 Akurasi Pengujian Nilai K Sapi Bali Betina

Nilai K	Akurasi Citra 32×32 piksel (%)	Akurasi Citra 64×64 piksel (%)	Akurasi Citra 128×128 piksel (%)	Akurasi Rata-Rata (%)
3	60,00	66,67	53,33	60,00
6	60,00	73,33	53,33	62,22
9	60,00	66,67	46,67	57,78
12	60,00	66,67	46,67	57,78
15	53,33	66,67	46,67	55,56
18	60,00	66,67	53,33	60,00
21	60,00	66,67	53,33	60,00
24	53,33	60,00	53,33	55,56
27	53,33	60,00	53,33	55,56
30	53,33	66,67	53,33	57,78

Sumber: [Pengujian][Lampiran 6]

Hasil akurasi pada skenario uji coba 1 direpresentasikan pada grafik perbandingan akurasi nilai K antara sapi bali jantan dan sapi bali betina yang dapat dilihat pada gambar 6.2.



Gambar 6.2 Grafik Perbandingan Akurasi Pengujian Nilai K
 Sumber: [Pengujian]

6.1.1.2. Analisis Skenario Uji Coba 1

Dari kotak tebal yang ada di tabel 6.1 dapat dilihat bahwa hasil pengujian terhadap nilai K untuk sapi bali jantan didapatkan nilai K terbaik adalah 3 dengan akurasi rata-rata 86,67 %. Sedangkan dari kotak tebal yang ada di tabel 6.2 dapat dilihat bahwa hasil pengujian terhadap nilai K untuk sapi bali betina didapatkan nilai K terbaik adalah 6 dengan akurasi rata-rata 62,22 %. Nilai standar deviasi untuk sapi jantan dan sapi bali betina sangat berpengaruh dalam pemilihan nilai K untuk proses pengklasifikasian menggunakan metode KNN. Sapi bali jantan memiliki rata-rata nilai SR = 9,48, SG = 9,50, dan SB = 9,46. Sedangkan sapi Bali betina memiliki rata-rata nilai SR = 15,58, SG = 13,55, SB = 12,52. Kecilnya nilai standar deviasi pada sapi bali jantan membuat pemilihan nilai K yang kecil bisa menghasilkan akurasi yang bagus karena kemungkinan data latih berhimpitan sangat kecil. Sedangkan pada sapi bali betina kemungkinan data latih berhimpitan sangat besar jika dilihat dari nilai standar deviasinya yang besar.

6.1.2. Pengujian Ukuran Citra

Sub bab ini akan menjelaskan mengenai skenario uji coba kedua yang akan dilakukan yaitu menguji ukuran citra yang akan digunakan serta analisis hasil dari pengujian ukuran citra yang akan digunakan.

6.1.2.1. Skenario Uji Coba 2

Dalam skenario uji coba ini akan membahas mengenai tujuan, prosedur, dan hasil dari skenario uji coba kedua yaitu pengujian ukuran citra.

6.1.2.1.1. Tujuan

Skenario ini dilakukan dengan tujuan untuk mencari ukuran pixel yang mempunyai akurasi tertinggi untuk digunakan dalam proses klasifikasi.

6.1.2.1.2. Prosedur

Pengujian ukuran citra dilakukan terhadap nilai K terbaik yang telah diujikan sebelumnya. Pengujian dilakukan dengan cara mencari nilai akurasi tiap pixel pada nilai K = 3 untuk sapi bali jantan dan K = 6 untuk sapi bali betina. Pengujian ukuran citra masih menggunakan seluruh data latih.

6.1.2.1.3. Hasil

Hasil dari pengujian ukuran citra terhadap nilai K terbaik sapi bali jantan dapat dilihat pada tabel 6.3. Sedangkan untuk hasil pengujian ukuran citra terhadap nilai K terbaik sapi bali betina dapat dilihat pada tabel 6.4.

Tabel 6.3 Akurasi Pengujian Ukuran Citra Terhadap Nilai K Terbaik Sapi Bali Jantan

Nilai K	Akurasi Citra 32×32 piksel (%)	Akurasi Citra 64×64 piksel (%)	Akurasi Citra 128×128 piksel (%)
3	86,67	86,67	86,67

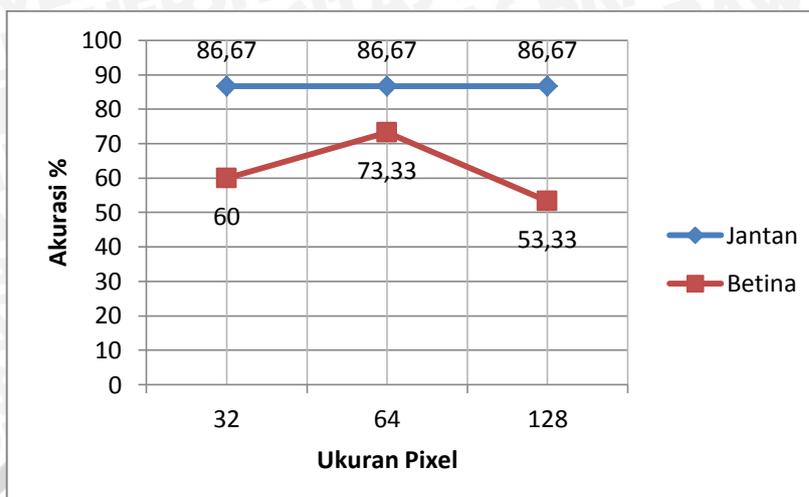
Sumber: [Pengujian]

Tabel 6.4 Akurasi Pengujian Ukuran Citra Terhadap Nilai K Terbaik Sapi Bali Betina

Nilai K	Akurasi Citra 32×32 piksel (%)	Akurasi Citra 64×64 piksel (%)	Akurasi Citra 128×128 piksel (%)
6	60,00	73,33	53,33

Sumber: [Pengujian]

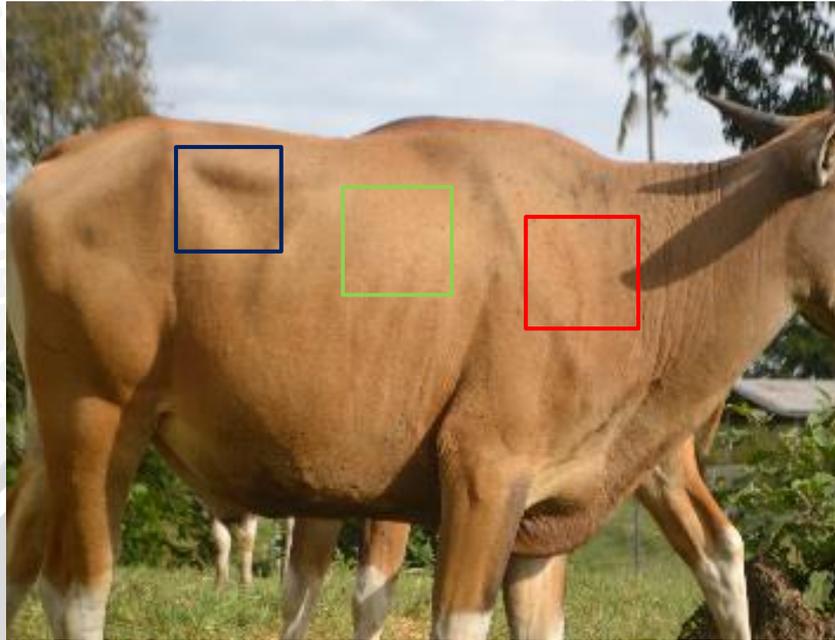
Hasil akurasi pada skenario uji coba 2 direpresentasikan pada grafik perbandingan hasil pengujian ukuran citra antara sapi bali jantan dan sapi bali betina dapat dilihat pada gambar 6.3.



Gambar 6.3 Grafik Perbandingan Akurasi Ukuran Citra
Sumber: [Pengujian]

6.1.2.2. Analisis Skenario Uji Coba 2

Tabel 6.3 mengacu baris yang telah ditandai pada tabel 6.1. Berdasarkan kotak tebal pada tabel 6.3 didapatkan hasil dari akurasi ukuran citra terbaik berada di semua ukuran piksel. Berdasarkan hasil tersebut dapat disimpulkan bahwa perbedaan ukuran citra tidak berpengaruh dalam proses mengklasifikasi kualitas sapi bali jantan berdasarkan warna kulit. Sehingga dipilih ukuran citra 128 yang akan digunakan dalam proses klasifikasi karena citra 128 merupakan ukuran citra terbesar yang sudah mewakili ukuran 64 dan 32. Tabel 6.4 mengacu baris yang telah ditandai pada tabel 6.2. Berdasarkan kotak tebal pada tabel 6.4 didapatkan hasil dari akurasi ukuran citra terbaik berada di ukuran citra 64×64 piksel dengan akurasi 73,33%. Terdapat banyak faktor yang mendukung terjadinya kesalahan, seperti bayangan dan cahaya yang tidak konstan. Letak dan posisi sewaktu melakukan pemotongan gambar sangat berpengaruh dalam keberhasilan proses klasifikasi. Semakin besar ukuran citra maka komponen warna yang ditangkap semakin banyak dan beragam.



Gambar 6.4 Posisi *Cropping*
Sumber: [Pengujian]

Gambar 6.4 menunjukkan posisi pemotongan gambar yang bisa saja terjadi pada saat pengujian. Jika pemotongan gambar seperti kotak merah dan kotak biru, memungkinkan terjadi kesalahan pada proses klasifikasi karena terdapat bayangan dari bagian sapi yang lain.

6.1.3. Pengujian Jumlah Data Latih

Sub bab ini akan menjelaskan mengenai skenario uji coba ketiga yang akan dilakukan yaitu menguji jumlah data latih yang akan digunakan serta analisis hasil dari pengujian jumlah data latih yang akan digunakan.

6.1.3.1. Skenario Uji Coba 3

Dalam skenario uji coba ini akan membahas mengenai tujuan, prosedur, dan hasil dari skenario uji coba ketiga yaitu pengujian jumlah data latih.

6.1.3.1.1. Tujuan

Skenario ini dilakukan dengan tujuan untuk mencari jumlah data latih yang mempunyai akurasi tertinggi untuk digunakan dalam proses klasifikasi.

6.1.3.1.2. Prosedur

Pengujian jumlah data latih dilakukan terhadap nilai K terbaik dan ukuran citra terbaik yang telah diujikan sebelumnya. Jumlah data latih yang akan diujikan adalah 30, 45, 60, 75, dan 90 data latih yang dipilih secara random oleh sistem.

Data uji yang digunakan sama dengan data uji yang digunakan pada pengujian nilai K yaitu 30 data uji yang berada di semua kelas pada kedua jenis kelamin.

6.1.3.1.3. Hasil

Hasil pengujian akurasi jumlah data latih terhadap nilai K dan ukuran pixel terbaik untuk sapi bali jantan dapat dilihat pada tabel 6.5. Sedangkan hasil akurasi pengujian jumlah data latih terhadap nilai K dan ukuran pixel terbaik sapi bali betina dapat dilihat pada tabel 6.6.

Tabel 6.5 Akurasi Pengujian Jumlah Data Latih Terhadap Nilai K dan Ukuran Citra Terbaik Sapi Bali Jantan

Jumlah Data Latih Yang Diujikan	Akurasi Citra 128×128 Piksel (%)
30	86,67
45	100,00
60	86,67
75	86,67
90	86,67

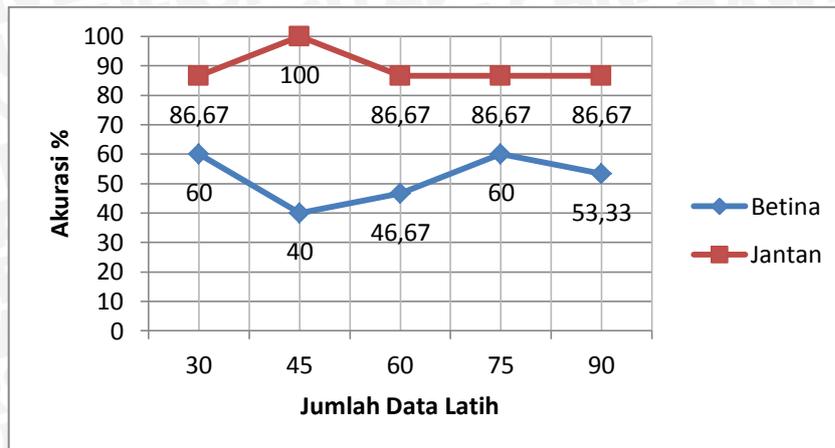
Sumber: [Pengujian][Lampiran 6]

Tabel 6.6 Hasil Akurasi Pengujian Jumlah Data Latih Terhadap Nilai K Dan Ukuran Citra Terbaik Sapi Bali Betina

Jumlah Data Latih Yang Diujikan	Akurasi Citra 64×64 Piksel (%)
30	60,00
45	40,00
60	46,67
75	60,00
90	53,33

Sumber: [Pengujian][Lampiran 6]

Hasil akurasi pada skenario uji coba 3 direpresentasikan pada grafik perbandingan hasil pengujian jumlah data latih antara sapi bali jantan dan sapi bali betina dapat dilihat pada gambar 6.4.



Gambar 6.4 Grafik Perbandingan Hasil Pengujian Jumlah Data Latih Antara Sapi Bali Jantan dan Sapi Bali Betina

Sumber: [Pengujian]

6.1.3.2. Analisis Skenario Uji Coba 3

Berdasarkan kotak tebal pada tabel 6.5 dapat dilihat jumlah data latih yang memiliki tingkat akurasi terbaik untuk sapi bali jantan adalah 45 data latih dengan akurasi 100%. Sedangkan berdasarkan kotak tebal pada tabel 6.6 diketahui nilai akurasi pengujian jumlah data latih untuk sapi bali betina memiliki nilai yang sama yaitu 60,00% pada jumlah data latih 30 dan 75. Sehingga dipilih jumlah data latih 30 yang digunakan dalam klasifikasi karena dengan jumlah data latih yang sedikit, kinerja sistem dalam mengklasifikasi lebih ringan.

6.2. Pengujian Akurasi Antara Pakar dan Sistem

Sub bab ini akan menjelaskan mengenai skenario uji coba keempat yang akan dilakukan yaitu menguji akurasi antara pakar dan sistem serta analisi dari pengujian akurasi antara pakar dan sistem ini.

6.2.1. Skenario Uji Coba 4

Dalam skenario uji coba ini akan membahas mengenai tujuan, prosedur, dan hasil dari skenario uji coba keempat yaitu pengujian akurasi antara pakar dan sistem.

6.2.1.1. Tujuan

Skenario ini dilakukan dengan tujuan untuk mengetahui hasil akurasi dari sistem dengan menggunakan nilai K terbaik, ukuran citra terbaik, dan jumlah data terbaik yang didapat dari skenario uji coba sebelumnya.

6.2.1.2. Prosedur

Pengujian terhadap sistem ini dilakukan dengan menggunakan hasil terbaik dari pengujian-pengujian sebelumnya. Pengujian sapi bali jantan dilakukan terhadap nilai $K=3$, ukuran citra 128×128 piksel, dan jumlah data latih 45. Sedangkan untuk pengujian sapi bali betina dilakukan terhadap nilai $K=6$, ukuran citra 64×64 piksel, dan jumlah data latih 30.

6.2.1.3. Hasil

Hasil pengujian akurasi antara pakar dan sistem untuk sapi bali jantan dapat dilihat pada tabel 6.7. Sedangkan hasil pengujian akurasi antara pakar dan sistem untuk sapi bali betina dapat dilihat pada tabel 6.8.

Tabel 6.7 Hasil Pengujian Akurasi Antara Pakar dan Sistem untuk Sapi Bali Jantan

ID Sapi	Weighted Voting			Prediksi Pakar	Klasifikasi Sistem	Hasil
	Baik	Sedang	Buruk			
SB-11	0,00478	7,31E-4	0	Baik	Baik	Benar
SB-12	0,00178	2,15E-4	0	Baik	Baik	Benar
SB-13	2,34E-4	5,76E-5	0	Baik	Baik	Benar
SB-14	0,00737	0	0	Baik	Baik	Benar
SB-15	2,18E-4	5,51E-5	0	Baik	Baik	Benar
SB-26	0	0,01858	0	Sedang	Sedang	Benar
SB-27	0	0,01856	0	Sedang	Sedang	Benar
SB-28	0	0,00300	0	Sedang	Sedang	Benar
SB-29	0	0,07756	0	Sedang	Sedang	Benar
SB-30	0,00384	0,00953	0	Sedang	Sedang	Benar
SB-41	0	0	0,18193	Buruk	Buruk	Benar
SB-42	4,30E-4	0	9,01E-4	Buruk	Buruk	Benar
SB-43	0,00130	0	0,00352	Buruk	Buruk	Benar
SB-44	0	0	4,13E-4	Buruk	Buruk	Benar
SB-45	0	0	3,88E-4	Buruk	Buruk	Benar

Sumber: [Pengujian]

Tabel 6.8 Hasil Pengujian Akurasi Antara Pakar dan Sistem untuk Sapi Bali Betina

ID Sapi	Weighted Voting			Prediksi Pakar	Klasifikasi Sistem	Hasil
	Baik	Sedang	Buruk			
SB-56	0,25852	0,00755	0,00614	Baik	Baik	Benar
SB-57	0,01391	0,00855	0,00325	Baik	Baik	Benar
SB-58	0,00179	0,00247	0,00422	Baik	Buruk	Salah
SB-59	0,00183	0,00248	0,00421	Baik	Buruk	Salah
SB-60	0,00781	0,00428	0,00475	Baik	Baik	Benar
SB-71	0,00821	0,01682	0,01327	Sedang	Sedang	Benar
SB-72	0,00479	0,01085	0,00977	Sedang	Sedang	Benar

SB-73	0,00888	0,01685	0,00303	Sedang	Sedang	Benar
SB-74	2,04E-4	4,57E-4	0,00381	Sedang	Buruk	Salah
SB-75	0,00805	0,00848	0,00133	Sedang	Sedang	Benar
SB-86	1,70E-4	5,17E-4	0,00199	Buruk	Buruk	Benar
SB-87	2,76E-4	0,00143	0,00130	Buruk	Sedang	Salah
SB-88	3,37E-4	7,30E-4	0,00488	Buruk	Buruk	Benar
SB-89	5,39E-4	0,00105	0,00424	Buruk	Buruk	Benar
SB-90	0,00445	0,05423	0	Buruk	Sedang	Salah

Sumber: [Pengujian]

Dengan menggunakan persamaan (2-5) didapatkan akurasi untuk pengujian sapi bali jantan adalah sebagai berikut:

$$\text{akurasi sapi bali jantan} = \frac{15}{15} \times 100\% = 100\%$$

Sedangkan untuk pengujian SPK sapi bali betina adalah sebagai berikut:

$$\text{akurasi sapi bali betina} = \frac{10}{15} \times 100\% = 66,67\%$$

6.2.2. Analisis Hasil Skenario Uji Coba 4

Hasil uji coba pengujian akurasi antara pakar dan sistem menunjukkan bahwa untuk klasifikasi Sapi Bali berjenis kelamin jantan didapatkan tingkat akurasi sebesar 100%, sedangkan untuk klasifikasi Sapi Bali berjenis kelamin betina didapatkan tingkat akurasi sebesar 66,67%. Klasifikasi sapi bali betina memiliki akurasi yang rendah dikarenakan gambar sapi bali betina yang diproses tidak bagus dan terdapat noise. Hal tersebut terjadi karena terdapat kendala dalam proses pengambilannya, seperti kamera yang hanya menggunakan lensa fix (tidak dapat digunakan untuk menangkap gambar objek yang berjarak jauh) dan cahaya matahari yang terlalu terik sehingga menimbulkan bayangan saat pengambilan gambar dan membuat perbedaan warna sapi bali betina tiap kelas tidak begitu jelas terlihat.

BAB VII

PENUTUP

Bab ini akan membahas mengenai kesimpulan dari penelitian skripsi dan saran untuk penelitian berikutnya.

7.1. Kesimpulan

Berdasarkan hasil analisa terhadap pengujian sistem yang dilakukan dalam penelitian ini dapat disimpulkan beberapa hal sebagai berikut:

1. Hasil yang diperoleh dari pengujian akurasi rata-rata nilai K untuk sapi bali jantan adalah nilai $K=3$ dengan akurasi 86,67%. Sedangkan hasil yang diperoleh dari pengujian rata-rata nilai K untuk sapi bali betina adalah nilai $K=6$ dengan akurasi 62,22%. Nilai standar deviasi untuk sapi jantan dan sapi bali betina sangat berpengaruh dalam pemilihan nilai K untuk proses pengklasifikasian menggunakan metode KNN. Sapi bali jantan memiliki rata-rata nilai $SR = 9,48$, $SG = 9,50$, dan $SB = 9,46$. Sedangkan sapi Bali betina memiliki rata-rata nilai $SR = 15,58$, $SG = 13,55$, $SB = 12,52$. Kecilnya nilai standar deviasi pada sapi bali jantan membuat pemilihan nilai K yang kecil bisa menghasilkan akurasi yang bagus karena kemungkinan data latih berhimpitan sangat kecil. Sedangkan pada sapi bali betina kemungkinan data latih berhimpitan sangat besar jika dilihat dari nilai standar deviasinya yang besar.
2. Hasil yang diperoleh dari pengujian akurasi ukuran citra di nilai $K=3$ untuk sapi bali jantan adalah sama di semua ukuran, yaitu 86,67%. Perbedaan ukuran citra tidak berpengaruh dalam proses mengklasifikasi kualitas sapi bali jantan berdasarkan warna kulit. Sehingga dipilih ukuran citra 128×128 piksel yang akan digunakan dalam proses klasifikasi karena citra 128×128 piksel merupakan ukuran citra terbesar yang sudah mewakili ukuran citra 64×64 piksel dan 32×32 piksel. Sedangkan hasil pengujian akurasi ukuran pixel terbaik untuk sapi bali betina adalah citra 64×64 piksel dengan nilai akurasi 73,33%.

3. Hasil yang diperoleh dari pengujian jumlah data latih yang memiliki tingkat akurasi terbaik untuk sapi bali jantan adalah 45 data latih dengan akurasi 100%. Sedangkan hasil yang diperoleh dari pengujian jumlah data latih untuk sapi bali betina adalah tingkat akurasi tertinggi berada pada jumlah data latih 30 dan 75 dengan nilai akurasi 60%. Sehingga dipilih jumlah data latih 30 yang digunakan dalam klasifikasi karena dengan jumlah data latih yang sedikit, kinerja sistem dalam mengklasifikasi lebih ringan.
4. Dari pengujian parameter didapatkan nilai-nilai yang paling optimal untuk digunakan pada pengujian sistem. Nilai-nilai tersebut adalah nilai $K=3$, ukuran citra 128×128 piksel, dan jumlah data latih 45 untuk sapi Bali jantan. Sedangkan untuk sapi Bali betina menggunakan nilai $K=6$, ukuran citra 64×64 piksel, dan jumlah data latih 30. Hasil uji coba pengujian akurasi antara pakar dan sistem menunjukkan bahwa untuk klasifikasi Sapi Bali berjenis kelamin jantan didapatkan tingkat akurasi sebesar 100%, sedangkan untuk klasifikasi Sapi Bali berjenis kelamin betina didapatkan tingkat akurasi sebesar 66,67%. Klasifikasi sapi bali betina memiliki akurasi yang rendah karena terdapat kendala dalam pengambilan gambar dan posisi daerah gambar sapi bali yang akan dipotong terdapat noise.

7.2. Saran

Saran untuk penelitian berikutnya dari skripsi ini adalah sebagai berikut:

1. Mengimplementasikan metode KNN dengan perhitungan jarak selain *euclidean distance* untuk dapat membandingkan tingkat akurasi.
2. Dapat melakukan penambahan jumlah keragaman dataset karena dataset yang digunakan dalam skripsi ini hanya sebanyak 15 data gambar untuk setiap kelas di jenis kelamin jantan dan betina.
3. Dalam proses pengambilan data gambar sebaiknya menggunakan peralatan yang sesuai dengan kondisi lapangan, seperti kamera lensa tele. Dikarenakan kondisi lapangan yang sangat luas dan kondisi objek yang selalu bergerak bila didekati.

4. Untuk kendala dalam pencahayaan sebaiknya tidak mengambil gambar sapi bali saat matahari terik untuk mengurangi kemungkinan terjadi bayangan. Bisa menggunakan format komponen warna lain yang lebih tahan atau tidak terpengaruh terhadap cahaya.



DAFTAR PUSTAKA

- [1] Andaruresmi, Ratri, 2009, "Indentifikasi Tipe Wilayah Berbasis Pengolahan Citra Penginderaan Jarak Jauh", hal. 121-126, Prosiding SENTIA 2009, Politeknik Negeri Malang
- [2] Bandini, Yusni, 2003, "Sapi Bali", Penebar Swadaya, Jakarta
- [3] Bradley, Julia Case, 2002, "Programming with Java", McGraw-Hill Companies, 1221 Avenue of the Americas, New York, NY, 10020
- [4] Bressan, Stephane dan Catania, Barbara, 2005, "Introduction to Database System", McGraw-Hill Education (Asia)
- [5] Gonzalez, Rafael C. Dan Woods, Richard E., 2001, "Digital Image Processing second Edition", Prentice Hall, Upper Saddle River, New Jersey 07458
- [6] Guntoro, Suprio, 2002, "Membudidayakan Sapi Bali", Kanisius, Yogyakarta
- [7] Han, Jiawei dan Kamber, Micheline, 2006, "Data Mining: Concepts and Techniques Second Edition", Elsevier, 500 Sansome Street, Suite 400, San Francisco, CA 94111
- [8] Kadir, Abdul, 2004, "Dasar Pemrograman JAVA™ 2", ANDI, Yogyakarta
- [9] Prasetyo, Didik Dwi, 2003, "Belajar Sendiri Administrasi Database Server MySQL", Elex Media Komputindo, Jakarta
- [10] Prasetyo, Eko, 2012, "Data Mining :Konsep dan Aplikasi Menggunakan MATLAB", ANDI, Yogyakarta
- [11] Safitri, Mutihera, 2012, "Deteksi Hutan Mangrovedi Pantai Utara Jakarta Menggunakan Citra dari Google *Earth* dengan Metode Curvelet", hal. 1-9, Fakultas Teknik Elektro dan Komunikasi IT Telkom, Bandung
- [12] Soekardono, dkk, 2009, "Identifikasi *Grade* Sapi Bali Betina Bibit dan Koefisien Reproduksi Sapi Betina di Propinsi Nusa Tenggara Barat, Buletin Peternakan Vol. 33(2): 74-80, Jurusan Ilmu Produksi Ternak, Fakultas Peternakan, Universitas Mataram
- [13] Spiegel, R. Murray, 1992, "*Schaum's Outline of Theory and Problems of Statistics 2/ed*", McGraw-Hill, Inc, United States of America

- [14] Turban, Efraim, 2005, "Decision Support System and Intelligent System – 7th Ed", Pearson Education, Inc. Upper Saddle River, New Jersey
- [15] Whidhiasih, Retno Nugroho, dkk, 2013, "Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan KNN dan LDA", Jurnal Penelitian Ilmu Komputer, System Embedded & Logic 1(1) : 29-35, Program Studi Teknik Komputer Universitas Islam "45" Bekasi
- [16] Wicaksono, Ady, 2002, "Dasar-Dasar Pemrograman Java 2", Elex Media Komputindo, Jakarta

