

**PENINGKATAN PERFORMA KLASIFIKASI JENIS
KOPRA MENGGUNAKAN ALGORITMA K-
NEAREST NEIGHBOUR BERDASARKAN
EKSTRAKSI FITUR**

Skripsi



Oleh

PUSPITASARI

NIM : 1119091000003

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH

JAKARTA

2023 M / 1445 H

KATA PENGANTAR

Assalamualaikum Wr. Wb

Puji syukur atas kehadiran Allah SWT yang telah memberikan rahmat dan hidayah-Nya, sehingga penulis diberikan kesabahan dan kemudahan selama proses penulisan skripsi ini dari awal hingga selesai. Tak lupa pula shalawat serta salam penulis haturkan kepada junjungan Nabi Muhammad SAW beserta keluarga dan para sahabatnya.

Skripsi ini ditulis sebagai syarat kelulusan untuk mendapatkan gelar Sarjana Komputer (S.Kom) pada program studi Teknik Informatika UIN Syarif Hidayatullah Jakarta. Proses penyusunan skripsi ini bisa berjalan dengan lancar salah satunya karena dukungan dari berbagai pihak yang selalu memberikan dukungan, motivasi, bantuan doa. Maka dari itu, dalam kesempatan kali ini penulis ingin mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan nikmat, rahmat, dan karunia-Nya kepada penulis.
2. Bapak Husni Teja Sukmana, S.T., M.Sc, Ph.D selaku dekan fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta.
3. Ibu Dewi Khairani, M.Sc. selaku Ketua Program Studi Teknik Informatika dan Bapak Saepul Aripiyanto, M.Kom selaku Sekretaris Program Studi Teknik Informatika.
4. Bapak Husni Teja Sukmana, S.T., M.Sc, Ph.D selaku Dosen Pembimbing I dan Ibu Dewi Khairani, M.Sc. selaku Dosen Pembimbing II yang telah bersedia meluangkan waktunya untuk membimbing, mengarahkan, dan memberikan masukan kepada penulis dalam proses pembuatan hingga penyelesaian skripsi ini.
5. Seluruh dosen di Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta yang telah memberikan banyak ilmu dan pengalaman yang bermanfaat bagi penulis selama masa perkuliahan.

6. Kedua orang tua penulis, Bapak Suparso dan Ibu Darti serta ke empat kakak dan abang penulis yang selalu mendoakan dan memberikan nasihat serta motivasi penulis untuk menyelesaikan skripsi dengan baik.
7. Muhammad Althaf yang selalu memberikan semangat, motivasi, dan bantuan kepada penulis dalam penyusunan skripsi hingga selesai.
8. Teman-taman seperjuangan dalam revisi dan mengejar wisuda, khususnya Azzahra Putri Komara dan Iva Alfiyanti yang telah membantu penulis untuk menyelesaikan skripsi ini.
9. Muhammad Landy Hakim dan William Santoso yang telah memberikan dukungan teknis kepada penulis untuk mengerjakan skripsi dengan cepat dan tepat, serta seluruh teman-teman Teknik Informatika Angkatan 2019, yang secara langsung maupun tidak langsung telah membantu penulis dalam proses perkuliahan.
10. Seluruh pihak yang tidak dapat disebutkan satu persatu yang telah mendukung penulisan skripsi ini.
11. Diri sendiri yang telah berjuang dan berusaha semaksimal mungkin dalam proses pembuatan skripsi ini.

Penulis menyadari bahwa masih banyak kekurangan dan hal yang harus diperbaiki pada skripsi ini karena beberapa keterbatasan yang penulis miliki. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan sehingga dapat dilakukan perbaikan di masa yang akan datang. Penulis berharap skripsi ini dapat bermanfaat bagi penulis dan pembaca serta memberikan kontribusi pada keilmuan.

Wassalamualaikum Wr. Wb.

Tangerang Selatan, 21 Agustus 2023



Puspitasari
11190910000003

Nama : Puspitasari
Program Studi : Teknik Infomatika
Judul : Peningkatan Performa Klasifikasi Jenis Kopra Menggunakan Algoritma K-Nearest Neighbour Berdasarkan Ekstraksi Fitur

ABSTRAK

Kopra adalah daging buah kelapa yang dikeringkan. Berdasarkan kualitas dan penggunaannya, kopra terbagi menjadi tiga jenis, yaitu kopra edible, reguler, dan reject. Pemisahan dan pengklasifikasian jenis kopra penting dilakukan untuk memenuhi kebutuhan hilirisasi pada industri kopra. Saat ini, berdasarkan hasil wawancara yang dilakukan dengan salah satu pedagang kopra di Kabupaten Indragiri Hilir, Provinsi Riau, pengklasifikasian jenis kopra umumnya masih dilakukan secara manual. Proses ini tentunya memerlukan tenaga, waktu, dan biaya yang cukup tinggi. Penelitian ini akan membuat model klasifikasi jenis kopra berdasarkan ekstraksi fitur menggunakan algoritma klasifikasi k-nearest neighbour. Sumber data yang digunakan dalam penelitian ini berasal dari Gudang kopra yang terletak di Kabupaten Indragili Hilir, Provinsi Riau. Dataset terdiri dari tiga jenis kopra non telungkup, yaitu edible, regular, dan reject dengan masing-masing jenis berjumlah 613 data citra. Ekstraksi fitur yang digunakan meliputi warna, bentuk, dan tekstur. Untuk ekstraksi fitur warna, digunakan metode *color moment* yang mencakup *mean RGB*, *HSV*, dan *Grayscale*. Sedangkan ekstraksi fitur bentuk menggunakan *area* dan *perimeter*. Dan untuk ekstraksi fitur tekstur menggunakan *GLCM* (*Gray Level Co-occurrence Matrix*) dengan fitur-fitur seperti *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*. Klasifikasi dilakukan pada 7 skenario yang masing-masing skenario mempertimbangkan variasi fitur. Hasil akurasi tertinggi yang berhasil dihasilkan sebesar 84% menggunakan kombinasi fitur warna & tekstur, serta gabungan dari ketiga fitur: warna, bentuk, dan tekstur.

Kata Kunci : Kopra, Klasifikasi, Ekstraksi Fitur, Warna, Bentuk, Tekstur, K-Nearest Neighbour
Jumlah Pustaka : 27 jurnal dan 6 buku
Jumlah Halaman : VI Bab + viii halaman + 143 halaman + 34 gambar + 10 tabel

Name : Puspitasari
Study Program : Informatics Engineering
Title : Performance Improvement of Copra Classification Using K-Nearest Neighbor Algorithm Based on Feature Extraction

ABSTRACT

Copra is the dried flesh of the coconut fruit. Based on the quality and use, copra is divided into three types, namely edible, regular and rejected copra. Separation and classification of copra types is important to meet the downstream needs of the copra industry. Currently, based on the results of an interview conducted with a copra trader in Indragiri Hilir Regency, Riau Province, the classification of copra species is generally still done manually. This process certainly requires a lot of effort, time, and costs. This research will create a classification model for copra species based on feature extraction using the k-nearest neighbor classification algorithm. The data source used in this study came from the copra warehouse located in Indragili Hilir Regency, Riau Province. The dataset consists of three types of non-faced copra, namely edible, regular, and reject with 613 image data for each type. The feature extraction used includes color, shape, and texture. For color feature extraction, the color moment method is used which includes the mean RGB, HSV, and Grayscale. While extraski feature shape using area and perimeter. And for texture feature extraction using GLCM (Gray Level Co-occurrence Matrix) with features such as contrast, dissimilarity, homogeneity, energy, and correlation. Classification is carried out in 7 scenarios, each of which considers the variety of features. The highest accuracy results were successfully produced at 84% using a combination of color & texture features, as well as a combination of the three features: color, shape, and texture.

Keywords : Copra, Classification, Feature Extraction, Color, Shape, Texture, K-Nearest Neighbor
Bibliography : 27 journal dan 6 book
Number of Pages : VI Chapter + viii pages + 143 pages + 34 images + 10 table

DAFTAR ISI

KATA PENGANTAR	i
ABSTRAK	iii
ABSTRACT	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR	ix
DAFTAR TABEL.....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	6
1.3 Batasan Masalah.....	6
1.3.1 Proses	6
1.3.2 Metode.....	7
1.3.3 <i>Tools</i>	7
1.4 Tujuan Penelitian.....	7
1.5 Manfaat Penelitian.....	8
1.5.1 Manfaat Bagi Universitas.....	8
1.5.2 Manfaat Bagi Pembaca	8
1.6 Metodologi Penelitian	8
1.6.1 Metode Pengumpulan Data	8
1.6.2 Metode Pengembangan Model.....	8
1.7 Sistematika Penelitian	9
BAB II LANDASAN TEORI	11
2.1 Kelapa.....	11

2.2	Jenis-jenis Kelapa.....	11
2.3	Kopra	12
2.4	Pengolahan Citra	13
2.5	<i>Preprocessing</i> Citra.....	14
2.6	Citra Digital	15
2.6.1	Definisi Citra Digital.....	15
2.6.2	Pembentukan Citra Digital	16
2.6.3	Jenis-jenis Citra Digital.....	17
2.7	Ekstraksi Fitur	19
2.7.1	Ekstraksi Fitur Warna	19
2.7.2	Ekstraksi Fitur Bentuk.....	22
2.7.3	Ekstraksi Fitur Tekstur.....	22
2.8	<i>Independent dan Dependent Value</i>	25
2.9	<i>Feature Scaling</i>	25
2.10	<i>Artificial Intelligence</i>	26
2.11	<i>K-Nearest Neighbors</i> (K-NN)	31
2.12	Mengukur Performansi	33
2.12.1	Akurasi	33
2.12.2	<i>Confusion Matrix</i>	34
2.12.3	<i>Classification Report</i>	35
2.13	Metode Simulasi.....	35
2.14	Studi Pustaka	39
BAB III METODE PENELITIAN	45
3.1	Metode Pengumpulan Data	45
3.1.1	Studi Pustaka.....	45

3.1.2	Wawancara.....	45
3.2	Metode Pengembangan Model	45
3.2.1	<i>Problem Formulation</i>	45
3.2.2	<i>Conceptual Model</i>	46
3.2.3	<i>Collection of Input Data</i>	46
3.2.4	<i>Modeling</i>	46
3.2.5	<i>Simulation</i>	47
3.2.6	<i>Verification and Validation</i>	47
3.2.7	<i>Experiment</i>	47
3.2.8	<i>Output Analysis</i>	48
3.3	Alur Penelitian.....	48
BAB IV IMPLEMENTASI		50
4.1	Identifikasi Kebutuhan Sistem	50
4.2	<i>Problem Formulation</i>	51
4.3	<i>Conceptual Model</i>	51
4.4	<i>Collection of Input Data</i>	57
4.5	<i>Modeling</i>	58
4.6.1	Perhitungan Manual Ekstraksi Fitur Warna	58
4.6.2	Perhitungan Manual Ekstraksi Fitur Bentuk	61
4.6.3	Perhitungan Manual Ekstraksi Fitur Tekstur	63
4.6.4	Pengklasifikasian dengan K-Nearest Nighbour	67
4.7	<i>Simulation</i>	74
BAB V HASIL DAN PEMBAHASAN		86
5.1	<i>Verify and Validation</i>	86
5.2	<i>Design of Experiment</i>	87

5.3	<i>Output Analysis</i>	88
BAB VI PENUTUP		119
6.1	Kesimpulan.....	119
6.2	Saran.....	119
DAFTAR PUSTAKA		121

DAFTAR GAMBAR

Gambar 2. 1 Pengolahan Citra	14
Gambar 2. 2 Koordinat Citra Digital.....	16
Gambar 2. 3 Perbedaan Pemograman Tradisional dan <i>Machine Learning</i>	28
Gambar 2. 4 <i>Supervised Learning</i>	29
Gambar 2. 5 <i>Unsupervised Learning</i>	30
Gambar 2. 6 <i>Reinforcement Learning</i>	30
Gambar 2. 7 Ilustrasi Algoritma K-NN.....	31
Gambar 3. 1 Alur Penelitian.....	49
Gambar 4. 1 Pra-Proses Simulasi Sistem.....	52
Gambar 4. 2 Proses Akuisisi Citra	53
Gambar 4. 3 Hasil Resize Citra Kopra Edible	54
Gambar 4. 4 Hasil Resize Citra Kopra Reguler	54
Gambar 4. 5 Hasil Resize Citra Kopra Reject.....	54
Gambar 4. 6 Hasil Penghapusan Latar Belakang Citra Kopra Edible	55
Gambar 4. 7 Hasil Penghapusan Latar Belakang Citra Kopra Reguler	55
Gambar 4. 8 Hasil Penghapusan Latar Belakang Citra Kopra Reject.....	55
Gambar 4. 9 Hasil Cropping Citra Kopra Edible	55
Gambar 4. 10 Hasil Cropping Citra Kopra Reguler.....	55
Gambar 4. 11 Hasil Cropping Citra Kopra Reject	56
Gambar 4. 12 Contoh Sampel Kopra Edible.....	57
Gambar 4. 13 Contoh Sampel Kopra Reguler	58
Gambar 4. 14 Contoh Sampel Kopra Reject.....	58
Gambar 4. 15 Citra RGB 3×3	59
Gambar 4. 16 Citra HSV 3×3	60
Gambar 4. 17 Citra Grayscale 3×3	61
Gambar 4. 18 Citra Grayscale 5×5	62
Gambar 4. 19 Citra Biner 5×5	62
Gambar 5. 1 Confusion Matrix Skenario 1	89
Gambar 5. 2 Classification Report Skenario 1	90

Gambar 5. 3 Confusion Matrix Skenario 2	92
Gambar 5. 4 Classification Report Skenario 2.....	93
Gambar 5. 5 Confusion Matrix Skenario 3	95
Gambar 5. 6 Classification Report Skenario 3.....	96
Gambar 5. 7 Confusion Matrix Skenario 4	99
Gambar 5. 8 Classification Report Skenario 4.....	100
Gambar 5. 9 Confusion Matrix Skenario 5	102
Gambar 5. 10 Classification Report Skenario 5.....	103
Gambar 5. 11 Confusion Matrix Skenario 6	105
Gambar 5. 12 Classification Report Skenario 6.....	106
Gambar 5. 13 Confusion Matrix Skenario 7	108
Gambar 5. 14 Classification Report Skenario 7.....	109
Gambar 5. 15 Hasil Evaluasi Model berdasarkan Akurasi	112
Gambar 5. 16 Hasil Evaluasi Model berdasarkan Precision	113
Gambar 5. 17 Hasil Evaluasi Model berdasarkan Recall.....	114
Gambar 5. 18 Hasil Evaluasi Model berdasarkan F1-score	115
Gambar 5. 19 Hasil Evaluasi Model Secara Keseluruhan	116

DAFTAR TABEL

Tabel 2. 1 <i>Confusion Matrix</i>	34
Tabel 2. 2 Studi Pustaka.....	39
Tabel 4. 1 Spesifikasi Perangkat Keras.....	50
Tabel 4. 2 Spesifikasi Perangkat Lunak.....	50
Tabel 4. 3 Contoh Data Training Citra Kopra.....	68
Tabel 4. 4 Contoh Data Testing Citra Kopra	68
Tabel 4. 5 Nilai Hasil Perhitungan Jarak Data <i>Training</i> dengan Data <i>Testing</i>	72
Tabel 4. 6 Hasil Pengurutan Nilai Jarak.....	72
Tabel 5.1 Skenario Pengujian	87
Tabel 5. 2 Hasil Akurasi Pengujian.....	111

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia merupakan negara yang memiliki iklim tropis. Terdapat banyak tanaman yang dapat tumbuh pada negara yang beriklim tropis, salah satunya adalah tanaman kelapa. Tanaman kelapa merupakan salah satu tanaman yang sering kali dibudidayakan oleh masyarakat sekitar karena memiliki nilai ekonomis yang tinggi. Hampir semua bagian tanaman kelapa dapat memberikan manfaat, mulai dari akar, batang, buah, hingga daunnya memiliki manfaat tersendiri (Rahayu Marlis et al., 2021). Buah kelapa adalah bagian dari tanaman kelapa yang jika diolah memberikan manfaat untuk kehidupan manusia. Salah satu bentuk pengolahannya adalah dengan menjadikannya kopra. Kopra merupakan kelapa yang telah dikeluarkan isinya lalu dikeringkan (Adang et al., 2020).

Berdasarkan kualitas dan penggunaanya, kopra terbagi menjadi tiga jenis yaitu kopra edible, reguler, dan reject (Lahay et al., 2023). Kopra edible adalah kopra yang memenuhi standar kebersihan dan keamanan pangan yang ditetapkan oleh badan pengawas makanan. Jenis kopra ini dapat dijadikan sebagai bahan baku dalam produksi minyak kelapa, makanan ringan, produk susu kelapa, dan sebagainya. Kopra reguler adalah kopra yang tidak memenuhi standar kebersihan dan keamanan untuk dikonsumsi langsung, tetapi masih dapat digunakan dalam berbagai industri lainnya. Kopra reguler biasanya digunakan dalam industri kosmetik, sabun, minyak sayur, dan produk-produk non-pangan lainnya. Kopra reject adalah kopra yang tidak memenuhi standar kualitas yang diperlukan untuk digunakan dalam industri makanan, minuman, atau industri lainnya (Zubair et al., 2020).

Pemisahan dan pengklasifikasian jenis kopra berdasarkan kualitas dan penggunaannya sangat penting dilakukan untuk kebutuhan hilirisasi pada

industry kelapa khusunya pada bagian kopra. Kesalahan dalam menentukan ini dapat berakibat adanya undang-undang yang dilanggar, seperti yang terjadi pada industri kelapa sawit (M. Perdagangan dan R. Indonesia, 2022).

Selama ini, berdasarkan hasil wawancara yang dilakukan terhadap seorang pedagang kopra di Kabupaten Indragiri Hilir, Provinsi Riau, dapat diketahui bahwa pengklasifikasian jenis kopra umumnya masih dilakukan secara manual oleh para pedagang kopra berdasarkan intuisi (feeling). Cara membedakan jenis-jenis kopra biasanya dilakukan dengan mengamati ciri-ciri tertentu. Sebagai contoh, kopra edible memiliki diameter kurang dari 9 cm, berbentuk seperti mangkuk, bebas dari cacat pada kulit ari, tidak berlubang, tidak berwarna kecoklatan, bebas dari pertumbuhan jamur, serta berwarna putih pada bagian dagingnya. Sementara itu, untuk kopra regular memiliki diameter lebih dari 9 cm dan bentuknya tidak harus seperti mangkuk. Namun, ciri-ciri lainnya tetap sama dengan kopra edible. Adapun jenis kopra reject muncul setelah proses penyortiran Edible dan Reguler. Kopra dengan jenis reject umumnya mengandung jamur, memiliki warna kecokelatan, dan mungkin memiliki lubang atau keretakan.

Proses ini tentunya membutuhkan tenaga, waktu, dan biaya yang cukup tinggi. Selain itu perbedaan persepsi setiap pedagang dalam melihat jenis kopra membuat klasifikasi menjadi tidak konsisten. Hal ini menyebabkan kinerja petani menjadi kurang efisien apalagi jika dilakukan dalam jumlah besar (Ismail et al., 2023). Oleh karena itu, dibutuhkan pengembangan metode yang dapat mengklasifikasikan jenis kopra secara otomatis dan akurat untuk mendukung proses ini. Klasifikasi yang akurat dan efisien dapat membantu mengelola pasokan kopra dengan standar kualitas yang berlaku, serta memenuhi kebutuhan beragam pengguna kopra di berbagai sektor industri dengan cara yang lebih tepat (Rachmawati & Gunawan, 2020).

Penelitian terkait telah banyak dilakukan, diantara penelitian yang dilakukan oleh (Adang et al., 2020), penelitian telah membuat metode untuk

klasifikasi kematangan kopra menggunakan *naïve bayes* berdasarkan warna daging kopra dengan cara membandingkan nilai variabel penduga setiap model, hasil yang diperoleh dari model 1-3 rata-rata akurasinya mencapai 91,12% (Adang et al., 2020). Penelitian lainnya yang dilakukan oleh Idam et al (2023), membuat sistem untuk menentukan *grade* kopra berdasarkan kadar air, kadar minyak, dan jamur menggunakan logika fuzy. Sistem ini mampu memberikan kemudahan, menghemat waktu, dan menimilisir kesalahan yang di akibatkan oleh *human error*. Akurasi yang didapat dari sistem ini sebesar 95% (Lahay et al., 2023).

Penelitian ini akan fokus membuat model klasifikasi jenis kopra berdasarkan ekstraksi fitur menggunakan metode *k-nearest neighbour*. Untuk mendukung proses klasifikasi, perlu adanya pengolahan citra digital untuk mempersiapkan datanya. Pengolahan citra digital adalah sebuah disiplin ilmu yang mempelajari tentang teknik-teknik mengolah citra yang bertujuan untuk memperbaiki kualitas citra agar mudah di interpretasi oleh manusia atau mesin komputer (Rilo Pambudi et al., 2020). Pada pengolahan citra tentunya tidak lepas dari proses pengenalan objek untuk memisahkan bagian tertentu dalam suatu citra agar dapat mengenali objek dengan akurat. Ekstraksi fitur pada citra merupakan tahap yang sangat penting dalam pengolahan citra digital (Nurul Huda, 2019).

Ekstraksi fitur adalah proses untuk mengekstrak atau mengambil nilai unik dari setiap citra yang ingin dikenali untuk dapat dibedakan dengan objek lainnya (Priadana, 2018). Salah satu proses ekstraksi fitur adalah menganalisa berdasarkan isi visual seperti warna, bentuk, dan tekstur (Luthfi et al, 2019). Ekstraksi fitur warna merupakan suatu proses yang dilakukan untuk mengambil informasi warna dari suatu citra untuk mengidentifikasi, membedakan, serta mengklasifikasikan objek-objek berdasarkan karakteristik warnanya. Warna memiliki atribut yang mudah dikenali dan mampu memberikan informasi yang sangat signifikan dalam hal pemisahan citra (Paramita et al., 2019). Mengekstraksi warna dalam format RGB, HSV, dan

Grayscale merupakan langkah yang umum digunakan dalam analisis citra dan pemrosesan gambar. Setiap model warna ini memberikan informasi yang berbeda tentang citra. Misalnya, dalam model warna RGB, kita dapat mengambil informasi tentang tingkat intensitas merah, hijau, dan biru pada setiap piksel. Dalam model warna HSV, kita dapat memperoleh informasi tentang tingkat kecerahan (nilai), serta hue dan saturasi warna pada setiap piksel. Sementara dalam model grayscale, dapat digunakan dalam proses ekstraksi fitur lanjutan (Farokah, 2022). Untuk merepresentasikan warna RGB, HSV, dan Grayscale dalam citra, metode *color moment* telah terbukti menjadi pendekatan yang efektif dan efisien (Rakhmawati, 2018).

Ekstraksi fitur bentuk adalah proses yang digunakan untuk mengidentifikasi dan mewakili karakteristik geometris dari objek dalam citra. Metode ini melibatkan analisis bentuk objek, seperti panjang tepi, *area*, *perimeter*, kompaktitas, atau aspek rasio. Penelitian sebelumnya telah menggunakan ekstraksi fitur bentuk berupa area dan perimeter dalam mengidentifikasi citra kopra putih. Representasi ini dapat memberikan informasi dasar tentang ukuran dan kompleksitas bentuk yang ada dalam citra (Rahayu Marlis et al., 2021). Ekstraksi fitur tekstur merupakan proses pengambilan informasi tentang pola dan struktur tekstur dari sebuah citra. Salah satu ciri tekstur yang dapat digunakan adalah dengan menggunakan metode *Gray Level Co-Occurrence Matrix* (GLCM). *Gray Level Co-Occurrence Matrix* (GLCM) merupakan metode yang efektif untuk mengekstraksi informasi tekstur dalam citra dengan menganalisis hubungan spasial antara pasangan piksel dengan tingkat keabuan yang sama (Shandy et al., 2019).

Penelitian sebelumnya telah mengaplikasikan pendekatan *color moment* dan GLCM (*Gray-Level Co-occurrence Matrix*) dalam mengklasifikasikan berbagai jenis material pasir yang umumnya digunakan dalam konstruksi bangunan. Metode ini diterapkan dengan menggunakan algoritma SVM (Support Vector Machine), yang menghasilkan tingkat akurasi

mencapai 94% (Astutik et al., 2022). Selain itu, penelitian lainnya juga telah melibatkan ekstraksi fitur bentuk, seperti area dan perimeter, serta ekstraksi warna dalam bentuk model RGB, dalam usaha untuk memprediksi kualitas kopra putih. Penelitian tersebut berhasil mencapai tingkat akurasi tertinggi mencapai 93% (Rahayu Marlis et al., 2021). Berdasarkan beberapa faktor yang telah diuraikan sebelumnya, penulis akan menggunakan ekstraksi fitur warna, bentuk, dan tekstur untuk menganalisis citra. Untuk fitur warna, penulis akan menggunakan metode color moment yang mencakup *mean* RGB, HSV, dan standar deviasi pada Grayscale. Fitur bentuk melibatkan *area* dan *perimeter*. Sementara fitur tekstur akan diekstraksi melalui metode GLCM dengan fitur seperti *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*. Dengan menggabungkan ketiga pendekatan ini, diharapkan dapat menghasilkan nilai ekstraksi fitur yang akurat dan mampu menggambarkan citra secara optimal. Hasil fitur-fitur yang telah diekstraksi dapat digunakan sebagai *input-an* untuk proses klasifikasi (Sugiarto & Budiman, 2021).

Terdapat banyak metode yang dapat digunakan untuk proses klasifikasi, salah satunya adalah *K-Nearest Neighbour*. *K-Nearest Neighbour* merupakan metode yang dapat digunakan untuk klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut (Siti Raysyah et al, 2021). Metode ini memiliki prinsip yang sederhana dan mampu bekerja dengan maksimal terhadap data yang mempunyai kesalahan acak (Setiawan & Triayudi, 2022)(Sari, 2020). Selain itu, metode ini juga mudah untuk dimplementasikan, membuat banyak penelitian yang menggunakan *K-Nearest Neighbour* sebagai metode klasifikasi (Nabila et al., 2021). Penelitian sebelumnya telah menggunakan *K-Nearest Neighbour* sebagai metode klasifikasi terhadap dataset citra penyakit *pneumonia*. Dimana hasil dari klasifikasi tersebut mendapat akurasi yang tinggi yaitu sekitar 96% (Halim & Anraeni, 2021).

Penelitian lainnya yang dilakukan oleh Sandi Prayoga dkk pada tahun 2020, telah membuat sistem klasifikasi jenis bunga mawar menggunakan

metode *K-Nearest Neighbour* (K-NN). Pengujian yang dilakukan menggunakan 250 citra yang dibagi dengan perbandingan 90% data *training* dan 10% data *testing*, sehingga data latih yang digunakan sebanyak 150 citra bunga mawar dan 75 citra bukan bunga mawar, serta 25 citra sisanya digunakan sebagai data uji. Data citra tersebut diuji menggunakan *confusion matrix* dan berhasil mendapatkan hasil akurasi sebesar 92%, presisi sebesar 93%, dan *recall* sebesar 92% (Prayogo et al., 2022).

Dengan merujuk pada penjelasan sebelumnya, penulis berencana untuk melakukan penelitian dengan judul "**Peningkatan Performa Klasifikasi Jenis Kopra Menggunakan Algoritma K-Nearest Neighbour Berdasarkan Ekstraksi Fitur**". Melalui penelitian ini, diharapkan dapat meningkatkan performa klasifikasi jenis kopra dengan menerapkan Algoritma *K-Nearest Neighbour* (K-NN) yang didukung oleh proses ekstraksi fitur. Hal ini diharapkan dapat mengoptimalkan proses pengolahan dan identifikasi jenis kopra dengan tingkat akurasi dan efisiensi yang lebih tinggi di dalam industri kopra. Oleh karena itu, penulis juga berharap bahwa penelitian ini akan memberikan kontribusi yang signifikan dalam meningkatkan efisiensi operasional serta pengambilan keputusan yang lebih baik di industri tersebut.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dikemukakan di atas, maka penulis merumuskan masalah yaitu bagaimana meningkatkan performa klasifikasi jenis kopra menggunakan metode *K-Nearest Neighbour* berdasarkan hasil ekstraksi fitur.

1.3 Batasan Masalah

Sebagai pembatasan pengembangan penelitian dan sesuai dengan tujuan yang ditetapkan, maka penulis memberikan ruang lingkup batasan sebagai berikut:

1.3.1 Proses

1. Dataset yang digunakan adalah citra berwarna.
2. Dataset citra terdiri atas tiga jenis kopra yaitu edible, regular, dan reject.
3. Citra yang digunakan yaitu kopra non telungkup.
4. Format citra yang digunakan adalah JPG (*Joint Photographic Expert Group*).

1.3.2 Metode

1. Klasifikasi jenis kopra ini dilakukan berdasarkan hasil 7 percobaan dengan mempertimbangkan kombinasi fitur.
2. Ekstraksi fitur warna menggunakan RGB, HSV, *GrayScale* dengan dua *color moment* yaitu *mean* dan standar deviasi
3. Ekstraksi fitur bentuk menggunakan *area* dan *perimeter*.
4. Ekstraksi fitur tekstur menggunakan metode GLCM (*Gray Level Co-occurrence Matrix*) dengan jarak 1 dan sudut 0 serta beberapa fitur seperti *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*.
5. Proses klasifikasi jenis kopra menggunakan metode *K-Nearest Neighbour*.

1.3.3 Tools

1. *Tools* yang digunakan untuk mengekstraksi ciri dan mengklasifikasikan adalah *Visual studio code* dengan Bahasa pemrograman *python*.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk meningkatkan performa klasifikasi jenis kopra edible, regular, dan reject dengan menggunakan hasil ekstraksi citra dan metode *k-nearest neighbour* sebagai metode klasifikasinya.

1.5 Manfaat Penelitian

1.5.1 Manfaat Bagi Universitas

1. Dapat dijadikan referensi literatur penelitian pada Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.
2. Memberikan gambaran sejauh mana kemampuan mahasiswa dalam menguasai ilmu yang telah diberikan selama masa perkuliahan.

1.5.2 Manfaat Bagi Pembaca

1. Menjadi referensi dalam hal pengolahan citra digital dengan ekstraksi fitur pada citra.
2. Memberikan pengetahuan kepada pembaca mengenai metode klasifikasi menggunakan algoritma K-Nearest Neighbour (K-NN).

1.6 Metodologi Penelitian

Pada penyusunan penelitian ini, penulis menggunakan metode-metode sebagai berikut:

1.6.1 Metode Pengumpulan Data

1. Studi Pustaka

Studi pustaka ini dilakukan dengan mengumpulkan data dengan membaca buku referensi, karya ilmiah, dan jurnal yang berhubungan dengan topik yang dibahas.

2. Wawancara

Wawancara dilakukan dengan pihak yang bersangkutan terkait dengan objek skripsi yang diteliti oleh penulis.

1.6.2 Metode Pengembangan Model

Dalam penelitian ini, penulis menggunakan metode simulasi. Adapun tahap-tahap pada metode simulasi adalah sebagai berikut (Gandomi et al., 2010):

1. *Problem Formulation*

2. *Model Conceptual*
3. *Collection of Input Data*
4. *Modelling*
5. *Simulation*
6. *Verification and Validation*
7. *Design of Experiment*
8. *Execute Simulation and Analysis Output*

1.7 Sistematika Penelitian

Pada penyusunan penelitian ini, penulis melakukan sistematika penelitian yang tersusun ke dalam enam bab. Dimana setiap bab-nya memiliki beberapa sub bab yang secara keseluruhan saling berkaitan satu sama lain. Berikut merupakan uraian dari masing-masing bab:

BAB I : PENDAHULUAN

Bab ini membahas mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penelitian.

BAB II : TINJAUAN PUSTAKA DAN LANDASAN TEORI

Bab ini membahas mengenai berbagai literatur yang dijadikan sebagai bahan acuan dan teori-teori yang berkaitan dengan topik yang dibahas.

BAB III : METODOLOGI PENELITIAN

Bab ini membahas mengenai metodologi yang akan digunakan dan langkah-langkah untuk melakukan penelitian.

BAB IV : IMPLEMENTASI

Bab ini membahas tentang simulasi sistem dari mulai perancangan sampai pengujian sistem yang sesuai dengan metode yang digunakan.

BAB V : HASIL DAN PEMBAHASAN

Bab ini membahas mengenai hasil penelitian dan pembahasan terhadap hasil penelitian yang dilakukan.

BAB VI : PENUTUP

Bab ini membahas mengenai kesimpulan dari penelitian yang telah dilakukan serta saran dari penulis.

BAB II

LANDASAN TEORI

2.1 Kelapa

Kelapa (*Cocos nucifera*) merupakan anggota tunggal dalam marga *Cocos* dari suku *Aren-arenan* atau *Arecaceae* yang tersebar di seluruh daerah tropis maupun subtropis (Abdullah & Usman, 2016). Tanaman ini berasal dari daerah pesisir Asia Tenggara (Indonesia, Malaysia, Filipina) dan Melanesia. Kemudian menyebar ke Amerika Latin, Karibia hingga ke Afrika. Saat ini tanaman kelapa telah tersebar di 200 negara di dunia (Majda, 2016).

Bagi masyarakat Indonesia, kelapa juga memiliki sejarah tersendiri yang merupakan bagian penting dari kehidupan karena kelapa memiliki banyak manfaat dan nilai ekonomi yang cukup tinggi. Mulai dari bagian batang hingga buah kelapa memiliki manfaat tersendiri. Buah kelapa adalah bagian yang paling bernilai ekonominya (Khatib et al., 2022). Arti penting tanaman kelapa bagi masyarakat juga tercermin dari luasnya area perkebunan rakyat yang mencapai 98% dari 3,74 juta Ha perkebunan kelapa (Novrianto, 2008).

2.2 Jenis-jenis Kelapa

Kelapa dibagikan menjadi tiga varietas yaitu:

1) Kelapa Varietas Dalam

Kelapa jenis ini mempunyai ciri-ciri dengan batang tinggi dan besar, dimana tingginya dapat mencapai 30 meter atau lebih. Berbuahnya tergolong agak lambat sebab, kelapa ini baru berbuah ketika sudah berumur 6-8 tahun setelah ditanam. Umur dari tanaman kelapa jenis ini bisa mencapai 100 tahun lebih. Adapun keunggulan dari jenis varietas ini adalah produksi kopranya tinggi, yaitu pertahun bisa menghasilkan 1 ton kopra/Ha, daging buahnya tebal dan keras dengan kadar minyak yang tinggi, serta lebih tahan terhadap hama dan penyakit. Beberapa jenis kelapa yang masuk ke dalam varietas jenis ini antara lain *viridis*

(kelapa hijau), *rubescens* (kelapa merah), *macrocorpu* (kelapa kelabu), *sakarina* (kelapa manis)(Sirnawati, 2023).

2) Kelapa Varietas Genjah

Kelapa varietas genjah ini mempunyai ciri-ciri dengan batang besar namun tidak terlalu tinggi, berbuah lebat tetapi mudah dipengaruhi oleh iklim dan lingkungan. Memiliki ukuran buah yang relatif kecil dengan kadar kopra rendah, yaitu 130 gr/buah, sementara kadar minyaknya 65% dari bobot kering daging buah. Beberapa jenis kelapa yang termasuk dalam varietas ini antara lain yaitu *eburnea* (kelapa gading), *regia* (kelapa raja), *pumila* (kelapa puyuh), dan *pretiosa* (kelapa raja malabar) (Sirnawati, 2023).

3) Kelapa Varietas Hibrida

Kelapa varietas hibrida merupakan jenis kelapa yang diperoleh dari hasil persilangan antara varietas Genjah dan varietas Dalam. Hasil persilangan itu merupakan kombinasi yang baik dari kedua jenis varietas aslinya. Dimana keunggulan dari jenis kelapa varietas ini adalah lebih cepat berbuah dengan waktu sekitar 3-4 tahun setelah tanam, produksi kopra tinggi sekitar 6-7 ton/ha/tahun pada umur 10 tahun dengan produktivitas sekitar 140/butir/pohon/tahun, produktivitas tandan buah sekitar 12 tandan yang berisi sekitar 10-20 butir kelapa, daging buahnya keras dan tebal dengan ketebalan sekitar 1,5cm, serta kandungan minyaknya tinggi (Sirnawati, 2023).

2.3 Kopra

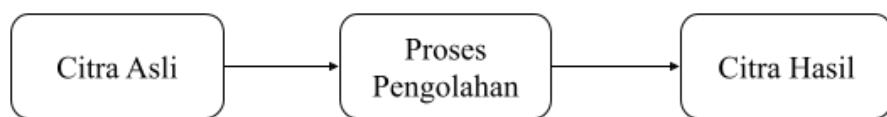
Kopra berasal dari daging buah kelapa (*Cocos nucifera*) yang diolah secara konvensional dengan cara dijemur atau di asapkan. Buah kelapa yang sudah kering harus segera dikeringkan menjadi kopra. Jika tetap berair maka permukaan daging buah kelapa akan berlendir dan berwarna kuning(Abdullah et al., 2017). Cara pengeringan buah kelapa digolongkan dengan dua cara, yaitu cara pertama dengan Cahaya matahari yaitu daging kelapa yang masih melekat pada tempurung dijemur langsung dibawah sinar matahari. Sedangkan cara kedua dengan pemanasan api yaitu dengan mengadakan kontak langsung

daging buah kelapa dengan panas api yang ada dari pembakaran. Umumnya cara ini disebut sebagai pengasapan (Abdullah et al., 2017).

Terdapat tiga jenis kopra, yaitu edible, regular, dan reject (Lahay et al., 2023). Kopra edible adalah kopra yang memenuhi standar kebersihan dan keamanan pangan yang ditetapkan oleh badan pengawas makanan. Kopra jenis ini dapat dijadikan sebagai bahan baku dalam produksi minyak kelapa, makanan ringan, produk susu kelapa, dan sebagainya. Kopra regular adalah kopra yang tidak memenuhi standar kebersihan dan keamanan untuk dikonsumsi langsung, tetapi masih dapat digunakan dalam berbagai industri lainnya. Kopra reguler biasanya digunakan dalam industri kosmetik, sabun, minyak sayur, dan produk-produk non-pangan lainnya. Kopra reject adalah kopra yang tidak memenuhi standar kualitas yang diperlukan untuk digunakan dalam industri makanan, minuman, atau industri lainnya. Kopra ini mungkin memiliki kualitas yang rendah, terkontaminasi, atau rusak secara fisik. Biasanya, kopra reject dianggap tidak dapat digunakan untuk tujuan apa pun dan harus dibuang.

2.4 Pengolahan Citra

Pengolahan citra adalah proses manipulasi dan analisa citra dengan bantuan komputer dengan menggunakan deteksi tepi yang menentukan titik-titik tepi dari objek (Putri, 2016). Pada pengolahan citra terdapat proses input dan output, dimana ketika citra yang di input memiliki kualitas yang kurang baik karena ada noda atau bintik-bintik hitam, maka hasil output dari citra yang dihasilkan menjadi kurang jelas informasinya. Maka dari itu, perlu dilakukan pengolahan citra untuk memperbaiki kualitas citra dengan menghilangkan noda dan bintik-bintik hitam sehingga hasil citra yang baru dapat lebih baik dan juga dengan informasi yang lengkap (Kusumanto et al., 2011).



Gambar 2. 1 Pengolahan Citra

Tujuan dari pengolahan citra yaitu untuk meningkatkan kualitas citra dari asli menjadi lebih baik. Di dalam proses pengolahan citra terdapat Analisa citra yang bertujuan untuk mengidentifikasi ciri-ciri yang akan digunakan untuk dijadikan parameter dalam penelitian dengan citra tersebut. Tahapan-tahapan dalam menganalisa suatu objek citra terdiri dari preprocessing, segmentasi dan ekstraksi ciri. Deteksi tepi merupakan salah satu faktor yang penting untuk ekstraksi ciri suatu citra. Deteksi tepi kurang bagus jika tidak digunakan dengan tepat karena mengandung nilai dari citra tersebut. Setiap objek citra memiliki batas tepi yang memiliki kegunaan sebagai proses untuk tahapan segmentasi pada citra (Soepomo, 2013).

2.5 *Preprocessing* Citra

Preprocessing citra adalah proses yang dilakukan pada citra agar dapat diproses pada tahap selanjutnya. Tujuan dari *Preprocessing* citra yaitu untuk meningkatkan kualitas citra. Pada penulisan ini, untuk tahap *preprocessing* citra dilakukan *resize* ukuran citra untuk memperkecil ukuran agar penulisan dapat berjalan lebih cepat (Wuryani & Agustiani, 2021), penghapusan *background* yang tidak dibutuhkan dari citra dan, pemotongan bagian berdasarkan *bounding box*.

1) *Resize*

Proses *resize* dilakukan untuk mengubah ukuran data yang sebelumnya memiliki dimensi yang berbeda-beda. Tujuan utama dari proses ini adalah untuk mengurangi waktu pelatihan model. Misalnya, jika kita memiliki ukuran gambar yang besar, dengan mengubah ukuran gambar, waktu yang dibutuhkan untuk melatih model dapat dikurangi tanpa mengurangi kinerja model secara keseluruhan (Febriawan, 2022).

2) *Remove Background*

Proses ini dilakukan untuk menghapus latar belakang yang tidak relevan atau mengganggu sebuah gambar atau citra. Tujuannya untuk

memfokuskan objek dan menyeragamkan latar belakang agar terlihat perbedaan antara objek dan latar belakang (Sutanto et al., 2022).

3) *Cropping*

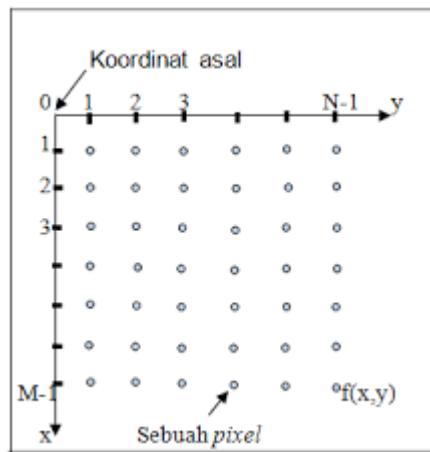
Proses *cropping* dilakukan untuk memperkecil ukuran sebuah citra dengan cara memotong citra pada koordinat yang telah ditentukan pada suatu area pada citra. Hasil proses *cropping* dapat memudahkan peneliti fokus pada obyek yang diperlukan (Yuhandri, 2019).

2.6 Citra Digital

2.6.1 Definisi Citra Digital

Berdasarkan bentuk sinyal penyusunnya, citra dapat digolongkan menjadi dua jenis yaitu citra digital dan citra analog. Citra analog adalah citra yang dibentuk dari sinyal analog bersifat kontinyu, sedangkan citra digital adalah citra yang dibentuk dari sinyal digital yang bersifat diskrit (Yuhandri et al., 2022).

Citra digital merupakan representasi dari fungsi intensitas cahaya dalam bentuk diskrit pada bidang dua dimensi. Citra tersusun oleh sekumpulan piksel yang memiliki koordinat (x,y) dan amplitudo $f(x,y)$. koordinat (x,y) menunjukkan letak atau posisi piksel dalam suatu citra, sedangkan amplitudo $f(x,y)$ menunjukkan nilai intensitas warna citra. Gambar berikut menunjukkan posisi koordinat citra digital (Riontopah, 2021).



Gambar 2. 2 Koordinat Citra Digital

Nilai pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut dengan picture elements, image elements, pels, atau pixels. Istilah pixel paling sering digunakan pada citra digital.

2.6.2 Pembentukan Citra Digital

Pembentukan citra digital terdiri dari beberapa proses, yaitu akuisisi citra, *sampling*, dan kuantisasi.

1) Akuisisi Citra

Proses akuisisi citra adalah pemetaan suatu pandangan menjadi citra kontinu dengan menggunakan sensor. Ada beberapa macam sensor yang dapat digunakan pada akuisisi citra, yaitu sensor tunggal, sensor garis, dan sensor larik. Sensor tersebut nantinya akan mengubah intensitas dan frekuensi cahaya menjadi gelombang analog. Selanjutnya gelombang analog akan diubah menjadi sinyal digital (Edha et al., 2020).

Pada setiap baris sinyal, diawali oleh *colorburst* sebagai informasi gelombang warna dan antar baris dibatasi oleh sinyal sinkronisasi (*sync pulse*). Sementara itu, sinyal tersebut terbentuk dalam fungsi waktu t dan amplitudo A . Untuk mendapatkan citra digital, sinyal analog harus menjalani proses digitalisasi yang terdiri dari proses *sampling* dan kuantisasi (Kartika, 2021).

2) Sampling

Proses sampling adalah proses konversi data analog ke dalam format digital pada koordinat x,y. Pada proses ini dilakukan penentuan jumlah sampel piksel. Dimana kecepatan proses sampling dipengaruhi oleh resolusi citra yang diinginkan. Seperti yang disebutkan di atas, hasil dari sensor pada tahap akuisisi citra masih berupa citra kontinu. Dimana hasil tersebut merupakan fungsi kontinu $f(x,y)$. fungsi tersebut merupakan sinyal kontinu pada nilai x,y dan juga amplitudonya. Nilai x,y yang kontinyu akan diubah menjadi bentuk diskrit (Kartika, 2021)..

3) Kuantisasi

Proses kuantisasi adalah proses perubahan nilai amplitudo kontinu menjadi nilai baru berupa nilai diskrit. Nilai amplitudo yang di kuantisasi adalah nilai-nilai pada koordinat diskrit hasil dari proses sampling (Kartika, 2021).

2.6.3 Jenis-jenis Citra Digital

Berdasarkan representasi ruang warna, citra terbagi menjadi tiga jenis yaitu, citra berwarna, citra keabuan, dan citra biner. Berikut merupakan penjelasan dari masing-masing jenis citra.

1) Citra berwarna atau RGB

RGB merupakan jenis citra yang masing-masing pikselnya memiliki warna tertentu, warna tersebut terdiri dari merah (*Red*), Hijau (*Green*), dan biru (*Blue*) (Farokhah, 2020). RGB memiliki nilai interval [0,255], dimana 255 merepresentasikan warna putih, sedangkan 0 mempresentasikan warna hitam. Citra berwarna atau RGB memiliki nilai yang lebih besar daripada citra keabuan (*grayscale*) karena memiliki z layer. Penyimpanan warna asli ini sebesar 1 byte = 8 bit atau sebaliknya, yang berarti masing-masing warna memiliki tingkat atau penyebaran sampai 256 warna. Terdapat 16 jutaan atau sama dengan 2^{24} kombinasi warna citra. Citra berwarna ini terdiri dari tiga matriks

yang mewakili nilai-nilai merah, hijau, dan biru untuk setiap pikselnya (Siregar et al., 2021).

2) Citra Keabuan (*grayscale*)

Citra keabuan merupakan jenis citra yang setiap pikselnya mempunyai warna gradasi mulai dari putih sampai hitam. Citra keabuan memiliki interval nilai [0,255], dimana 255 mempresentasikan pada warna putih dan 0 cenderung pada warna hitam. Citra keabuan dikodekan dengan 8 bit nilai atau 1 byte (2^8) sehingga memiliki nilai yang lebih besar daripada citra biner (Siregar et al., 2021). Citra keabuan merupakan hasil dari rata-rata color image, dengan demikian maka persamaannya dapat dituliskan sebagai berikut:

$$I_{BW}(x, y) = \frac{I_R(x, y) + I_G(x, y) + I_B(x, y)}{3} \quad (2.2)$$

Dimana $I_R(x, y)$ adalah nilai piksel *Red* titik (x, y) , $I_G(x, y)$ adalah nilai piksel *Green* titik (x, y) , $I_B(x, y)$ adalah nilai piksel *Blue* titik (x, y) , dan $I_{BW}(x, y)$ adalah nilai piksel *Greyscale* titik (x, y)

3) Citra Biner

Citra biner merupakan citra yang setiap pikselnya hanya terdiri dari warna hitam atau putih, dimana hanya perlu 1 bit per piksel 1 (putih) dan 0 (hitam). Apabila dikodekan dalam 8 bit menjadi 0 dan 255, sehingga sangat efisien dalam hal penyimpanan. Citra yang direpresentasikan dengan biner sangat cocok untuk teks, sidik jari, atau gambar arsitektur (Siregar et al., 2021). Citra biner merupakan hasil pengolahan dari hitam dan putih gambar, dengan menggunakan fungsi sebagai berikut.

$$I_{Bin}(x, y) = \begin{cases} 0 & I_{BW}(x, y) < T \\ 255 & I_{BW}(x, y) \geq T \end{cases} \quad (2.3)$$

Dan dalam bentuk *floating point*

$$I_{Bin}(x, y) = \begin{cases} 0 & I_{BW}(x, y) < T \\ 1 & I_{BW}(x, y) \geq T \end{cases} \quad (2.4)$$

Dimana $I_{BW}(x, y)$ merupakan nilai piksel gray titik (x,y), sedangkan $I_{Bin}(x, y)$ merupakan nilai piksel biner titik (x,y), dan T merupakan *threshold* (Kusumanto et al., 2011).

2.7 Ekstraksi Fitur

Ekstraksi fitur merupakan proses pengindeksan suatu database berupa citra dengan isinya. Salah satu proses ekstraksi fitur adalah menganalisa berdasarkan isi visual seperti warna, bentuk, dan tekstur (Luthfi et al, 2019).

2.7.1 Ekstraksi Fitur Warna

Ekstraksi fitur warna merupakan proses pengambilan informasi warna dari sebuah citra dengan tujuan untuk mengidentifikasi, membedakan, dan mengklasifikasikan objek berdasarkan karakteristik warnanya. Warna merupakan ciri yang mudah dikenali dan dapat memberikan informasi yang cukup signifikan untuk membedakan citra (Paramita et al., 2019).

2.7.1.1 Warna RGB

Warna RGB merupakan model warna yang terdiri dari tiga komponen warna primer yaitu Red, Green, dan Blue. Jika ketiga komponen warna tersebut digabungkan, maka akan menghasilkan satu warna baru (Farokhah, 2020). Dalam model warna RGB, sebuah warna direpresentasikan dengan menentukan sejauh mana warna merah (red), hijau (green), dan biru (blue) dicampurkan. Warna ini dituliskan dalam bentuk triplet RGB, di mana setiap komponen dapat memiliki nilai dari 0 hingga nilai maksimum yang telah ditetapkan, biasanya 255. Misalnya, warna putih dalam RGB direpresentasikan dengan nilai maksimum pada ketiga komponen warna, yaitu (255, 255, 255), sementara warna hitam direpresentasikan dengan nilai minimum pada ketiga komponen warna, yaitu (0, 0, 0). Warna lainnya dapat dihasilkan dengan mengatur kombinasi nilai-nilai antara 0 hingga 255 untuk setiap komponen warna. Dengan menggunakan model warna RGB, berbagai warna dapat direproduksi dengan presisi

yang tinggi. Penggunaan triplet RGB memungkinkan pengontrolan yang fleksibel terhadap intensitas setiap komponen warna, sehingga memungkinkan penciptaan berbagai nuansa dan variasi warna (Rahmanto et al., 2022).

2.7.1.2 Warna HSV

Warna HSV menunjukkan ruang warna dalam tiga kategori komponen utama yaitu *hue*, *saturation*, dan *value*. *Hue* adalah sudut 0 sampai 360 derajat, biasanya 0 adalah merah, 60 derajat adalah kuning, 120 derajat adalah hijau, 180 derajat adalah *cyan*, 240 derajat adalah biru, dan 300 derajat adalah warna *magenta*. *Hue* menunjukkan jenis warna (seperti merah, biru, atau kuning) atau corak warna yang menjadi tempat asal warna ditemukan dalam spektrum warna. *Saturation* merupakan ukuran seberapa besar kemurnian warna tersebut (Areni et al., 2019). Sebagai contoh suatu warna yang semuanya merah tanpa putih adalah saturasi penuh. Jika ditambahkan putih ke merah, warna merah akan digeser ke merah muda, maka akan menghasilkan warna yang berwarna-warni. *Hue* masih tetap merah tetapi nilai saturasinya berkurang. Saturasi biasanya bernilai 0 sampai 1 dan menunjukkan nilai keabu-abuan warna di mana 0 menunjukkan abu-abu dan 1 menunjukkan warna primer murni. Komponen ketiga dari warna HSV adalah *value* atau disebut juga intensitas yaitu ukuran seberapa besar kecerahan dari suatu warna atau seberapa besar cahaya datang dari suatu warna. Value dapat bernilai 0 sampai 100% (Areni et al., 2019).

2.7.1.3 Warna Grayscale

Warna *grayscale* merupakan model yang setiap pikselnya mempunyai warna gradasi mulai dari putih sampai hitam. Proses pengubahan citra berwarna menjadi *grayscale* dapat dilakukan dengan menyederhanakan citra berwarna (RGB). Setiap proses pengubahan citra dilakukan perhitungan dengan ketiga layernya, dimana ketiga perhitungan tersebut sama pada tiap layer (Munantri et al., 2020).

Untuk mendapatkan citra dalam bentuk *grayscale*, dapat menggunakan rumus berikut:

$$\text{Grayscale} = 0,3R + 0,59G + 0,11B \quad (2.5)$$

2.7.1.4 Momen Warna

Moment warna merupakan representasi suatu nilai ciri warna untuk memberikan karakter pada setiap warna agar dapat dibedakan atau dipisahkan sesuai dengan ciri-ciri yang dimilikinya. Terdapat tiga momen warna yang terbukti efisien dan efektif untuk mewakili distribusi warna dalam gambar yaitu *mean*, standar deviasi, *skewness* (Rakhmawati et al., 2018). Pada penelitian ini hanya mengekstrak dua momen warna yaitu *mean*, Standar Deviasi dan variasi. *Mean* merupakan rata-rata nilai piksel pada masing-masing *channel* R, G, B, H, S, V, dan Gray. *Mean* digunakan untuk merepresentasikan nilai rata-rata pada masing-masing komponen warna. Untuk menghitung nilai *mean* dari momen warna menggunakan persamaan berikut (Kadafi, 2018).

$$\mu = \frac{x_1+x_2+x_3\dots+x_n}{n} \quad (2.6)$$

Keterangan:

μ : Nilai *mean*

x_i : nilai *sample* ke-i

n : Jumlah piksel

Selanjutnya persamaan yang digunakan untuk menghitung nilai standar deviasi adalah sebagai berikut.

$$\sigma = \sqrt{\frac{N \sum_{i=1}^n (x_i - \mu)^2}{n-1}} \quad (2.7)$$

Keterangan:

σ : Standar deviasi

x_i : nilai x ke-i

μ : nilai *mean*

n : Jumlah piksel

2.7.2 Ekstraksi Fitur Bentuk

Bentuk merupakan ciri fisik yang dapat dilihat dengan menggunakan indera manusia. Agar bentuk dari suatu objek dapat dibedakan dengan objek lainnya, digunakan beberapa parameter dari ekstraksi fitur bentuk diantaranya *area* dan *perimeter* (Maulidin Yusuf et al., 2021).

- 1) *Area* atau luas, berfungsi untuk menghitung luas daerah berdasarkan jumlah *pixel* yang menempati objek gambar.
- 2) *Perimeter* atau keliling, berfungsi untuk menghitung banyaknya *pixel* yang ada pada wilayah tepi dari objek.

2.7.3 Ekstraksi Fitur Tekstur

Tekstur adalah sifat karakteristik yang dimiliki oleh suatu daerah (di dalam citra) yang cukup besar sehingga secara alami sifat-sifat tadi dapat berulang dalam daerah tersebut. Pengertian dari tekstur dalam hal ini kurang lebih adalah keteraturan pola-pola tertentu yang terbentuk dari susunan piksel-piksel dalam citra digital (Kadir, A et al, 2012). Untuk membedakan tekstur objek satu dengan objek yang lainnya, perlu dilakukan proses ekstraksi ciri tekstur. Salah satu ciri tekstur yang dapat digunakan adalah dengan menggunakan metode *Gray Level Co-Occurrence Matrix* (GLCM).

Gray Level Co-Occurrence Matrix (GLCM) adalah metode yang yang melakukan analisis terhadap piksel pada citra untuk mengetahui tingkat keabuan yang sering terjadi (Mukti et al., 2013). Metode ini juga digunakan untuk mendapatkan nilai statistic orde ke-2 dengan menghitung nilai probabilitas hubungan kedekatan antara dua buah piksel pada jarak (*d*) dan sudut (θ) tertentu (Asery et al., 2016). Metode GLCM bekerja dengan membentuk korelasi pada data citra dan menentukan ciri fungsi matriks piksel. Korelasi adalah kejadian ketika satu level piksel saling bertetangga dengan nilai berdasarkan jarak dan orientasi sudut. Jarak dapat berupa 1-10 piksel, sementara orientasi sudut terdiri dari 0° , 45° , 90° , dan 135° (Mukherjee et al., 2016). Menurut penelitian (Tampinongkol, 2023) penggunaan jarak 1

dan sudut 0 dalam GLCM dapat menghasilkan matriks dengan ukuran yang lebih kecil dibandingkan dengan menggunakan jarak dan sudut yang lebih besar. Hal ini dapat mengurangi kompleksitas komputasi dan kebutuhan penyimpanan data, serta mempercepat proses ekstraksi fitur GLCM. Dalam kasus ini, informasi tekstur yang dihasilkan dengan menggunakan jarak 1 dan sudut 0 telah berhasil memberikan informasi terkait dengan perbedaan dua jenis penyakit yang terdapat pada daun tomat.

Beberapa fitur yang ada pada *Gray Level Co-Occurrence Matrix* (GLCM) diantaranya sebagai berikut:

- 1) *Contrast*, *contrast* adalah suatu metrik yang mengukur tingkat perbedaan nilai keabuan di sekitar area pada citra. Jika terdapat variasi warna atau skala keabuan yang rendah dalam citra, dan nilai piksel yang bertetanggaan memiliki nilai yang sama, maka kontras akan memiliki nilai 0. Berikut merupakan rumus untuk menghitung *contrast*:

$$\text{Contrast} = \sum_{i,j} (i - j)^2 P(i,j) \quad (2.8)$$

Dimana i adalah baris matriks, j adalah kolom matriks, $P(i,j)$ adalah elemen matriks korelasi baris (i) dan kolom (j)

- 2) *Dissimilarity*, *dissimilarity* merupakan ukuran yang mendefinisikan variasi tingkat intensitas pasangan piksel dalam citra, yang didefinisikan pada rumus berikut:

$$\text{Dissimilarity} = \sum_{i,j} |i - j| P(i,j) \quad (2.9)$$

Dimana i adalah baris matriks, j adalah kolom matriks, $P(i,j)$ adalah elemen matriks korelasi baris (i) dan kolom (j)

- 3) *Homogeneity*, *homogeneity* merupakan ukuran kemiripan citra. Nilai homogenitas akan bernilai tinggi jika semua piksel memiliki kesamaan nilai. Berikut merupakan rumus untuk menghitung *homogeneity*:

$$\text{Homogeneity} = \sum_{i_1} \sum_{i_2} \frac{P(i_1 i_2)}{1 + (i - j)^2} \quad (2.10)$$

Dimana i adalah baris matriks, j adalah kolom matriks, $P(i,j)$ adalah elemen matriks korelasi baris (i) dan kolom (j)

- 4) *Energy, energy* menyatakan ukuran ketidakteraturan tingkat keabuan dalam citra, di mana nilai yang tinggi menunjukkan elemen-elemen GLCM memiliki nilai yang relatif serupa, sementara nilai yang rendah terjadi ketika elemen-elemen GLCM memiliki nilai 0 atau 1. Berikut merupakan rumus untuk menghitung *contrast*:

$$\text{Energy} = \sum_{i,j} P(i,j)^2 \quad (2.11)$$

Dimana i adalah baris matriks, j adalah kolom matriks, $P(i,j)$ adalah elemen matriks korelasi baris (i) dan kolom (j)

- 5) *Correlation, correlation* merupakan ukuran ketergantungan linear dari derajat tingkat keabuan dalam citra yang dapat menunjukkan struktur linear gambar pada setiap piksel yang saling bertetangga dengan objek citra. Nilai yang terdapat pada korelasi antara -1 sampai dengan 1. Berikut merupakan rumus untuk menghitung *energy*:

$$\text{Correlation} = \sum_{i,j} \frac{(i-\mu_i)(j-\mu_j)P(i,j)}{\sigma_i\sigma_j} \quad (2.12)$$

$$\mu_i = \sum_i \sum_j i p(i,j)$$

$$\mu_j = \sum_i \sum_j j p(i,j)$$

$$\sigma_i = \sqrt{\sum_i \sum_j (i - \mu_i)^2 P(i,j)}$$

$$\sigma_j = \sqrt{\sum_i \sum_j (j - \mu_j)^2 P(i,j)}$$

Dengan i adalah baris matriks, j adalah kolom matriks, $P(i,j)$ adalah elemen matriks korelasi baris (i) dan kolom (j), μ_i , μ_j adalah rata-rata elemen pada baris dan kolom matriks, σ_i , σ_j adalah standar deviasi pada baris dan kolom matriks.

2.8 *Independent dan Dependent Value*

Independent dan *dependent value* adalah konsep yang mendasar dalam statistik dan analisis data. Mereka digunakan untuk menggambarkan hubungan antara dua atau lebih variabel dalam suatu studi atau analisis. Penentuan apakah suatu variabel adalah dependent atau independent bergantung pada konteks masalah dan hipotesis yang diuji. Berikut adalah penjelasan dasar tentang penentuan keduanya (Ghodang & Hantono, 2020):

1) *Independent Value*

Independent Value adalah variabel yang dipercaya dapat memberikan perpengaruh terhadap *dependent value*. Dengan kata lain, *Independent Value* merupakan variabel yang mempengaruhi atau yang menjadi sebab perubahannya atau timbulnya *dependent value*. Biasanya *independent value* digunakan sebagai inputan atau atribut yang digunakan untuk melatih suatu model (Hafiz et al., 2021).

2) *Dependent Value*

Dependent Value adalah nilai yang dapat diukur dalam suatu penelitian dan faktor apa yang dapat mempengaruhinya. Dengan kata lain, *Dependent Value* merupakan variabel yang dipengaruhi atau yang menjadi akibat karena adanya *independent value*. *Dependent value* mengandung informasi tentang suatu kategori atau kelas tertentu dari suatu model (Hafiz et al., 2021).

2.9 *Feature Scaling*

Feature scaling adalah suatu teknik yang digunakan dalam pengolahan data untuk mengubah skala atau rentang nilai dari fitur-fitur yang pada dataset. Rentang nilai yang tidak seimbang pada setiap atribut dapat mempengaruhi kualitas hasil dataset (Nasution et al., 2019). Oleh karena itu, dibutuhkan *feature scaling* untuk menghasilkan distribusi nilai yang seragam pada setiap fitur, sehingga fitur-fitur tersebut dapat diolah dan dibandingkan dengan baik. Terdapat beberapa metode yang dapat digunakan, pada penelitian ini menggunakan Min-Max scaling.

Min-Max *scaling* atau sering disebut dengan Min-max *normalization*, merupakan suatu teknik normalisasi data yang menggunakan transformasi linier pada data awal untuk mengimbangi nilai perbandingan sebelum dan sesudah proses. Min-max *scaling* mengubah ukuran dari rentang asli, sehingga semua nilai berada pada rentang 0 dan 1 (Rifatama et al., 2023). Berikut merupakan rumus dari min-max *scaling*.

$$X_{sc} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2.13)$$

2.10 Artificial Intelligence

Istilah *Artificial Intelligence* diperkenalkan pertama kali pada tahun 1956 dalam sebuah konferensi di bidang *Computer Science* di Dartmouth. Pada awalnya *Artificial Intelligence* dirumuskan sebagai sebuah pendekatan untuk memodelkan cara kerja otak manusia yang selanjutnya digunakan sebagai dasar perancangan sistem komputer yang lebih cerdas. Menurut Poole et al (1998), mereka mendefinisikan istilah *Artificial Intelligence* sebagai sebuah bidang ilmu mengenai pengembangan *intelligence agents* yang dirumuskan lebih lanjut sebagai:

“..any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals.”

Berdasarkan definisi diatas, bidang kajian utama dari *Artificial Intelligence* adalah entitas yang mempunyai kemampuan untuk berinteraksi dengan lingkungan tempatnya berada dalam bentuk menerima data dari lingkungan tempatnya berada dan melakukan sejumlah aksi di dalam lingkungannya dengan tujuan untuk memaksimumkan probabilitas pencapaian tujuan yang ditetapkan.

Selanjutnya, pendapat lain dari Russell & Norvig (2009) mendefinisikan *Artificial Intelligence* sebagai berikut:

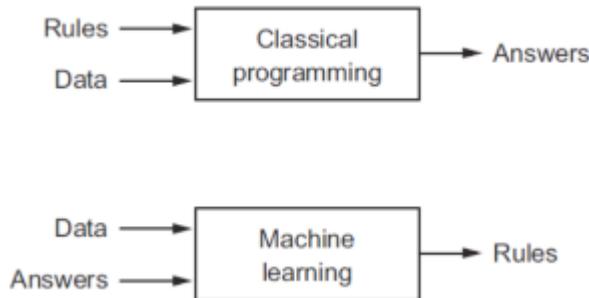
“..machine (or computer) that mimic cognitive functions that humans associate with the human mind, such as learning and problem solving.”

Berdasarkan definisi di atas, kajian *Artificial Intelligence* ini lebih berfokus kepada kemampuan komputer untuk meniru kemampuan berfikir manusia yaitu melakukan pembelajaran dan mengatasi masalah. Dalam dua puluh tahun terakhir, *Artificial Intelligence* telah berkembang menjadi beberapa cabang ilmu. Masing-masing cabang ilmu ini memfokuskan kepada bidang yang lebih spesifik dari kajian *Artificial Intelligence*. Salah satu cabang *Artificial Intelligence* yang berkembang dengan cepat dan banyak diimplementasikan ke dalam sejumlah teknologi aplikatif di berbagai industri adalah *Machine Learning* dan *Deep Learning*.

2.1.1 Machine Learning

Machine Learning adalah metode analisis yang membantu menangani data besar dengan cara mengembangkan algoritma komputer. Dengan menggunakan data, pembelajaran mesin memungkinkan komputer menemukan wawasan tersembunyi tanpa diprogram secara eksplisit saat mencarinya. Contohnya, ketika ada email baru, algoritma tersebut akan menghasilkan prediksi apakah email baru itu masuk kedalam spam atau tidak (Maulana et al., 2023).

Machine Learning merupakan salah satu cabang dari *Artificial Intelligence* yang fokus pada pengembangan sebuah sistem yang mampu belajar “sendiri” tanpa harus berulang kali di program oleh manusia. *Machine Learning* menyediakan sistem kinerja secara otomatis serta belajar memperbaiki diri dari pengalaman tanpa diprogram secara eksplisit. Hal tersebut sesuai dengan pendapat dari Chollet Francois (2017), yang menjelaskan bahwa ciri utama dari algoritma *Machine Learning* dibandingkan dengan pemrograman konvensional lainnya adalah kemampuannya untuk membuat komputer memprogram dirinya sendiri.



Gambar 2. 3 Perbedaan Pemograman Tradisional dan *Machine Learning*

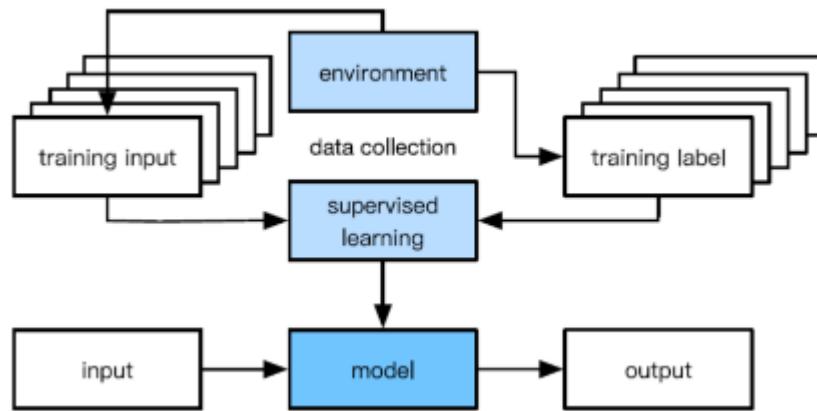
(Sumber : Francois, 2017)

Berdasarkan Gambar 2.4 Pada pemrograman konvensional, data dan *rules* (program) dijalankan pada sebuah komputer untuk menghasilkan *output*. Sedangkan pada *machine learning*, data dan *output* dijalankan didalam sebuah komputer untuk menghasilkan *rules* (program). Program hasil pembelajaran algoritma *machine learning* selanjutnya digunakan untuk memproses *input* menjadi *output* sesuai dengan tugas yang harus dilakukan.

Metode pembelajaran model *machine learning* dikelompokan kedalam tiga kategori utama sebagai berikut:

1) *Supervised learning*

Merupakan metode pembelajaran yang bertujuan untuk memprediksi variabel target data baru berdasarkan model yang sudah ditraining menggunakan himpunan data berlabel (Handayani, 2022). Dalam proses pembelajaran *machine learning*, sebuah algoritma digunakan untuk melakukan penelusuran nilai *parameter* model yang mengoptimalkan sebuah fungsi objektif. Model hasil pembelajaran *machine learning* merupakan hasil generalisasi dari data training sehingga dapat digunakan untuk memprediksi label dari sebuah data baru.



Gambar 2. 4 *Supervised Learning*
 (Sumber: Francois, 2017)

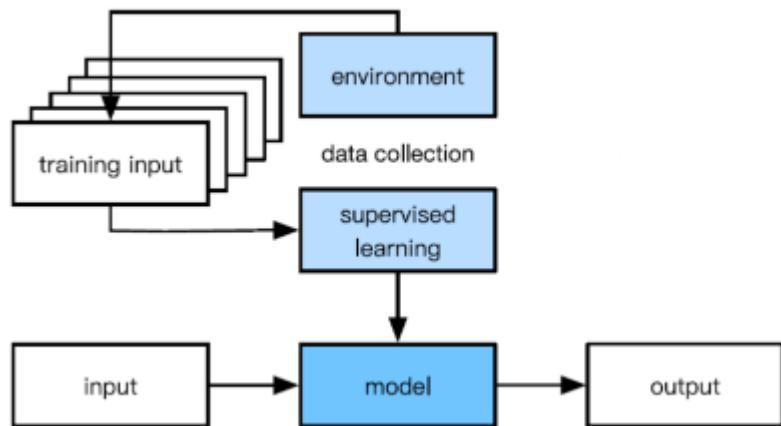
Teknik pembelajaran *supervised learning* biasanya digunakan untuk menyelesaikan dua kategori masalah komputasi yaitu:

- Masalah klasifikasi, digunakan untuk memprediksi variabel target berupa data kategorikal.
- Masalah regresi, digunakan untuk memprediksi variabel target berupa

2) *Unsupervised learning*

Merupakan metode pembelajaran yang menggunakan sejumlah contoh data tanpa label. Teknik unsupervised learning ini biasanya digunakan untuk menyelesaikan dua kategori masalah yaitu (Angga et al, 2023):

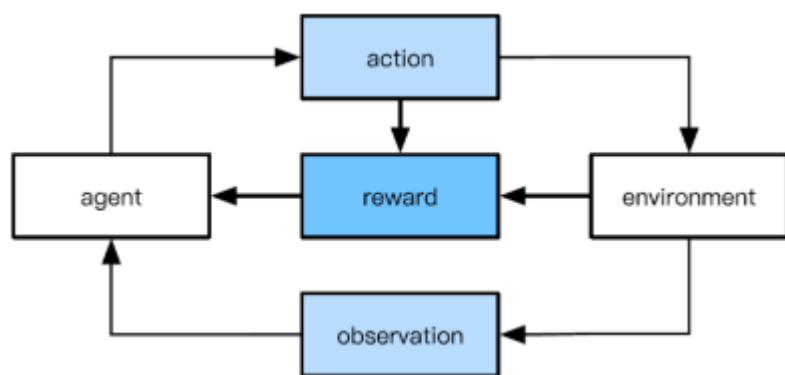
- Klastering, digunakan untuk mengelompokan data training kedalam beberapa klaster berdasarkan kesamaan fitur.
- Pengurangan dimensi data (dimensional reduction), digunakan untuk mengelompokan variabel input data kedalam beberapa kelompok variabel yang tidak saling berkorelasi.



Gambar 2. 5 *Unsupervised Learning*
 (Sumber : Francois, 2017)

3) Reinforcement learning

Merupakan metode pembelajaran yang didasarkan pada pemberian penghargaan terhadap keputusan yang diinginkan dan atau pemberian penalti terhadap keputusan yang tidak diinginkan. Metode ini mengintegrasikan metode pembelajaran dengan sebuah aplikasi yang memberi umpan balik bagi setiap keputusan yang telah di ambil. Dengan mekanisme ini, algoritma pembelajaran dapat belajar dari kesalahan dalam membuat sebuah keputusan (Angga et al, 2023).

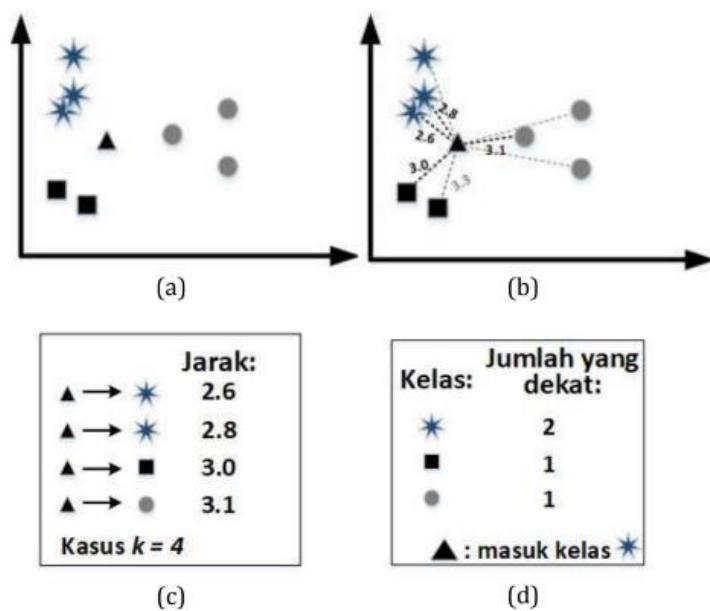


Gambar 2. 6 *Reinforcement Learning*
 (Sumber : Francois, 2017)

2.11 K-Nearest Neighbors (K-NN)

K-Nearest Neighbors (K-NN) merupakan metode klasifikasi yang sangat sederhana dalam mengklasifikasikan sebuah gambar berdasarkan jarak terdekat dengannya. Data yang memiliki jarak fitur vector terdekat akan menjadi satu kelas atau label klasifikasi. Walaupun sangat sederhana dan terdapat kelemahan jika sebaran datanya terlalu dekat namun metode K-NN ini mudah dipahami (Farokhah, 2020).

K-Nearest Neighbors (K-NN) masuk ke dalam kategori *supervised machine learning* yang dapat digunakan untuk klasifikasi dan regresi. Cara kerja metode ini dengan menggunakan data latih (*training data*) dan menghasilkan data tes berdasarkan jarak ke data latih tersebut. Tujuannya mencari sejumlah k tetangga terdekat yang ditentukan melalui label kelas terbanyak pada jangkauan k tetangga terdekat tersebut. Ilustrasi metode ini seperti pada **Gambar 2.8**. Untuk kemudahan ilustrasi digunakan sumbu dua dimensi dengan koordinat x_1 dan x_2 . Hal tersebut mewakili dua nilai sensor (Romy, 2022). Dalam ilustrasi ini terdapat tiga kelas yang menjadi target klasifikasi.



Gambar 2. 7 Ilustrasi Algoritma K-NN

(Sumber : Romi, 2022)

Gambar 2.7c) Dalam contoh ilustrasi ini, akan dicari apakah data tes \blacktriangle masuk ke suatu kelas, apakah masuk ke dalam kelas \star atau \blacksquare atau \bullet ?

Metode KNN menghitung jarak antara data tes \blacktriangle dengan semua data latih. Pada Gambar 2.7b) terlihat perhitungan yang menghasilkan jarak 2.6, 2.8, 3.0, 3.1, dan 3.3. Pada Gambar 2.7c) digunakan nilai $k = 4$ sehingga diurutkan empat jarak yang paling dekat sesuai dengan rankingnya. Terlihat bahwa jarak paling dekat adalah 2.7 Gambar 2.7d) pada langkah ini dihitung berapa jumlah label yang dekat pada kelas-kelas tersebut. Terlihat ada 2 label pada kelas \star , 1 label pada kelas \blacksquare , dan 1 label pada kelas \bullet . Dari sini dapat dilihat bahwa data tes diklasifikasikan ke dalam kelas \star .

Metode KNN secara umum terdiri atas dua langkah, yaitu pelatihan untuk menyimpan setiap pola latih dan klasifikasi. Pada saat klasifikasi sebuah pola, KNN memeriksa semua pola latih untuk menemukan sejumlah k pola terdekat. Proses pelatihan pada KNN menghasilkan k yang memberikan akurasi tertinggi dalam menggeneralisasi data-data yang akan datang. Salah satu tantangan pada proses pelatihan adalah menentukan nilai k yang optimum.

Nilai k pada K-NN harus menggunakan nilai ganjil jika digunakan untuk proses klasifikasi beda halnya jika digunakan untuk melakukan prediksi nilai k pada K-NN dapat berupa bilangan ganjil ataupun genap (Khairi et al., 2021). Nilai k yang kecil, misalnya $k = 1$, memiliki kecenderungan untuk memiliki bias yang rendah namun variasi yang tinggi. Hal ini dapat menyebabkan model menjadi sangat responsif terhadap noise atau outlier dalam data pelatihan, yang pada akhirnya dapat mengakibatkan prediksi yang tidak stabil. Sebaliknya, ketika nilai k lebih besar, model cenderung menjadi lebih umum dan dapat membantu mencegah overfitting. Namun, jika nilai k terlalu besar, ada risiko bahwa batas keputusan antar kelas yang sebenarnya menjadi tidak stabil, yang mengakibatkan dapat mengurangi akurasi, terutama

pada data yang kompleks (ADMINLP2M, 2023). Penelitian sebelumnya telah menggunakan nilai $k = 3$ dalam penerapan metode KNN untuk melakukan klasifikasi tingkat kematangan buah kopi berdasarkan deteksi warna, dan berhasil mencapai tingkat akurasi sebesar 97,7% (Raysyah et al., 2021) kopi. Berdasarkan beberapa pertimbangan tersebut, penulis memutuskan untuk menggunakan nilai $k = 3$ dalam konteks penelitian ini.

Pada Gambar 2.7b) perhitungan jarak menggunakan *Euclidean distance* untuk dua titik, dengan rumus persamaan seperti berikut.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.14)$$

Langkah-langkah klasifikasi dirangkum sebagai berikut:

- 1) Menentukan nilai k .
- 2) Menghitung jarak antara data tes yang baru dan tetangga terdekat sejumlah k pada latih.
- 3) Memeriksa dengan *voting* di kelas manakah yang terbanyak jumlah tetangga. Yang memiliki jumlah tetangga terdekat yang terbanyak akan dipilih sebagai hasil klasifikasi.

2.12 Mengukur Performansi

Sebuah sistem yang akan melakukan klasifikasi, diharapkan dapat melakukan klasifikasi semua data dengan benar, namun kenyataannya kinerja sistem terkadang tidak bisa benar 100%. Oleh karena itu, suatu sistem klasifikasi harus diukur kinerjanya (Hanafi et al., 2019). Pada umumnya, pengukuran performansi kerja suatu model untuk klasifikasi menggunakan metrik akurasi, *confusion matrix*, dan *classification report* (Romy, 2022).

2.12.1 Akurasi

Skor akurasi adalah metrik yang paling sederhana, cara kerja pengukuran ini dengan membagi jumlah kasus prediksi yang benar dengan

jumlah seluruh prediksi. Jika semua prediksi benar, akan menghasilkan skor akurasi = 1, dan jika semua prediksi salah maka skor akurasi = 0. Namun, terdapat kelamahan pada skor akurasi ini, sebab tidak memberikan informasi pengaruh ketidakseimbangan jumlah *false positif* dan *false negative*. Skor akurasi dapat dihitung dengan persamaan berikut:

$$\text{Akurasi} = \frac{\text{Jumlah Prediksi yang Benar}}{\text{Jumlah Seluruh Prediksi}} \quad (2.15)$$

2.12.2 Confusion Matrix

Confusion Matrix adalah tabel yang menyatakan klasifikasi jumlah data uji yang benar dan jumlah data uji yang salah (Normawati & Prayogi, 2021). Contoh *confusion matrix* untuk klasifikasi *multiclass* ditunjukkan pada tabel berikut.

Tabel 2. 1 *Confusion Matrix*

		Nilai Prediksi		
		1	2	3
Nilai Aktual	1	A	B	C
	2	D	E	F
	3	G	H	I

Pada *confusion matrix* untuk multi kelas, fokus pembahasan dilakukan per kelas. Misal saat ini fokus pada kelas 1, maka TP adalah A. Berikut nilai TN, FP, dan FN jika fokus pada kelas 1.

True Positif: Jika kelas 1, maka TP = A, Jika fokus pembahasan kelas 2, maka TP=E. Jika fokus pada kelas 3, maka TP=I. *True Negative*: Jumlah sel kecuali sel-sel pada kolom dan baris 1, maka TN=E+F+H+I. *False Positif*: Jumlah semua nilai aktual selain kelas 1 yang diprediksi sebagai kelas 1, maka FP=D+G. *False Negative*: Jumlah semua nilai prediksi yang bukan kelas 1 saat nilai aktualnya 1, maka FN=B+C (Romy, 2022).

Keterangan

TP (<i>True Positive</i>)	= Jumlah dokumen dari kelas 1 yang benar diklasifikasikan sebagai kelas 1.
TN (<i>True Negative</i>)	= Jumlah dokumen dari kelas 0 yang benar diklasifikasikan sebagai kelas 0.
FP (<i>False Positive</i>)	= Jumlah dokumen dari kelas 0 yang salah diklasifikasikan sebagai kelas 1.
FN (<i>False Negative</i>)	= Jumlah dokumen dari kelas 1 yang salah diklasifikasikan sebagai kelas 0.

2.12.3 Classification Report

Classification report menghasilkan tiga metrik evaluasi yaitu *precision*, *recall*, dan *F1 skor*. *Precision* adalah rasio prediksi benar positif terhadap total hasil prediksi positif. *Recall* adalah rasio prediksi benar positif terhadap keseluruhan data yang sebenarnya positif. *F1 skor* adalah rata-rata harmonik dari presisi dan recall (Ridhovan & Suharso, 2022). Akurasi dan *confusion matrix* dapat dibentuk dengan persamaan berikut.

$$Precision = \frac{True\ Positif}{True\ Positif + False\ Positif} \quad (2.16)$$

$$Recall = \frac{True\ Positif}{True\ Positif + False\ Negative} \quad (2.17)$$

$$f1\ Score = \frac{2 \times Recall \times Presisi}{Recall + Presisi} \quad (2.18)$$

2.13 Metode Simulasi

Metode simulasi adalah metode yang digunakan untuk simulasi seperti mengikuti tingkah laku dari suatu sistem tertentu. Metode ini dapat digunakan untuk berbagai keperluan seperti desain sistem, pembangunan kebijakan sistem operasi, dan dalam penelitian untuk mengembangkan pemahaman sistem

(Sargen, 2013). Penelitian sebelumnya telah menggunakan metode simulasi dalam mengembangkan model klasifikasi daun mangga berdasarkan fitur daun (Pitoyo, 2020). Hal serupa juga dilakukan dalam penelitian (Amalia, 2023), di mana pada penelitian ini menggunakan metode simulasi untuk melakukan analisis sentimen terkait perpindahan ibu kota Indonesia.

Menurut (Gandomi et al., 2010) yang dikutip dari skripsi (Artyani, 2019) metode simulasi mempunyai beberapa tahapan, tahapan tersebut di atur dalam satu siklus. Siklus tersebut dapat menyesuaikan dengan kompleksitas sistem sehingga tahapannya tidak selalu berurutan, berulang, maupun berlawanan arah. Terkadang, beberapa tahapan dapat dilangkahi oleh tahapan berikutnya tergantung dengan rumit atau tidaknya sistem tersebut. Berikut merupakan tahapan-tahapan dalam metode simulasi (Gandomi et al., 2010).

1. *Problem Formulation*

Proses simulasi dimulai dengan adanya masalah yang harus dipecahkan. Misalnya sebuah perusahaan kargo ingin mencoba untuk mengembangkan strategi baru untuk pengiriman truk. Pada tahap ini kita harus memahami perilaku dari sistem dan mengatur operasi sistem. Maka kita perlu menganalisa berbagai solusi dengan menyelidiki hasil sebelumnya dengan masalah yang sama.

2. *Model Conceptual*

Pada tahap ini dilakukan deskripsi tingkat tinggi dari struktur dan perilaku sebuah sistem serta mengidentifikasi semua benda dengan atribut dan antarmuka mereka. Pada tahap ini juga dilakukan penentuan *variabel state* dan aspek-aspek kunci dari *requirement*. Terakhir, mendokumentasikan informasi *non-fungsional*, misalnya seperti perubahan pada masa yang akan datang, perilaku *non-intuitive* atau *non-formal*, dan hubungan dengan lingkungan.

3. *Collection of Input Data*

Pada tahap ini dilakukan pembelajaran sistem untuk mendapatkan data *input* atau *output*. Untuk melakukannya, dapat mengumpulkan dan

mengamati atribut yang telah ditentukan pada tahap sebelumnya. Isu penting lainnya pada tahap ini adalah memilih ukuran sampel yang valid secara statistic dan format data yang dapat diproses dengan komputer.

4. *Modelling*

Pada tahap ini dilakukan pemodelan dengan membangun representasi yang rinci dari sistem berdasarkan *conceptual model* dan *input/output* data yang dikumpulkan. Model ini dibangun dengan mendefinisikan objek, atribut, dan metode yang ingin digunakan. Pada tahap ini, spesifikasi model juga dibuat dengan set persamaan yang mendefinisikan perilaku dan struktur.

5. *Simulation*

Pada tahap ini, kita harus memilih mekanisme untuk menerapkan model (dalam banyak kasus menggunakan computer dan Bahasa pemograman serta alat-alat yang memadai) dan model simulasi yang dibangun.

6. *Verification and Validation*

Pada tahap ini, kita perlu memverifikasi dan memvalidasi ketiga model yang telah dibangun pada tahap sebelumnya yaitu model konseptual, sistem model, dan model simulasi. Validasi difokuskan pada korespondensi antara model dan realitas terkait hasil simulasi yang konsisten dengan sistem yang dianalisis dan membangun model yang tepat berdasarkan hasil yang diperoleh selama tahap ini. Verifikasi menurut Kamus Besar Bahasa Indonesia (KBBI), merupakan suatu pemeriksaan tentang atau mengenai kebenaran laporan, pernyataan, perhitungan uang dan juga lain sebagainya. Sedangkan validasi adalah pengujian kebenaran atas sesuatu.

7. *Design of Experiment*

Pada tahap ini, dilakukan evaluasi *output* dari simulator menggunakan korelasi statistik untuk menentukan tingkat presisi untuk metrik kerja. Fase ini dimulai dengan desain eksperimen, menggunakan

teknik yang berbeda. Beberapa Teknik ini meliputi analisis sensitive, optimasi, dan seleksi.

8. *Execute Simulation and Analysis Output*

Pada tahap ini dilakukan untuk memahami perilaku sistem. *Output* digunakan untuk mendapatkan tanggapan tentang perilaku sistem yang asli. Pada tahap ini, alat visualisasi dapat digunakan untuk membantu proses tersebut.

2.14 Studi Pustaka

Tabel 2. 2 Studi Pustaka

No	Judul	Penulis/ Tahun	Objek Penelitian	Lokasi Penelitian	Parameter Kualitas	Metode Klasifikasi	Jumlah Dataset	Evaluasi Model	Hasil
1	Klasifikasi Tingkat Kematangan Kopra Menggunakan Metode <i>naïve bayes</i>	Yunaldi M.Z. Masi Adang, Abd.Rabi,M. Kom dan Rahman Arifuddin (2020)	Tingkat kematangan kopra	Malang, Jawa Timur	Metode transformasi warna HSI dan RGB	<i>Naïve Bayes</i>	Menggunakan 30 data training dan 12 data testing	Metrik Akurasi	Berdasarkan nilai input yang digunakan yaitu r, g, b, v, *a, *b, entropi, energy, homogenitas dan metode naïve bayes dalam proses pengklasifikasian dengan membandingkan nilai probabilitas diperoleh rata-rata akurasi terbesar mencapai 91,12%.
2	Sistem Prediksi Kualitas Kopra Putih Menggunakan	Rosi Rahayu Marlis, Abdullah, dan Fitri Yunita (2021)	Prediksi kualitas kopra putih	Riau	Ekstraksi ciri warna (RGB) dan bentuk (<i>Area and Perimeter</i>)	<i>K-Nearest Neighbor – Euclidean Distance</i>	Menggunakan 120 data training dan 60 data testing	<i>Confusion Matrix</i>	Berdasarkan pengujian dari sistem prediksi kopra putih menggunakan metode <i>Holdout</i> , maka dapat ditarik kesimpulan bahwa Metode k-Nearest

	k-Nearest Neighbor (k-NN)								Neighbor dapat melakukan prediksi terhadap jenis kopra putih berdasarkan warna dan bentuk. Akurasi yang diperoleh pada saat menggunakan $k = 1$ sebesar 93,33%, pada $k = 3$ sebesar 83,33%, dan pada $k = 5$ sebesar 81,67%. Akurasi tertinggi didapat pada $k = 1$, yakni sebesar 93,33%
3	Penentuan <i>Grade</i> Kopra dengan Penerapan Logika <i>Fuzzy</i>	Idham Halid Lahay, Jamal Darusalam Giu, Hasanuddin, dan Malfrin Bawole (2023)	<i>Grade</i> Kopra	Pohuwato, Gorontalo	Kadar air, kadar minyak, dan jamur.	Logika <i>Fuzzy</i>	Menggunakan 40 data training dan 10 data testing	Metrik Akurasi	Berdasarkan pengujian yang telah dilakukan dengan software matlab dengan logika <i>fuzzy</i> didapatkan tingkat akurasi sebesar 95%.
4	Sistem Prediksi Kualitas Santan Kelapa	Masparudin, Abdullah, dan Usman (2020)	Kualitas Santan Kelapa	Riau	Ekstraksi Warna RGB	<i>Nearest Mean Classifier</i> (NMC)	Menggunakan 90 data training dan 45 data testing	Metrik Akurasi	Evaluasi dilakukan dengan 3 jenis kamera smartphone yaitu kamera 1 Xiaomi Mi 8 Lite, Oppo F7, dan

	Menggunakan <i>Nearest Mean Classifier</i> (NMC)								Samsung Galaxy J3 Pro. Pada pengujian kamera pertama memiliki tingkat akurasi tertinggi yaitu 86,66% dibandingkan dengan kamera 2 dengan akurasi 60% dan kamera 3 dengan akurasi 46%.
5	Klasifikasi Jeruk Nipis Terhadap Tingkat Kematangan Buah Berdasarkan Fitur Warna Menggunakan K- <i>Nearest Neighbor</i>	Cintya Paramita, Eko Hari R, Christy Atika S, dan De Rosal M. S. (2019)	Klasifikasi Jeruk Nipis terhadap tingkat kematangan yang	Semarang	Ekstraksi fitur warna RGB	K- <i>Nearest Neighbor – Euclidean Distance</i> dan <i>Cityblock Distance</i>	Menggunakan 50 data training dan 25 data testing	Metrik Akurasi	Berdasarkan hasil pengujian yang sudah dilakukan, dari 25 data citra uji, akurasi terbaik sebesar 92% saat menggunakan <i>euclidean distance</i> adalah dengan nilai $k=7$ dan $k=3$. Sedangkan menggunakan <i>cityblock distance</i> akurasi sebesar sebesar 88% dengan nilai $k=3$ dan $k=1$.

6	Sistem Klasifikasi Jenis Pisang Berdasarkan Ciri Warna HSV menggunakan Metode K-NN	Zenni Dwi Lestari dan Nur Nafiyah (2019)	Klasifikasi Jenis Pisang	Lamongan, Jawa Timur	Ekstraksi fitur warna HSV	<i>K-Nearest Neighbor - Euclidean Distance</i>	Menggunakan 100 data training dan 50 data testing	Metrik Akurasi	Berdasarkan hasil uji coba yang telah dilakukan, akurasi yang didapatkan dari pengujian data testing menggunakan k=3 sebesar 82 %.
7	Penerapan Metode <i>Grey Level Co-Occurrence Matrix</i> (GLCM) dan <i>K-Nearest Neighbor</i> (K-NN) untuk Mendeteksi Tingkat Kematangan Buah	Qurnia Shandy, Sudirman S. Panna, Yurisyanto Malago (2019)	Tingkat kematangan buah belimbing bintang	Gorontalo	<i>Grey Level Co-Occurrence Matrix</i> (GLCM)	<i>K-Nearest Neighbor - Euclidean Distance</i>	Menggunakan 50 data training dan 10 data testing	<i>Confusion Matrix</i>	Kinerja model deteksi tingkat kematangan buah belimbing bintang menggunakan metode K-Nearest Neighbor dengan fitur ekstraksi Gray Level Co-Occurrence Matriks setelah diukur menggunakan Confusion Matrix menghasilkan akurasi sebesar 90%.

	Belimbing Bintang								
8	Klasifikasi Jenis Kopra Berdasarkan Ekstraksi Fitur Menggunakan Metode K-Nearest Neighbour	Penelitian penulis	Jenis Kopra	Indragiri Hilir, Riau	Ekstraksi fitur warna RGB, HSV dan Grayscale, Ekstraksi fitur bentuk berupa area dan perimeter serta Ekstraksi fitur tekstur menggunakan GLCM	<i>K-Nearest Neighbor - Euclidean Distance</i>	Menggunakan 1.655 data training dan 184 data testing	Metrik Akurasi, <i>confusion matrix</i> , dan <i>classification report</i>	Metode KNN berhasil mengklasifikasikan jenis kopra dengan akurasi optimum sebesar 84% menggunakan kombinasi fitur warna & tekstur, serta gabungan dari ketiga fitur: warna, bentuk, dan tekstur. Fitur tekstur memiliki peran yang signifikan dalam mengidentifikasi data citra kopra dibandingkan dengan fitur warna dan bentuk.

Berdasarkan hasil kajian terdahulu yang relevan, penulis akan menggunakan beberapa metode dan kriteria yang telah dijabarkan sebelumnya untuk digunakan sebagai penunjang penelitian ini. Berikut merupakan kebaruan yang akan di terapkan dalam penelitian ini sebagai berikut:

1. Pada penelitian ini menggunakan dataset kopra yang terbagi menjadi tiga jenis yaitu edible, regular, dan reject dengan masing-masing jenis memiliki 613 citra sehingga total dataset yang digunakan sebanyak 1.839 citra.
2. Ekstraksi fitur yang digunakan meliputi warna, bentuk, dan tekstur. Untuk ekstraksi fitur warna, digunakan metode *color moment* yang mencakup *mean* RGB, HSV, *Grayscale* dan standar deviasi. Sedangkan ekstraksi fitur bentuk menggunakan *area* dan *parameter*. Dan untuk ekstraksi fitur tekstur, metode yang digunakan adalah GLCM (*Gray Level Co-occurrence Matrix*) dengan menggunakan fitur-fitur seperti *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*.
3. Klasifikasi dilakukan pada 7 skenario yang masing-masing skenario mempertimbangkan variasi fitur, seperti hanya fitur warna, hanya fitur bentuk, hanya fitur tekstur, kombinasi warna dan bentuk, kombinasi warna dan tekstur, kombinasi bentuk dan tekstur, serta kombinasi ketiganya, yaitu warna, bentuk, dan tekstur
4. Pengujian model klasifikasi menggunakan metrik akurasi, *confusion matrix*, dan *classification report*.

BAB III

METODOLOGI PENELITIAN

3.1 Metode Pengumpulan Data

Pengumpulan data merupakan salah satu prosedur sistematis yang harus dilakukan untuk memperoleh data yang dibutuhkan, dengan adanya data maka suatu penelitian dapat menjadi optimal dan efektif (Fajar, 2021).

3.1.1 Studi Pustaka

Studi pustaka dilakukan dengan membaca sumber-sumber media seperti jurnal, *e-book*, website, dan penelitian skripsi yang relevan dengan penelitian mengenai pengolahan citra digital, klasifikasi kopra, dan algoritma K-NN. Dengan tujuan untuk memperoleh informasi yang digunakan sebagai landasan teori dan referensi untuk menyelesaikan masalah yang ada.

3.1.2 Wawancara

Wawancara dilakukan dengan narasumber pedagang kopra dan kelapa yaitu bapak Mohd Rifai yang berada di daerah kabupaten Indragiri Hilir, Provinsi Riau pada tanggal 25 Juli 2023. Tujuan dari wawancara ini untuk memperoleh wawasan yang mendalam terkait dengan berbagai aspek yang berkaitan dengan kopra. Dari hasil wawancara ini penulis akan mengumpulkan data mengenai penjelasan dari kopra edible, regular, dan reject.

3.2 Metode Pengembangan Model

Metode pengembangan model yang digunakan dalam penelitian ini adalah metode simulasi. Berikut merupakan tahapan-tahapan dari metode simulasi:

3.2.1 *Problem Formulation*

Tahap pertama dalam metode simulasi ini, yaitu merumuskan permasalahan yang akan dianalisa dan menentukan penggunaan simulasi

atau tidak untuk mendapatkan solusi dari permasalahan yang telah dirumuskan. Setelah mengumpulkan data, penulis merumuskan masalah bagaimana mengklasifikasikan jenis kopra berdasarkan ekstraksi fitur warna, bentuk, dan tekstur menggunakan metode *K-Nearest Neighbour*. Dari rumusan masalah tersebut, penulis memutuskan untuk menggunakan metode simulasi.

3.2.2 Conceptual Model

Setelah permasalahan dirumuskan, penulis membuat pemodelan yang akan digunakan untuk melakukan klasifikasi jenis kopra menggunakan metode *K-Nearest Neighbour*.

3.2.3 Collection of Input Data

Sumber data yang digunakan dalam penelitian ini berasal dari Gudang kopra dan kelapa yang terletak di Kabupaten Indragiri Hilir, Provinsi Riau. Dataset terdiri dari tiga jenis kopra non telungkup, yaitu edible, regular, dan reject. Terdapat 613 data citra untuk masing-masing jenis kopra, sehingga total seluruhnya berjumlah 1.839 data citra yang akan digunakan. Proses pengambilan citra dilakukan menggunakan berbagai jenis kamera *handphone*, termasuk dari merek-merek seperti Infinix, Samsung S23, Poco M3, dan Oppo Reno 8. Selanjutnya, dataset tersebut diletakkan di dalam folder yang sesuai dengan jenis kopra masing-masing.

3.2.4 Modeling

Penulis membuat sebuah model simulasi yang melibatkan metode ekstraksi fitur warna, bentuk, dan tekstur secara manual. Untuk ekstraksi fitur warna, model ini menggunakan *color moment* yang mencakup *mean* RGB, HSV, dan standar deviasi pada Grayscale. Sedangkan untuk ekstraksi fitur bentuk, model ini menggunakan *area* dan *perimeter*. Selain itu, dalam proses ekstraksi fitur tekstur, model ini menggunakan GLCM (Gray Level Co-occurrence Matrix) dengan jarak 1 dan sudut 0,

serta fitur-fitur seperti *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*. Dalam penelitian ini, algoritma klasifikasi yang digunakan adalah K-*Nearest Neighbour*.

3.2.5 *Simulation*

Pada tahap ini, penulis menjalankan proses simulasi berdasarkan *conceptual model* menggunakan teknik pengolahan citra yang telah dibuat pada tahap sebelumnya.

3.2.6 *Verification and Validation*

Pada tahap *verification and validation* dilakukan pengulangan kembali mengenai kegiatan yang dilakukan pada tahapan-tahapan sebelumnya apakah sudah sesuai dengan yang diharapkan dan saling berhubungan.

3.2.7 *Experiment*

Pada tahap eksperimen, penulis melaksanakan serangkaian percobaan yang terdiri dari tujuh skenario berbeda. Setiap skenario dirancang dengan mempertimbangkan variasi fitur, yang membantu dalam menganalisis dampak masing-masing fitur terhadap hasil eksperimen. Keberagaman skenario ini memungkinkan penulis untuk melakukan pengevaluasian terhadap hubungan antara fitur-fitur tersebut dan hasinya. Eksperimen yang dilakukan dalam penelitian ini terdiri dari tujuh skenario yang masing-masing skenario mempertimbangkan variasi fitur, seperti hanya fitur warna, hanya fitur bentuk, hanya fitur tekstur, kombinasi warna dan bentuk, kombinasi warna dan tekstur, kombinasi bentuk dan tekstur, serta kombinasi ketiganya, yaitu warna, bentuk, dan tekstur.

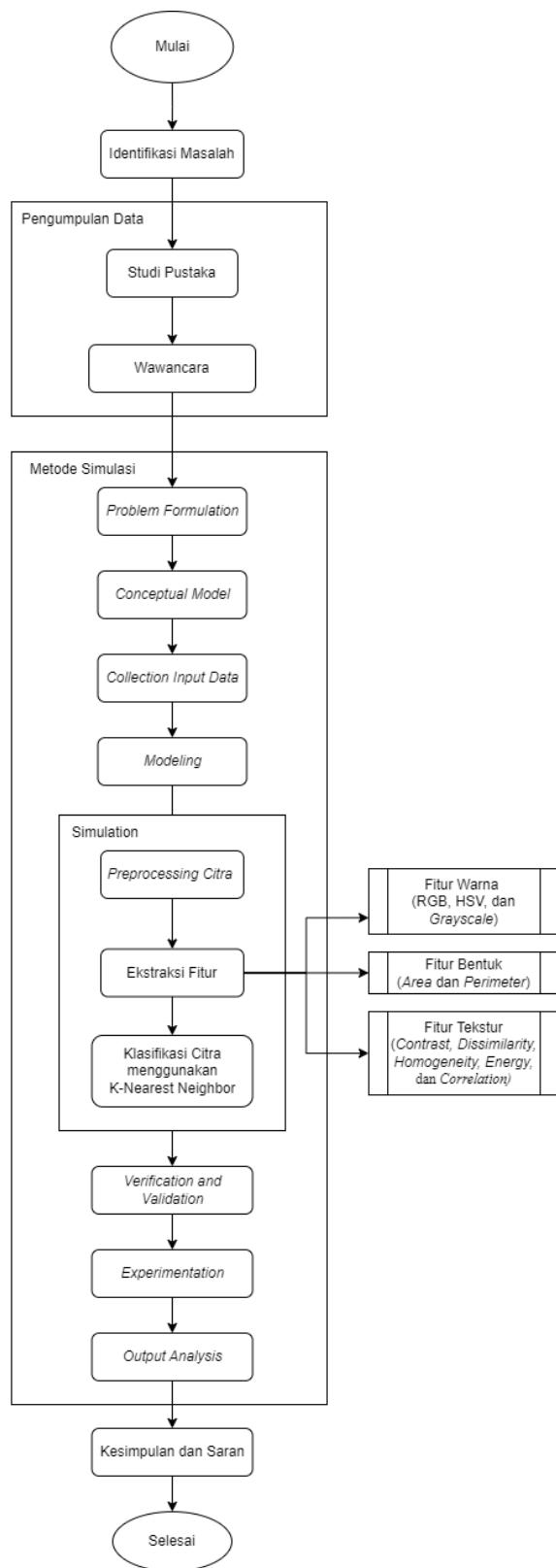
Dengan memanfaatkan variasi dalam ekstraksi fitur ini, penulis berharap dapat meningkatkan performa klasifikasi dalam penelitian ini. Dengan cara ini, diharapkan hasil eksperimen dapat memberikan wawasan yang lebih dalam mengenai kontribusi setiap fitur terhadap

klasifikasi yang lebih akurat dan dapat diandalkan. Pembagian dataset dilakukan dengan rasio 90:10 dimana di mana 90% digunakan untuk pelatihan (training) dan 10% untuk pengujian (testing) model. Penelitian sebelumnya telah menerapkan berbagai perbandingan pembagian data, seperti 66:34, 75:25, dan 90:10. Hasil yang paling baik tercapai saat menggunakan pembagian data 90:10, yang juga berhasil mencapai akurasi tertinggi (Riska & Farokhah, 2021). Selain itu, penelitian sebelumnya juga telah melakukan variasi pembagian data lainnya, seperti 70:30, 80:20, dan 90:10. Namun, hasil terbaik tetap diperoleh pada pembagian data 90:10 (Muningsih, 2022).

3.2.8 *Output Analysis*

Tahap ini merupakan tahap akhir dari metode simulasi. Pada tahapan ini, penulis menganalisis *output* dari hasil percobaan yang dilakukan. Hasil analisis yang diuraikan berupa hasil akurasi yang diperoleh dalam penelitian sesuai dengan permasalahan utama.

3.3 Alur Penelitian



Gambar 3. 1 Alur Penelitian

BAB IV

IMPLEMENTASI

Pada bab ini, penulis akan menjelaskan seluruh proses implementasi dari mulai pemrosesan dataset hingga evaluasi kinerja model, serta identifikasi perangkat yang digunakan dalam eksperimen penulisan ini. Selain itu, penulis juga menyajikan penjelasan implementasi dari *code* yang digunakan, termasuk tangkapan layar dari baris kode yang dieksekusi, serta perhitungan manual yang dilakukan.

4.1 Identifikasi Kebutuhan Sistem

Kebutuhan sistem yang digunakan dalam implementasi penelitian ini dibagi menjadi dua yaitu spesifikasi perangkat keras yang dijelaskan pada Tabel 4.1 dan spesifikasi perangkat lunak yang dijelaskan pada Tabel 4.2

Tabel 4. 1 Spesifikasi Perangkat Keras

No	Nama	Spesifikasi
1	Jenis Laptop	Lenovo ideapad 330
2	<i>Processor</i>	Intel Core i3-6006U
3	RAM	4 GB
4	VGA	Intel UHD <i>Graphics</i> 600
5	<i>System Type</i>	64-bit <i>Operating System</i>
6	<i>OS Edition</i>	Windows 10 Home Single Language

Tabel 4. 2 Spesifikasi Perangkat Lunak

No	Nama	Spesifikasi
1	Bahasa Pemograman	Python 3.11.3

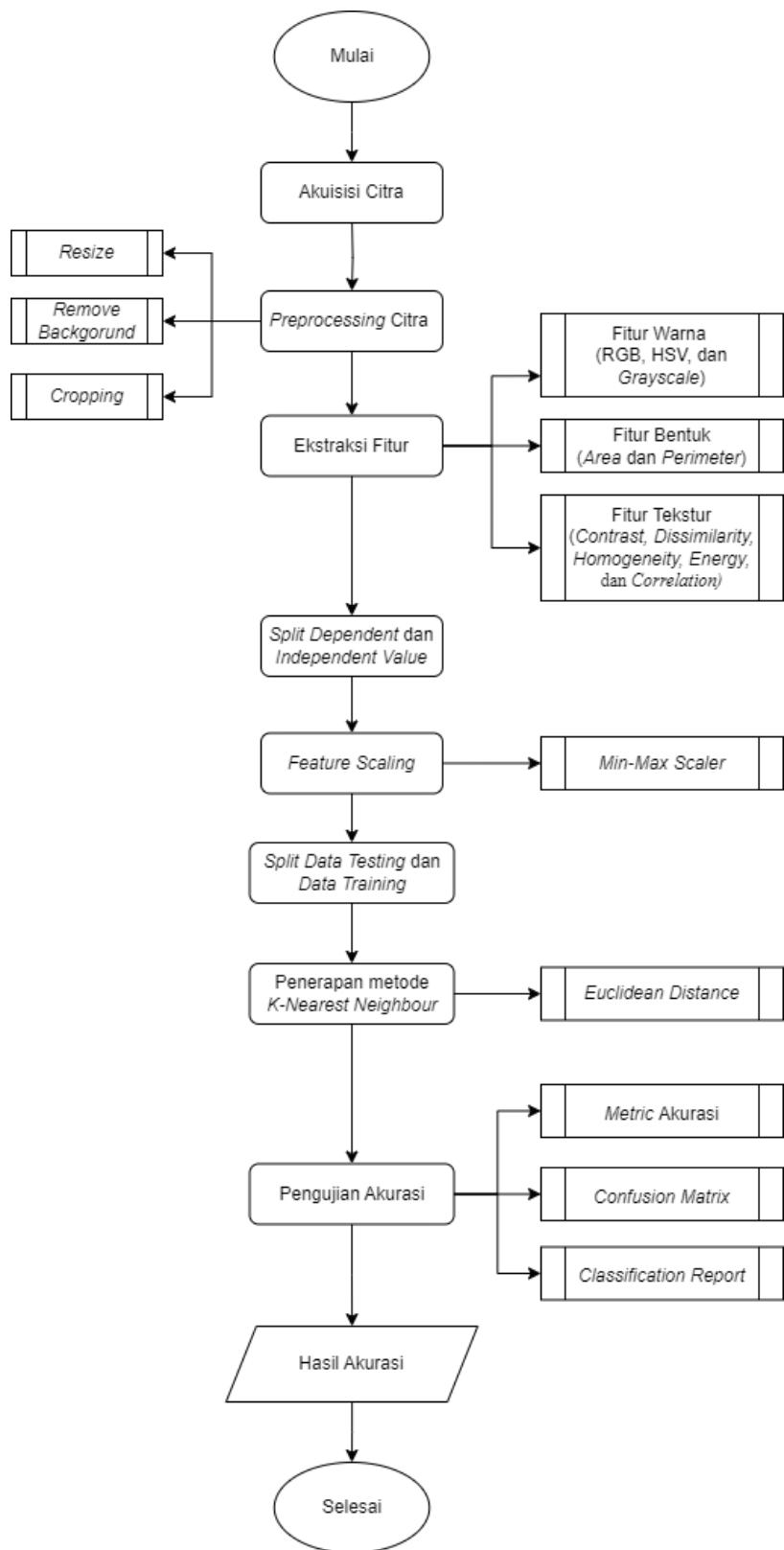
2	Library	<ul style="list-style-type: none"> - Numpy - Pandas - Pillow - Scikit-image - Rembg
3	<i>Tools</i>	<i>Visual Studio Code</i>

4.2 Problem Formulation

Dalam merumuskan permasalahan pada penlitian ini penulis melakukan studi literatur dan wawancara kepada pihak terkait. Berdasarkan hasil wawancara yang dilakukan kepada salah satu pedagang kopra dan kelapa di daerah kabupaten Indragiri Hilir Provinsi Riau, pengklasifikasian jenis kopra umumnya masih dilakukan secara manual. Tentunya proses ini membutuhkan tenaga, waktu, dan biaya yang cukup tinggi. Perbedaan persepsi antar pedang kopra juga membuat klasifikasi tidak konsisten. Oleh karena itu, dibutuhkan pengembangan metode yang dapat mengklasifikasikan jenis kopra secara otomatis dan akurat untuk mendukung proses ini. Klasifikasi yang akurat dan efisien dapat membantu mengelola pasokan kopra dengan standar kualitas yang berlaku, serta memenuhi kebutuhan beragam pengguna kopra di berbagai sektor industri dengan cara yang lebih tepat.

4.3 Conceptual Model

Pada tahapan ini, penulis membuat konsep model yang berisi langkah-langkah yang dilakukan dalam penelitian untuk mengklasifikasikan jenis kopra berdasarkan ekstraksi fiturnya. Tujuan dari tahapan ini, yaitu untuk menjawab pertanyaan permasalahan yang telah didefinisikan sebelumnya pada penulisan ini. Berikut merupakan langkah-langkahnya:



Gambar 4. 1 Pra-Proses Simulasi Sistem

1) Akuisisi Citra

Tahap awal dalam proses akuisisi citra melibatkan penentuan data yang diperlukan untuk mendapatkan citra digital. Dalam konteks penelitian ini, langkah akuisisi citra dilaksanakan dengan memanfaatkan kamera yang terdapat pada perangkat smartphone. Proses ini mencakup pengambilan gambar menggunakan kamera smartphone untuk memperoleh citra dalam bentuk digital.



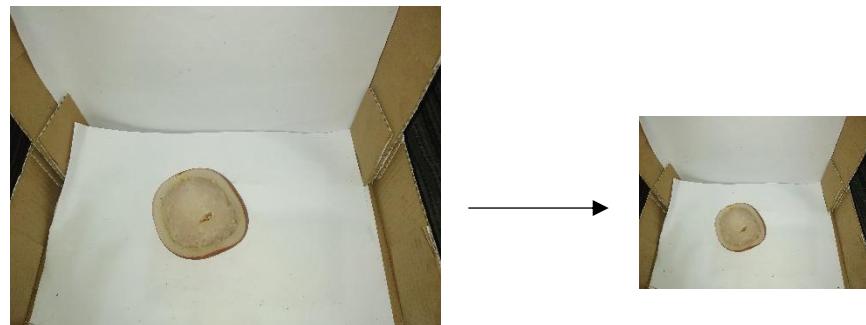
Gambar 4. 2 Proses Akuisisi Citra

2) *Preprocessing* Citra

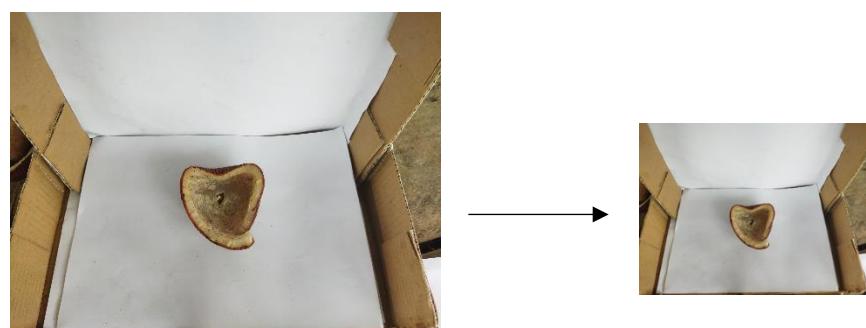
Tahap ini dilakukan untuk memproses citra sebelum digunakan untuk pemodelan. Pada penelitian ini pemrosesan citra dilakukan dengan normalisasi ukuran citra (*resize image*), menghapus latar belakang (*remove background*), dan pemotongan citra (*cropping*).

a. *Resize*

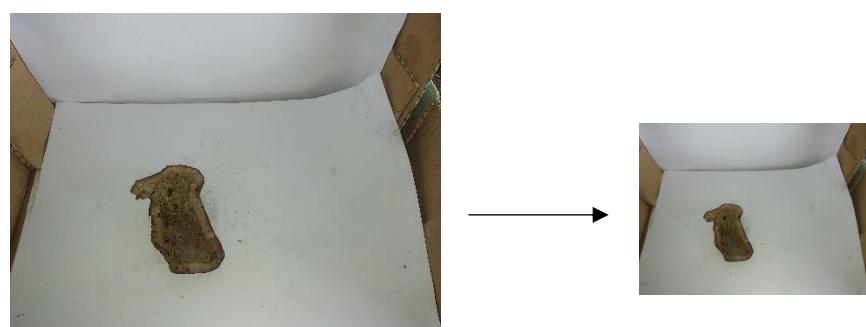
Proses *resize* dilaksanakan dengan mengubah dimensi citra-citra yang sebelumnya memiliki ukuran yang bervariasi dan besar agar dapat seragam dan kecil. Tujuan utama dari langkah ini untuk mengoptimalkan durasi pelatihan model yang digunakan. Citra-citra yang telah melalui proses *resize* akan memiliki ukuran seragam yaitu 800x600 piksel. Berikut merupakan contoh dari hasil *resize* citra.



Gambar 4. 3 Hasil *Resize* Citra Kopra Edible



Gambar 4. 4 Hasil *Resize* Citra Kopra Reguler



Gambar 4. 5 Hasil *Resize* Citra Kopra Reject

b. *Remove Background*

Setelah proses *resize* ukuran citra, maka tahapan selanjutnya adalah penghapusan latar belakang (*remove background*). Proses ini dilakukan untuk menghapus latar belakang yang tidak relevan atau mengganggu sebuah gambar atau citra. Tujuannya untuk memfokuskan objek dan menyeragamkan latar belakang agar terlihat perbedaan antara objek dan latar belakang. Berikut merupakan contoh hasil penghapusan latar belakang citra.



Gambar 4. 6 Hasil Penghapusan Latar Belakang Citra Kopra Edible



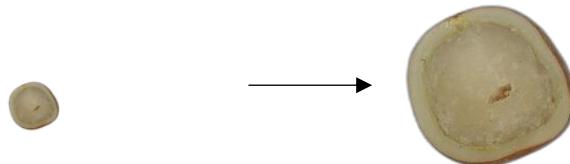
Gambar 4. 7 Hasil Penghapusan Latar Belakang Citra Kopra Reguler



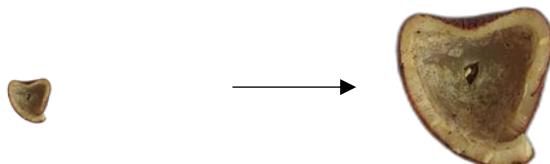
Gambar 4. 8 Hasil Penghapusan Latar Belakang Citra Kopra Reject

c. *Cropping*

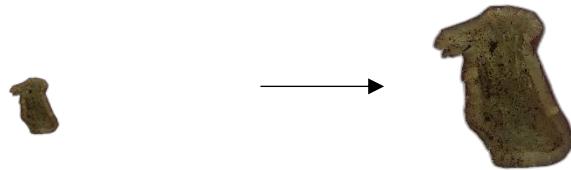
Setelah melakukan penghapusan latar belakang, maka tahapan selanjutnya adalah pemotongan citra (*cropping*). Proses *cropping* dilakukan untuk memfokuskan citra. Dilakukan pemotongan citra dengan memanfaatkan kotak pembatas yang mencakup seluruh isi gambar. Berikut merupakan contoh hasil *cropping* citra.



Gambar 4. 9 Hasil *Cropping* Citra Kopra Edible



Gambar 4. 10 Hasil *Cropping* Citra Kopra Reguler



Gambar 4. 11 Hasil *Cropping* Citra Kopra Reject

3) Ekstraksi Fitur

Tahap ini dilakukan untuk merubah data citra yang dikonversi menjadi data numerik. Ekstraksi fitur yang digunakan pada penelitian ini yaitu ekstraksi fitur warna, bentuk, dan tekstur. Untuk ekstraksi fitur warna, model ini menggunakan *color moment* yang mencakup *mean* RGB, HSV, Grayscale dan standar deviasi. Sedangkan untuk ekstraksi fitur bentuk, model ini menggunakan area dan perimeter. Selain itu, dalam proses ekstraksi fitur tekstur, model ini menggunakan GLCM (Gray Level Co-occurrence Matrix) dengan jarak 1 dan sudut 0, serta fitur-fitur seperti *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*. Hasil dari ekstraksi fitur ini akan dijadikan sebagai parameter untuk proses klasifikasi.

4) *Split Dependent* dan *Independent value*

Pada tahap ini dilakukan pemisahan data yaitu *dependent value* dan *independent value*. Dalam penelitian ini, *dependent value* berisi informasi tentang kategori atau kelas dari jenis kopra yaitu, kopra edible, regular, dan reject. Sedangkan *independent value* digunakan sebagai inputan atau atribut yang digunakan untuk melatih model. *Independent value* dalam penelitian ini meliputi ekstraksi fitur yaitu *meanR*, *meanG*, *meanB*, *meanH*, *meanS*, *meanV*, *meanGrayscale*, Standar-Deviasi, Luas, *Parimeter*, *Contrast*, *Dissimililarity*, *Homogenity*, *Energy*, *Correlation*.

5) *Feature Scaling*

Setelah mendapatkan nilai ekstraksi ciri pada masing-masing citra pada tahap sebelumnya, pada tahap ini akan dilakukan normalisasi data untuk mengubah skala atau rentang nilai dari fitur-fitur yang ada pada dataset.

6) *Train-Test Split*

Pada tahap ini dilakukan pembagian dataset menjadi dua bagian yaitu data *training* dan data *testing*. Adapun pembagian antara data *training* dan data *testing* adalah 90: 10 dimana 90 % untuk data *training* dan 10% untuk data *testing*.

7) Penerapan Metode K-Nearest Neighbour

Pada tahap ini dilakukan klasifikasi citra berdasarkan hasil ekstraksi yang didapat pada tahap sebelumnya. Algoritma K-*Nearest Neighbor* digunakan untuk melakukan klasifikasi citra dengan membandingkan data *testing* dan data *training* menggunakan rumus *Euclidean distance* dengan nilai $k = 3$ dari metode K-*Nearest Neighbor*.

8) Pengujian

Pada tahap ini dilakukan pengujian perfomansi untuk mengevaluasi hasil klasifikasi dari metode yang diusulkan. Untuk mengetahui performa dari algoritma K-*Nearest Neighbor*, penelitian ini menggunakan metrik akurasi, *confusion matrix*, dan *classification report*.

4.4 Collection of Input Data

Input yang akan diproses pada simulasi ini yaitu tiga folder jenis kopra yang masing-masing folder tersebut terdiri dari citra dengan format .jpg. Berikut merupakan contoh sampel citra yang penulis gunakan sebagai data input:



Gambar 4. 12 Contoh Sampel Kopra Edible



Gambar 4. 13 Contoh Sampel Kopra Reguler



Gambar 4. 14 Contoh Sampel Kopra Reject

4.5 Modeling

4.6.1 Perhitungan Manual Ekstraksi Fitur Warna

Berikut ini adalah tahapan perhitungan ekstraksi fitur warna RGB, HSV, dan Grayscale:

1) Ekstraksi Fitur Warna RGB

Langkah pertama untuk melakukan ekstraksi ciri warna RGB pada suatu citra yaitu dengan mengkonversinya dari BGR (default dari OpenCV) menjadi citra RGB. Selanjutnya, melakukan pengambilan nilai *mean* pada setiap citra. Nilai *mean* RGB ini akan digunakan sebagai salah satu ciri warna dari jenis kopra yang di ekstraksi. Misalkan terdapat citra RGB 3×3 yang dapat digunakan untuk mencari nilai ciri warna dari suatu citra.

R : 80	R : 85	R : 92
--------	--------	--------

G : 81 B : 76	G : 86 B : 80	G : 92 B : 85
R : 97 G : 97 B : 89	R : 103 G : 102 B : 92	R : 106 G : 104 B : 94
R : 107 G : 107 B : 94	R : 110 G : 108 B : 97	R : 111 G : 109 B : 96

Gambar 4. 15 Citra RGB 3×3

Berikut merupakan proses perhitungan untuk mencari nilai *mean* berdasarkan komponen nilai citra warna RGB dengan persamaan (2.6):

$$\begin{aligned} MeanR &= \frac{1}{9}(80 + 85 + 92 + 97 + 103 + 106 + 107 + 110 \\ &\quad + 111) \\ &= 99 \end{aligned}$$

$$\begin{aligned} MeanG &= \frac{1}{9}(81 + 86 + 92 + 97 + 102 + 104 + 107 + 108 \\ &\quad + 109) \\ &= 98,4444 \end{aligned}$$

$$\begin{aligned} MeanB &= \frac{1}{9}(76 + 80 + 85 + 89 + 92 + 94 + 94 + 97 + 96) \\ &= 89,2222 \end{aligned}$$

Berdasarkan hasil perhitungan untuk mencari nilai *mean* dari momen warna RGB diperoleh tiga nilai ciri warna yaitu: *MeanR* = 99, *MeanG* = 98,4444, dan *MeanB* = 89,222.

2) Ekstraksi Fitur Warna RGB

Langkah pertama untuk melakukan ekstraksi ciri warna HSV pada suatu citra yaitu dengan mengkonversinya dari BGR (default dari OpenCV) menjadi citra HSV. Selanjutnya, melakukan pengambilan nilai *mean* pada setiap citra. Nilai *mean* HSV ini akan digunakan sebagai salah satu ciri warna dari jenis kopra yang di ekstraksi. Misalkan terdapat citra HSV 3×3 yang dapat digunakan untuk mencari nilai ciri warna dari suatu citra.

H : 36 S : 16 V : 81	H : 35 S : 18 V : 86	H : 30 S : 19 V : 92
H : 30 S : 21 V : 97	H : 27 S : 27 V : 103	H : 25 S : 29 V : 106
H : 30 S : 31 V : 107	H : 25 S : 30 V : 110	H : 26 S : 34 V : 111

Gambar 4. 16 Citra HSV 3x3

Berikut merupakan proses perhitungan untuk mencari nilai *mean* berdasarkan komponen nilai citra warna HSV dengan persamaan (2.6):

$$\begin{aligned} MeanH &= \frac{1}{9}(36 + 35 + 30 + 30 + 27 + 25 + 30 + 25 + 26) \\ &= 29,3333 \end{aligned}$$

$$\begin{aligned} MeanS &= \frac{1}{9}(16 + 18 + 19 + 21 + 27 + 29 + 31 + 30 + 34) \\ &= 25 \end{aligned}$$

$$\begin{aligned} MeanV &= \frac{1}{9}(81 + 86 + 92 + 97 + 103 + 106 + 107 + 110 \\ &\quad + 111) \\ &= 99,2222 \end{aligned}$$

Berdasarkan hasil perhitungan untuk mencari nilai *mean* dari momen warna HSV diperoleh tiga nilai ciri warna yaitu: *MeanH* = 29,3333, *MeanS* = 25, dan *MeanV* = 99,222.

3) Ekstraksi ciri warna *Grayscale*

Langkah pertama untuk melakukan ekstraksi ciri warna *Grayscale* pada suatu citra yaitu dengan mengkonversinya dari BGR (default dari OpenCV) menjadi citra *Grayscale*. Berikut merupakan contoh perhitungan untuk mendapatkan citra berwarna *grayscale* menggunakan persamaan (2.5):

$$GrayScale_{11} = (0,3 \times 80) + (0,59 \times 81) + (0,11 \times 76)$$

$$GrayScale_{11} = 80,15$$

Selanjutnya, melakukan pengambilan nilai *mean* pada setiap citra. Nilai *mean* Grayscale ini akan digunakan sebagai salah satu ciri warna dari jenis kopra yang di ekstraksi. Misalkan terdapat citra Greyscale 3×3 yang dapat digunakan untuk mencari nilai ciri warna dari suatu citra.

80	85	91
96	101	103
106	107	108

Gambar 4. 17 Citra *Grayscale* 3×3

Berikut merupakan proses perhitungan untuk mencari nilai *mean* berdasarkan komponen nilai citra warna *Grayscale* dengan persamaan (2.6):

$$\begin{aligned} MeanGray &= \frac{1}{9}(80 + 85 + 91 + 96 + 101 + 103 + 106 + 107 \\ &\quad + 108) \\ &= 97,4444 \end{aligned}$$

Kemudian, melakukan perhitungan standar deviasi dari nilai *Grayscale* dengan persamaan (2.7):

$$\begin{aligned} Standar - Deviasi &= \sqrt{\frac{(80 - 97,4444)^2 + (85 - 97,4444)^2 + (91 - 97,4444)^2 + (96 - 97,4444)^2 + (101 - 97,4444)^2 + (103 - 97,4444)^2 + (106 - 97,4444)^2 + (107 - 97,4444)^2 + (108 - 97,4444)^2}{9 - 1}} \\ &= \sqrt{\frac{+ (107 - 97,4444)^2 + (108 - 97,4444)^2}{9 - 1}} \\ &= 58,94838 \end{aligned}$$

Terakhir, melakukan perhitungan variasi dari nilai *Grayscale*:

$$\begin{aligned} Variasi &= 58,94838^2 \\ &= 3474,912 \end{aligned}$$

4.6.2 Perhitungan Manual Ekstraksi Fitur Bentuk

Ekstraksi bentuk yang digunakan pada penilitian ini yaitu *area* dan *perimeter*. Dalam hal ini penulis menggunakan contoh citra *grayscale*.

80	85	91	25	50
96	101	103	75	111
106	150	200	96	91
70	135	167	104	35
50	100	120	83	89

Gambar 4. 18 Citra *Grayscale* 5×5

Selanjutnya melakukan *thresholding* dengan nilai ambang 127, Hasil dari operasi *thresholding* ini akan menghasilkan citra biner di mana piksel-piksel yang awalnya memiliki nilai di atas 127 akan menjadi putih (nilai 255) untuk mewakili objek, sedangkan piksel-piksel yang memiliki nilai di bawah atau sama dengan 127 akan menjadi hitam (nilai 0) untuk mewakili latar belakang.

0	0	0	0	0
0	0	0	0	0
0	255	255	0	0
0	255	255	0	0
0	0	0	0	0

Gambar 4. 19 Citra Biner 5×5

Selanjutnya, melakukan pencarian kontur. Setiap segmen garis yang membentuk kontur akan memberikan kontribusi terhadap *area* dan *perimeter* kontur. Jika ditemukan kontur, maka akan dihitung luas *area* dan panjang keliling (*perimeter*) dari kontur terbesar. Jika tidak ditemukan kontur, *area* dan *perimeter* akan diatur sebagai 0.

Dalam hal ini kontur ditemukan dengan nilai 255 di dalam kotak tengah. Kemudian, mencari kontur terbesar. Karena dalam contoh yang diberikan, kontur yang ditemukan hanya 1, maka kontur tersebut dijadikan sebagai kontur terbesar. Kontur inilah yang dipilih sebagai kontur utama dalam objek. Berikutnya adalah menghitung nilai *area* dan *perimeter*.

$$\begin{aligned} \text{Area} &= \text{jumlah piksel dengan nilai 255 pada kontur terbesar} \\ &= 4 \text{ Piksel} \end{aligned}$$

$$\begin{aligned}
 \text{Perimeter} &= 2 \times (\text{jumlah piksel horizontal} + \text{jumlah piksel vertikal}) \\
 &= 2 \times (2 + 4) = 12 \text{ Piksel}
 \end{aligned}$$

4.6.3 Perhitungan Manual Ekstraksi Fitur Tekstur

Ekstraksi fitur yang digunakan pada penilitian ini yaitu metode GLCM (*Gray Level Co-occurrence Matrix*) dengan beberapa fitur seperti *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*. Berikut merupakan tahapan untuk perhitungan manualnya:

- 1) Dalam hal ini, menggunakan contoh matriks 3×3 .

4	3	1
7	6	4
6	4	3

- 2) Menentukan hubungan spasial antar piksel dengan inisiasi nilai jarak dan sudut. Dalam hal ini menggunakan jarak 1 dan sudut 0° .
- 3) Membuat matiks GLCM untuk setiap pikselnya dengan menghitung seberapa sering pasangan piksel muncul pada jarak dan arah yang ditentukan.

0 1 2 3 4 5 6 7

0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0
4	0	0	0	2	0	0	0

5	0	0	0	0	0	0	0	0
6	0	0	0	0	2	0	0	0
7	0	0	0	0	0	0	1	0

- 4) Normalisasi matriks untuk mendapatkan distribusi probabilitasnya. Hal ini dilakukan dengan membagi setiap elemen matriks dengan jumlah total pasangan piksel yang dihitung.

0 1 2 3 4 5 6 7

0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0,167	0	0	0	0	0	0
4	0	0	0	0,333	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0,333	0	0	0
7	0	0	0	0	0	0	0,167	0

- 5) Menghitung nilai ekstraksi fitur ekstraksi GLCM dari hasil normalisasi.

- a. *Contrast*, berikut merupakan perhitungan contrast menggunakan persamaan (2.8):

$$\text{Contrast}_{(3,1)} = (3 - 1)^2 \times 0,167 = 0,668$$

$$\text{Contrast}_{(4,3)} = (4 - 3)^2 \times 0,333 = 0,333$$

$$\text{Contrast}_{(6,4)} = (6 - 4)^2 \times 0,333 = 1,332$$

$$\text{Contrast}_{(7,6)} = (7 - 6)^2 \times 0,167 = 0,167$$

$$Contrast = 0,668 + 0,333 + 1,332 + 0,167 = 2,333$$

- b. *Dissimilarity*, berikut merupakan perhitungan *dissimilarity* menggunakan persamaan (2.9):

$$Dissimilarity_{(3,1)} = |3 - 1| \times 0,167 = 0,333$$

$$Dissimilarity_{(4,3)} = |4 - 3| \times 0,333 = 0,333$$

$$Dissimilarity_{(6,4)} = |6 - 4| \times 0,333 = 0,667$$

$$Dissimilarity_{(7,6)} = |7 - 6| \times 0,167 = 0,167$$

$$Dissimilarity = 0,333 + 0,333 + 0,667 + 0,167 = 1,5$$

- c. *Homogeneity*, berikut merupakan perhitungan *homogeneity* menggunakan persamaan (2.10):

$$Homogeneity_{(3,1)} = \frac{0,167}{1 + (3 - 1)^2} = 0,033$$

$$Homogeneity_{(4,3)} = \frac{0,333}{1 + (4 - 3)^2} = 0,167$$

$$Homogeneity_{(6,4)} = \frac{0,333}{1 + (6 - 4)^2} = 0,067$$

$$Homogeneity_{(7,6)} = \frac{0,167}{1 + (7 - 6)^2} = 0,084$$

$$Homogeneity = 0,033 + 0,167 + 0,067 + 0,084 = 0,267$$

- d. *Energy*, berikut merupakan perhitungan *energy* menggunakan persamaan (2.11):

$$\begin{aligned} Energy &= 0,167^2 + 0,333^2 + 0,333^2 + 0,167^2 \\ &= 0,278 \end{aligned}$$

e. *Correlation*, berikut merupakan perhitungan *homogeneity* menggunakan persamaan (2.12):

$$\mu i = (3 \times 0,167) + (4 \times 0,333) + (6 \times 0,333) \\ + (7 \times 0,167)$$

$$\mu i = 5$$

$$\mu j = (1 \times 0,167) + (3 \times 0,333) + (4 \times 0,333) \\ + (6 \times 0,167)$$

$$\mu j = 3,5$$

$$\sigma i_{(3,1)} = (3 - 5)^2 \times 0,167 = 0,668$$

$$\sigma i_{(4,3)} = (4 - 5)^2 \times 0,333 = 0,333$$

$$\sigma i_{(6,4)} = (6 - 5)^2 \times 0,333 = 0,333$$

$$\sigma i_{(7,6)} = (7 - 5)^2 \times 0,167 = 0,668$$

$$\sigma i = 0,668 + 0,333 + 0,333 + 0,668 = 2,002$$

$$\sigma j_{(3,1)} = (1 - 3,5)^2 \times 0,167 = 1,044$$

$$\sigma j_{(4,3)} = (3 - 5)^2 \times 0,333 = 0,083$$

$$\sigma j_{(6,4)} = (4 - 5)^2 \times 0,333 = 0,083$$

$$\sigma j_{(7,6)} = (6 - 5)^2 \times 0,167 = 1,044$$

$$\sigma j = 0,668 + 0,333 + 0,333 + 0,668 = 2,254$$

$$\text{Correlation}_{(3,1)} = \frac{(3 - 5) \times (1 - 3,5) \times 0,167}{0,668 \times 1,044} \\ = 1,198$$

$$\text{Correlation}_{(4,3)} = \frac{(4 - 5) \times (3 - 3,5) \times 0,333}{0,333 \times 0,083} \\ = 6,006$$

$$\text{Correlation}_{(6,4)} = \frac{(6 - 5) \times (4 - 3,5) \times 0,333}{0,333 \times 0,083} \\ = 6,006$$

$$\text{Correlation}_{(7,6)} = \frac{(3 - 5) \times (1 - 3,5) \times 0,167}{0,668 \times 1,044} \\ = 1,198$$

$$\text{Correlation} = 1,198 + 6,006 + 6,006 + 1,198$$

$$= 14,407$$

4.6.4 Pengklasifikasian dengan K-Nearest Nighbour

Untuk melakukan klasifikasi citra kopra, penulis menggunakan metode *K-Nearest Neighbour*. Klasifikasi ini menentukan jenis kopra dari data *testing* yang sudah tersedia. Berikut merupakan contoh langkah-langkah untuk mengklasifikasikan jenis kopra berdasarkan ekstraksi fitur yang telah diperoleh pada tahap sebelumnya.

Diketahui data *training* dan data *testing* ekstraksi fitur pada citra kopra yang terdiri dari ekstraksi warna, bentuk, dan tekstur.

Tabel 4. 3 Contoh Data *Training* Citra Kopra

Mean -R	Mean -G	Mean -B	Mean -H	Mean -V	Mean -S	Mean -Gray	Standar -Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homogeneity	Energy	Correlation	Jenis Kopra
0,664	0,641	0,677	0,084	0,664	0,291	0,652	0,485	0,046	0,339	0,070	0,137	0,508	0,148	0,951	Edible
0,650	0,752	0,834	0,172	0,653	0,043	0,728	0,609	0,258	0,181	0,048	0,108	0,492	0,207	0,987	Edible
0,517	0,610	0,576	0,169	0,525	0,257	0,578	0,566	0,098	0,191	0,118	0,196	0,455	0,227	0,942	Reguler
0,652	0,765	0,759	0,205	0,525	0,208	0,729	0,646	0,167	0,359	0,155	0,229	0,370	0,154	0,942	Reguler
0,837	0,797	0,509	0,139	0,838	0,690	0,782	0,694	0,425	0,887	0,530	0,661	0,145	0,192	0,794	Reject
0,490	0,399	0,177	0,102	0,490	0,853	0,406	0,341	0,008	0,084	0,147	0,288	0,324	0,224	0,824	Reject

Tabel 4. 4 Contoh Data *Testing* Citra Kopra

Mean -R	Mean -G	Mean -B	Mean -H	Mean -V	Mean -S	Mean -Gray	Standar -Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homogeneity	Energy	Correlation	Jenis Kopra
0,696	0,769	0,864	0,143	0,697	0,079	0,755	0,555	0,274	0,186	0,065	0,102	0,530	0,110	0,969	?

- 1) Langkah pertama yang harus dilakukan yaitu menentukan nilai k. Nilai k yang digunakan pada penelitian ini adalah 3.
- 2) Selanjutnya menghitung rentang atau jarak data *training* dengan data *testing* menggunakan rumus (2.13):

$$X1 = \sqrt{(0,6636 - 0,6959)^2 + (0,6414 - 0,7686)^2 + (0,6765 - 0,8640)^2 + (0,0835 - 0,1432)^2 + (0,6637 - 0,6971)^2 + (0,2913 - 0,0787)^2 + (0,6519 - 0,7552)^2 + (0,4846 - 0,5548)^2 + (0,0461 - 0,2743)^2 + (0,3391 - 0,1861)^2 + (0,0698 - 0,0646)^2 + (0,1371 - 0,1020)^2 + (0,5084 - 0,5296)^2 + (0,1479 - 0,1098)^2 + (0,9514 - 0,9695)^2}$$

$$X1 = \sqrt{0,1963}$$

$$X1 = 0,4431$$

$$X2 = \sqrt{(0,6497 - 0,6959)^2 + (0,7520 - 0,7686)^2 + (0,8343 - 0,8640)^2 + (0,1725 - 0,1432)^2 + (0,6532 - 0,0435)^2 + (0,0435 - 0,0787)^2 + (0,7281 - 0,7552)^2 + (0,6086 - 0,5548)^2 + (0,2575 - 0,2743)^2 + (0,1806 - 0,1861)^2 + (0,0475 - 0,0646)^2 + (0,1080 - 0,1020)^2 + (0,4915 - 0,5296)^2 + (0,2068 - 0,1098)^2 + (0,9870 - 0,9650)^2}$$

$$X2 = \sqrt{0,0228}$$

$$X2 = 0,1510$$

$$X3 = \sqrt{(0,5168 - 0,6959)^2 + (0,6100 - 0,7686)^2 + (0,5764 - 0,8640)^2 + (0,1685 - 0,1432)^2 + (0,5246 - 0,0435)^2 + (0,2569 - 0,0787)^2 + (0,5780 - 0,7552)^2 + (0,5663 - 0,5548)^2 + (0,0978 - 0,2743)^2 + (0,1912 - 0,1861)^2 + (0,1180 - 0,646)^2 + (0,1964 - 0,1020)^2 + (0,4554 - 0,5296)^2 + (0,2269 - 0,1098)^2 + (0,9421 - 0,9695)^2}$$

$$X3 = \sqrt{0,2963}$$

$$X3 = 0,5444$$

$$X4 = \sqrt{(0,6552 - 0,6959)^2 + (0,7648 - 0,7686)^2 + (0,7586 - 0,8640)^2 + (0,2050 - 0,1432)^2 + (0,5246 - 0,0435)^2 + (0,2075 - 0,0787)^2 + (0,7295 - 0,7552)^2 + (0,6460 - 0,5548)^2 + (0,1671 - 0,2743)^2 + (0,3585 - 0,1861)^2 + (0,1547 - 0,646)^2 + (0,2292 - 0,1020)^2 + (0,3701 - 0,5296)^2 + (0,1542 - 0,1098)^2 + (0,9421 - 0,9695)^2}$$

$$X4 = \sqrt{0,1659}$$

$$X4 = 0,4073$$

$$X5 = \sqrt{(0,8366 - 0,6959)^2 + (0,7970 - 0,7686)^2 + (0,5094 - 0,8640)^2 + (0,1393 - 0,1432)^2 + (0,8379 - 0,0435)^2 + (0,6902 - 0,0787)^2 + (0,7818 - 0,7552)^2 + (0,6936 - 0,5548)^2 + (0,4246 - 0,2743)^2 + (0,8871 - 0,1861)^2 + (0,5299 - 0,646)^2 + (0,6612 - 0,1020)^2 + (0,1453 - 0,5296)^2 + (0,1919 - 0,1098)^2 + (0,7937 - 0,9695)^2}$$

$$X5 = \sqrt{1,7884}$$

$$X5 = 1,3373$$

$$X6 = \sqrt{(0,4903 - 0,6959)^2 + (0,3987 - 0,7686)^2 + (0,1768 - 0,8640)^2 + (0,1020 - 0,1432)^2 + (0,4904 - 0,0435)^2 + (0,8527 - 0,0787)^2 + (0,4062 - 0,7552)^2 + (0,3414 - 0,5548)^2 + (0,0084 - 0,2743)^2 + (0,0836 - 0,1861)^2 + (0,1473 - 0,646)^2 + (0,2883 - 0,1020)^2 + (0,3241 - 0,5296)^2 + (0,2245 - 0,1098)^2 + (0,8240 - 0,9695)^2}$$

$$X6 = \sqrt{1,6614}$$

$$X6 = 1,2890$$

Tabel 4. 5 Nilai Hasil Perhitungan Jarak Data *Training* dengan Data *Testing*

<i>Mea n-R</i>	<i>Mea n-G</i>	<i>Mea n-B</i>	<i>Mea n-H</i>	<i>Mea n-V</i>	<i>Mea n-S</i>	<i>Mea n- Gray</i>	<i>Standa r- Deviasi</i>	<i>Luas</i>	<i>Peri mete r</i>	<i>Contr ast</i>	<i>Dissi milari ty</i>	<i>Homo geneit y</i>	<i>Energ y</i>	<i>Corre lation</i>	<i>Jenis Kopra</i>	<i>Nilai Jarak (Eucli dean)</i>
0,664	0,641	0,677	0,084	0,664	0,291	0,652	0,485	0,046	0,339	0,070	0,137	0,508	0,148	0,951	Edible	0,443
0,650	0,752	0,834	0,172	0,653	0,043	0,728	0,609	0,258	0,181	0,048	0,108	0,492	0,207	0,987	Edible	0,151
0,517	0,610	0,576	0,169	0,525	0,257	0,578	0,566	0,098	0,191	0,118	0,196	0,455	0,227	0,942	Reguler	0,544
0,652	0,765	0,759	0,205	0,525	0,208	0,729	0,646	0,167	0,359	0,155	0,229	0,370	0,154	0,942	Reguler	0,407
0,837	0,797	0,509	0,139	0,838	0,690	0,782	0,694	0,425	0,887	0,530	0,661	0,145	0,192	0,794	Reject	1,337
0,490	0,399	0,177	0,102	0,490	0,853	0,406	0,341	0,008	0,084	0,147	0,288	0,324	0,224	0,824	Reject	1,289

- 3) Setelah mendapatkan nilai jarak antara data *training* dengan data *testing*, selanjutnya jarak tersebut diurutkan dari nilai terkecil sampai yang terbesar.

Tabel 4. 6 Hasil Pengurutan Nilai Jarak

<i>Mea n-R</i>	<i>Mea n-G</i>	<i>Mea n-B</i>	<i>Mea n-H</i>	<i>Mea n-V</i>	<i>Mea n-S</i>	<i>Mea n- Gray</i>	<i>Standa r- Deviasi</i>	<i>Luas</i>	<i>Peri mete r</i>	<i>Contr ast</i>	<i>Dissi milari ty</i>	<i>Homo geneit y</i>	<i>Energ y</i>	<i>Corre lation</i>	<i>Jenis Kopra</i>	<i>Nilai Jarak (Eucli dean)</i>
0,650	0,752	0,834	0,172	0,653	0,043	0,728	0,609	0,258	0,181	0,048	0,108	0,492	0,207	0,987	Edible	0,151
0,652	0,765	0,759	0,205	0,525	0,208	0,729	0,646	0,167	0,359	0,155	0,229	0,370	0,154	0,942	Reguler	0,407
0,664	0,641	0,677	0,084	0,664	0,291	0,652	0,485	0,046	0,339	0,070	0,137	0,508	0,148	0,951	Edible	0,443
0,517	0,610	0,576	0,169	0,525	0,257	0,578	0,566	0,098	0,191	0,118	0,196	0,455	0,227	0,942	Reguler	0,544
0,490	0,399	0,177	0,102	0,490	0,853	0,406	0,341	0,008	0,084	0,147	0,288	0,324	0,224	0,824	Reject	1,289

0,837	0,797	0,509	0,139	0,838	0,690	0,782	0,694	0,425	0,887	0,530	0,661	0,145	0,192	0,794	Reject	1,337
-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	-------

Dari hasil pengurutan tersebut, kita akan melihat tiga kelas tetangga terdekatnya (pada langkah pertama kita menentukan nilai $k=3$).

Dalam kasus ini, kita menemukan bahwa terdapat 1 kelas regular dan 2 kelas edible. Berdasarkan mayoritas kelas dari tetangga terdekat, kita dapat menyimpulkan bahwa data baru tersebut masuk ke dalam kelas edible.

4.7 Simulation

Proses simulasi dilakukan berdasarkan model konseptual yang telah dibuat pada tahap sebelumnya. Adapun penjelasan simulasi dengan langkah-langkah beserta potongan baris kode sebagai berikut:

1) Preprocessing Citra

Langkah pertama yang dilakukan adalah meng-*import Library* yang dibutuhkan seperti Image, imageOps, os, dan remove.

```
from PIL import Image, ImageOps
import os
from rembg import remove
```

Pemrosesan citra yang pertama yaitu *resize* ukuran citra. Dimulai dengan membuat direktori yang dibutuhkan untuk menampung hasil proses *resize* dengan menggunakan fungsi **os.makedirs()**. Menggunakan **exist_ok=True** untuk memastikan bahwa direktori yang akan dibuat jika belum ada dan jika sudah ada, tidak akan menimbulkan masalah. Lalu melakukan iterasi pada setiap folder di dalam direktori yang dituju dengan fungsi **os.listdir()** dan **os.path.join()** untuk membentuk path lengkap ke setiap folder. Kemudian, melakukan *resize* citra menjadi 800×600 menggunakan fungsi **img.resize(800×600)**. Setelah itu, menyimpan hasil resize pada folder yang sudah dibuat sebelumnya menggunakan **img.save()**.

```
os.makedirs("Kopra_Resize", exist_ok=True)
os.makedirs("Kopra_Resize/Kopra Edible", exist_ok=True)
os.makedirs("Kopra_Resize/Kopra Reguler", exist_ok=True)
os.makedirs("Kopra_Resize/Kopra Reject", exist_ok=True)

for folder in os.listdir("Kopra"):
    for filename in os.listdir(os.path.join("Kopra", folder)):
        img = Image.open(os.path.join("Kopra", folder,
filename))
        img = img.resize((800, 600))
        img.save(os.path.join("Kopra_Resize", folder,
filename))
```

```
    print("Image resized to:",
os.path.join("Kopra_Resize", folder, filename))
```

Pemrosesan citra yang kedua yaitu *remove background* citra. Sama seperti proses sebelumnya, proses ini dimulai dengan membuat direktori yang dibutuhkan dan melakukan iterasi melalui setiap folder dan file di dalam direktori yang dituju. Lalu setiap gambar akan dibuka dan hapus latar belakangnya dengan fungsi **remove()**. Setelah latar belakang dihapus, nama file diubah dengan mengganti ekstensi .jpg menjadi .png. Setelah itu, menyimpan hasil *remove background* pada folder yang sudah dibuat sebelumnya menggunakan **result.save()**.

```
os.makedirs("Kopra_Rembg", exist_ok=True)
os.makedirs("Kopra_Rembg/Kopra Edible", exist_ok=True)
os.makedirs("Kopra_Rembg/Kopra Reguler", exist_ok=True)
os.makedirs("Kopra_Rembg/Kopra Reject", exist_ok=True)

for folder in os.listdir("Kopra_Resize"):
    for filename in os.listdir(os.path.join("Kopra_Resize",
folder)):
        img = Image.open(os.path.join("Kopra_Resize", folder,
filename))
        result = remove(img)
        filename = filename.replace(".jpg", ".png")
        result.save(os.path.join("Kopra_Rembg", folder,
filename))
        print("Background image removed on:",
os.path.join("Kopra_Rembg", folder, filename))
```

Pemrosesan citra yang terakhir adalah *cropping*. Proses ini pun sama seperti sebelum-sebelumnya, dimulai dengan membuat direktori yang dibutuhkan dan melakukan iterasi melalui setiap folder dan file di dalam direktori yang dituju. Lalu setiap gambar akan dibuka dan di *crop* dengan fungsi **img.crop(img.getbbox())**. *Crop* dilakukan dengan memanfaatkan **img.getbbox** untuk mendapatkan kotak pembatas yang mencakup seluruh isi gambar. Gambar yang telah di *crop* disimpan menggunakan **img.save()**..

```
os.makedirs("Kopra_Crop", exist_ok=True)
os.makedirs("Kopra_Crop/Kopra Edible", exist_ok=True)
os.makedirs("Kopra_Crop/Kopra Reguler", exist_ok=True)
```

```

os.makedirs("Kopra_Crop/Kopra Reject", exist_ok=True)

for folder in os.listdir("Kopra_Rembg"):
    for filename in os.listdir(os.path.join("Kopra_Rembg",
    folder)):
        img = Image.open(os.path.join("Kopra_Rembg", folder,
        filename))
        img = img.crop(img.getbbox())
        img.save(os.path.join("Kopra_Crop", folder, filename))
        print("Image resized to:", os.path.join("Kopra_Crop",
        folder, filename))

```

2) Ekstraksi fitur

Langkah pertama yang dilakukan adalah meng-*import Library* yang dibutuhkan seperti pandas, numpy, openCv, os, dan csv.

```

import pandas as pd
import numpy as np
import cv2
import os
import csv
from skimage.feature import graycomatrix, graycoprops

```

Selanjutnya, menginisialisasi variabel ‘path’ yang berisi lokasi tempat citra-citra yang akan di ekstraksi fiturnya dan mengambil daftar file dalam direktori. Kemudian membuat beberapa list kosong untuk menyimpan hasil ekstraksi fitur dari setiap citra.

```

path = 'E:/A A Skripsi/SKRIPSI/AA SKRIPSI FIX
BISMILLAH/dataset/Kopra_Crop/Kopra Edible'
data = os.listdir(path)

rataR = []
rataG = []
rataB = []
rataH = []
rataS = []
rataV = []
ratagray = []
stand = []
luas = []
perimeter = []
contrast = []
dissimilarity = []

```

```
homogeneity = []
energy = []
correlation = []
label = []
```

Kemudian melakukan iterasi ekstraksi fitur pada setiap citra. Ekstraksi fitur pertama yang dilakukan adalah ekstraksi fitur warna. Pada setiap iterasi, gambar dibaca terlebih dahulu, lalu diubah ke format RGB. Langkah ini bertujuan untuk memastikan bahwa urutan karnal warna yang digunakan adalah RGB. Fungsi yang digunakan untuk mengubah warna citra dari BGR menjadi RGB yaitu **cvtColor** dari *library* OpenCV. Selanjutnya memisahkan komponen warna R (*Red*), G (*Green*), dan B (*Blue*) dari citra menggunakan fungsi **split** dari *library* OpenCV. Kemudian menghitung rata-rata nilai piksel untuk masing-masing komponen RGB menggunakan fungsi **mean** dari *library* numpy. Dan terakhir menambahkan nilai rata-rata pada setiap komponen ke dalam suatu daftar dengan fungsi **append**. Nilai *mean* RGB ini akan digunakan sebagai salah satu ciri warna dari jenis kopra yang di ekstraksi.

Setelah melakukan ekstraksi fitur warna RGB sebelumnya, berikut merupakan proses ekstraksi fitur warna HSV. Citra yang telah dibaca sebelumnya diubah ke format HSV. Hal ini dilakukan untuk mendapatkan komponen warna H (*Hue*), S (*Saturation*), dan V (*Value*). Setelah citra diubah menjadi format HSV, komponen warna H, S, dan V diekstraksi secara terpisah. Hal ini dilakukan dengan mengakses matriks HSV menggunakan indexing `[:, :, 0]` untuk H, `[:, :, 1]` untuk S, `[:, :, 2]` untuk V. Selanjutnya menghitung nilai rata-rata dari masing-masing komponen warna H, S, dan V. Nilai rata-rata tersebut kemudian ditambahkan ke dalam list ‘rataH’, ‘rataS’, dan ‘rataV’ menggunakan fungsi **append**. Hal ini dilakukan untuk mengumpulkan nilai rata-rata dari setiap komponen warna H, S, dan V dari citra yang sedang diproses ke dalam list terpisah

Ekstraksi fitur warna yang terakhir adalah warna *grayscale*. Citra yang sudah dibaca sebelumnya diubah menjadi citra *grayscale*. Langkah ini bertujuan untuk mengkonversi citra menjadi skala keabuan dengan hanya satu saluran warna. Setelah citra merubah menjadi *grayscale*, nilai rata-rata dari citra tersebut dihitung menggunakan fungsi **mean** dan **std** dari *library* numpy. Hal ini memberikan nilai rata-rata intensitas piksel dari citra *grayscale* yang sedang diproses. Dan terakhir menambahkan nilai rata-rata dan standar deviasi dalam suatu daftar dengan fungsi **append**. Nilai *mean* dan standar deviasi *Grayscale* ini akan digunakan sebagai salah satu ciri warna dari jenis kopra yang di ekstraksi.

```

for gbr in data:
    gbr_read = cv2.imread(os.path.join(path, gbr))
    gbr_rgb = cv2.cvtColor(gbr_read, cv2.COLOR_BGR2RGB)
    (R, G, B) = cv2.split(gbr_rgb)
    meanR = np.mean(R)
    rataR.append(meanR)
    meanG = np.mean(G)
    rataG.append(meanG)
    meanB = np.mean(B)
    rataB.append(meanB)

    gbr_hsv = cv2.cvtColor(gbr_read, cv2.COLOR_BGR2HSV)
    H = gbr_hsv[:, :, 0]
    S = gbr_hsv[:, :, 1]
    V = gbr_hsv[:, :, 2]
    meanH = np.mean(H)
    rataH.append(meanH)
    meanS = np.mean(S)
    rataS.append(meanS)
    meanV = np.mean(V)
    rataV.append(meanV)

    gbr_gray = cv2.cvtColor(gbr_read, cv2.COLOR_BGR2GRAY)
    rata = np.mean(gbr_gray)
    ratagray.append(rata)
    standar = np.std(gbr_gray)
    stand.append(standar)

```

Setelah melakukan ekstraksi fitur wana, selanjutnya melakukan ekstraksi fitur bentuk dari citra dalam format *grayscale*. Pertama, melakukan *thresholding* pada gambar *grayscale* dengan nilai **threshold** 127. Hasil thresholding menghasilkan citra biner di mana piksel di atas *threshold* bernilai maksimum (255) dan piksel di bawah *threshold* bernilai minimum (0). Selanjutnya, dilakukan pencarian kontur dalam gambar *threshold* menggunakan fungsi **findContours** dengan Parameter **RETR_EXTERNAL** yang digunakan untuk mengambil hanya kontur eksternal, dan **CHAIN_APPROX_SIMPLE** untuk menyimpan titik-titik kontur. Kontur-kontur yang ditemukan disimpan dalam variabel *contours*. Jika terdapat kontur yang ditemukan, selanjutnya mencari kontur terbesar berdasarkan luas menggunakan fungsi **max** dan **contourArea**. Luas dan panjang perimeter kontur terbesar dihitung menggunakan fungsi **contourArea** dan **arcLength**. Hasil perhitungan luas dan panjang perimeter tersebut disimpan dalam list luas dan perimeter. Jika tidak ada kontur yang ditemukan, maka list luas dan perimeter akan berisi nilai 0.

```
_ , thresh = cv2.threshold(gbr_gray, 127, 255, 0)
contours, _ = cv2.findContours(thresh, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)

if len(contours) > 0:
    contour = max(contours, key=cv2.contourArea)
    area = cv2.contourArea(contour)
    perimeter_val = cv2.arcLength(contour, True)
else:
    area = 0
    perimeter_val = 0

luas.append(area)
perimeter.append(perimeter_val)
```

Ekstraksi fitur yang terakhir adalah tekstur menggunakan metode GLCM (*Gray Level Co-occurrence Matrix*). Langkah pertama yang dilakukan adalah membentuk GLCM dari gambar *grayscale* menggunakan fungsi **graycomatrix**. Selanjutnya menggunakan Parameter [1] dan [0]

untuk menunjukkan jarak piksel horizontal dan vertikal dalam membangun matriks GLCM serta Parameter 256 untuk menunjukkan jumlah tingkat keabuan yang digunakan dalam gambar grayscale. Kemudian melakukan ekstraksi fitur tekstur menggunakan fungsi **graycoprops**. Fungsi ini menerima matriks GLCM dan jenis fitur yang ingin diekstraksi. Terdapat beberapa jenis fitur yang diekstraksi, yaitu *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*. Kemudian masing-masing nilainya ditambahkan ke dalam list untuk mengumpulkan nilai-nilai tekstur dari setiap citra ke dalam list terpisah.

```
glcm = graycomatrix(gbr_gray, [1], [0], 256,
symmetric=True, normed=True)
contrast_val = graycoprops(glcm, 'contrast')[0, 0]
dissimilarity_val = graycoprops(glcm, 'dissimilarity')[0, 0]
homogeneity_val = graycoprops(glcm, 'homogeneity')[0, 0]
energy_val = graycoprops(glcm, 'energy')[0, 0]
correlation_val = graycoprops(glcm, 'correlation')[0, 0]

contrast.append(contrast_val)
dissimilarity.append(dissimilarity_val)
homogeneity.append(homogeneity_val)
energy.append(energy_val)
correlation.append(correlation_val)
```

Selanjutnya mengatur jenis kopra sebagai ‘1’ dan menambahkannya ke dalam list label. Langkah ini bertujuan untuk memberikan label pada data yang sedang diproses ‘1’ disini merepresentasikan jenis kopra edible. ‘2’ merupakan jenis kopra regular, dan ‘3’ merupakan jenis kopra reject.

```
Jenis_Kopra = '1'
label.append(Jenis_Kopra)
```

Membuat Dataframe yang merupakan wadah untuk menyimpan data fitur yang telah diekstraksi sebelumnya. Setiap fitur memiliki Dataframe tersendiri yang dibentuk menggunakan library pandas. Dataframe data1 hingga data17 dibentuk untuk masing-masing fitur. Setiap Dataframe mewakili satu fitur dan kolomnya dinamai sesuai dengan fitur yang

bersangkutan. Misalnya, Dataframe data1 memiliki kolom '*Mean-R*' yang berisi nilai rata-rata komponen merah (R) dari gambar-gambar yang diproses.

Selanjutnya, semua Dataframe fitur digabungkan menjadi satu Dataframe gabungan. List listdata1 berisi semua Dataframe fitur yang telah dibentuk sebelumnya. Penggunaan axis=1 dalam fungsi concat mengindikasikan bahwa penggabungan dilakukan berdasarkan kolom. Dengan demikian, hasil dari penggabungan Dataframe-fitur ini adalah Dataframe gabung1, yang berisi semua fitur yang telah diekstraksi sebagai kolom-kolomnya.

```
data1 = pd.DataFrame(rataR, columns=['Mean-R'])
data2 = pd.DataFrame(rataG, columns=['Mean-G'])
data3 = pd.DataFrame(rataB, columns=['Mean-B'])
data4 = pd.DataFrame(rataH, columns=['Mean-H'])
data5 = pd.DataFrame(rataV, columns=['Mean-V'])
data6 = pd.DataFrame(rataS, columns=['Mean-s'])
data7 = pd.DataFrame(ratagray, columns=['Mean-Gray'])
data8 = pd.DataFrame(stand, columns=['Standar-Deviasi'])
data9 = pd.DataFrame(luas, columns=['Luas'])
data10 = pd.DataFrame(perimeter, columns=['Perimeter'])
data11 = pd.DataFrame(contrast, columns=['Contrast'])
data12 = pd.DataFrame(dissimilarity,
columns=['Dissimilarity'])
data13 = pd.DataFrame(homogeneity, columns=['Homogeneity'])
data14 = pd.DataFrame(energy, columns=['Energy'])
data15 = pd.DataFrame(correlation, columns=['Correlation'])
data16 = pd.DataFrame(label, columns=['Jenis Kopra'])

listdata1 = [data1, data2, data3, data4, data5, data6, data7,
data8, data9, data10, data11, data12, data13, data14, data15,
data16]
gabung1 = pd.concat(listdata1, axis=1)
```

Setelah semua fitur diekstraksi dan disimpan dalam Dataframe-fitur terpisah, langkah selanjutnya adalah menggabungkan semua Dataframe-fitur tersebut menjadi satu Dataframe tunggal. Pertama, list ‘listdata’ dibentuk untuk menyimpan Dataframe-fitur yang telah dibuat sebelumnya. Dataframe-fitur dalam ‘listdata’ digabungkan menjadi satu Dataframe

tunggal yang akan disimpan dalam variabel total. Penggunaan ignore_index=True memastikan indeks pada Dataframe gabungan diatur ulang secara otomatis. Selanjutnya, Dataframe ‘total’ disimpan dalam format CSV dan Excel.

```
listdata = [gabung1, gabung2, gabung3]
total = pd.concat(listdata, ignore_index=True)
total.to_csv("ekstrasi-fitur-kopra-nonTelungkup.csv",
index=False)
total.to_excel("ekstrasi-fitur-kopra-nonTelungkup.xlsx",
index=False)

print("tahap ekstraksi fitur telah selesai dibuat")
```

3) Split Dependent and Independent Value

Langkah pertama yang dilakukan untuk split dependent dan independent value adalah membaca dataset dengan mengimpor *library* yang dibutuhkan. Selanjutnya, menggunakan fungsi read_csv untuk membaca file CSV dan memuatnya ke dalam variabel def_kopra.

```
import pandas as pd
def_Kopra = pd.read_csv('ekstrasi-fitur-kopra-
nonTelungkup.csv')
```

Setelah data dibaca, data perlu dipisahkan ke dalam dua jenis yaitu *dependent* dan *independent value*. *Independent value* merupakan atribut atau fitur yang digunakan untuk melatih model, pemilihan *Independent value* ini disesuaikan dengan skenario yang ingin dilakukan. Berikut merupakan contoh untuk skenario yang menggunakan fitur warna, bentuk, dan tekstur. Semua fitur dimasukan dalam variabel x dengan memilih semua baris dan kolom dengan indeks 0 hingga 14. Sedangkan *dependent value* berisi informasi tentang kategori atau kelas dari jenis kopra yaitu, kopra edible, regular, dan reject, dimana semua kelas ini dimasukan dalam variabel y dengan memilih semua baris dan kolom dengan indeks terakhir yaitu -1. Penelitian ini menggunakan fungsi **iloc** untuk mengindeks baris dan kolom berdasarkan posisi numerik. Selanjutnya menggunakan metode **.values** untuk mengonversi data menjadi array Numpy.

```
x = def_Kopra.iloc[:,0:15].values
y = def_Kopra.iloc[:,[-1]].values
```

4) Feature Scaling

Setelah data dibagi ke dalam dua jenis yaitu *dependent* dan *independent value*, maka langkah selanjutnya melakukan normalisasi data untuk mengubah skala atau rentang nilai dari fitur-fitur yang ada pada dataset. Penelitian ini menggunakan fungsi **MinMaxScaler** dari *library Scikit-learn* untuk menormalisasi nilai dalam atribut menjadi antara 0 dan 1.

```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
x = scaler.fit_transform(x)
```

5) Train-Test Split

Selanjutnya data dibagi menjadi dua bagian yaitu data *training* dan *testing*. Pembagian ini dilakukan menggunakan fungsi yang telah tersedia dari *library scikit-learn*. Adapun pembagian antara data *training* dan data *testing* dibagi dengan perbandingan 90:10. Untuk membagi dataset menjadi dua bagian, penelitian ini menggunakan fungsi **train_test_split** dari model **model_selection** dalam *library Scikit-learn*.

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test =
train_test_split(x,y,test_size=0.1,random_state=0)
```

6) Model Klasifikasi

Pada tahap ini dilakukan klasifikasi citra berdasarkan hasil ekstraksi fitur yang didapat pada tahap sebelumnya. Algoritma *K-Nearest Neighbor* digunakan untuk melakukan klasifikasi citra dengan membandingkan data *testing* dan data *training*. Pertama, mengimport *library* yang diperlukan. Kemudian, membuat objek dengan mengatur parameter *n_neighbors* menjadi 3, yang menentukan jumlah tetangga terdekat. Selain itu, juga

mengatur parameter *metric* menjadi 'euclidean', yang menandakan penggunaan metrik jarak *Euclidean* untuk mengukur kedekatan antara data. Setelah itu, objek knn tersebut dilatih menggunakan data pelatihan x_train dan label kelas y_train. Pada penerapan metode K-Nearest Neighbor penelitian ini menggunakan *library Scikit-Learn*

```
from sklearn.neighbors import KNeighborsClassifier
knn =
KNeighborsClassifier(n_neighbors=3, metric='euclidean').fit(x_t
rain,y_train)
knn
```

7) Mengukur Performansi

Pada tahap ini dilakukan pengujian performansi untuk mengevaluasi hasil klasifikasi dari metode yang diusulkan. Untuk mengetahui performa dari algoritma K-Nearest Neighbor, penelitian ini menggunakan metrik akurasi, *confusion matrix*, dan *classification report*.

a. Metrik akurasi

Hal pertama yang dilakukan adalah meng-import fungsi dan *library* yang dibutuhkan. Kemudian, menghitung jumlah prediksi yang benar berdasarkan perbandingan antara y_test dan y_prediksi. **Normalize=False** digunakan untuk mendapatkan jumlah prediksi yang benar dalam bentuk bilangan bulat, bukan dalam bentuk presentase. Selanjutnya menghitung panjang dari y_test untuk mendapatkan jumlah seluruh prediksi yang dilakukan.

Setelah itu, menghitung nilai akurasi menggunakan fungsi **accuracy_score** dan dua variabel yang digunakan sebagai inputan yaitu y_test yang merujuk pada data target yang sebenarnya dan y_prediksi yang merupakan hasil prediksi. Dan terakhir, menampilkan pesan yang berisi jumlah prediksi yang benar, jumlah seluruh prediksi dan nilai akurasi yang dihasilkan.

```
jumlah_benar = accuracy_score(y_test, y_prediksi,
normalize=False)
jumlah_seluruh = len(y_test)
```

```

nilai_akurasi = accuracy_score(y_test, y_prediksi)

print("Jumlah prediksi yang benar:", jumlah_benar)
print("Jumlah seluruh prediksi:", jumlah_seluruh)
print("Nilai akurasi", nilai_akurasi)

```

b. *Confusion Matrix* dan *Classification Report*

Hal pertama yang dilakukan adalah meng-*import* fungsi dan *library* yang dibutuhkan. kemudian, menggunakan fungsi heatmap untuk membuat heatmap berdasarkan *confusion matrix* yang dihasilkan dari pemanggilan fungsi `confusion_matrix(y_test, y_prediksi)`. Heatmap digunakan untuk memvisualisasikan *confusion matrix* dengan warna yang berbeda-beda untuk setiap selnya. Parameter `annot=True` digunakan untuk menampilkan angka pada setiap sel heatmap. Selanjutnya, mencetak laporan klasifikasi untuk presisi, *recall*, *f1-score*, dan *support* untuk setiap kelas yang ada dalam data target yang sebenarnya (`y_test`) dan hasil prediksi (`y_prediksi`)

```

from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
import seaborn as sb
import numpy as np
sb.heatmap(confusion_matrix(y_test, y_prediksi), annot=True)
print(classification_report(y_test, y_prediksi))

```

BAB V

HASIL DAN PEMBAHASAN

5.1 Verify and Validation

Pada tahap ini, penulis melakukan verifikasi dan validasi dari tahapan-tahapan yang dilakukan pada bab 4. Verifikasi dilakukan dengan mengulas kembali setiap tahapan sebelumnya dengan tujuan memastikan setiap tahapan yang dikerjakan sesuai dan memiliki keterkaitan. Dimulai dari tahap *problem formulation*, penulis melakukan identifikasi masalah untuk dirumuskan sehingga tercipta konsep pemodelan untuk mengatasi masalah tersebut. Selanjutnya, pada tahap *conceptual model*, penulis membuat alur proses pengembangan model berdasarkan hasil rumusan masalah pada tahap sebelumnya. Kemudian, pada tahap *collection input*, penulis menjelaskan jenis citra yang akan digunakan sebagai *input-an*. Selanjutnya, pada tahap *modelling*, penulis melakukan pemodelan dengan menjelaskan perhitungan manual untuk mendapatkan fitur dari suatu citra dengan mengekstrak fitur warna, bentuk, dan tekstur dari citra tersebut. Dari hasil ekstraksi fitur ini, penulis juga menjelaskan proses klasifikasinya, dengan memanfaatkan fitur-fitur tersebut sebagai nilai untuk melakukan klasifikasi. Selanjutnya, proses *simulation*, penulis melakukan simulasi berdasarkan pemodelan manual yang telah dibuat sebelumnya ke dalam *Visual Studio Code* dengan Bahasa pemrograman python.

Berdasarkan penjelasan tersebut, dapat disimpulkan bahwa setiap tahapan yang telah dilakukan oleh penulis telah sesuai dan terdapat keterkaitan antara tahapan-tahapan tersebut. Hal ini dikarenakan setiap tahapan memiliki pengaruh terhadap tahapan-tahapan berikutnya. Oleh karena itu, dapat dipastikan bahwa tahapan-tahapan yang telah dibahas oleh penulis telah melalui proses verifikasi yang baik. Sedangkan untuk *Validation*, penulis melakukan serangkaian pengujian terhadap model yang telah dibangun. Pengujian ini melibatkan penggunaan berbagai metrik evaluasi, seperti akurasi, serta analisis yang mendalam melalui matriks

konfusi dan laporan klasifikasi. Langkah ini memastikan bahwa performa model dinilai secara menyeluruh dan memberikan kesempatan bagi penulis untuk memahami serta menganalisis hasilnya dengan lebih mendalam.

5.2 Design of Experiment

Pada tahap ini, penulis melakukan pengukuran kinerja dari hasil klasifikasi dengan 7 skenario. Dataset yang digunakan terdiri dari 1.839 data, dengan perbandingan pembagian 90:10 di mana 90% data digunakan sebagai *data training* (1.655 data) dan 10% sebagai data *testing* (184 data). Berikut merupakan skenario yang dilakukan pada penelitian ini.

Tabel 5.1 Skenario Pengujian

No	Parameter	Tujuan
1	<i>mean R, mean G, mean B, mean H, mean S, mean V, mean Gray, dan standar deviasi,</i>	Mengetahui akurasi klasifikasi jika menggunakan fitur warna
2	<i>area, dan perimeter</i>	Mengetahui akurasi klasifikasi jika menggunakan fitur bentuk
3	<i>contrast, dissimilarity, homogeneity, energy, dan correlation</i>	Mengetahui akurasi klasifikasi jika menggunakan fitur tekstur
4	<i>mean R, mean G, mean B, mean H, mean S, mean V, mean Gray, standar deviasi, area, dan perimeter</i>	Mengetahui akurasi klasifikasi jika menggunakan fitur warna dan bentuk
5	<i>mean R, mean G, mean B, mean H, mean S, mean V, mean Gray, standar deviasi, variasi, contrast, dissimilarity, homogeneity, energy, dan correlation</i>	Mengetahui akurasi klasifikasi jika menggunakan fitur warna dan tekstur

6	luas, <i>perimeter</i> , <i>contrast</i> , <i>dissimilarity</i> , <i>homogeneity</i> , <i>energy</i> , dan <i>correlation</i>	Mengetahui akurasi klasifikasi jika menggunakan fitur bentuk dan tekstur
7	<i>mean R</i> , <i>mean G</i> , <i>mean B</i> , <i>mean H</i> , <i>mean S</i> , <i>mean V</i> , <i>mean Gray</i> , standar deviasi, variasi, luas, <i>perimeter</i> , <i>contrast</i> , <i>dissimilarity</i> , <i>homogeneity</i> , <i>energy</i> , dan <i>correlation</i>	Mengetahui akurasi klasifikasi jika menggunakan fitur warna, bentuk, dan tekstur

Melalui pengukuran ini, penulis dapat mengetahui skenario mana yang memiliki tingkat akurasi tertinggi. Analisis akurasi tersebut akan dibahas lebih lanjut pada tahap *analysis output*.

5.3 Output Analysis

Dalam tahap *output analysis*, penulis melakukan analisa pengujian tingkat akurasi dari hasil klasifikasi dengan menggunakan algoritma *k-nearest neighbour* dalam 7 skenario yang berbeda serta pembahasan mengenai perbandingan hasil dengan penelitian terdahulu. Untuk mendapatkan tingkat akurasi, penulis menggunakan rumus metrik akurasi, *confusion matrix*, dan *classification report*.

5.3.1 Pengujian Tingkat Akurasi

Untuk mendapatkan tingkat akurasi dari hasil klasifikasi, penulis menggunakan rumus metrik akurasi, *confusion matrix*, dan *classification report*.

1. Skenario 1

Pada skenario 1 dilakukan percobaan klasifikasi menggunakan ekstraksi fitur warna yang terdiri *mean R*, *mean G*, *mean B*, *mean H*, *mean S*, *mean V*, *mean Gray*, dan standar deviasi dengan menggunakan perbandingan dataset 90:10 yaitu

90% untuk data *testing* dan 10% untuk data *training*. Berikut merupakan hasil pengujian tingkat akurasinya.

a. Metrik Akurasi

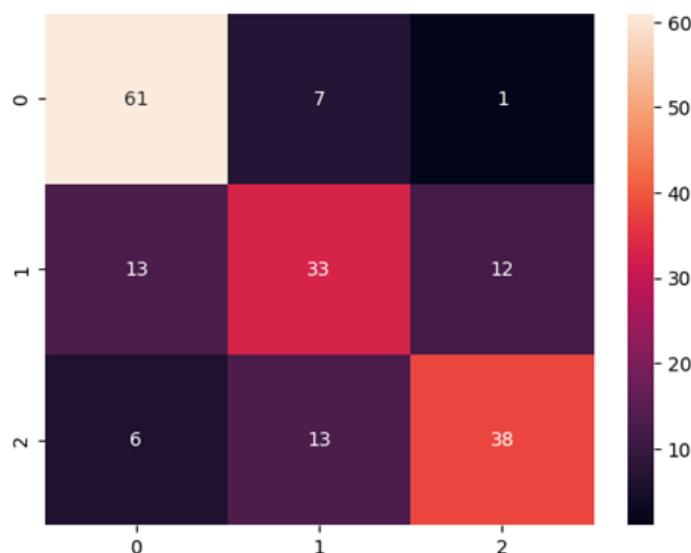
Perhitungan akurasi dengan rumus (2.14) sebagai berikut:

$$\text{Akurasi} = \frac{132}{184} \times 100\% = 72\%$$

Pada pengujian data testing menggunakan metrik akurasi didapatkan akurasi sebesar 72%.

b. *Confusion Matrix*

Berikut merupakan hasil pengujian menggunakan *confusion matrix*.



Gambar 5. 1 *Confusion Matrix* Skenario 1

Berdasarkan Gambar 5.1 *confusion matrix* 3×3 terdiri atas dua bagian yaitu hasil prediksi dan hasil aktual. Hasil aktual terdiri atas tiga kelas yaitu kelas 0, 1, dan 2. Dimana 0 merepresentasikan jenis kopra edile, 1 merupakan jenis kopra regular, dan 2 merupakan jenis kopra reject. Sedangkan hasil prediksi juga terdiri dari kelas 0, 1, dan 2.

Dalam kasus kelas jenis kopra edible, kita memiliki nilai *True Positive* (TP) sebesar 61. Nilai *True Negative*

(TN) sebesar $33+13+12+38 = 96$. Nilai *False Positive* (FP) sebesar $13+6 = 19$. Nilai *False Negative* (FN) sebesar $7+1 = 8$. Sedangkan untuk kelas kopra regular, kita memiliki *True Positive* (TP) sebesar 33. Nilai *True Negative* (TN) sebesar $61+6+1+38 = 106$. Nilai *False Positive* (FP) sebesar $7+13 = 20$. Nilai *False Negative* (FN) sebesar $13+12 = 25$. Dan untuk kelas kopra reject, kita memiliki *True Positive* (TP) sebesar 38. Nilai *True Negative* (TN) sebesar $61+13+7+33 = 114$. Nilai *False Positive* (FP) sebesar $1+12 = 13$. Nilai *False Negative* (FN) sebesar $6+13 = 19$.

c. *Classification Report*

Berikut merupakan hasil pengujian menggunakan *classification report*.

	precision	recall	f1-score	support
1	0.76	0.88	0.82	69
2	0.62	0.57	0.59	58
3	0.75	0.67	0.70	57
accuracy			0.72	184
macro avg	0.71	0.71	0.71	184
weighted avg	0.71	0.72	0.71	184

Gambar 5. 2 *Classification Report* Skenario 1

Berdasarkan Gambar 5.2 kita dapat melihat nilai presisi, *recall*, dan f1-score pada masing-masing kelas.

- 1) *Precision* dapat dihitung menggunakan rumus (2.15).

$$P(\text{Edible}) = \frac{61}{61 + (13 + 6)} = 0,76$$

$$P(\text{Reguler}) = \frac{33}{33 + (7 + 13)} = 0,62$$

$$P(\text{Reject}) = \frac{38}{38 + (1 + 12)} = 0,75$$

Nilai *precision* merupakan persentase kasus positif yang diidentifikasi dengan benar dari keseluruhan kasus yang diidentifikasi sebagai positif oleh model. Nilai *precision* yang diperoleh untuk kopra edible sebesar 76%, kopra regular sebesar 62%, dan kopra reject sebesar 75%. Nilai *precision* tertinggi diperoleh oleh kopra edible.

- 2) *Recall* dapat dihitung menggunakan rumus (2.16).

$$R(\text{Edible}) = \frac{61}{61 + (7 + 1)} = 0,88$$

$$R(\text{Reguler}) = \frac{33}{33 + (13 + 12)} = 0,57$$

$$R(\text{Reject}) = \frac{38}{38 + (6 + 13)} = 0,67$$

Nilai *recall* merupakan persentase kasus positif yang diidentifikasi dengan benar oleh model dari keseluruhan kasus positif yang sebenarnya. Nilai *recall* yang diperoleh untuk kopra edible sebesar 88%, kopra regular sebesar 57%, dan kopra reject sebesar 67%. Nilai *recall* tertinggi diperoleh oleh kopra edible.

- 3) *F1-score* dapat dihitung menggunakan rumus (2.17).

$$F1 - score(\text{Edible}) = \frac{2 \times 0,76 \times 0,88}{0,76 + 0,88} = 0,82$$

$$F1 - score(\text{Reguler}) = \frac{2 \times 0,62 \times 0,57}{0,62 + 0,57} = 0,59$$

$$F1 - score(\text{Reject}) = \frac{2 \times 0,75 \times 0,67}{0,75 + 0,67} = 0,70$$

F1-score merupakan rata-rata harmonik dari presisi dan recall untuk setiap kelas. Nilai *F1-score* yang diperoleh untuk kopra edible sebesar 82%, kopra regular sebesar 59%, dan kopra reject sebesar 70%. Nilai *F1-score* tertinggi diperoleh oleh kopra edible.

2. Skenario 2

Pada skenario 2 dilakukan percobaan klasifikasi menggunakan ekstraksi fitur bentuk yang terdiri *area* dan *perimeter* dengan menggunakan perbandingan dataset 90:10 yaitu 90% untuk data *testing* dan 10% untuk data *training*. Berikut merupakan hasil pengujian tingkat akurasinya.

a. Metrik Akurasi

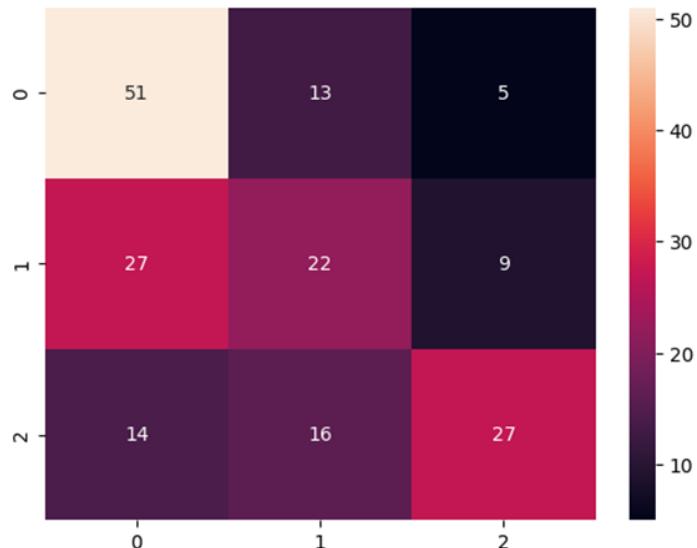
Perhitungan akurasi dengan rumus (2.14) sebagai berikut:

$$Akurasi = \frac{100}{184} \times 100\% = 54\%$$

Pada pengujian data testing menggunakan metrik akurasi didapatkan akurasi sebesar 54%.

b. *Confusion Matrix*

Berikut merupakan hasil pengujian menggunakan *confusion matrix*.



Gambar 5. 3 *Confusion Matrix* Skenario 2

Berdasarkan Gambar 5.3 *confusion matrix* 3×3 terdiri atas dua bagian yaitu hasil prediksi dan hasil aktual. Hasil aktual terdiri atas tiga kelas yaitu kelas 0, 1, dan 2. Dimana 0 merepresentasikan jenis kopra edile, 1 merupakan

jenis kopra regular, dan 2 merupakan jenis kopra reject. Sedangkan hasil prediksi juga terdiri dari kelas 0, 1, dan 2.

Dalam kasus kelas jenis kopra edible, kita memiliki nilai *True Positive* (TP) sebesar 51. Nilai *True Negative* (TN) sebesar $22+16+9+27 = 74$. Nilai *False Positive* (FP) sebesar $27+14 = 41$. Nilai *False Negative* (FN) sebesar $13+5 = 18$. Sedangkan untuk kelas kopra regular, kita memiliki *True Positive* (TP) sebesar 22. Nilai *True Negative* (TN) sebesar $51+14+5+27 = 87$. Nilai *False Positive* (FP) sebesar $13+16 = 29$. Nilai *False Negative* (FN) sebesar $27+9 = 36$. Dan untuk kelas kopra reject, kita memiliki *True Positive* (TP) sebesar 27. Nilai *True Negative* (TN) sebesar $51+27+13+22 = 113$. Nilai *False Positive* (FP) sebesar $5+9 = 14$. Nilai *False Negative* (FN) sebesar $14+16 = 30$.

c. *Classification Report*

Berikut merupakan hasil pengujian menggunakan *classification report*.

	precision	recall	f1-score	support
1	0.55	0.74	0.63	69
2	0.43	0.38	0.40	58
3	0.66	0.47	0.55	57
accuracy			0.54	184
macro avg	0.55	0.53	0.53	184
weighted avg	0.55	0.54	0.54	184

Gambar 5. 4 *Classification Report* Skenario 2

Berdasarkan Gambar 5.4 kita dapat melihat nilai presisi, *recall*, dan f1-score pada masing-masing kelas.

1) *Precision* dapat dihitung menggunakan rumus (2.15).

$$P (\text{Edible}) = \frac{51}{51 + (27 + 14)} = 0,55$$

$$P (\text{Reguler}) = \frac{22}{22 + (13 + 16)} = 0,43$$

$$P (\text{Reject}) = \frac{27}{27 + (5 + 9)} = 0,66$$

Nilai *precision* merupakan persentase kasus positif yang diidentifikasi dengan benar dari keseluruhan kasus yang diidentifikasi sebagai positif oleh model. Nilai *precision* yang diperoleh untuk kopra edible sebesar 55%, kopra regular sebesar 43%, dan kopra reject sebesar 66%. Nilai *precision* tertinggi diperoleh oleh kopra reject.

- 2) *Recall* dapat dihitung menggunakan rumus (2.16).

$$R (\text{Edible}) = \frac{51}{51 + (13 + 5)} = 0,74$$

$$R (\text{Reguler}) = \frac{22}{22 + (27 + 9)} = 0,38$$

$$R (\text{Reject}) = \frac{27}{27 + (14 + 16)} = 0,47$$

Nilai *recall* merupakan persentase kasus positif yang diidentifikasi dengan benar oleh model dari keseluruhan kasus positif yang sebenarnya. Nilai *recall* yang diperoleh untuk kopra edible sebesar 74%, kopra regular sebesar 38%, dan kopra reject sebesar 47%. Nilai *recall* tertinggi diperoleh oleh kopra edible.

- 3) *F1-score* dapat dihitung menggunakan rumus (2.17).

$$F1 - score (\text{Edible}) = \frac{2 \times 0,55 \times 0,74}{0,55 + 0,74} = 0,63$$

$$F1 - score (\text{Reguler}) = \frac{2 \times 0,43 \times 0,38}{0,43 + 0,38} = 0,40$$

$$F1 - score (\text{Reject}) = \frac{2 \times 0,66 \times 0,47}{0,66 + 0,47} = 0,55$$

F1-score merupakan rata-rata harmonik dari presisi dan recall untuk setiap kelas. Nilai *F1-score* yang diperoleh

untuk kopra edible sebesar 63%, kopra regular sebesar 40%, dan kopra reject sebesar 55%. Nilai *F1-score* tertinggi diperoleh oleh kopra edible.

3. Skenario 3

Pada skenario 2 dilakukan percobaan klasifikasi menggunakan ekstraksi fitur tekstur yang terdiri *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation* dengan menggunakan perbandingan dataset 90:10 yaitu 90% untuk data *testing* dan 10% untuk data *training*. Berikut merupakan hasil pengujian tingkat akurasinya.

a. Metrik Akurasi

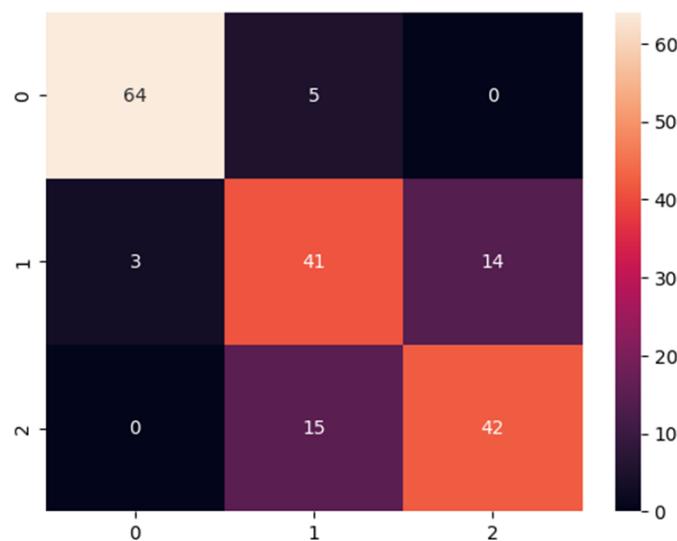
Perhitungan akurasi dengan rumus (2.14) sebagai berikut:

$$\text{Akurasi} = \frac{147}{184} \times 100\% = 80\%$$

Pada pengujian data testing menggunakan metrik akurasi didapatkan akurasi sebesar 80%.

b. *Confusion Matrix*

Berikut merupakan hasil pengujian menggunakan *confusion matrix*.



Gambar 5. 5 *Confusion Matrix* Skenario 3

Berdasarkan Gambar 5.5 *confusion matrix* 3×3 terdiri atas dua bagian yaitu hasil prediksi dan hasil aktual. Hasil aktual terdiri atas tiga kelas yaitu kelas 0, 1, dan 2. Dimana 0 merepresentasikan jenis kopra edile, 1 merupakan jenis kopra regular, dan 2 merupakan jenis kopra reject. Sedangkan hasil prediksi juga terdiri dari kelas 0, 1, dan 2.

Dalam kasus kelas jenis kopra edible, kita memiliki nilai *True Positive* (TP) sebesar 64. Nilai *True Negative* (TN) sebesar $41+15+14+42 = 112$. Nilai *False Positive* (FP) sebesar $3+0 = 3$. Nilai *False Negative* (FN) sebesar $5+0 = 5$. Sedangkan untuk kelas kopra regular, kita memiliki *True Positive* (TP) sebesar 41. Nilai *True Negative* (TN) sebesar $64+0+0+42 = 106$. Nilai *False Positive* (FP) sebesar $5+15 = 20$. Nilai *False Negative* (FN) sebesar $3+14 = 17$. Dan untuk kelas kopra reject, kita memiliki *True Positive* (TP) sebesar 42. Nilai *True Negative* (TN) sebesar $64+3+5+41 = 113$. Nilai *False Positive* (FP) sebesar $14+0 = 4$. Nilai *False Negative* (FN) sebesar $0+15 = 15$.

c. *Classification Report*

Berikut merupakan hasil pengujian menggunakan *classification report*.

	precision	recall	f1-score	support
1	0.96	0.93	0.94	69
2	0.67	0.71	0.69	58
3	0.75	0.74	0.74	57
accuracy			0.80	184
macro avg	0.79	0.79	0.79	184
weighted avg	0.80	0.80	0.80	184

Gambar 5. 6 *Classification Report* Skenario 3

Berdasarkan Gambar 5.6 kita dapat melihat nilai presisi, *recall*, dan f1-score pada masing-masing kelas.

- 1) *Precision* dapat dihitung menggunakan rumus (2.15).

$$P(\text{Edible}) = \frac{64}{64 + (3 + 0)} = 0,96$$

$$P(\text{Reguler}) = \frac{41}{41 + (5 + 15)} = 0,67$$

$$P(\text{Reject}) = \frac{42}{42 + (0 + 14)} = 0,75$$

Nilai *precision* merupakan persentase kasus positif yang diidentifikasi dengan benar dari keseluruhan kasus yang diidentifikasi sebagai positif oleh model. Nilai *precision* yang diperoleh untuk kopra edible sebesar 96%, kopra regular sebesar 67%, dan kopra reject sebesar 75%. Nilai *precision* tertinggi diperoleh oleh kopra edible.

- 2) *Recall* dapat dihitung menggunakan rumus (2.16).

$$R(\text{Edible}) = \frac{64}{64 + (5 + 0)} = 0,93$$

$$R(\text{Reguler}) = \frac{41}{41 + (3 + 14)} = 0,71$$

$$R(\text{Reject}) = \frac{42}{42 + (0 + 15)} = 0,74$$

Nilai *recall* merupakan persentase kasus positif yang diidentifikasi dengan benar oleh model dari keseluruhan kasus positif yang sebenarnya. Nilai *recall* yang diperoleh untuk kopra edible sebesar 93%, kopra regular sebesar 71%, dan kopra reject sebesar 74%. Nilai *recall* tertinggi diperoleh oleh kopra edible.

- 3) *F1-score* dapat dihitung menggunakan rumus (2.17).

$$F1 - score(\text{Edible}) = \frac{2 \times 0,96 \times 0,93}{0,96 + 0,93} = 0,94$$

$$F1 - score(\text{Reguler}) = \frac{2 \times 0,67 \times 0,71}{0,67 + 0,71} = 0,69$$

$$F1 - score(\text{Reject}) = \frac{2 \times 0,75 \times 0,74}{0,75 + 0,74} = 0,74$$

F1-score merupakan rata-rata harmonik dari presisi dan recall untuk setiap kelas. Nilai *F1-score* yang diperoleh untuk kopra edible sebesar 94%, kopra regular sebesar 69%, dan kopra reject sebesar 74%. Nilai *F1-score* tertinggi diperoleh oleh kopra edible dan reject.

4. Skenario 4

Pada skenario 4 dilakukan percobaan klasifikasi menggunakan ekstraksi fitur warna dan bentuk yang terdiri *mean R*, *mean G*, *mean B*, *mean H*, *mean S*, *mean V*, *mean Gray*, standar deviasi, *area* dan perimeter dengan menggunakan perbandingan dataset 90:10 yaitu 90% untuk data *testing* dan 10% untuk data *training*. Berikut merupakan hasil pengujian tingkat akurasinya.

a. Metrik Akurasi

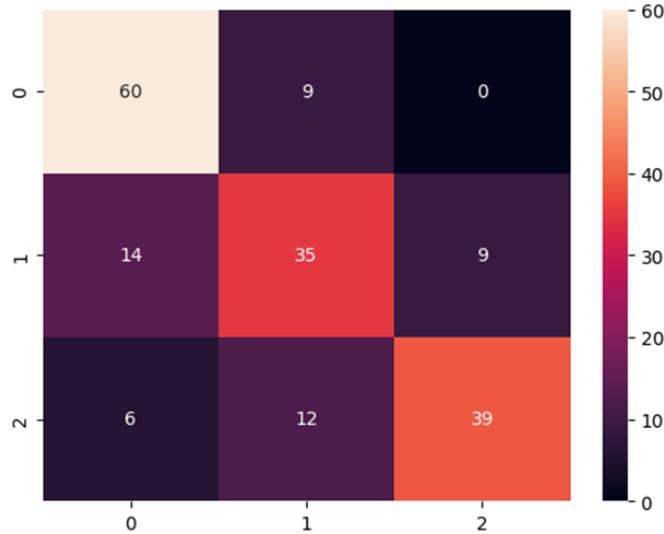
Perhitungan akurasi dengan rumus (2.14) sebagai berikut:

$$\text{Akurasi} = \frac{134}{184} \times 100\% = 73\%$$

Pada pengujian data testing menggunakan metrik akurasi didapatkan akurasi sebesar 73%.

b. *Confusion Matrix*

Berikut merupakan hasil pengujian menggunakan *confusion matrix*.



Gambar 5. 7 *Confusion Matrix Skenario 4*

Berdasarkan Gambar 5.7 *confusion matrix* 3×3 terdiri atas dua bagian yaitu hasil prediksi dan hasil aktual. Hasil aktual terdiri atas tiga kelas yaitu kelas 0, 1, dan 2. Dimana 0 merepresentasikan jenis kopra edile, 1 merupakan jenis kopra regular, dan 2 merupakan jenis kopra reject. Sedangkan hasil prediksi juga terdiri dari kelas 0, 1, dan 2.

Dalam kasus kelas jenis kopra edible, kita memiliki nilai *True Positive* (TP) sebesar 60. Nilai *True Negative* (TN) sebesar $35+12+9+39 = 95$. Nilai *False Positive* (FP) sebesar $14+6 = 20$. Nilai *False Negative* (FN) sebesar $9+0 = 9$. Sedangkan untuk kelas kopra regular, kita memiliki *True Positive* (TP) sebesar 35. Nilai *True Negative* (TN) sebesar $60+6+0+39 = 105$. Nilai *False Positive* (FP) sebesar $9+12 = 21$. Nilai *False Negative* (FN) sebesar $14+9 = 23$. Dan untuk kelas kopra reject, kita memiliki *True Positive* (TP) sebesar 39. Nilai *True Negative* (TN) sebesar $60+14+9+35 = 118$. Nilai *False Positive* (FP) sebesar $0+9 = 9$. Nilai *False Negative* (FN) $6+12 = 18$.

c. *Classification Report*

Berikut merupakan hasil pengujian menggunakan *classification report*.

	precision	recall	f1-score	support
1	0.75	0.87	0.81	69
2	0.62	0.60	0.61	58
3	0.81	0.68	0.74	57
accuracy			0.73	184
macro avg	0.73	0.72	0.72	184
weighted avg	0.73	0.73	0.73	184

Gambar 5. 8 *Classification Report* Skenario 4

Berdasarkan Gambar 5.8 kita dapat melihat nilai presisi, *recall*, dan f1-score pada masing-masing kelas.

- P) *Precision* dapat dihitung menggunakan rumus (2.15).

$$P(\text{Edible}) = \frac{60}{60 + (14 + 6)} = 0,75$$

$$P(\text{Reguler}) = \frac{35}{35 + (9 + 12)} = 0,62$$

$$P(\text{Reject}) = \frac{39}{39 + (0 + 9)} = 0,81$$

Nilai *precision* merupakan persentase kasus positif yang diidentifikasi dengan benar dari keseluruhan kasus yang diidentifikasi sebagai positif oleh model. Nilai *precision* yang diperoleh untuk kopra edible sebesar 75%, kopra regular sebesar 62%, dan kopra reject sebesar 81%. Nilai *precision* tertinggi diperoleh oleh kopra reject.

- R) *Recall* dapat dihitung menggunakan rumus (2.16).

$$R(\text{Edible}) = \frac{60}{60 + (9 + 0)} = 0,87$$

$$R(\text{Reguler}) = \frac{35}{35 + (14 + 9)} = 0,60$$

$$R(\text{Reject}) = \frac{39}{39 + (6 + 12)} = 0,68$$

Nilai *recall* merupakan persentase kasus positif yang diidentifikasi dengan benar oleh model dari keseluruhan kasus positif yang sebenarnya. Nilai *recall* yang diperoleh untuk kopra edible sebesar 87%, kopra regular sebesar 60%, dan kopra reject sebesar 68%. Nilai *recall* tertinggi diperoleh oleh kopra edible.

- 3) *F1-score* dapat dihitung menggunakan rumus (2.17).

$$F1 - score (Edible) = \frac{2 \times 0,75 \times 0,87}{0,75 + 0,87} = 0,81$$

$$F1 - score (Reguler) = \frac{2 \times 0,62 \times 0,60}{0,62 + 0,60} = 0,61$$

$$F1 - score (Reject) = \frac{2 \times 0,81 \times 0,68}{0,81 + 0,68} = 0,74$$

F1-score merupakan rata-rata harmonik dari presisi dan recall untuk setiap kelas. Nilai *F1-score* yang diperoleh untuk kopra edible sebesar 81%, kopra regular sebesar 61%, dan kopra reject sebesar 74%. Nilai *F1-score* tertinggi diperoleh oleh kopra edible.

5. Skenario 5

Pada skenario 5 dilakukan percobaan klasifikasi menggunakan ekstraksi fitur warna dan tekstur yang terdiri *mean R*, *mean G*, *mean B*, *mean H*, *mean S*, *mean V*, *mean Gray*, standar deviasi, *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation* dengan menggunakan perbandingan dataset 90:10 yaitu 90% untuk data *testing* dan 10% untuk data *training*.

Berikut merupakan hasil pengujian tingkat akurasinya.

a. Metrik Akurasi

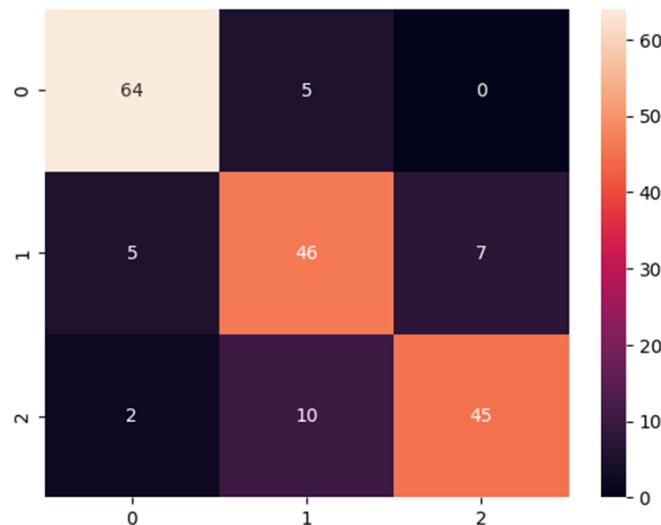
Perhitungan akurasi dengan rumus (2.14) sebagai berikut:

$$Akurasi = \frac{155}{184} \times 100\% = 84\%$$

Pada pengujian data testing menggunakan metrik akurasi didapatkan akurasi sebesar 84%.

b. *Confusion Matrix*

Berikut merupakan hasil pengujian menggunakan *confusion matrix*.



Gambar 5. 9 *Confusion Matrix* Skenario 5

Berdasarkan Gambar 5.9 *confusion matrix* 3×3 terdiri atas dua bagian yaitu hasil prediksi dan hasil aktual. Hasil aktual terdiri atas tiga kelas yaitu kelas 0, 1, dan 2. Dimana 0 merepresentasikan jenis kopra edile, 1 merupakan jenis kopra regular, dan 2 merupakan jenis kopra reject. Sedangkan hasil prediksi juga terdiri dari kelas 0, 1, dan 2.

Dalam kasus kelas jenis kopra edible, kita memiliki nilai *True Positive* (TP) sebesar 64. Nilai *True Negative* (TN) sebesar $46+10+7+45 = 108$. Nilai *False Positive* (FP) sebesar $5+2 = 7$. Nilai *False Negative* (FN) sebesar $5+0 = 5$. Sedangkan untuk kelas kopra regular, kita memiliki *True Positive* (TP) sebesar 46. Nilai *True Negative* (TN) sebesar $64+2+0+45 = 111$. Nilai *False Positive* (FP) sebesar $5+10 = 15$. Nilai *False Negative* (FN) sebesar $5+7 = 12$. Dan untuk kelas kopra reject, kita memiliki *True Positive* (TP) sebesar

45. Nilai *True Negative* (TN) sebesar $64+5+5+46 = 111$.

Nilai *False Positive* (FP) sebesar $0+7 = 7$. Nilai *False Negative* (FN) sebesar $2+10 = 12$.

c. *Classification Report*

Berikut merupakan hasil pengujian menggunakan *classification report*.

	precision	recall	f1-score	support
1	0.90	0.93	0.91	69
2	0.75	0.79	0.77	58
3	0.87	0.79	0.83	57
accuracy			0.84	184
macro avg	0.84	0.84	0.84	184
weighted avg	0.84	0.84	0.84	184

Gambar 5. 10 *Classification Report* Skenario 5

Berdasarkan Gambar 5.10 kita dapat melihat nilai presisi, *recall*, dan f1-score pada masing-masing kelas.

1) *Precision* dapat dihitung menggunakan rumus (2.15).

$$P(\text{Edible}) = \frac{64}{64 + (5 + 2)} = 0,90$$

$$P(\text{Reguler}) = \frac{46}{46 + (5 + 10)} = 0,75$$

$$P(\text{Reject}) = \frac{45}{45 + (0 + 7)} = 0,87$$

Nilai *precision* merupakan persentase kasus positif yang diidentifikasi dengan benar dari keseluruhan kasus yang diidentifikasi sebagai positif oleh model. Nilai *precision* yang diperoleh untuk kopra edible sebesar 88%, kopra regular sebesar 77%, dan kopra reject sebesar 85%. Nilai *precision* tertinggi diperoleh oleh kopra edible.

2) *Recall* dapat dihitung menggunakan rumus (2.16).

$$R (\text{Edible}) = \frac{64}{64 + (5 + 0)} = 0,93$$

$$R (\text{Reguler}) = \frac{46}{46 + (5 + 7)} = 0,79$$

$$R (\text{Reject}) = \frac{45}{45 + (2 + 10)} = 0,79$$

Nilai *recall* merupakan persentase kasus positif yang diidentifikasi dengan benar oleh model dari keseluruhan kasus positif yang sebenarnya. Nilai *recall* yang diperoleh untuk kopra edible sebesar 93%, kopra regular sebesar 79%, dan kopra reject sebesar 79%. Nilai *recall* tertinggi diperoleh oleh kopra edible.

- 3) *F1-score* dapat dihitung menggunakan rumus (2.17).

$$F1 - score (\text{Edible}) = \frac{2 \times 0,90 \times 0,93}{0,90 + 0,93} = 0,91$$

$$F1 - score (\text{Reguler}) = \frac{2 \times 0,75 \times 0,79}{0,75 + 0,79} = 0,77$$

$$F1 - score (\text{Reject}) = \frac{2 \times 0,87 \times 0,79}{0,87 + 0,79} = 0,83$$

F1-score merupakan rata-rata harmonik dari presisi dan recall untuk setiap kelas. Nilai *F1-score* yang diperoleh untuk kopra edible sebesar 91%, kopra regular sebesar 77%, dan kopra reject sebesar 83%. Nilai *F1-score* tertinggi diperoleh oleh kopra edible.

6. Skenario 6

Pada skenario 6 dilakukan percobaan klasifikasi menggunakan ekstraksi fitur bentuk dan tekstur yang terdiri *area*, *perimeter*, *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation* dengan menggunakan perbandingan dataset 90:10 yaitu 90% untuk data *testing* dan 10% untuk data *training*. Berikut merupakan hasil pengujian tingkat akurasinya.

- a. Metrik Akurasi

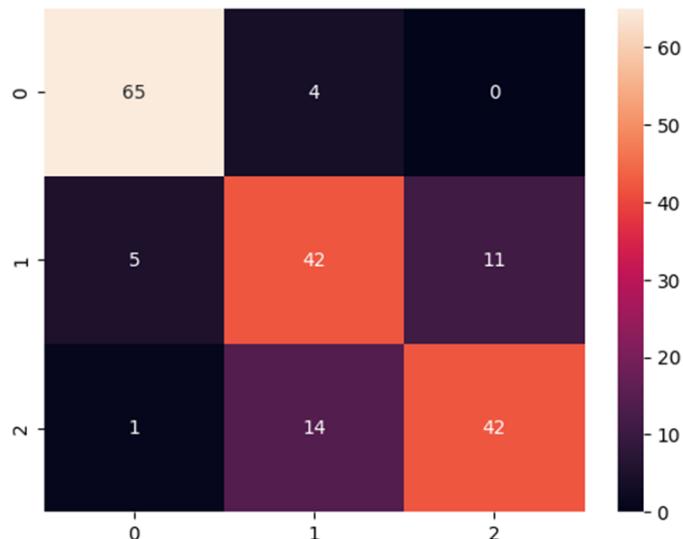
Perhitungan akurasi dengan rumus (2.14) sebagai berikut:

$$Akurasi = \frac{149}{184} \times 100\% = 81\%$$

Pada pengujian data testing menggunakan metrik akurasi didapatkan akurasi sebesar 81%.

b. *Confusion Matrix*

Berikut merupakan hasil pengujian menggunakan *confusion matrix*.



Gambar 5. 11 *Confusion Matrix* Skenario 6

Berdasarkan Gambar 5.11 *confusion matrix* 3×3 terdiri atas dua bagian yaitu hasil prediksi dan hasil aktual. Hasil aktual terdiri atas tiga kelas yaitu kelas 0, 1, dan 2. Dimana 0 merepresentasikan jenis kopra edile, 1 merupakan jenis kopra regular, dan 2 merupakan jenis kopra reject. Sedangkan hasil prediksi juga terdiri dari kelas 0, 1, dan 2.

Dalam kasus kelas jenis kopra edile, kita memiliki nilai *True Positive* (TP) sebesar 65. Nilai *True Negative* (TN) sebesar $45+14+11+42= 109$. Nilai *False Positive* (FP) sebesar $5+1 = 6$. Nilai *False Negative* (FN) sebesar $4+0 = 4$.

Sedangkan untuk kelas kopra regular, kita memiliki *True Positive* (TP) sebesar 42. Nilai *True Negative* (TN) sebesar $65+1+0+42 = 108$. Nilai *False Positive* (FP) sebesar $4+14 = 18$. Nilai *False Negative* (FN) sebesar $5+11 = 16$. Dan untuk kelas kopra reject, kita memiliki *True Positive* (TP) sebesar 42. Nilai *True Negative* (TN) sebesar $65+1+4+42 = 112$. Nilai *False Positive* (FP) sebesar $0+11 = 11$. Nilai *False Negative* (FN) sebesar $1+14 = 15$.

c. *Classification Report*

Berikut merupakan hasil pengujian menggunakan *classification report*.

	precision	recall	f1-score	support
1	0.92	0.94	0.93	69
2	0.70	0.72	0.71	58
3	0.79	0.74	0.76	57
accuracy			0.81	184
macro avg	0.80	0.80	0.80	184
weighted avg	0.81	0.81	0.81	184

Gambar 5. 12 *Classification Report* Skenario 6

Berdasarkan Gambar 5.12 kita dapat melihat nilai presisi, *recall*, dan f1-score pada masing-masing kelas.

- 1) *Precision* dapat dihitung menggunakan rumus (2.15).

$$P(\text{Edible}) = \frac{65}{65 + (5 + 1)} = 0,92$$

$$P(\text{Reguler}) = \frac{42}{42 + (4 + 14)} = 0,70$$

$$P(\text{Reject}) = \frac{42}{42 + (0 + 11)} = 0,79$$

Nilai *precision* merupakan persentase kasus positif yang diidentifikasi dengan benar dari keseluruhan kasus yang diidentifikasi sebagai positif oleh model. Nilai *precision*

yang diperoleh untuk kopra edible sebesar 92%, kopra regular sebesar 70%, dan kopra reject sebesar 79%. Nilai *precision* tertinggi diperoleh oleh kopra edible.

- 2) *Recall* dapat dihitung menggunakan rumus (2.16).

$$R (\text{Edible}) = \frac{65}{65 + (4 + 0)} = 0,94$$

$$R (\text{Reguler}) = \frac{42}{42 + (5 + 11)} = 0,72$$

$$R (\text{Reject}) = \frac{42}{42 + (1 + 14)} = 0,74$$

Nilai *recall* merupakan persentase kasus positif yang diidentifikasi dengan benar oleh model dari keseluruhan kasus positif yang sebenarnya. Nilai *recall* yang diperoleh untuk kopra edible sebesar 94%, kopra regular sebesar 72%, dan kopra reject sebesar 74%. Nilai *recall* tertinggi diperoleh oleh kopra edible.

- 3) *F1-score* dapat dihitung menggunakan rumus (2.17).

$$F1 - score (\text{Edible}) = \frac{2 \times 0,92 \times 0,94}{0,92 + 0,94} = 0,93$$

$$F1 - score (\text{Reguler}) = \frac{2 \times 0,70 \times 0,72}{0,70 + 0,72} = 0,71$$

$$F1 - score (\text{Reject}) = \frac{2 \times 0,79 \times 0,74}{0,79 + 0,74} = 0,76$$

F1-score merupakan rata-rata harmonik dari presisi dan recall untuk setiap kelas. Nilai *F1-score* yang diperoleh untuk kopra edible sebesar 93%, kopra regular sebesar 71%, dan kopra reject sebesar 76%. Nilai *F1-score* tertinggi diperoleh oleh kopra edible.

7. Skenario 7

Pada skenario 7 dilakukan percobaan klasifikasi menggunakan ekstraksi fitur warna, bentuk dan tekstur yang terdiri dari *mean R*, *mean G*, *mean B*, *mean H*, *mean S*, *mean V*,

mean Gray, standar deviasi, variasi, luas, perimeter, contrast, dissimilarity, homogeneity, energy, dan correlation, area, perimeter, contrast, dissimilarity, homogeneity, energy, dan correlation dengan menggunakan perbandingan dataset 90:10 yaitu 90% untuk data *testing* dan 10% untuk data *training*. Berikut merupakan hasil pengujian tingkat akurasinya.

a. Metrik Akurasi

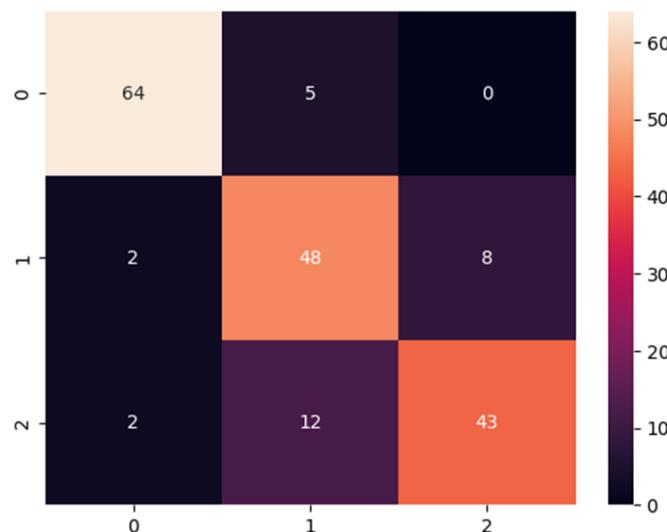
Perhitungan akurasi dengan rumus (2.14) sebagai berikut:

$$\text{Akurasi} = \frac{155}{184} \times 100\% = 84\%$$

Pada pengujian data testing menggunakan metrik akurasi didapatkan akurasi sebesar 84%.

b. *Confusion Matrix*

Berikut merupakan hasil pengujian menggunakan *confusion matrix*.



Gambar 5. 13 *Confusion Matrix* Skenario 7

Berdasarkan Gambar 5.13 *confusion matrix* 3×3 terdiri atas dua bagian yaitu hasil prediksi dan hasil aktual. Hasil aktual terdiri atas tiga kelas yaitu kelas 0,

1, dan 2. Dimana 0 merepresentasikan jenis kopra edile, 1 merupakan jenis kopra regular, dan 2 merupakan jenis kopra reject. Sedangkan hasil prediksi juga terdiri dari kelas 0, 1, dan 2.

Dalam kasus kelas jenis kopra edible, kita memiliki nilai *True Positive* (TP) sebesar 64. Nilai *True Negative* (TN) sebesar $48+12+8+43 = 111$. Nilai *False Positive* (FP) sebesar $2+2 = 4$. Nilai *False Negative* (FN) sebesar $5+0 = 5$. Sedangkan untuk kelas kopra regular, kita memiliki *True Positive* (TP) sebesar 48. Nilai *True Negative* (TN) sebesar $64+2+0+43 = 109$. Nilai *False Positive* (FP) sebesar $5+2 = 7$. Nilai *False Negative* (FN) sebesar $2+8 = 10$. Dan untuk kelas kopra reject, kita memiliki *True Positive* (TP) sebesar 43. Nilai *True Negative* (TN) sebesar $64+2+5+48 = 119$. Nilai *False Positive* (FP) sebesar $0+8 = 8$. Nilai *False Negative* (FN) sebesar $2+12 = 14$.

c. *Classification Report*

Berikut merupakan hasil pengujian menggunakan *classification report*.

	precision	recall	f1-score	support
1	0.94	0.93	0.93	69
2	0.74	0.83	0.78	58
3	0.84	0.75	0.80	57
accuracy			0.84	184
macro avg	0.84	0.84	0.84	184
weighted avg	0.85	0.84	0.84	184

Gambar 5. 14 *Classification Report* Skenario 7

Berdasarkan Gambar 5.14 kita dapat melihat nilai presisi, *recall*, dan f1-score pada masing-masing kelas.

- 1) *Precision* dapat dihitung menggunakan rumus (2.15).

$$P(\text{Edible}) = \frac{64}{64 + (2 + 2)} = 0,94$$

$$P(\text{Reguler}) = \frac{48}{48 + (5 + 12)} = 0,74$$

$$P(\text{Reject}) = \frac{43}{43 + (0 + 8)} = 0,84$$

Nilai *precision* merupakan persentase kasus positif yang diidentifikasi dengan benar dari keseluruhan kasus yang diidentifikasi sebagai positif oleh model. Nilai *precision* yang diperoleh untuk kopra edible sebesar 94%, kopra regular sebesar 74%, dan kopra reject sebesar 84%. Nilai *precision* tertinggi diperoleh oleh kopra edible.

- 2) *Recall* dapat dihitung menggunakan rumus (2.16).

$$R(\text{Edible}) = \frac{64}{64 + (5 + 0)} = 0,93$$

$$R(\text{Reguler}) = \frac{48}{48 + (2 + 8)} = 0,83$$

$$R(\text{Reject}) = \frac{43}{43 + (2 + 12)} = 0,75$$

Nilai *recall* merupakan persentase kasus positif yang diidentifikasi dengan benar oleh model dari keseluruhan kasus positif yang sebenarnya. Nilai *recall* yang diperoleh untuk kopra edible sebesar 93%, kopra regular sebesar 83%, dan kopra reject sebesar 75%. Nilai *recall* tertinggi diperoleh oleh kopra edible.

- 3) *F1-score* dapat dihitung menggunakan rumus (2.17).

$$F1 - score(\text{Edible}) = \frac{2 \times 0,94 \times 0,93}{0,94 + 0,93} = 0,93$$

$$F1 - score(\text{Reguler}) = \frac{2 \times 0,74 \times 0,83}{0,74 + 0,83} = 0,78$$

$$F1-score (Reject) = \frac{2 \times 0,84 \times 0,75}{0,84 + 0,75} = 0,80$$

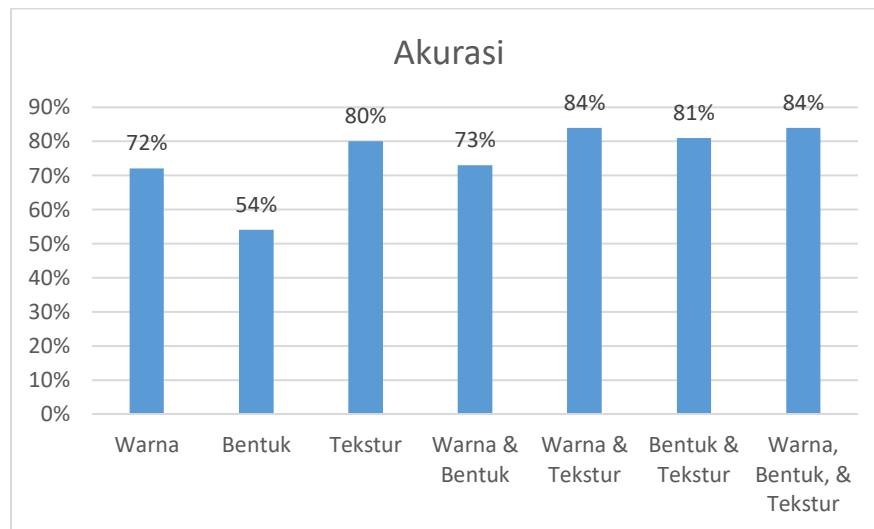
F1-score merupakan rata-rata harmonik dari presisi dan recall untuk setiap kelas. Nilai *F1-score* yang diperoleh untuk kopra edible sebesar 93%, kopra regular sebesar 78%, dan kopra reject sebesar 80%. Nilai *F1-score* tertinggi diperoleh oleh kopra edible.

Setelah melakukan simulasi dengan 12 skenario, didapatkan hasil dari setiap skenario. Hasil eksperimen simulasi dapat dilihat pada tabel dibawah ini.

Tabel 5. 2 Hasil Akurasi Pengujian

No	Skenario	Akurasi	Precision	Recall	<i>F1-score</i>
1	Skenario 1	72%	71%	71%	71%
2	Skenario 2	54%	55%	53%	53%
3	Skenario 3	80%	79%	79%	79%
4	Skenario 4	73%	73%	72%	72%
5	Skenario 5	84%	84%	84%	84%
6	Skenario 6	81%	80%	80%	80%
7	Skenario 7	84%	84%	84%	84%

Hasil evaluasi model juga dapat dilihat dalam bentuk grafik. Berikut merupakan hasil evaluasi akurasi terhadap model klasifikasi:

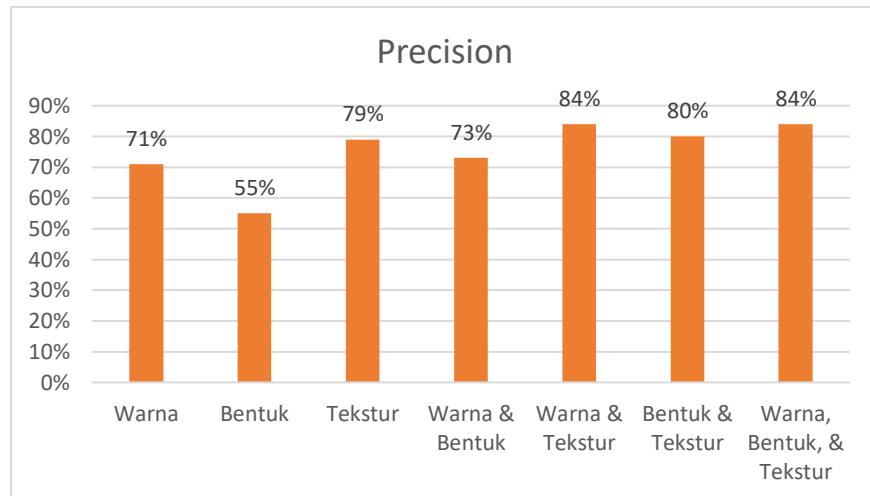


Gambar 5. 15 Hasil Evaluasi Model berdasarkan Akurasi

Grafik diatas menyajikan hasil akurasi dari sejumlah skenario yang diuji dalam penelitian. Skenario-skenario tersebut mencakup variasi dalam penggunaan ekstraksi fitur tertentu, yaitu warna, bentuk, dan tekstur, serta beberapa kombinasi di antaranya. Hasil menunjukkan bahwa skenario yang menggabungkan ekstraksi fitur warna dan tekstur, serta skenario yang melibatkan semua ekstraksi fitur (warna, bentuk, dan tekstur), memiliki akurasi tertinggi, masing-masing mencapai 84%. Hal ini menunjukkan bahwa kombinasi ekstraksi fitur tersebut secara bersama-sama memberikan kontribusi yang signifikan dalam meningkatkan akurasi hasil. Di sisi lain, skenario yang hanya mengandalkan ekstraksi fitur bentuk memiliki akurasi yang lebih rendah, yaitu 54%, sedangkan skenario dengan ekstraksi fitur tekstur memiliki akurasi sebesar 80%.

Selanjutnya penulis juga melakukan pengujian terhadap *precision*. Dengan demikian, analisis yang disajikan tidak hanya berfokus pada seberapa tepat model dalam mengklasifikasikan secara keseluruhan, namun juga pada kemampuannya dalam memberikan hasil yang benar dari prediksi positif secara spesifik. Berdasarkan evaluasi model yang telah dilakukan, berikut

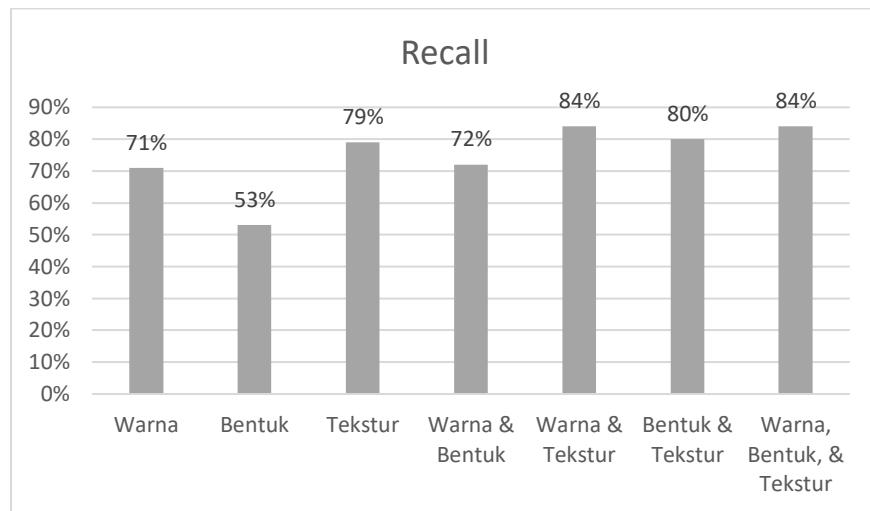
merupakan evaluasi *precision* terhadap model klasifikasi yang telah dibuat.



Gambar 5. 16 Hasil Evaluasi Model berdasarkan *Precision*

Dari grafik di atas, dapat dilihat bahwa skenario penggunaan fitur warna dan tekstur secara terpisah maupun kombinasi memberikan hasil *precision* yang cukup baik. Skenario yang melibatkan fitur tekstur saja menunjukkan *precision* tertinggi, mencapai 79%. Kombinasi fitur warna dan tekstur, serta penggunaan semua tiga fitur (warna, bentuk, dan tekstur) menghasilkan *precision* yang lebih tinggi, dengan masing-masing mencapai 84%. Di sisi lain, skenario yang hanya mempertimbangkan fitur bentuk menunjukkan *precision* yang lebih rendah, yakni 53%.

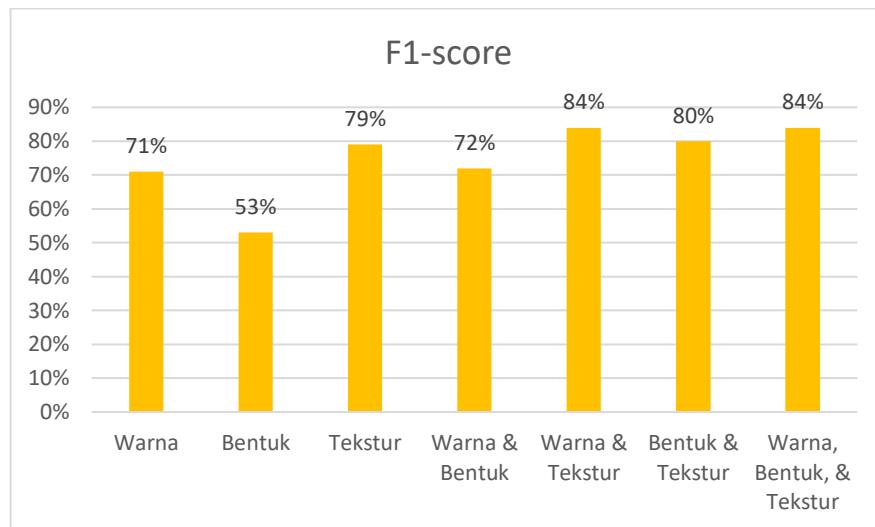
Selain menghitung nilai *precision*, penulis juga melakukan perhitungan terhadap nilai *recall*. *Recall*, digunakan untuk mengukur kemampuan sistem untuk mengidentifikasi keseluruhan jumlah sampel positif yang sebenarnya dari total keseluruhan sampel positif yang ada. Berdasarkan evaluasi model yang telah dilakukan, berikut merupakan evaluasi *recall* terhadap model klasifikasi yang telah dibuat.



Gambar 5. 17 Hasil Evaluasi Model berdasarkan *Recall*

Dari tabel di atas, dapat dilihat bahwa skenario penggunaan fitur warna dan tekstur secara terpisah maupun kombinasi memberikan hasil *recall* yang cukup baik. Skenario yang melibatkan fitur tekstur saja menunjukkan *recall* tertinggi, mencapai 79%. Kombinasi fitur warna dan tekstur, serta penggunaan semua tiga fitur (warna, bentuk, dan tekstur) menghasilkan *recall* yang lebih tinggi, dengan masing-masing mencapai 84%. Di sisi lain, skenario yang hanya mempertimbangkan fitur bentuk menunjukkan *recall* yang lebih rendah, yakni 53%.

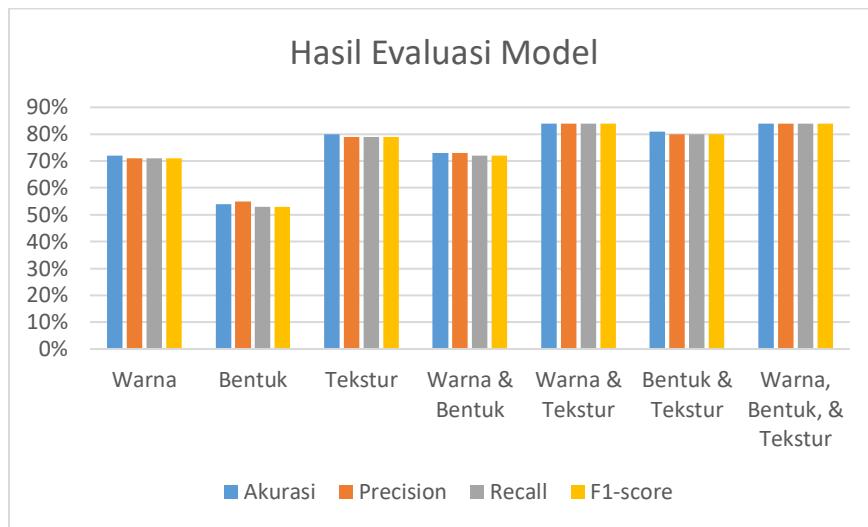
Penulis juga melakukan penghitungan *f1-score* sebagai metrik evaluasi tambahan. *F1-score* menggabungkan antara *precision* dan *recall* dalam satu angka, sehingga memberikan pandangan yang lebih luas mengenai keseimbangan antara kemampuan sistem dalam mengidentifikasi positif sejati (*recall*) dan menghindari kesalahan klasifikasi positif palsu (*precision*). Berdasarkan evaluasi model yang telah dilakukan, maka berikut ini hasil evaluasi *f1-score* terhadap model klasifikasi dalam bentuk grafik:



Gambar 5. 18 Hasil Evaluasi Model berdasarkan *F1-score*

Dari tabel di atas, dapat dilihat bahwa skenario penggunaan fitur warna dan tekstur secara terpisah maupun kombinasi memberikan hasil *recall* yang cukup baik. Skenario yang melibatkan fitur tekstur saja menunjukkan *f1-score* tertinggi, mencapai 79%. Kombinasi fitur warna dan tekstur, serta penggunaan semua tiga fitur (warna, bentuk, dan tekstur) menghasilkan *f1-score* yang lebih tinggi, dengan masing-masing mencapai 84%. Di sisi lain, skenario yang hanya mempertimbangkan fitur bentuk menunjukkan *f1-score* yang lebih rendah, yakni 53%.

Berdasarkan evaluasi model yang telah dilakukan, berikut merupakan rekapitulasi hasil evaluasi terhadap model klasifikasi menggunakan metode *k-nearest neighbour* dalam bentuk grafik.



Gambar 5. 19 Hasil Evaluasi Model Secara Keseluruhan

Skenario yang melibatkan kombinasi ekstraksi fitur warna dan tekstur, serta skenario yang melibatkan semua tiga ekstraksi fitur (warna, bentuk, dan tekstur), mencapai hasil yang konsisten dan optimal dengan nilai akurasi, presisi, recall, dan F1-score sebesar 84%. Hal ini menunjukkan bahwa penggunaan kombinasi fitur warna dan tekstur saja mampu memberikan performa setara dengan penggunaan semua fitur yang tersedia.

Skenario yang berfokus hanya pada atribut bentuk memiliki performa yang lebih rendah dibandingkan dengan skenario lainnya, dengan nilai akurasi, presisi, recall, dan F1-score di bawah 60%. Hal ini menunjukkan bahwa fitur bentuk mungkin kurang efektif dalam mengklasifikasikan data dalam kerangka penelitian ini.

Ekstraksi fitur tekstur menunjukkan kontribusi yang cukup kuat dalam semua skenario, dengan akurasi, presisi, recall, dan F1-score yang konsisten di atas 75%. Hal ini menandakan bahwa tekstur memiliki pengaruh yang signifikan dalam membuat prediksi yang akurat.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa penggunaan fitur warna dan tekstur, baik secara terpisah maupun bersama-sama, memiliki performa yang paling baik dalam konteks ini. Ekstraksi fitur bentuk memiliki dampak yang lebih rendah dalam kemampuan klasifikasi.

5.3.2 Perbandingan Hasil dengan Penelitian Terdahulu

Penulis melakukan penelitian dengan topik yang serupa dengan penelitian yang dilakukan oleh (Adang et al., 2020) yaitu mengenai klasifikasi kopra namun pada fokus spesifik yang berbeda. Pada penelitian (Adang et al., 2020) bertujuan untuk memprediksi presentase kekeringan sampel daging kopra guna membedakan kualitas kopra yang masuk dalam kategori kopra standar, bagus, dan sangat bagus. Pengujian akurasi yang dilakukan hanya menggunakan metrik akurasi.

Sementara itu, penelitian yang dilakukan oleh penulis lebih berfokus pada klasifikasi jenis kopra, yaitu kopra edible, reguler, dan reject. Jumlah dataset yang digunakan dalam penelitian penulis sebanyak 1.839 citra. Dengan menggunakan jumlah dataset yang lebih besar, penulis memiliki lebih banyak data untuk melatih dan menguji model klasifikasi. Selanjutnya, dalam menguji akurasi hasil klasifikasi, penelitian penulis menggunakan metrik akurasi, *confusion matrix*, dan *classification report*. Dengan penggunaan ketiga jenis pengujian tersebut, penelitian penulis dapat memberikan informasi yang lebih rinci mengenai performa model klasifikasi, termasuk *precision*, *recall*, dan *f1-score* untuk setiap kelas.

Penelitian yang dilakukan oleh (Suhendra et al., 2022) telah menggabungkan variasi fitur dalam eksperimennya. Dalam penelitian tersebut, dilakukan eksperimen dengan beberapa variasi fitur, seperti hanya fitur warna, hanya fitur tekstur, dan fitur gabungan antara fitur warna dan tekstur. Di sisi lain, penelitian penulis juga melakukan eksperimen dengan penggabungan variasi

fitur, namun dengan variasi yang berbeda. Variasi fitur yang diuji meliputi hanya fitur warna, hanya fitur bentuk, hanya fitur tekstur, kombinasi warna dan bentuk, warna dan tekstur, bentuk dan tekstur, serta kombinasi warna, bentuk, dan tekstur. Penelitian (Suhendra et al., 2022) dan penelitian penulis keduanya menguji pengaruh variasi fitur terhadap hasil eksperimen. Dengan menggabungkan variasi fitur yang berbeda, diharapkan dapat menghasilkan pemahaman yang lebih komprehensif tentang efektivitas metode yang digunakan dalam analisis data.

Dalam penelitian sebelumnya, metode *K-Nearest Neighbor* (K-NN) telah digunakan untuk melakukan klasifikasi jenis pisang dan mendapat akurasi sebesar 82% (Lestari et al., 2019). Kemudian, (Prayogo et al., 2022) juga melakukan penelitian yang fokus pada pengembangan sistem klasifikasi untuk jenis bunga mawar menggunakan metode *K-Nearest Neighbor* (K-NN). Hasil klasifikasinya menunjukkan akurasi sebesar 92%. Selanjutnya, penelitian yang dilakukan oleh (Halim & Anraeni, 2021) melakukan klasifikasi pada dataset citra penyakit pneumonia. Penelitian ini mencapai hasil klasifikasi dengan akurasi tinggi, yaitu sekitar 96%. Terakhir, (Rahayu Marlis et al., 2021) melakukan penelitian yang berfokus untuk memprediksi kualitas kopra putih menggunakan metode *K-Nearest Neighbor* (K-NN) dengan akurasi sebesar 83,3%.

Sementara itu, dalam penelitian penulis, metode *K-Nearest Neighbor* (K-NN) digunakan untuk klasifikasi jenis kopra berdasarkan ekstraksi fitur dan berhasil mencapai akurasi sebesar 84%. Meskipun akurasi penulis hanya mencapai 84%, namun model ini telah berhasil melakukan klasifikasi dengan baik. Perbedaan akurasi ini disebabkan oleh beberapa faktor, antara lain jumlah dataset, pengolahan data, dan penggunaan ekstraksi fitur yang berbeda.

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan hasil pembahasan dari penelitian yang telah penulis lakukan, berikut merupakan kesimpulan pada penelitian ini:

Pada penelitian ini, dilakukan ekstraksi fitur warna, bentuk, dan tekstur dalam analisis data citra. Fitur warna yang diekstraksi meliputi HSV, RGB, dan *grayscale*. Fitur bentuk yang diekstraksi meliputi luas dan *perimeter*, sedangkan fitur tekstur meliputi *contrast*, *dissimilarity*, *homogeneity*, *energy*, dan *correlation*. Hasil penelitian menunjukkan bahwa metode tersebut berhasil mencirikan data citra dengan baik. Selain itu, ditemukan bahwa penggunaan kombinasi fitur warna dan tekstur saja sudah mampu memberikan performa setara dengan penggunaan semua fitur yang tersedia. Fitur tekstur memiliki peran yang signifikan dalam mengidentifikasi data citra kopra dibandingkan dengan fitur warna dan bentuk.

Klasifikasi jenis kopra menggunakan ekstraksi fitur dengan metode *k-nearest neighbour* menghasilkan akurasi terbaik pada penggunaan kombinasi ekstraksi fitur warna dan tekstur, serta skenario yang melibatkan semua tiga ekstraksi fitur (warna, bentuk, dan tekstur), mencapai hasil yang konsisten dan optimal dengan nilai akurasi, presisi, recall, dan F1-score sebesar 84%.

6.2 Saran

Dalam penelitian ini, penulis menyadari masih banyak kekurangan dan keterbatasan yang perlu diperbaiki. Oleh sebab itu berdasarkan penelitian ini, penulis menyarankan untuk mengembangkan penelitian ini menjadi lebih baik dengan beberapa poin sebagai berikut:

1. Penelitian selanjutnya dapat mempertimbangkan penambahan fitur lainnya yang dapat memberikan kontribusi lebih lanjut dalam menggambarkan karakteristik citra.
2. Mencoba metode klasifikasi lainnya untuk membandingkan performa dan akurasi yang lebih tinggi.
3. Menggunakan jumlah data testing yang seimbang untuk masing-masing kelas agar dapat menghindari bias dalam evaluasi performa klasifikasi.
4. Mempertimbangkan penerapan teknik atau algoritma lain untuk meningkatkan kinerja klasifikasi jenis kopra, seperti penggunaan *ensemble learning* atau pengolahan data yang lebih canggih.

DAFTAR PUSTAKA

- A. S. Abdul Kadir. (2013). Teori dan Aplikasi Pengolahan Citra. Yogyakarta:CV Andi Offset
- Abdullah, A., Usman, U., & Efendi, M. (2017). Sistem Klasifikasi Kualitas Kopra Berdasarkan Warna dan Tekstur Menggunakan Metode Nearest Mean Classifier (NMC). *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 4(4), 297–303. <https://doi.org/10.25126/jtiik.201744479>
- Abdullah, & Usman. (2016). Sistem Cerdas untuk Klasifikasi Buah Kelapa Menggunakan Metode Backpropagation. *Selondang Mayang*, 87–94.
- Adang, Y., Rabi, A., & Arifuddin, R. (2020). Klasifikasi Tingkat Kematangan Kopra Menggunakan Metode naïve bayes. *Cyclotron*, 3(1). <https://doi.org/10.30651/cl.v3i1.4307>
- Adi Nugraha, K. (2019). Deteksi Area Parkir Mobil Berbasis Marker Menggunakan Moment Invariants dan K-NN. *Jurnal Teknik Informatika Dan Sistem Informasi*, 5(1), 112–121. <https://doi.org/10.28932/jutisi.v5i1.1586>
- Budi Putranto, B. Y., Hapsari, W., & Wijana, K. (2011). Segmentasi Warna Citra Dengan Deteksi Warna Hsv Untuk Mendeteksi Objek. *Jurnal Informatika*, 6(2). <https://doi.org/10.21460/inf.2010.62.81>
- Farokhah, L. (2020). Implementasi K-Nearest Neighbor untuk Klasifikasi Bunga Dengan Ekstraksi Fitur Warna RGB. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 7(6), 1129. <https://doi.org/10.25126/jtiik.2020722608>
- Ghodang, H., & Hantono. (2020). *Metodologi Penelitian Kuantitatif: Konsep Dasar & Aplikasi Analisis Regresi dan Jalur dengan SPSS*. PT. Penerbit Mitra Grup.
- Hadi, H. P., & Rachmawanto, E. H. (2022). Analisa Fitur Ekstraksi Ciri Dan Warna Dalam Proses Klasifikasi Kematangan Buah Rambutan Berbasis K-Nearest Neighbor. *Skanika*, 5(2), 177–189. <https://doi.org/10.36080/skanika.v5i2.2944>
- Halim, A. A. D., & Anraeni, S. (2021). Analisis Klasifikasi Dataset Citra Penyakit Pneumonia menggunakan Metode K-Nearest Neighbor (KNN). *Indonesian Journal of Data and Science*, 2(1), 01–12. <https://doi.org/10.33096/ijodas.v2i1.23>
- Heryadi, Y & Edy Irwansyah. (2020). Deep Learning: Aplikasinya di Bidang Geospasial. Depok: PT Artfisia Wahana Informa Teknologi
- Ilyas, M., Syahpawi, & Jonnius. (2022). Analisis Kelayakan Usaha Kopra Putih Di Kabupaten Indragiri Hilir Menurut Perspektif Ekonomi Syariah. *Islamic Business and Finance (IBF)*, 3(2), 158–178.

- Jenis Jenis Kopra, dan Teknik Pengeringan Kopra Asalan, Regular, Edible.* (n.d.). Retrieved May 13, 2023, from <https://rockrealms.com/2023/02/24/jenis-jenis-kopra/>
- Khatib, J., Dalam, S., Satria, B., Sidauruk, A., Wardhana, R., Akbar, A. Al, Ihsan, A., Gama, A. M., Yogyakarta, U. A., Bengkulu, U. D., Selatan, P. A., & Kunci, K. (2022). Implementasi Learning Vector Quantization untuk Klasifikasi Jenis Buah Kelapa menggunakan Image Processing. *Indonesian Journal of Computer Science*, 11(1), 566–576.
- Kirana, K. C. (2021). Pengolahan Citra Digital Teori Penerapan Pengolahan Citra Digital pada Deteksi Wajah. Malang: Ahlimedia Book
- Kusumanto, R. D., Tompunu, A. N., & Pambudi, S. (2011). Klasifikasi Warna Menggunakan Pengolahan Model Warna HSV Abstrak. *Jurnal Ilmiah Teknik Elektro*, 2(2), 83–87.
- Panggabean, A. K., Syahfaridzah, A., & Ardiningih, N. A. (2021). Mendeteksi Objek Berdasarkan Warna Dengan Segmentasi Warna Hsv Menggunakan Aplikasi Matlab. *METHOMIKA Jurnal Manajemen Informatika Dan Komputerisasi Akuntansi*, 4(2), 94–97. <https://doi.org/10.46880/jmika.vol4no2.pp94-97>
- Putra, Darma. (2010). Pengolahan Citra Digital (Edisi 1). Yogyakarta: Andi.
- Putri, A. R. (2016). Pengolahan Citra Dengan Menggunakan Web Cam Pada Kendaraan Bergerak Di Jalan Raya. *JIPI (Jurnal Ilmiah Penulisan Dan Pembelajaran Informatika)*, 1(01), 1–6. <https://doi.org/10.29100/jipi.v1i01.18>
- Rahayu Marlis, R., Abdullah, & Yunita, F. (2021). Sistem Prediksi Kualitas Kopra Putih Menggunakan k-Nearest Neighbor (k-NN). *SISTEMASI: Jurnal Sistem Informasi*, 10(2), 290–299. <http://sistemas.ftik.unisi.ac.id>
- Rakhmawati, P. U., Pranoto, Y. M., & Setyati, E. (2018). Klasifikasi Penyakit Daun Kentang Berdasarkan Fitur Tekstur dan Fitur Warna Menggunakan Support Vector Machine. *Seminar Nasional Teknologi Dan Rekayasa (SENTRA)*, 1–8.
- Rilo Pambudi, A., Garno, & Purwantoro. (2020). JIP (Jurnal Informatika Polinema) DETEKSI KEASLIAN UANG KERTAS BERDASARKAN WATERMARK DENGAN PENGOLAHAN CITRA DIGITAL. *Jurnal Informatika Polinema*, 6(4), 69–74.
- Soepomo, P. (2013). SISTEM IDENTIFIKASI CITRA JENIS CABAI (Capsicum Annum L.) MENGGUNAKAN METODE KLASIFIKASI CITY BLOCK DISTANCE. *JSTIE (Jurnal Sarjana Teknik Informatika) (E-Journal)*, 1(2), 409–418.
- Salamah, U. G., & Risma E. (2021). Pengolahan Citra Digital. Bandung: Media Sains Indonesia.
- Susilowati, E. (2018). Konversi Citra RGB Ke Citra HSV Dan HCL Pada Citra

- Jeruk Medan. *Seminar Nasional Teknologi Informasi Dan Komunikasi STI&K (SeNTIK)*, 2, 67–71.
- Tingkat, U. M., Buah, K., Bintang, B., Shandy, Q., Panna, S. S., & Malago, Y. (2019). Penerapan Metode Grey Level Co-Occurrence Matriks (GLCM) dan K-Nearest Neighbor (K-NN). *Jurnal Nasional CosPhi*, 3(1), 2597–9329.
- W, Romy Budhi. (2022). Machine Learning Metode k-Nearest Neighbors Klasifikasi Angka Bahasa Isyarat. Malang: Media Nusa Creative.
- Wuryani, N., & Agustiani, S. (2021). Random Forest Classifier untuk Deteksi Penderita COVID-19 Berbasis Citra CT Scan. *Jurnal Teknik Komputer AMIK BSI*, 7(2), 187–193. <https://doi.org/10.31294/jtk.v4i2>

LAMPIRAN

Lampiran 1

Hasil Wawancara Pihak Pedagang Kelapa/Kopra di Indragiri Hilir, Provinsi Riau

Narasumber : Mohd Rifai
 Tanggal Wawancara : 25 Juli 2023
 Waktu Wawancara : 13.10 WIB
 Pekerjaan : Pedagang Kelapa/Kopra

Pertanyaan wawancara

1. Apa definisi kopra dan fungsinya dalam dunia industri?
 = Kopra adalah daging buah kelapa yang dikeringkan. Umumnya, kopra yang dapat dikonsumsi (edible) digunakan dalam berbagai makanan dan biasanya dieksport ke Dubai. Sementara untuk kopra yang memenuhi standar, biasanya dijadikan sebagai bahan kosmetik. Sedangkan kopra yang tidak memenuhi standar (reject) dapat diolah menjadi minyak kopra (CCO).
2. Bagaimana klasifikasi kopra saat ini dilakukan? Apakah ada metode atau teknologi terbaru yang digunakan?
 = Saat ini, proses klasifikasi kopra masih dilakukan secara manual atau berdasarkan intuisi (feeling) dalam memilahnya.
3. Apa saja kualitas yang terkandung dalam kopra?
 = Terdapat tiga kualitas utama dalam kopra, yaitu Edible (konsumsi), Reguler, dan Reject (kelas yang ditolak).
4. Bagaimana kita dapat mengidentifikasi ciri-ciri unik dari setiap kualitas kopra?
 = Edible merupakan kualitas kopra yang memiliki diameter kurang dari 9 cm, berbentuk seperti mangkok, bebas dari cacat pada kulit ari, tidak berlubang, tidak berwarna kecoklatan, tidak memiliki jamur, serta berwarna putih pada bagian dagingnya. Sementara itu, untuk kualitas Reguler, kopra memiliki diameter lebih dari 9 cm dan bentuknya tidak diwajibkan untuk seperti mangkok. Selain itu, ciri-ciri lainnya sama dengan kualitas Edible. Adapun

kualitas Reject muncul setelah proses penyortiran Edible dan Reguler. Kopra dengan kualitas Reject biasanya mengandung jamur, berwarna kecokelatan, dan memiliki lubang atau keretakan.

5. Dari mana sumber dataset kopra tersebut diperoleh dan bagaimana proses pengumpulannya?
= Sumber dataset kopra tersebut diperoleh dari gudang kelapa/kopra yang terletak di Kota Tembilahan, Kabupaten Indragiri Hilir, Provinsi Riau. Wilayah gudang tersebut mencakup lahan seluas kurang lebih 1 hektar.
6. Dalam pengumpulan dataset ini, bagaimana proses pengambilan gambar kopra dilakukan?
= Proses pengambilan gambar dilakukan menggunakan berbagai jenis kamera handphone, termasuk dari merek-merek seperti Infinix, Samsung S23, Poco M3, dan Oppo Reno 8.

Lampiran 2

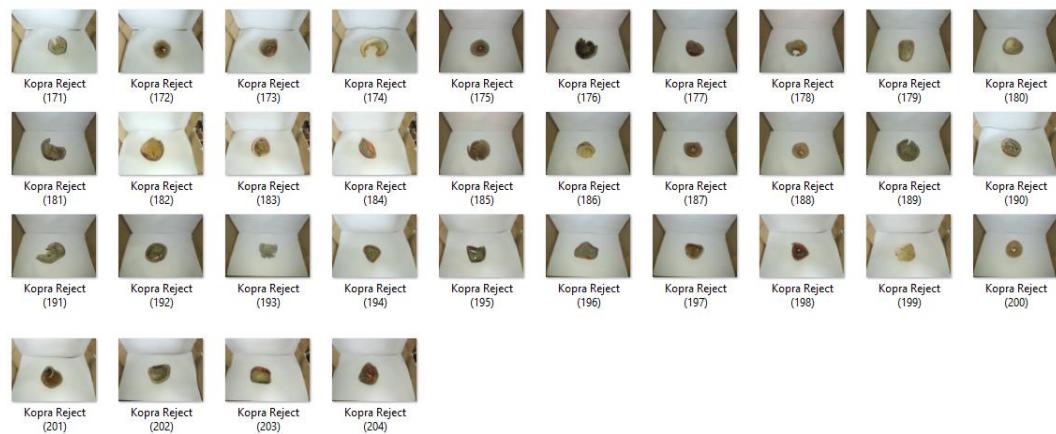
Dataset Asli











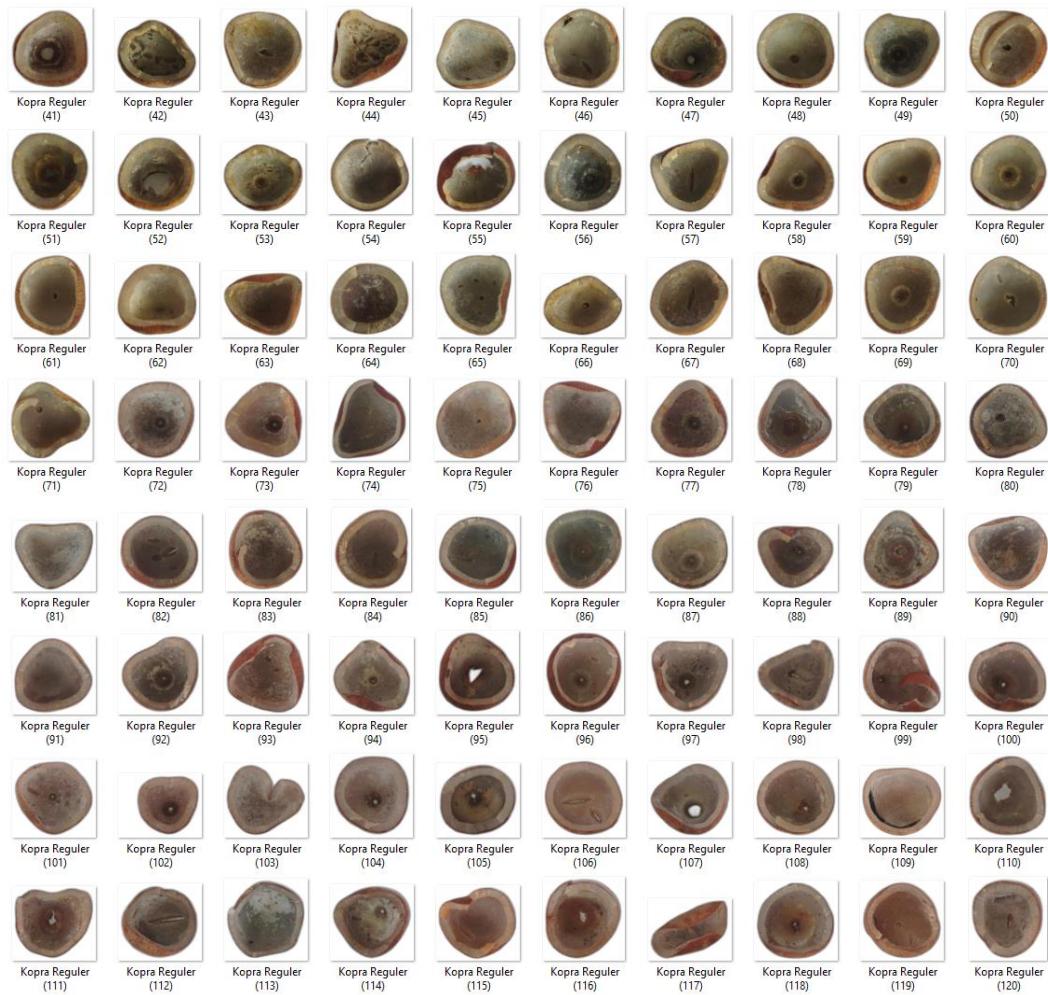
Note: Dataset yang terlampir hanya 1/3 dari dataset yang digunakan.

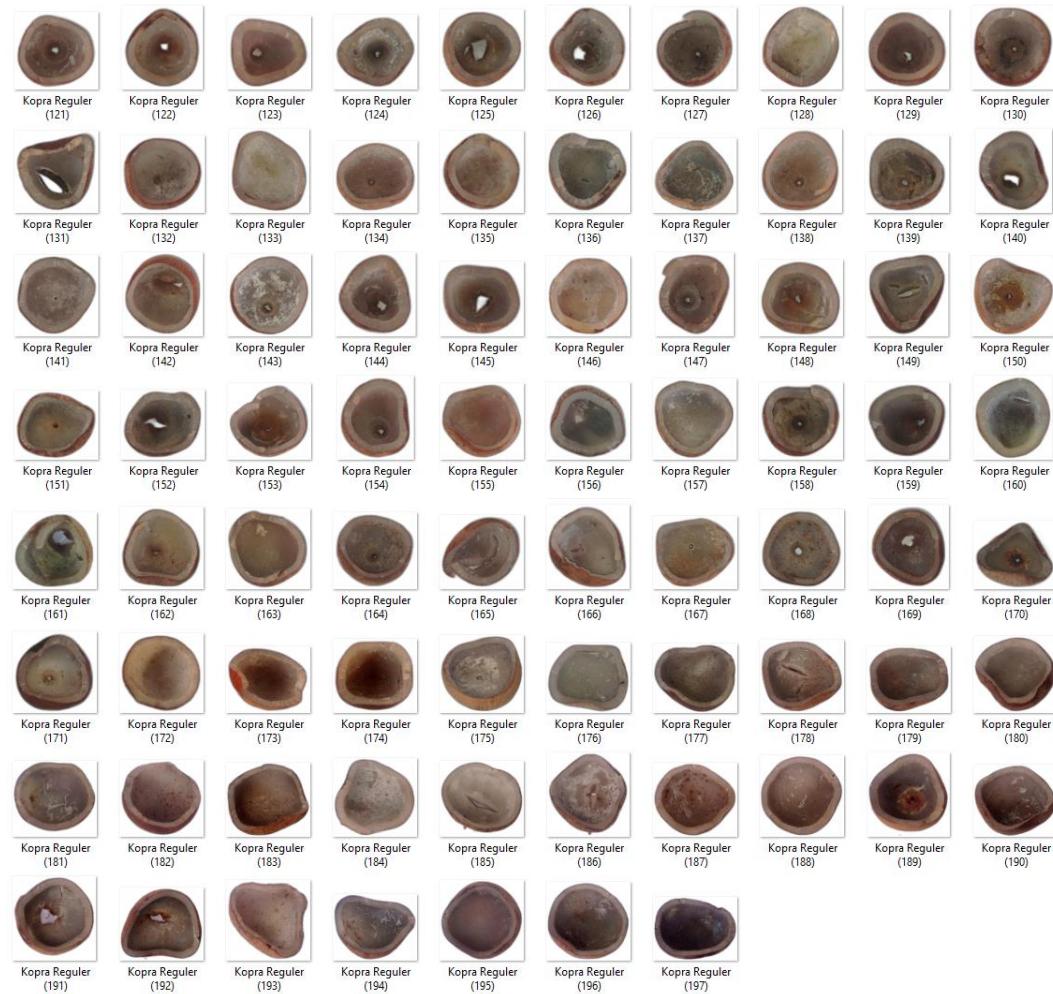
Lampiran 3

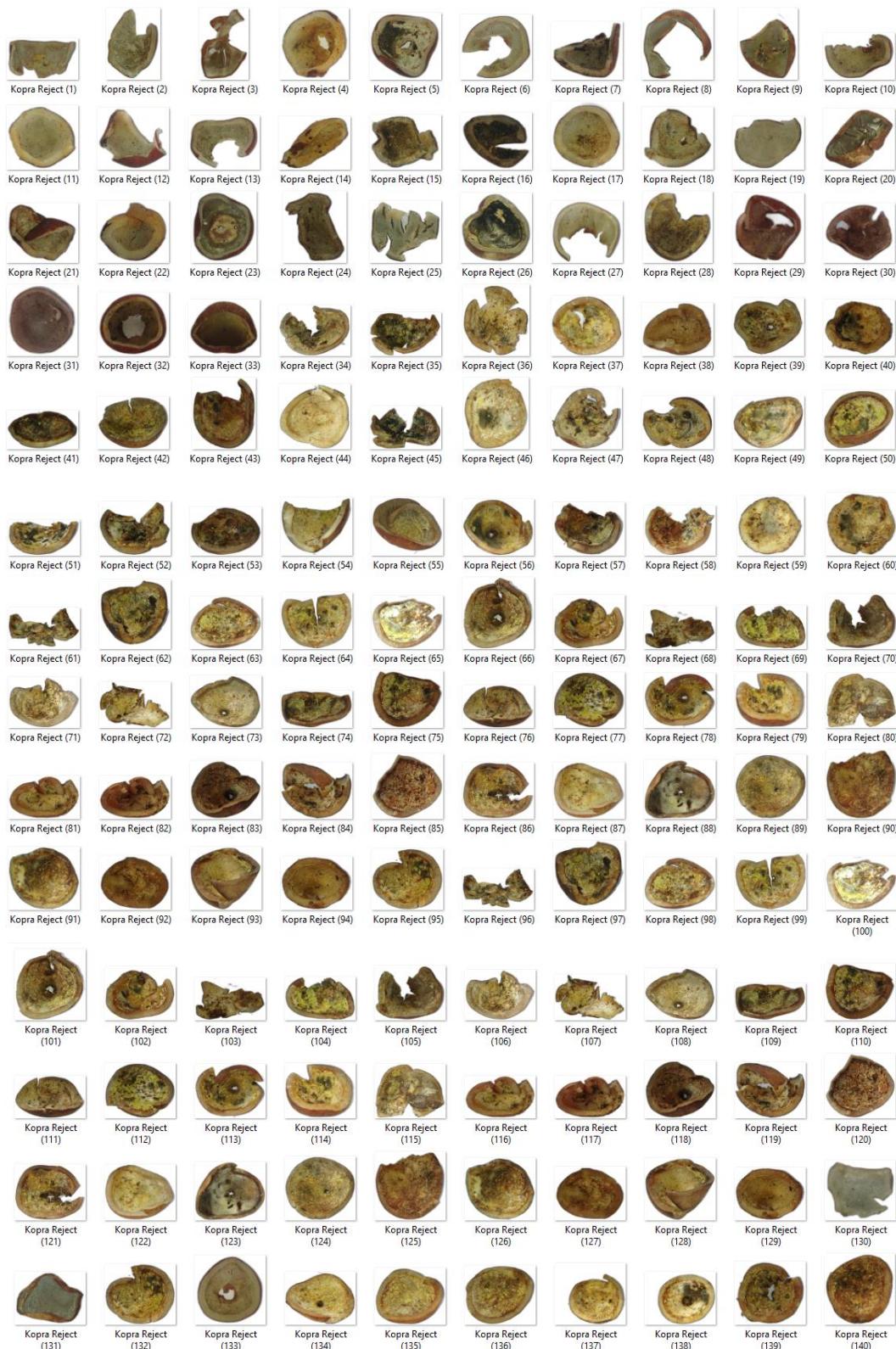
Dataset setelah *preprocessing*

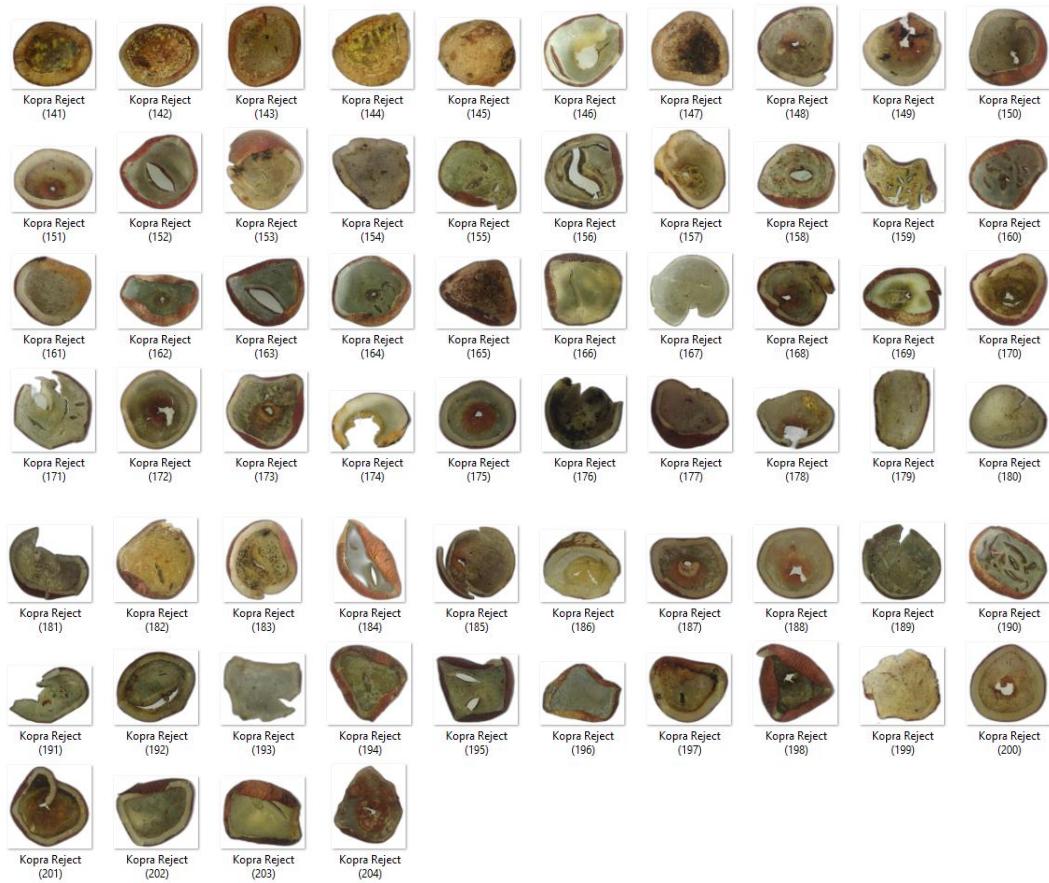












Note: Dataset yang terlampir hanya sebagian dari dataset yang digunakan.

Lampiran 4

Nilai Hasil Ekstraksi Fitur

Mean-R	Mean-G	Mean-B	Mean-H	Mean-V	Mean-S	Mean-Gray	Standar-Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homo-geneity	Energy	Corre-lation	Jenis Kopra
99,774	85,299	72,579	11,272	99,791	51,024	88,194	54,589	7934	2053,30	43,447	3,341	0,482	0,228	0,993	1
114,163	110,425	91,146	19,004	114,828	36,872	109,358	79,939	33489	1334,85	68,707	4,141	0,474	0,276	0,995	1
86,759	77,096	58,220	14,374	86,760	58,790	77,834	52,085	1959	573,85	37,735	2,917	0,522	0,236	0,993	1
80,398	71,179	55,915	12,919	80,399	52,578	72,237	52,309	2908	862,97	34,472	2,606	0,565	0,264	0,994	1
98,964	89,047	68,022	14,364	98,967	56,024	89,634	63,758	31868	706,14	51,522	3,058	0,541	0,258	0,994	1
87,446	76,920	59,293	13,503	87,447	59,273	78,085	52,617	7023	1290,92	33,878	2,786	0,511	0,241	0,994	1
80,187	71,379	53,273	14,589	80,189	59,821	71,980	48,879	723	259,66	39,559	2,848	0,539	0,228	0,992	1
91,190	79,020	60,322	12,773	91,192	59,090	80,544	57,829	21637	853,38	49,444	3,304	0,521	0,259	0,993	1
71,674	54,416	41,880	9,396	71,678	77,006	58,163	42,450	156	139,44	53,777	3,322	0,558	0,233	0,985	1
96,604	84,657	67,755	13,042	96,607	53,946	86,296	58,899	15980	2071,20	38,650	2,741	0,552	0,241	0,994	1
81,808	73,721	55,597	14,646	81,811	56,800	74,069	52,398	7270	1599,03	53,331	3,261	0,533	0,259	0,990	1
77,521	68,087	49,085	13,561	77,523	62,684	68,763	51,794	2963	671,35	44,021	3,191	0,513	0,288	0,992	1
95,425	93,474	70,487	20,414	95,979	48,328	91,431	61,297	35905	790,12	51,865	3,253	0,510	0,244	0,993	1
106,768	92,931	72,231	13,849	106,774	61,007	94,712	58,789	32614	1137,89	47,787	3,055	0,546	0,214	0,993	1
96,763	84,131	58,786	14,781	96,765	71,830	85,030	54,812	6377	2090,64	47,073	3,568	0,439	0,220	0,992	1
82,707	61,559	44,572	10,012	82,708	84,203	65,965	43,960	186	120,23	46,931	3,481	0,488	0,231	0,988	1
73,760	62,155	47,075	10,903	73,761	57,817	63,914	51,175	1942	761,01	34,756	2,459	0,617	0,321	0,993	1
95,363	86,866	65,598	15,380	95,365	56,780	86,985	58,866	11491	2049,73	48,307	3,473	0,478	0,237	0,993	1
93,172	76,046	58,448	11,312	93,173	68,912	79,175	50,837	1389	536,90	48,835	3,412	0,488	0,212	0,990	1
69,908	65,951	50,101	17,495	69,961	49,081	65,310	46,059	124	80,08	33,353	2,327	0,642	0,265	0,992	1

Mean-R	Mean-G	Mean-B	Mean-H	Mean-V	Mean-S	Mean-Gray	Standar-Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homo-geneity	Energy	Corre-lation	Jenis Kopra
82,322	76,850	65,503	15,001	82,367	37,006	77,186	53,200	6037	974,25	41,492	2,820	0,568	0,240	0,993	1
97,095	85,619	67,373	13,823	97,098	57,347	87,024	54,073	36475	981,80	45,557	3,044	0,506	0,214	0,992	1
98,126	94,195	76,388	18,720	98,284	41,117	93,329	61,261	28939	735,80	56,018	3,537	0,497	0,226	0,992	1
105,496	92,185	59,341	16,190	105,532	80,217	92,413	62,550	19637	1561,76	62,841	3,972	0,479	0,231	0,992	1
104,807	95,949	75,622	14,267	104,809	47,703	96,233	69,637	23688	629,41	44,961	3,028	0,536	0,270	0,995	1
105,768	94,195	72,447	14,425	105,769	58,775	95,154	61,220	31363	809,88	40,126	3,096	0,476	0,222	0,995	1
96,405	85,073	66,704	13,538	96,408	56,029	86,391	57,666	26367	637,13	49,666	3,092	0,547	0,231	0,992	1
93,130	84,543	66,545	14,149	93,131	50,483	85,081	58,421	14863	1821,67	36,253	2,423	0,614	0,260	0,995	1
91,197	81,594	64,198	14,197	91,198	54,303	82,492	54,892	5836	1478,22	44,147	3,099	0,519	0,224	0,993	1
78,814	63,760	47,521	11,527	78,819	71,917	66,432	46,148	881	470,32	49,217	3,232	0,511	0,243	0,988	1
82,861	76,938	58,464	16,823	82,869	53,118	76,582	53,888	2957	542,84	45,103	3,247	0,494	0,241	0,992	1
95,772	91,348	72,035	17,840	95,779	45,728	90,437	60,257	18052	1475,69	53,123	3,395	0,522	0,233	0,993	1
116,237	105,117	90,779	12,892	116,239	41,744	106,790	67,688	32010	751,21	68,279	3,809	0,479	0,208	0,992	1
79,624	75,050	55,628	17,587	79,688	54,805	74,190	51,403	1607	521,57	35,642	2,684	0,545	0,239	0,993	1
99,774	85,299	72,579	11,272	99,791	51,024	88,194	54,589	7934	2053,30	43,447	3,341	0,482	0,228	0,993	1
100,198	90,585	71,278	14,956	100,200	54,788	91,255	58,253	16349	1530,50	43,194	2,909	0,535	0,213	0,994	1
79,085	71,987	55,976	14,999	79,095	51,244	72,276	52,795	3437	663,45	55,863	3,481	0,513	0,252	0,990	1
100,952	91,635	74,014	13,703	100,960	45,487	92,416	67,364	18118	551,29	69,162	3,817	0,507	0,263	0,992	1
86,759	77,096	58,220	14,374	86,760	58,790	77,834	52,085	1959	573,85	37,735	2,917	0,522	0,236	0,993	1
80,398	71,179	55,915	12,919	80,399	52,578	72,237	52,309	2908	862,97	34,472	2,606	0,565	0,264	0,994	1
98,964	89,047	68,022	14,364	98,967	56,024	89,634	63,758	31868	706,14	51,522	3,058	0,541	0,258	0,994	1
87,446	76,920	59,293	13,503	87,447	59,273	78,085	52,617	7023	1290,92	33,878	2,786	0,511	0,241	0,994	1
91,804	81,237	59,719	14,813	91,810	63,043	81,947	54,642	6613	1228,64	38,228	2,746	0,537	0,233	0,994	1

Mean-R	Mean-G	Mean-B	Mean-H	Mean-V	Mean-S	Mean-Gray	Standar-Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homo-geneity	Energy	Corre-lation	Jenis Kopra
79,311	75,710	57,640	17,085	79,513	46,491	74,721	56,639	4316	837,90	51,375	3,432	0,520	0,294	0,992	1
84,942	79,956	64,400	16,346	84,955	42,446	79,644	57,007	9319	1372,62	39,264	2,603	0,584	0,270	0,994	1
98,325	96,614	85,834	19,915	98,694	23,787	95,881	63,308	44291	1093,54	36,599	3,087	0,476	0,251	0,995	1
81,554	71,642	56,683	13,626	81,564	53,878	72,935	50,133	373	214,65	33,236	2,747	0,525	0,241	0,993	1
75,256	68,082	53,644	15,784	75,264	51,213	68,580	46,768	20	42,04	34,014	2,639	0,549	0,243	0,992	1
114,587	99,295	74,279	13,892	114,602	62,523	100,989	68,537	25227	814,16	67,231	4,003	0,465	0,241	0,993	1
103,057	93,789	71,516	14,698	103,061	52,732	93,963	66,892	26271	680,04	40,119	2,943	0,526	0,266	0,995	1
101,232	98,327	78,365	19,096	101,974	41,248	96,919	71,879	31118	1152,56	59,623	3,712	0,488	0,281	0,994	2
97,432	88,982	66,263	16,978	97,459	61,455	88,914	56,556	13326	2305,41	78,802	4,451	0,428	0,215	0,988	2
67,823	54,449	38,857	11,097	67,824	76,060	56,677	43,893	205	191,24	78,350	4,457	0,459	0,264	0,980	2
68,330	47,682	30,328	10,474	68,332	94,354	51,887	42,976	1010	339,61	59,260	3,954	0,482	0,281	0,984	2
70,990	66,444	50,359	17,864	71,091	56,149	65,964	45,645	400	155,10	100,592	5,893	0,371	0,206	0,976	2
69,483	60,029	42,829	14,038	69,489	71,077	60,895	47,928	2237	636,78	64,936	4,101	0,454	0,246	0,986	2
57,085	46,834	34,181	13,328	57,089	76,038	48,474	41,396	106	97,05	74,855	4,362	0,446	0,233	0,978	2
105,524	92,349	68,041	14,602	105,529	67,411	93,515	64,477	11509	1269,95	89,248	4,856	0,414	0,217	0,989	2
81,299	72,949	45,643	16,001	81,324	79,772	72,317	57,581	8706	1252,52	118,509	5,845	0,403	0,260	0,982	2
91,768	79,577	63,563	12,403	91,771	56,300	81,378	62,508	18948	749,64	94,755	4,989	0,439	0,253	0,988	2
89,691	79,043	60,035	13,899	89,695	58,636	80,051	63,961	14141	530,74	72,738	4,306	0,451	0,268	0,991	2
94,931	87,896	64,408	16,771	94,994	59,477	87,317	63,707	11242	1249,83	85,728	4,912	0,414	0,243	0,989	2
90,813	85,529	60,688	18,624	90,937	61,844	84,265	57,633	39693	1243,71	96,008	5,250	0,426	0,244	0,985	2
105,687	96,591	71,124	16,715	105,734	62,033	96,401	69,102	22583	628,78	94,651	5,021	0,430	0,240	0,990	2
80,149	70,457	49,650	15,783	80,357	68,401	70,987	55,559	4433	836,06	129,401	6,311	0,384	0,255	0,979	2
88,221	74,593	47,251	14,980	88,230	86,281	75,564	54,032	5939	1173,24	74,186	4,686	0,418	0,228	0,987	2

Mean-R	Mean-G	Mean-B	Mean-H	Mean-V	Mean-S	Mean-Gray	Standar-Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homo-geneity	Energy	Corre-lation	Jenis Kopra
81,224	73,054	56,289	14,560	81,240	56,596	73,567	55,418	6208	1277,45	99,578	5,254	0,423	0,249	0,984	2
88,344	78,765	60,220	14,535	88,365	57,950	79,509	60,175	7097	1282,48	107,352	6,024	0,389	0,247	0,985	2
102,934	92,856	64,451	15,605	102,940	67,058	92,648	69,692	17093	525,39	104,950	5,169	0,421	0,252	0,989	2
87,770	80,665	57,368	16,782	87,787	64,173	80,117	59,538	5619	917,28	93,568	5,174	0,408	0,240	0,987	2
95,186	88,803	65,710	17,537	95,303	60,905	88,063	62,549	12587	1198,98	73,084	4,392	0,429	0,229	0,991	2
85,408	69,157	47,430	12,000	85,413	78,011	71,534	57,677	4972	901,28	93,594	4,843	0,457	0,268	0,986	2
72,936	60,524	48,555	10,798	72,939	60,440	62,879	51,475	2736	695,31	92,882	5,192	0,417	0,223	0,982	2
84,507	78,723	59,953	17,964	84,592	53,560	78,304	54,417	9078	2314,83	122,862	6,107	0,402	0,252	0,979	2
78,971	66,702	53,919	10,926	78,973	59,266	68,908	56,804	6215	1013,06	67,476	4,346	0,443	0,236	0,990	2
46,332	38,459	29,643	11,779	46,364	59,608	39,816	33,344	4	8,24	37,852	3,255	0,502	0,297	0,983	2
53,140	51,207	37,573	18,569	53,780	56,947	50,222	40,002	839	267,38	34,439	3,067	0,491	0,268	0,989	2
83,419	78,770	58,836	17,552	83,499	51,843	77,871	58,414	9082	1827,95	73,051	4,790	0,434	0,273	0,989	2
80,062	64,905	50,287	10,654	80,063	66,661	67,779	52,687	3033	721,05	63,470	4,150	0,446	0,244	0,989	2
95,320	88,836	67,655	17,044	95,361	52,707	88,336	61,678	23377	741,49	62,335	4,182	0,444	0,242	0,992	2
78,425	67,167	53,893	12,255	78,446	57,501	69,034	55,642	5227	836,21	86,448	5,191	0,395	0,236	0,986	2
95,801	89,027	67,125	16,873	95,850	54,387	88,542	61,896	26787	699,15	94,956	5,461	0,393	0,245	0,988	2
60,534	45,409	36,121	8,515	60,534	68,338	48,872	40,785	517	223,97	58,922	3,809	0,512	0,293	0,982	2
56,644	44,727	33,860	8,611	56,644	63,655	47,051	43,573	195	120,23	38,792	3,072	0,539	0,314	0,990	2
84,507	78,723	59,953	17,964	84,592	53,560	78,304	54,417	9078	2314,83	122,862	6,107	0,402	0,252	0,979	2
107,278	98,596	73,106	17,637	107,286	61,689	98,275	60,368	27532	1283,51	94,299	5,548	0,364	0,195	0,987	2
99,128	91,003	69,667	15,981	99,136	54,253	90,996	63,426	11585	1208,15	106,646	5,900	0,387	0,235	0,987	2
109,415	103,877	77,903	18,568	109,622	53,932	102,565	69,505	27855	720,24	84,112	4,713	0,424	0,232	0,991	2
89,048	72,910	56,603	10,522	89,049	65,055	75,862	57,772	6442	981,66	60,200	4,109	0,433	0,250	0,991	2

Mean-R	Mean-G	Mean-B	Mean-H	Mean-V	Mean-S	Mean-Gray	Standar-Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homo-geneity	Energy	Corre-lation	Jenis Kopra
97,009	89,765	66,503	17,095	97,028	57,445	89,263	63,537	15285	674,34	95,125	5,336	0,406	0,229	0,988	2
70,158	60,052	48,256	11,027	70,158	57,175	61,730	52,427	3663	690,62	68,637	4,029	0,503	0,247	0,987	2
90,763	80,706	55,472	15,839	90,768	75,006	80,826	58,286	26038	786,36	120,772	6,175	0,390	0,224	0,982	2
68,394	54,597	40,852	10,420	68,396	70,170	57,157	47,242	1425	308,29	57,875	3,941	0,448	0,244	0,987	2
83,044	67,991	51,699	10,475	83,044	64,587	70,641	58,494	5931	1000,23	65,819	4,251	0,462	0,294	0,990	2
107,771	105,634	73,466	21,383	108,767	66,921	102,593	69,369	35107	811,25	118,730	5,860	0,399	0,210	0,988	2
74,957	71,264	53,506	18,494	75,157	50,737	70,340	51,111	1106	273,71	63,217	4,157	0,460	0,270	0,988	2
68,484	56,206	45,039	9,471	68,488	56,678	58,607	51,599	2339	459,59	57,821	3,457	0,544	0,290	0,989	2
108,533	101,771	78,140	17,073	108,544	50,619	101,063	69,680	31743	849,18	57,799	3,958	0,453	0,252	0,994	2
116,426	109,045	85,106	17,746	116,496	51,145	108,501	70,301	43494	873,19	66,793	4,331	0,440	0,228	0,993	2
60,052	48,211	36,685	9,183	60,052	62,620	50,440	47,874	1483	460,46	52,324	3,852	0,494	0,334	0,989	2
95,713	79,925	42,999	15,501	95,725	101,435	80,449	58,514	20647	1863,69	152,742	7,268	0,366	0,252	0,978	3
81,625	72,361	43,722	14,006	81,655	72,045	71,863	65,851	23159	2849,43	108,335	5,363	0,479	0,367	0,988	3
64,101	55,719	39,173	16,762	64,229	73,027	56,350	36,622	124	61,94	89,760	5,545	0,356	0,196	0,966	3
90,737	73,366	46,534	13,385	90,740	89,511	75,543	52,558	3703	994,17	78,530	4,898	0,407	0,230	0,986	3
56,827	42,709	30,236	11,262	56,895	89,690	45,509	36,976	1123	130,91	58,437	3,678	0,450	0,230	0,979	3
43,918	36,175	27,442	9,597	43,956	51,596	37,491	40,125	917	175,44	91,396	4,566	0,533	0,413	0,972	3
91,333	78,261	52,789	15,330	91,368	80,786	79,286	54,543	2639	327,85	90,965	5,249	0,389	0,200	0,985	3
66,721	61,937	48,567	13,553	66,726	41,449	61,858	52,769	127	113,88	82,216	4,574	0,479	0,339	0,985	3
54,228	51,131	37,627	13,119	54,280	38,319	50,522	53,117	363	189,58	43,837	2,928	0,600	0,450	0,992	3
57,369	49,161	34,750	13,207	57,375	66,453	49,994	38,391	67	61,60	62,111	3,896	0,456	0,271	0,979	3
44,329	33,616	23,351	10,246	44,334	76,688	35,644	29,978	39	28,14	48,490	3,629	0,476	0,311	0,973	3
73,319	63,044	45,190	14,326	73,330	72,045	64,092	46,167	881	353,40	74,043	4,941	0,385	0,215	0,983	3

Mean-R	Mean-G	Mean-B	Mean-H	Mean-V	Mean-S	Mean-Gray	Standar-Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homo-geneity	Energy	Corre-lation	Jenis Kopra
52,825	29,702	22,811	18,878	52,825	111,707	35,836	33,561	833	285,95	121,798	6,307	0,384	0,251	0,946	3
50,689	41,783	27,281	10,513	50,693	62,093	42,798	41,234	3	8,83	41,565	3,043	0,571	0,409	0,988	3
46,204	36,370	24,942	11,961	46,266	87,339	38,013	34,910	1415	201,14	57,461	3,845	0,440	0,255	0,976	3
60,665	51,070	37,880	13,834	60,733	71,531	52,433	35,977	15	23,56	76,161	5,279	0,358	0,203	0,970	3
65,716	55,402	37,070	12,804	65,716	74,769	56,406	43,704	28	42,04	48,110	3,777	0,469	0,278	0,987	3
52,518	47,494	35,970	13,839	52,522	52,714	47,673	38,472	1	4,00	39,481	3,251	0,500	0,305	0,987	3
70,511	59,035	44,305	12,608	70,512	68,891	60,789	41,436	44	30,97	64,651	4,366	0,421	0,223	0,981	3
64,850	53,424	38,141	11,995	64,878	72,907	55,095	42,616	366	95,84	67,156	4,346	0,440	0,263	0,981	3
79,365	68,170	44,724	15,269	79,435	81,031	68,851	51,037	3664	532,84	73,849	4,754	0,402	0,211	0,986	3
83,699	77,192	49,501	18,185	83,728	76,182	75,973	52,164	2218	1040,23	57,301	4,289	0,417	0,230	0,989	3
79,158	68,282	47,807	13,412	79,161	70,357	69,218	51,733	1883	416,17	72,546	4,450	0,445	0,260	0,986	3
77,445	63,389	39,469	13,472	77,457	82,505	64,874	50,825	2110	1424,10	88,461	5,344	0,433	0,312	0,983	3
62,285	56,500	46,613	14,881	62,448	47,408	57,100	45,659	2242	191,97	58,925	3,837	0,472	0,267	0,986	3
73,447	66,742	42,728	16,126	73,463	70,843	66,017	51,083	886	545,23	83,377	5,325	0,404	0,262	0,984	3
60,782	48,470	26,845	13,728	60,834	93,582	49,688	38,248	36	43,80	57,190	4,531	0,410	0,261	0,980	3
52,703	44,425	31,070	13,655	52,718	78,686	45,380	37,337	531	102,47	104,858	4,710	0,438	0,227	0,962	3
54,115	43,852	36,303	9,714	54,118	54,208	46,057	39,671	87	54,04	49,309	3,863	0,461	0,304	0,984	3
62,985	47,409	38,448	8,258	62,985	76,496	51,038	49,828	3696	572,54	76,965	4,868	0,442	0,315	0,984	3
81,225	62,191	52,669	8,146	81,225	65,720	66,804	65,590	8646	1337,55	99,291	4,986	0,465	0,316	0,988	3
68,891	42,682	31,141	6,839	68,891	100,644	49,208	45,380	2188	649,07	63,773	4,143	0,474	0,291	0,985	3
78,647	62,023	52,753	7,399	78,647	61,905	65,957	59,206	8432	1729,78	77,170	4,408	0,458	0,292	0,989	3
88,742	61,274	50,862	8,044	88,742	89,832	68,304	57,122	26425	746,22	97,961	5,408	0,385	0,214	0,985	3
67,756	48,520	35,419	14,862	67,759	82,720	52,786	50,476	7932	814,21	92,763	4,975	0,464	0,313	0,982	3

Mean-R	Mean-G	Mean-B	Mean-H	Mean-V	Mean-S	Mean-Gray	Standar-Deviasi	Luas	Perimeter	Contrast	Dissimilarity	Homo-geneity	Energy	Corre-lation	Jenis Kopra
62,673	35,242	27,057	6,305	62,673	107,739	42,539	42,124	282	173,92	105,777	5,358	0,443	0,257	0,970	3
62,287	46,663	36,586	8,017	62,287	75,181	50,183	48,004	2165	674,62	85,814	4,919	0,440	0,302	0,981	3
83,361	60,984	50,994	11,042	83,361	79,237	66,547	57,337	4824	1350,04	87,629	4,989	0,428	0,243	0,987	3
72,165	53,521	44,017	6,809	72,165	80,760	58,014	59,010	6710	1279,41	74,219	4,339	0,460	0,277	0,989	3
78,543	56,221	43,797	10,464	78,543	95,819	61,496	60,159	29256	716,14	93,718	4,891	0,426	0,235	0,987	3
75,056	53,337	41,254	7,278	75,056	86,441	58,455	59,054	5455	1190,14	99,226	4,627	0,472	0,303	0,986	3
95,576	67,417	52,876	7,668	95,576	85,518	74,193	57,633	6470	1103,77	131,338	6,425	0,371	0,240	0,980	3
65,872	41,540	29,079	6,604	65,872	94,045	47,405	43,270	255	109,74	94,317	5,092	0,457	0,308	0,975	3
97,566	69,545	53,737	8,054	97,566	93,386	76,134	57,982	12451	2764,98	106,874	5,179	0,432	0,210	0,984	3
89,259	71,182	58,829	9,118	89,259	75,539	75,195	59,291	12486	1694,53	78,439	4,407	0,434	0,234	0,989	3
65,434	40,378	28,763	18,343	65,436	98,203	46,565	37,760	5	10,24	75,382	5,189	0,402	0,274	0,974	3
80,270	51,577	31,652	7,793	80,270	117,856	57,889	57,777	6514	1369,27	103,491	5,499	0,401	0,254	0,984	3
99,425	64,992	39,562	8,609	99,425	107,264	72,412	60,149	10976	1870,39	89,178	4,934	0,439	0,281	0,988	3
68,099	43,976	30,843	7,910	68,099	96,904	49,698	46,826	1467	366,09	174,491	6,651	0,446	0,307	0,960	3
90,047	66,278	54,986	9,884	90,047	82,868	72,106	59,553	7833	1143,99	98,615	5,256	0,388	0,206	0,986	3

Note: Nilai ekstraksi fitur yang terlampir hanya 1/12 dari dataset yang digunakan, terdapat 50 citra untuk masing-masing jenis kopra.