

**PENGENALAN CITRA JENIS MAKANAN MENGGUNAKAN
KLASIFIKASI NAIVE BAYES DENGAN EKSTRAKSI FITUR HUE
SATURATION INTENSITY COLOR MOMENTS DAN
MORPHOLOGICAL SHAPE DESCRIPTORS**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
IAN LORD PERDANA
NIM: 155150200111095



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019

PENGESAHAN

Pengenalan Citra Jenis Makanan Menggunakan Klasifikasi Naïve Bayes dengan Ekstraksi Fitur Hue Saturation Intensity Color Moments dan Morphological Shape Descriptors

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun Oleh :
Ian Lord Perdana
NIM: 155150200111095

Skripsi ini telah diuji dan dinyatakan lulus pada
26 Juni 2019

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing 1

Yuita Arum Sari, S.Kom., M.Kom
NIK: 201609 880715 2 001

Dosen Pembimbing 2

Sutrisno, Ir., M.T.
NIP: 19570325 198701 1 001

Mengetahui

Ketua Jurusan Teknik Informatika



H. Astoto Kurniawan, S.T., M.T., Ph.D
NIP: 19710518 200312 1 001



PERNYATAAN ORISINALITAS

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

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

Malang, 1 Mei 2019



Ian Lord Perdana

NIM: 155150200111095



PRAKATA

Puji syukur kehadiran Tuhan Yang Maha Esa yang telah melimpahkan berkat dan kasih-Nya sehingga laporan skripsi yang berjudul "Pengenalan Citra Jenis Makanan Menggunakan Klasifikasi Naïve Bayes Dengan Ekstraksi Fitur Hue Saturation Intensity Color Moments Dan Morphological Shape Descriptors" ini dapat terselesaikan. Penulis menyadari bahwa laporan ini tidak akan berhasil tanpa bantuan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Yuita Arum Sari, S.Kom., M.Kom. selaku dosen pembimbing satu yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan laporan ini.
2. Sutrisno, Ir., M.T. selaku dosen pembimbing dua yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan laporan ini.
3. Seluruh Dosen Fakultas Ilmu Komputer Universitas Brawijaya yang telah mengajarkan dan memberikan ilmu yang bermanfaat kepada penulis selama masa perkuliahan.
4. Mama dan Bapak beserta keluarga besar atas segala kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik penulis, serta yang senantiasa tiada henti-hentinya memberikan doa dan semangat demi terselesaikannya skripsi ini.
5. Teman dalam menjalani perkuliahan dan penulisan skripsi Jimmy Sohaga Dachi, Yosua Dwi Amerta, Muhammad Rizky Setiawan, Gabriel Mulyawan, Usman Adi Nugroho dan Muhammad Wafiq atas bantuan serta dukungan yang telah diberikan pada masa perkuliahan.
6. Pihak-pihak yang membantu menyelesaikan skripsi ini yang tidak dapat disebutkan satu persatu.

Penulis menyadari bahwa dalam penyusunan laporan ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Malang, 1 Mei 2019

Penulis

ianlordsianipar@student.ub.ac.id

ABSTRAK

Ian Lord Perdana, Pengenalan Citra Jenis Makanan Menggunakan Klasifikasi Naïve Bayes Dengan Ekstraksi Fitur Hue Saturation Intensity Color Moments Dan Morphological Shape Descriptors.

Pembimbing: Yuita Arum Sari, S.Kom., M.Kom. dan Sutrisno, Ir., M.T.

Proses klasifikasi jenis makanan menjadi penting karena akan menentukan makanan yang akan diproses pada sistem yang merekam makanan dengan tujuan kesehatan atau diet. Proses untuk menentukan jenis makanan terdiri dari proses *preprocessing*, yang dilanjutkan dengan mengubah model warna dari RGB menjadi HSI. Proses selanjutnya adalah pengekstraksian fitur warna dengan *Color Moment* yang akan menghasilkan fitur *mean*, standar deviasi, dan *skewness* dari masing-masing *channel* warna citra. Setelah itu pengekstraksian fitur bentuk dengan *Morphological Shape Descriptors* (MSD) yang akan menghasilkan fitur panjang, lebar, diameter, dan rasio aspek. Setelah fitur-fitur dari citra makanan diperoleh, dilakukan pengklasifikasian dengan metode klasifikasi *Naive Bayes* dibantu dengan fungsi *LogSumExp* untuk proses penghitungan probabilitasnya. Pada pengujian hasil akurasi menghasilkan hasil akurasi sebesar 78% dengan penggunaan jumlah data uji sebanyak 100 data uji. Pengujian terhadap dimensi citra menghasilkan hasil akurasi sebesar 81% dengan ukuran dimensi 300x300 piksel. Pengujian terhadap jumlah fitur menghasilkan hasil akurasi sebesar 83% saat menggunakan metode ekstraksi fitur *color moment* saja. Dari pengujian yang telah dilakukan, disimpulkan bahwa metode *color moment* dan *morphological shape descriptor* dengan metode klasifikasi *Naive Bayes* dapat digunakan untuk mengklasifikasikan citra makanan.

Kata kunci: jenis makanan, *naive bayes*, *color moment*, *morphological shape descriptors*, *logsumexp*, klasifikasi

ABSTRACT

Ian Lord Perdana, Food Image Recognition Using Naïve Bayes Classification with Hue Saturation Intensity Color Moments and Morphological Shape Descriptors Features Extraction.

Supervisors: Yuita Arum Sari, S.Kom., M.Kom. dan Sutrisno, Ir., M.T.

The process of determining some kind of food become important because it will determine the food that will be processed in the system that recorded the food for health and diet purpose. The process of determining food consist of preprocessing process and then changing the color model of the image from RGB to HSI. The next process is color feature extraction with Color Moment method that will generate the mean feature, standard deviation feature, and skewness feature from every color channel. Then, for shape feature extraction will using Morphological Shape Descriptors that will generate the length feature, width feature, diameter feature, and aspect ratio feature from the image. After the feature get extracted from the image, do the classification process with Naive Bayes Method with the help of LogSumExp for the probability calculation. The result in the testing of the effect of testing data generate 78% accuracy value when using 100 testing data. The result in the testing the effect of image dimension generate 81% accuracy value when using 300x300 pixel image for testing. The result in the testing the effect of number feature used generate 83% accuracy value when using feature from Color Moment only. The conclusion is, the feature extraction Color Moment and Morphological Shape Descriptors with Naive Bayes classification can be used to determine the kind of some food.

Keywords: kind of food, naive bayes, color moment, morphological shape descriptors, logsumexp, classification

DAFTAR ISI

PERSETUJUAN	Error! Bookmark not defined.
PERNYATAAN ORISINALITAS.....	Error! Bookmark not defined.
PRAKATA.....	iii
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN.....	xii
BAB 1 PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan masalah	3
1.3 Tujuan.....	3
1.4 Manfaat	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan	4
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Model Warna Citra	6
2.2.1 Model Warna <i>Red, Green, Blue</i> (RGB).....	6
2.2.2 Model Warna <i>Cyan, Magenta, Yellow</i> (CMY).....	6
2.2.3 Model Warna <i>Hue, Saturation, Intensity</i> (HSI).....	7
2.3 Ekstraksi Fitur Warna	8
2.3.1 Color Moment	8
Keterangan:	8
2.4 Ekstraksi Fitur Bentuk.....	9
2.4.1 Morphological Shape Descriptors	9
2.5 Metode Klasifikasi	10
2.5.1 Klasifikasi <i>Naive Bayes</i>	10
2.5.2 Fungsi LogSumExp	11

2.6 Metode Evaluasi.....	11
2.6.1 Akurasi.....	12
BAB 3 METODOLOGI.....	13
3.1 Tipe Penelitian.....	13
3.2 Metode Penelitian.....	13
3.3 Lokasi Penelitian.....	14
3.4 Partisipan Penelitian	14
3.5 Peralatan Pendukung	14
3.6 Teknik Pengumpulan Data	15
BAB 4 PERANCANGAN	17
4.1 Perancangan Algoritme Pada Proses Implementasi.....	17
4.1.1 Proses <i>Preprocessing</i>	18
4.1.2 Proses <i>Grayscale</i>	19
4.1.3 Proses Konversi RGB ke HSI.....	19
4.1.4 Proses <i>Gaussian Blur</i>	22
4.1.5 Proses <i>Thresholding</i>	23
4.1.6 Proses Erosi	24
4.1.7 Proses Dilasi.....	25
4.1.8 Proses <i>Color Moment</i>	26
4.1.9 Proses <i>Morphological Shape Descriptors</i>	27
4.1.10 Proses Klasifikasi <i>Naive Bayes</i>	28
4.2 Manualisasi Penghitungan	29
4.2.1 Penghitungan Konversi model warna RGB ke model warna HSI	30
4.2.2 Penghitungan <i>Color Moment</i>	32
4.2.3 Penghitungan <i>Morphological Shape Descriptors</i>	33
4.2.4 Penghitungan Klasifikasi Naive Bayes	34
4.2.5 Penghitungan Evaluasi.....	43
BAB 5 IMPLEMENTASI	44
5.1 Batasan Implementasi.....	44
5.2 Implementasi Algoritme.....	44
5.2.1 Implementasi <i>Preprocessing</i> Citra Makanan.....	44



5.2.2 Implementasi Proses Ekstraksi Fitur Warna Dan Bentuk Pada Citra Makanan	46
5.2.3 Implementasi Pengujian Klasifikasi Citra Makanan.....	53
BAB 6 PENGUJIAN DAN ANALISIS	57
6.1 Pengujian Hasil Akurasi Klasifikasi Citra Makanan.....	57
6.2 Pengujian Terhadap Dimensi Data Citra	61
6.3 Pengujian Terhadap Jumlah Metode Ekstraksi Fitur	66
BAB 7 KESIMPULAN	71
7.1 Kesimpulan.....	71
7.2 Saran.....	71
DAFTAR PUSTAKA.....	72
LAMPIRAN	74



DAFTAR TABEL

Tabel 2.1 Deskriptor MSD Beserta Deskripsinya	9
Tabel 3.1 Daftar Citra Data Latih	16
Tabel 3.2 Daftar Citra Data Uji.....	16
Tabel 4.1 Tabel Fitur <i>Color Moment</i>	33
Tabel 4.2 Fitur dan Kelas	34
Tabel 4.3 Mean Fitur	37
Tabel 4.4 Standar Deviasi Fitur.....	39
Tabel 6.1 Tabel Kode Makanan dan Nama Makanan.....	57
Tabel 6.2 Pengujian Data Uji Klasifikasi Makanan.....	58
Tabel 6.3 Kesalahan Klasifikasi Pada Pengujian	61
Tabel 6.4 Hasil Akurasi Pengujian Terhadap Dimensi Citra.....	61
Tabel 6.5 Pengujian Data Uji Berukuran 300x300 Piksel	61
Tabel 6.6 Kesalahan Klasifikasi Pada Pengujian Dimensi	65
Tabel 6.7 Hasil Akurasi Pengujian Terhadap Jumlah Fitur	66
Tabel 6.8 Pengujian Dengan 1 Metode Ekstraksi Fitur Warna.....	66
Tabel 6.9 Kesalahan Klasifikasi Pada Pengujian Jumlah Metode Ekstraksi Fitur	70

DAFTAR GAMBAR

Gambar 2.1 Model Warna RGB	6
Gambar 2.2 Hasil Penggabungan Model Warna CMY.....	7
Gambar 3.1 Proses Pelatihan	13
Gambar 3.2 Proses Pengujian	14
Gambar 3.3 Citra Hasil <i>Preprocessing</i>	15
Gambar 4.1 Alur Proses Pengujian.....	17
Gambar 4.2 Alur Proses Pengambilan Data Latih	18
Gambar 4.3 Alur Proses <i>Preprocessing</i>	19
Gambar 4.4 Alur Proses <i>Grayscale</i>	22
Gambar 4.5 Alur Proses Konversi RGB Ke HSI.....	21
Gambar 4.6 Alur Proses <i>Gaussian Blur</i>	23
Gambar 4.7 Alur Proses <i>Thresholding</i>	24
Gambar 4.8 Alur Proses Erosi.....	25
Gambar 4.9 Alur Proses Dilasi	26
Gambar 4.10 Alur Proses Klasifikasi Color Moment	27
Gambar 4.11 Alur Proses <i>Morphological Shape Descriptors</i>	28
Gambar 4.12 Alur Proses Klasifikasi Naive Bayes.....	29
Gambar 4.13 Sampel Data Citra Makanan.....	30
Gambar 6.1 Perbandingan Nilai Standar Deviasi Donat dan Rendang	60
Gambar 6.2 Perbandingan Hasil Akurasi Pengujian Dengan Dimensi Data	64
Gambar 6.3 Perbandingan Nilai Standar Deviasi Dimensi Citra.....	65
Gambar 6.4 Perbandingan Hasil Akurasi Pengujian Dengan Jumlah Fitur.....	69

DAFTAR LAMPIRAN

Lampiran 1 Hasil Uji Citra Dimensi 100x100	74
Lampiran 2 Hasil Uji Citra Dimensi 200x200	76
Lampiran 3 Hasil Uji Citra Dimensi 300x300	78
Lampiran 4 Hasil Uji Fitur Bentuk Dan Warna.....	81
Lampiran 5 Hasil Uji Fitur Bentuk.....	83
Lampiran 6 Hasil Uji Fitur Warna.....	85



BAB 1

PENDAHULUAN

1.1 Latar belakang

Makanan adalah salah satu kebutuhan pokok manusia yang berfungsi sebagai penghasil energi utama bagi tubuh manusia. Di dalam makanan terdapat tiga sumber energi bagi tubuh yaitu karbohidrat, lemak, dan protein (Hardinsyah, Riyaldi, & Napitupulu, 2013). Selain berfungsi sebagai sumber energi makanan juga berfungsi untuk pembentuk sel-sel baru pada tubuh dan pengatur metabolisme tubuh. Pentingnya makanan untuk dikonsumsi oleh manusia membuat kesadaran akan mengonsumsi makanan berdasarkan jenisnya semakin meningkat. Jenis yang ada pada makanan menjadi hal yang penting karena dapat memengaruhi kebutuhan sumber daya tubuh dari tubuh manusia. Dengan mengetahui jenis dan gizi dari suatu makanan, manusia dapat mengatur sendiri kebutuhan pangan yang akan dikonsumsi. (Mezgec & Seljak, 2017).

Proses untuk mengatur jenis makanan yang akan dimakan dapat dibantu dengan sistem bernama "FoodLog", yang dapat digunakan oleh pengguna untuk mengambil citra makanan yang dimakan setelah itu, sistem akan mendeteksi jenis makanan tersebut lalu menyimpan hasil deteksinya dan mengukur keseimbangan makanan (Maruyama, de Silva, Yamasaki, & Aizawa, 2010). Pada sistem "FoodLog" tahap pengenalan citra makanan menjadi penting karena tahap ini adalah tahap yang akan menentukan jenis makanan yang dikonsumsi dan prosesnya harus dapat dilakukan pada lingkungan yang berbeda-beda seperti restoran, kantin, dan pasar (Ciocca, Napoletano, & Schettini, 2017). Proses pengenalan citra dapat dilakukan dengan menggunakan proses klasifikasi karena proses klasifikasi dapat menghasilkan keluaran berupa suatu jenis kelas yang didapatkan dari hasil proses ekstraksi fitur citra (Alpaydin, 2010).

Metode klasifikasi yang dapat digunakan pada pengenalan citra seperti metode klasifikasi *Support Vector Machine* (SVM) dan juga metode klasifikasi *Naive Bayes*. Metode klasifikasi pengenalan citra dengan metode klasifikasi SVM telah dilakukan oleh Farinella pada tahun 2016. Penelitian tersebut menggunakan 4.754 gambar makanan dan penelitian tersebut menghasilkan akurasi sebesar 84,05% dan 85,82% dengan mengganti *Textons* dengan *Anti-Textons* (Farinella, Allegra, Moltisanti, & Stanco, 2016). Selain metode klasifikasi SVM, metode klasifikasi *Naive Bayes* juga merupakan salah satu aplikasi yang dapat digunakan untuk mengklasifikasikan citra. Metode *Naive Bayes* digunakan pada penelitian Dong-Chul pada tahun 2016 dan penelitian tersebut menghasilkan akurasi sebesar 77,2% dengan menggunakan 800 gambar data latih (Dong-Chul, 2016). Dibandingkan dengan metode SVM, metode *Naive Bayes* memiliki akurasi yang lebih rendah, tetapi metode SVM mempunyai beberapa kekurangan seperti hasil akurasi yang sangat bergantung dengan jenis kernel yang digunakan dan juga pada kecepatan waktu proses yang lambat dan besarnya memori yang digunakan pada saat melakukan proses klasifikasi (Burgess, 1998). Metode *Naive Bayes* juga

memiliki kelebihan seperti, tidak memerlukan data latih yang banyak untuk digunakan pada saat proses pelatihan dan dapat meminimalisasi waktu pelatihan karena menggunakan konsep probabilitas yang ada pada teorema Bayes (Murphy, 2006).

Pada sistem klasifikasi citra, selain metode klasifikasi dibutuhkan juga metode ekstraksi fitur untuk mengambil nilai fitur-fitur yang ada pada citra. Metode ekstraksi fitur yang dapat digunakan untuk mengambil nilai fitur citra adalah metode ekstraksi fitur warna dan bentuk. Metode ekstraksi fitur warna yang dapat digunakan pada pengenalan citra adalah metode *Color Histogram*. Metode tersebut digunakan pada penelitian tahun 2015 oleh Khatua. Penelitian tersebut menggunakan 1.000 data citra, menghasilkan hasil pencarian gambar yang memiliki nilai presisi di atas 0,6 dan recall di atas 0,4 (Khatua, Nayak, & Panda, 2015). Selain metode *Color Histogram*, juga ada metode ekstraksi warna *Color Moment*. Metode *Color Moment* digunakan pada penelitian tahun 2016 oleh Kusuma. Penelitian tersebut menggunakan 60 data latih dan menghasilkan akurasi sebesar 87% (Kusuma, Pawening, & Dijaya, 2016). Jika metode *Color Moment* dan metode *Color Histogram* dibandingkan, metode *Color Histogram* mempunyai kelemahan pada lama waktu komputasi yang lambat, penggunaan memori yang besar, dan memerlukan ruang penyimpanan yang besar (Lukac, 2013) dan untuk metode *Color Moment* mempunyai kelebihan yaitu, memiliki karakteristik yang efisien dan efektif pada saat pendistribusian warna pada citra (Soman, Ghorpade, & Sonone, 2012).

Proses klasifikasi citra juga dapat menggunakan metode ekstraksi bentuk untuk menambah nilai-nilai fitur yang akan diklasifikasikan. Metode ekstraksi fitur bentuk yang dapat digunakan untuk proses pengenalan citra adalah *Chain Codes*. Metode *Chain Codes* telah diterapkan di salah satu penelitian untuk mendeteksi wajah yang menggunakan metode ekstraksi bentuk oleh Nazmeen pada tahun 2013. Penelitian tersebut menggunakan data citra dari database wajah Universitas Essex dan menghasilkan akurasi sebesar 70% pada saat menggunakan deteksi tepi *Canny* dan deteksi kontur (Boodoo-Jahangeer & Baichoo, 2013). Selain metode *Chain Codes*, metode yang dapat digunakan untuk pengenalan citra adalah metode *Morphological Shape Descriptors* (MSD). Metode MSD adalah metode yang dapat menganalisis bentuk geometris dasar pada objek seperti diameter objek, luas objek, panjang maksimal dan minimal sumbu pada objek, dan daerah *centroid* pada objek. Metode MSD digunakan penelitian pengenalan citra daun pada tahun 2018 oleh Wäldchen. Penelitian tersebut menghasilkan akurasi sebesar 84,7% dengan metode klasifikasi SVM (Wäldchen & Mäder, 2017). Metode MSD memiliki kelebihan yaitu, mudah diimplementasikan karena metode MSD mengambil nilai-nilai dasar dari suatu bentuk objek seperti panjang, lebar, dan luas objek (Wäldchen & Mäder, 2017).

Dengan segala kelebihan-kelebihan yang telah disebutkan, metode-metode yang digunakan dalam proses penyusunan skripsi ini adalah metode klasifikasi *Naive Bayes* dan metode *Color Moment* beserta *Morphological Shape Descriptors* sebagai pengekstraksi fitur warna dan bentuk pada citra nantinya. Metode-

metode tersebut dipilih karena memiliki kelebihan yang dapat digabungkan pada penelitian ini.

1.2 Rumusan masalah

Sesuai dengan hasil dari uraian latar belakang, rumusan masalah yang akan dikaji di dalam penelitian ini adalah:

1. Bagaimana hasil nilai akurasi pada saat mengklasifikasikan citra dengan menggunakan ekstraksi fitur warna dan bentuk menggunakan algoritme *Naive Bayes*?
2. Bagaimana pengaruh hasil nilai akurasi pada saat mengklasifikasi citra dengan dimensi citra data uji yang berbeda dan algoritme *Naive Bayes*?
3. Bagaimana pengaruh hasil nilai akurasi pada saat mengklasifikasi citra dengan jumlah metode ekstraksi fitur yang berbeda dan algoritme *Naive Bayes*?

1.3 Tujuan

Sesuai dengan hasil dari uraian latar belakang, tujuan yang akan dituju dari diadakannya penelitian ini adalah:

1. Mengetahui hasil nilai akurasi pada saat mengklasifikasikan citra dengan menggunakan ekstraksi fitur warna dan bentuk menggunakan algoritme *Naive Bayes*.
2. Mengetahui pengaruh hasil nilai akurasi pada saat mengklasifikasikan citra dengan dimensi data uji yang berbeda dan algoritme *Naive Bayes*.
3. Mengetahui pengaruh hasil nilai akurasi pada saat mengklasifikasikan citra dengan jumlah metode ekstraksi fitur yang berbeda dan algoritme *Naive Bayes*.

1.4 Manfaat

Manfaat dari dilakukannya penelitian ini adalah dapat menggunakan algoritme *Color Moment* dan algoritme *Morphological Descriptors* untuk mengekstraksi fitur-fitur pada citra, dapat menggunakan algoritme *Naive Bayes* untuk mengklasifikasikan citra dengan parameter fitur warna dan fitur bentuk, dan dapat mengetahui hasil akurasi/pengujian hasil klasifikasi.

1.5 Batasan masalah

Untuk memperoleh hasil dari penelitian yang sesuai dan tidak meluas maka diperlukan batasan, di mana batasan tersebut adalah:

1. Data yang akan digunakan adalah data primer berupa citra makanan yang diambil dengan menggunakan kamera pada *handphone*.
2. Pencahayaan yang digunakan pada saat pengumpulan data citra adalah cahaya alami dari matahari pada pukul 10.00 WIB sampai dengan 14.00 WIB.

3. Jenis makanan yang digunakan sebanyak 10 jenis makanan yang terdiri dari: donat, telur dadar, rendang, selada, stroberi, pisang hijau, nasi kuning, dan mie soba.

1.6 Sistematika pembahasan

Agar penulisan skripsi menjadi lebih terarah dan terurut, maka penulis berusaha untuk menyusun tahap-tahap pembahasan secara sistematis dengan tujuan agar hubungan antara bab yang ada tampak jelas. Adapun isi dari tiap bab tersebut adalah:

BAB 1 PENDAHULUAN

Pada bab 1 berisi latar belakang penelitian, rumusan masalah penelitian, tujuan penelitian, manfaat, dan sistematika pembahasan skripsi.

BAB 2 LANDASAN KEPUSTAKAAN

Pada bab 2 berisi tentang landasan teori serta referensi pustaka yang akan mendasari proses penelitian.

BAB 3 METODOLOGI

Pada bab 3 berisi penjelasan metodologi dari penelitian yang akan dilakukan. Diantaranya adalah tipe penelitian, metode penelitian, lokasi penelitian, partisipan penelitian, peralatan pendukung, dan teknik pengumpulan data.

BAB 4 PERANCANGAN

Pada bab 4 berisi perancangan perangkat lunak yang akan dibangun meliputi perancangan data, perancangan proses, perancangan tabel, perancangan uji coba, dan penghitungan manualisasi.

BAB 5 IMPLEMENTASI

Pada bab 5 berisi implementasi dari perangkat lunak yang digunakan dan juga rancangan algoritme yang dirancang.

BAB 6 PENGUJIAN

Pada bab 6 berisi tentang penjelasan pengujian dan hasil pengklasifikasian citra makanan dari proses implementasi.

BAB 7 PENUTUP

Pada bab 7 berisi kesimpulan yang didapat dari hasil pembuatan dan pengujian yang dikembangkan dalam skripsi dan saran untuk pengembangan lebih lanjut.

BAB 2

LANDASAN KEPUSTAKAAN

Pada bab landasan kepastakaan akan membahas penelitian-penelitian yang berkaitan serta, mengkaji metode-metode yang dapat digunakan dalam penelitan Pengenalan Citra Jenis Makanan Menggunakan Klasifikasi Naive Bayes Dengan Ekstraksi Fitur Hue Saturation Intensity Color Moments dan Morphological Shape Descriptors. Di dalam kajian pustaka nantinya akan membahas penelitian yang sebelumnya telah dilakukan yang mempunyai hubungan dengan penelitian ini. Penelitian yang dipakai sebagai referensi adalah "*Image Classification Using Naive Bayes Classifier*" oleh Dong-Chul tahun 2016. Penelitian "*CBIR System using Color Moment and Color Auto-Correlogram with Block Truncation Coding*" oleh Vinayak pada tahun 2017. Penelitian "*Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review*" oleh Jana pada tahun 2017.

2.1 Kajian Pustaka

Penelitian pertama adalah penelitian yang dilakukan oleh Dong-Chul pada tahun 2016. Penelitian ini menggunakan 800 citra sebagai dataset yang dibagi menjadi 4 kelas, di mana masing-masing kelas mempunyai 200 citra. Penelitian ini menggunakan algoritme *Local Binary Pattern* (LBP), *Covariance Descriptor* (Cov.D), *Wavelet Packet Transform* (WPT) untuk mengekstraksi fitur-fitur di setiap citra. Algoritme klasifikasi yang dipakai pada penelitian adalah algoritme *Centroid Neural Network* yang menghasilkan akurasi sebesar 71,8% dan waktu latih selama 1,22 detik. Algoritme *Fuzzy C-Means* menghasilkan akurasi sebesar 66,5% dan waktu latih selama 1,86 detik. Algoritme *Multi-Layer Perceptron Neural Network* menghasilkan akurasi sebesar 72,6% dan waktu latih selama 4,22 detik. Algoritme *Naive Bayes* menghasilkan akurasi 77,2% dan waktu latih selama 0,42 detik (Dong-Chul, 2016).

Penelitian kedua adalah penelitian yang dilakukan oleh Vanda pada tahun 2017. Penelitian ini dilakukan dengan menggunakan 1.000 data citra dengan format JPEG. Setiap dataset dari citra tersebut dibagi ke dalam 10 kelas yang terdiri dari citra pemandangan alam, binatang, bus, bunga, dan air terjun. Tiap kelas di dalam dataset terdiri dari 100 citra. Penelitian ini menggunakan algoritme ekstraksi warna *Color Moment*, algoritme ekstraksi tekstur *Gabor Filter*, dan algoritme *Block Truncation Coding* (BTC). Penelitian ini menghasilkan nilai presisi di atas 0,6 untuk tiap-tiap kelas data citra (Vinayak & Sonika, 2017).

Penelitian ketiga adalah penelitian yang dilakukan oleh Jana Waldchen dan Patrick Mader pada tahun 2016. Penelitian yang dilakukan bertujuan untuk melakukan perancangan sistem untuk mengenali tanaman. Metode yang digunakan pada proses ekstraksi fitur bentuk adalah metode *Morphological Shape Descriptors* (MSD) dan beberapa metode klasifikasi seperti *Support Vector Machine* (SVM), *k-Nearest Neighbor* (k-NN). Penelitian ini menghasilkan tingkat akurasi dengan menggunakan metode MSD dan menggunakan metode klasifikasi

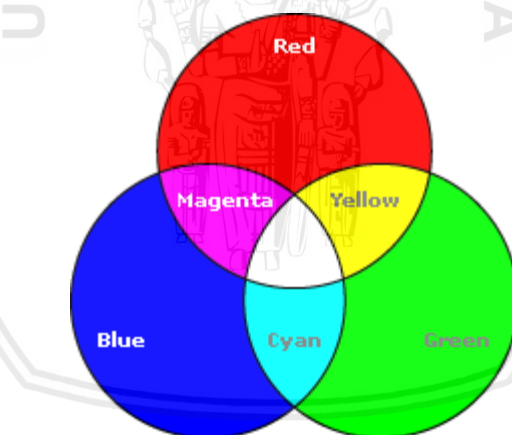
SVM menghasilkan akurasi sebesar 87,61%, sedangkan jika menggunakan metode MSD dengan metode klasifikasi k-NN menghasilkan akurasi sebesar 84,45% (Waldchen & Mäder, 2017).

2.2 Model Warna Citra

Model warna citra adalah metode untuk mendeskripsikan suatu jenis warna ke dalam bentuk sinyal warna agar warna tersebut dapat digunakan pada bidang multimedia, pemrosesan citra, grafik, dan aplikasi visi komputer (Kulkarni, Kurundkar, & Khare, 2015). Untuk menyimpan informasi tentang warna dibutuhkan setidaknya tiga parameter warna yaitu merah, hijau, dan biru. Model warna citra kemudian merepresentasikan nilai dari parameter warna-warna tersebut ke dalam format tiga dimensi. Model warna citra menggunakan fungsi matematika untuk merepresentasi posisi warna di dalam ruang tiga dimensi. Terdapat beberapa model citra warna yaitu sebagai berikut

2.2.1 Model Warna *Red, Green, Blue* (RGB)

Model warna RGB merepresentasikan masing-masing warna di dalam sebuah sistem koordinat, di mana setiap warna pada RGB memiliki nilai nol sampai dengan nilai maksimum. Desain model warna RGB tidak terlalu ideal untuk diaplikasikan ke beberapa sistem grafik komputer karena komponen warna merah, hijau, dan biru memiliki nilai korelasi yang tinggi (Kulkarni, Kurundkar, & Khare, 2015).



Gambar 2.1 Model Warna RGB

(Sumber: www.sketchpad.net)

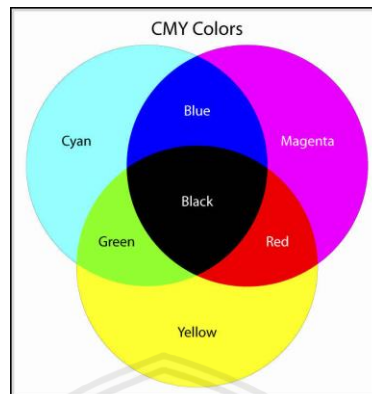
2.2.2 Model Warna *Cyan, Magenta, Yellow* (CMY)

Model warna CMY terdiri dari warna komplementer yaitu *Cyan*, *Magenta*, dan *Yellow*. Model warna CMY dapat dideskripsikan dengan fungsi berikut

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.1)$$

Dalam model warna CMY, jika warna *Yellow* dan *Cyan* digabungkan akan menghasilkan warna hijau. Warna *Yellow* digabungkan dengan warna *Magenta*

akan menghasilkan warna merah. Warna *Cyan* digabungkan dengan warna warna *Magenta* akan menghasilkan warna biru. Proses penggabungan warna ini ditunjukkan pada Gambar 2.2



Gambar 2.2 Hasil Penggabungan Model Warna CMY

(Sumber: www.stodioo.com)

2.2.3 Model Warna *Hue, Saturation, Intensity (HSI)*

Model warna HSI adalah model warna yang menggunakan nilai *Hue*, *Saturation*, dan *Intensity* dalam persebaran warna yang ada pada citra. Model warna HSI adalah hasil peningkatan dari model warna RGB di mana model warna HSI mempunyai kelebihan hampir menyerupai dengan warna penglihatan manusia, memiliki nilai pencahayaan yang berdiri sendiri sehingga dapat mengatasi citra yang mengalami perbedaan pencahayaan (Chien & Tseng, 2011) dan ideal untuk pengembangan algoritma pemrosesan citra yang berdasarkan deskripsi warna (Woods & Gonzalez, 2007). Komponen I pada model warna HSI mewakili intensitas cahaya pada citra, komponen H pada model warna HSI mewakili kejernihan warna pada citra, dan komponen S pada model warna HSI mewakili saturasi atau nilai warna yang menyerap warna putih pada citra. Untuk menkonversi model warna RGB menjadi model warna HSI dapat menggunakan fungsi berikut.

$$I = \frac{1}{3} (R + G + B) \quad (2.2)$$

$$H = \cos^{-1} \left\{ \frac{\frac{1}{2} [(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{\frac{1}{2}}} \right\} \quad (2.3)$$

$$S = 1 - \frac{3}{(R + G + B)} [\min(R, G, B)] \quad (2.4)$$

Keterangan:

H = Nilai warna H dari citra HSI

S = Nilai warna S dari citra HSI

I = Nilai warna I dari citra HSI

R = Nilai warna R dari citra RGB

G = Nilai warna G dari citra RGB

B = Nilai warna B dari citra RGB

2.3 Ekstraksi Fitur Warna

Ekstra fitur warna adalah salah satu metode untuk mengekstraksi nilai-nilai warna pada *pixel* yang ada pada gambar, sesuai dengan model warna dan metode ekstraksi fitur warna yang digunakan.

2.3.1 Color Moment

Color Moment adalah metode yang mengukur pendistribusian warna pada citra untuk digunakan mengkarakterisasikan warna pada citra yang digunakan (Kusuma, Pawening, & Dijaya, 2016). Metode *Color Moment* menggunakan nilai *Mean*, standar deviasi, dan *Skewness* sebagai komponen untuk mengkarakterisasi karena bersifat efisien dan efektif untuk distribusi warna pada citra (Soman, Ghorpade, & Sonone, 2012). Untuk penghitungan nilai *Mean*, standar deviasi, dan *Skewness* pada citra yang memiliki piksel dengan ukuran $A \times B$ dapat menggunakan fungsi berikut.

$$\text{Mean} = \bar{x}_i = \frac{\sum_{j=1}^{A \cdot B} X_{ij}}{A \times B} \quad (2.5)$$

$$\text{Standar deviasi} = \sigma_i = \sqrt{\frac{1}{A \times B} \sum_{j=1}^{A \cdot B} (X_{ij} - \bar{x}_i)^2} \quad (2.6)$$

$$\text{Skewness} = S_i = \sqrt{\frac{1}{A \times B} \sum_{j=1}^{A \cdot B} (X_{ij} - \bar{x}_i)^3} \quad (2.7)$$

Keterangan:

A = Panjang pixel citra

B = Lebar pixel citra

X_{ij} = Nilai dari piksel pada A dan B

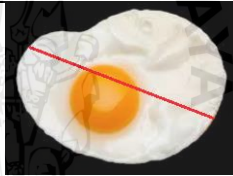
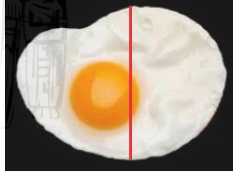
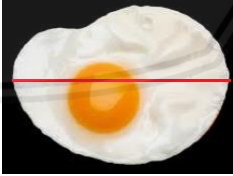
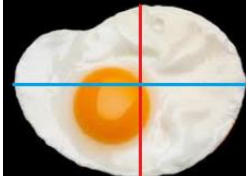
2.4 Ekstraksi Fitur Bentuk

Ekstra fitur bentuk adalah salah satu metode untuk mengekstraksi nilai-nilai bentuk yang ada pada citra berdasarkan metode yang akan digunakan.

2.4.1 Morphological Shape Descriptors

Metode *Morphological Shape Descriptors* adalah metode yang menganalisis bentuk geometris dasar pada objek seperti diameter objek, luas objek, panjang maksimal dan minimal sumbu pada objek, dan daerah *centroid* pada objek (Wäldchen & Mäder, 2017). Selain itu, *Morphological Shape Descriptors* dapat menggunakan beberapa deskriptor dasar seperti rasio, *Roundness*, *Compactness*. Pada deskriptor dasar terdapat rasio, rasio pada *Morphological Shape Descriptors* dapat digunakan menjadi deskriptor bentuk karena rasio mempunyai proses penghitungan yang sederhana dan mempunyai nilai yang invarian terhadap proses *translation*, rotasi, dan *scaling* membuat rasio tetap bernilai baik walaupun dengan representasi objek yang berbeda. Berikut adalah deskriptor yang ada pada *Morphological Shape Descriptors*.

Tabel 2.1 Deskriptor MSD Beserta Deskripsinya

Deskriptor	Deskripsi	Pictogram	Rumus
Diameter	Jarak terpanjang antara 2 titik pada margin objek		D
Sumbu Panjang	Panjang garis yang menghubungkan bagian atas dan bagian bawah objek		L
Sumbu Lebar	Panjang garis yang tegak lurus dengan sumbu panjang		W
Rasio aspek	Rasio dari sumbu lebar dan sumbu panjang, digunakan untuk menjelaskan seberapa lebar atau panjang objek		$AR = \frac{L}{W}$

2.5 Metode Klasifikasi

Metode klasifikasi adalah metode yang digunakan untuk mengelompokkan data berdasarkan ciri-ciri atau fitur yang secara analitis. Metode ini digunakan untuk memanfaatkan nilai-nilai fitur yang telah diekstraksi pada metode ekstraksi warna dan bentuk.

2.5.1 Klasifikasi *Naive Bayes*

Naive Bayes adalah salah satu metode klasifikasi yang mempunyai proses penghitungan sederhana (Taheri & Mammadov, 2013). Pada metode klasifikasi *Naive Bayes* menganggap efek dari nilai suatu kelas bersifat bebas dari nilai atribut lain. *Naive Bayes* mempunyai ciri utama kondisi dari suatu variabel memiliki independensi yang memiliki asumsi sangat kuat. Jika pada teorema Bayes mempunyai kejadian yang berbeda (X dan Y) maka teorema Bayes akan menjadi seperti persamaan 2.8.

$$P(X|Y) = \frac{P(Y|X) \cdot P(X)}{P(Y)} \quad (2.8)$$

Keterangan:

$P(X|Y)$ = Probabilitas posterior

$P(Y|X)$ = Nilai likelihood

$P(X)$ = Probabilitas kelas prior

$P(Y)$ = Probabilitas kelas prediktor

Probabilitas X dengan syarat Y didapatkan dari probabilitas Y dengan syarat X dikali dengan peluang X dibagi dengan peluang Y.

Jika *Naive Bayes* memproses data yang bersifat kontinu, data tersebut harus diasumsikan bahwa nilai-nilai berhubungan dengan masing-masing kelasnya yang didistribusikan menggunakan metode distribusi Gaussian. Distribusi dari nilai-nilai dideskripsikan dengan dua nilai yaitu mean dan variansi pada kelas y_j , peluang kelas X_i . Mean adalah nilai rata-rata diperoleh atribut data kontinu. Fungsi dari Mean dirumuskan pada persamaan 2.9.

$$\text{Mean} = \frac{\sum_{i=1}^n x_i}{n} \quad (2.9)$$

Sedangkan untuk fungsi standar deviasi pada persamaan 2.10.

$$\text{Standar Deviasi} = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \mu)^2} \quad (2.10)$$

Dari *Mean* dan variansi akan menghasilkan fungsi densitas untuk mendeskripsikan probabilitas relatif. Fungsi densitas dirumuskan pada persamaan 2.11.

$$Densitas = P(X|Y) = \varphi_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2.11)$$

Keterangan:

X_i = Nilai data ke-i

n = Jumlah data

μ = Nilai mean data

σ = Nilai standar deviasi data

Sedangkan jika metode *Naive Bayes* yang memiliki data diskrit mencari nilai likelihood suatu fitur dari masing-masing kelas dan kemudian dikalikan dengan prior dari setiap kelas.

2.5.2 Fungsi LogSumExp

Fungsi *LogSumExp* adalah fungsi yang dapat digunakan untuk mengatasi keadaan *numerical underflow*. *Numerical Flow* adalah keadaan di mana nilai dari hasil perkalian densitas atau proses pencarian *likelihood* tiap fitur memiliki nilai 10^{-x} sangat rendah sehingga hasil perkaliannya dibulatkan menjadi 0. Agar nilai *likelihood* tidak menjadi 0 maka dapat menggunakan fungsi *LogSumExp* (Nielsen & Sun, 2016). Fungsi dari *LogSumExp* dirumuskan pada Fungsi 2.12.

$$\begin{aligned} \text{LogSumExp} &= \log \sum_c e^{b_c} = \log \left[\left(\sum_c e^{b_c} \right) e^{-B} e^B \right] \\ &= \log \left[\left(\sum_c e^{b_c - B} \right) e^B \right] \\ &= \log \left[\left(\sum_c e^{b_c - B} \right) + B \right] \end{aligned} \quad (2.12)$$

Keterangan:

c = indeks data

b = nilai data

e = nilai data pada indeks

B = nilai data dengan nilai maksimum

2.6 Metode Evaluasi

Proses evaluasi adalah suatu proses yang dilakukan untuk menentukan nilai dari suatu objek yang sudah ada agar dapat mengetahui sejauh mana tingkat keberhasilan dari hasil yang didapatkan oleh objek tersebut.



2.6.1 Akurasi

Salah satu metode evaluasi yang dapat digunakan dalam melakukan proses pengukuran hasil klasifikasi adalah metode akurasi. Akurasi adalah selisih antara jumlah data uji dan jumlah data yang memiliki hasil yang tidak tepat dibagi dengan jumlah data uji (Riyanto, 2014) seperti yang dirumuskan pada fungsi 2.12.

$$Akurasi = \frac{K1 - K2}{K1} \times 100\% \quad (2.12)$$

Keterangan:

K1: Jumlah data uji

K2: Jumlah data tidak tepat



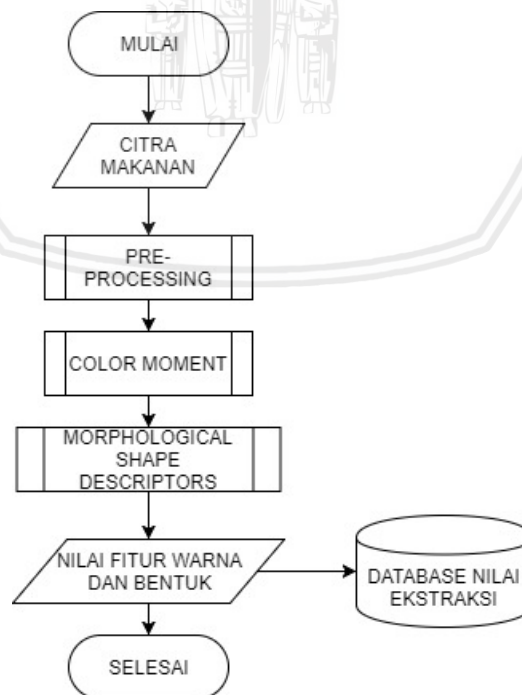
BAB 3 METODOLOGI

3.1 Tipe Penelitian

Tipe penelitian yang akan diterapkan pada penelitian ini adalah tipe penelitian non-implementatif analitik. Penelitian dengan tipe non-implementatif analitik mempunyai tujuan untuk mencari tahu hasil dari relasi di antara elemen di dalam suatu objek yang nantinya dapat digunakan sebagai acuan untuk pengambilan keputusan dalam suatu peristiwa. Setelah gambar telah dimasukkan tahap selanjutnya adalah melakukan proses *preprocessing* agar proses yang diolah fokus pada objek yang ada di dalam citra makanan. Setelah objek yang ada pada citra makanan dideteksi, proses selanjutnya adalah mengekstraksi fitur nilai warna dengan metode *color moment* dan mengekstraksi fitur nilai bentuk dengan metode *morphological shape descriptors*.

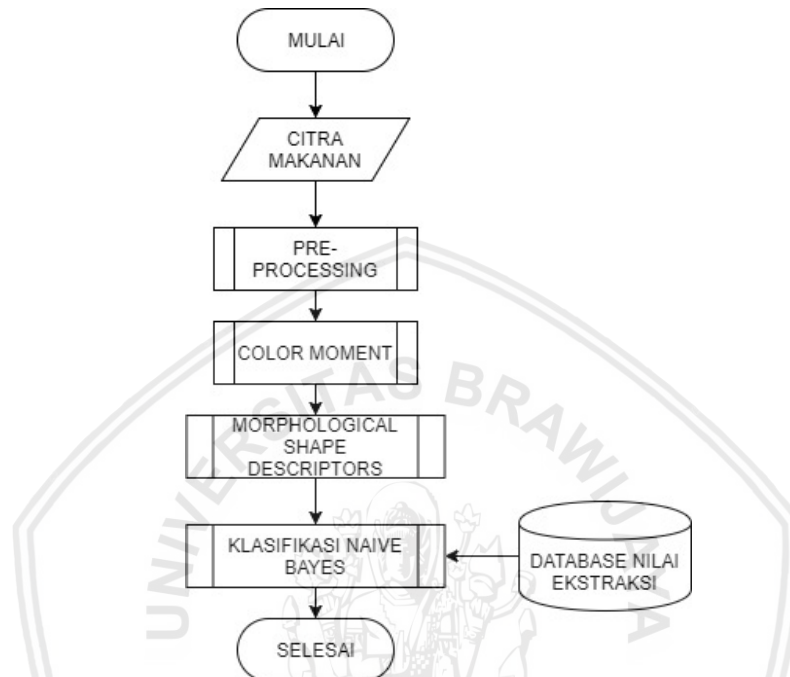
3.2 Metode Penelitian

Metode penelitian yang digunakan di dalam penelitian ini dibagi menjadi dua proses, yaitu proses pelatihan (*training*) dan pengujian (*testing*). Proses pelatihan diawali dengan memasukkan data citra makanan yang nilai-nilai fiturnya akan diekstraksi warna dan bentuknya. Setelah hasil dari ekstraksi fitur warna dan bentuk telah didapatkan, nilai-nilai tersebut disimpan ke dalam *database* agar dapat digunakan untuk proses pengujian. Proses pelatihan ini digambarkan pada Gambar 3.1.



Gambar 3.1 Proses Pelatihan

Proses pengujian diawali dengan memasukkan data citra yang diklasifikasikan. Proses pertama adalah melakukan *preprocessing* pada citra, setelah itu citra akan diekstraksi fitur warna dan bentuknya. Nilai-nilai hasil ekstraksi fitur kemudian akan diklasifikasikan dengan metode klasifikasi *naive bayes*, dengan menghitung probabilitas dari citra yang ingin diklasifikasi dengan masing-masing kelas. Proses pelatihan akan digambarkan pada Gambar 3.2.



Gambar 3.2 Proses Pengujian

3.3 Lokasi Penelitian

Lokasi tempat berlangsungnya penelitian dilakukan di Gedung F lantai 9 di ruangan F9.3 Fakultas Ilmu Komputer, Universitas Brawijaya, Malang.

3.4 Partisipan Penelitian

Partisipan yang mengikuti penelitian adalah mahasiswa yang mengambil mata kuliah kelas induksi riset F yang diajar oleh Ibu Yuita Arum Sari S.Kom, M.Kom. Penelitian ini menggunakan makanan sebagai objek penelitian, yang terdiri dari 30 kelas.

3.5 Peralatan Pendukung

Peralatan pendukung yang digunakan pada proses penelitian dalam proses pengumpulan data dan pemrograman sistem adalah:

1. Alat pengambilan data
 - *Smartphone* Samsung Galaxy S8
2. Alat pemrograman sistem
 - Spesifikasi *hardware*

- Laptop dengan *processor* Intel(R) Core (TM)i7-4750HQ CPU @ 2.00GHz (8 CPUs), ~2.0GHz, memori dengan kapasitas 4GB DDR3, dan kapasitas penyimpanan 1TB.
- Spesifikasi *software*
 - Microsoft Windows 10
 - Python 3.6

3.6 Teknik Pengumpulan Data

Proses pengumpulan data citra yang akan digunakan adalah mengambil data citra makanan menggunakan *smartphone*. Data citra makanan yang diambil dibagi menjadi 2 kelompok yaitu, kelompok makanan yang sudah dimakan dan kelompok makanan yang belum dimakan. Kelompok makanan yang belum dimakan terdiri dari citra yang makanannya masih berbentuk utuh. Sedangkan kelompok makanan yang sudah dimakan terdiri dari citra makanan yang telah dimakan sebanyak $\frac{1}{4}$, $\frac{1}{2}$, dan $\frac{3}{4}$ bagian makanan. Kondisi pada saat pengambilan data penelitian adalah:

1. Makanan yang digunakan untuk menjadi dataset memiliki bentuk padat dan terdiri dari x jenis.
2. Tingkat pencahayaan yang digunakan pada saat pengambilan dataset menggunakan cahaya matahari pada pukul 10.00 – 13.00 WIB di Universitas Brawijaya.
3. Jenis *handphone* yang digunakan adalah Samsung Galaxy S8 dengan ukuran kamera 12 megapiksel dan f/1.9.
4. Posisi pengambilan citra makanan adalah tegak lurus.

Data yang dikumpulkan setelah melewati proses *preprocessing* akan digambarkan pada Gambar 3.3.



Gambar 3.3 Citra Hasil *Preprocessing*

5. Data yang digunakan dibagi ke dalam dua jenis yaitu data latih dan data uji. Jumlah data citra makanan yang digunakan terdapat pada Tabel 3.1 untuk data latih dan Tabel 3.2 untuk data uji

Tabel 3.1 Daftar Citra Data Latih

Kelas Makanan	Jumlah Data
001 (Donat)	20 Citra
002 (Roti Gandum)	21 Citra
004 (Mie Goreng)	20 Citra
007 (Telur Dadar)	21 Citra
009 (Rendang)	20 Citra
013 (Selada)	20 Citra
016 (Stroberi)	20 Citra
017 (Pisang Hijau)	20 Citra
021 (Nasi Kuning)	20 Citra
025 (Mie Soba)	20 Citra

Tabel 3.2 Daftar Citra Data Uji

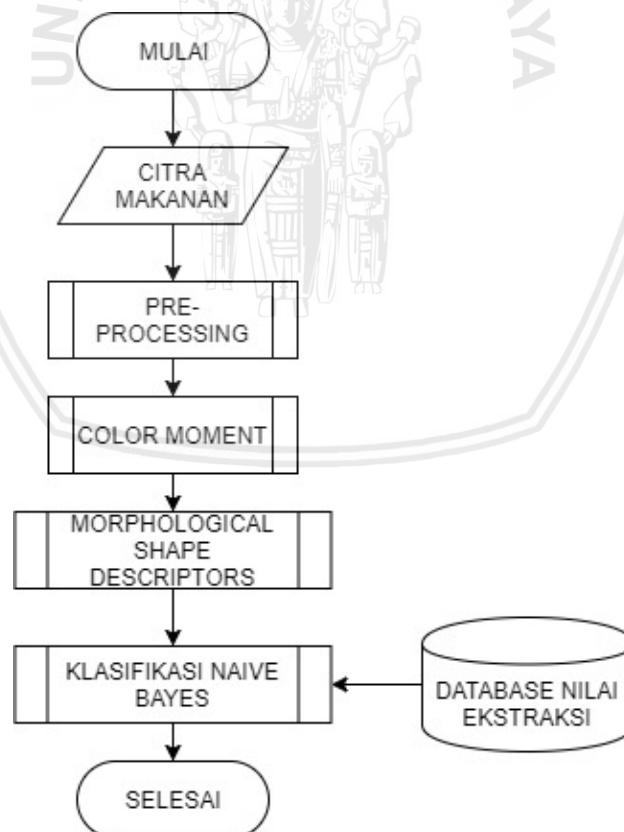
Kelas Makanan	Jumlah Data
001 (Donat)	10 Citra
002 (Roti Gandum)	10 Citra
004 (Mie Goreng)	10 Citra
007 (Telur Dadar)	10 Citra
009 (Rendang)	10 Citra
013 (Selada)	10 Citra
016 (Stroberi)	10 Citra
017 (Pisang Hijau)	10 Citra
021 (Nasi Kuning)	10 Citra
025 (Mie Soba)	10 Citra

BAB 4 PERANCANGAN

Di dalam bab ini akan membahas penerapan perancangan pada sistem pengenalan citra jenis makanan menggunakan klasifikasi *Naive Bayes* dengan ekstraksi fitur HSI *color moment* dan *morphological shape descriptors* sampai dengan menampilkan hasil dari proses klasifikasi.

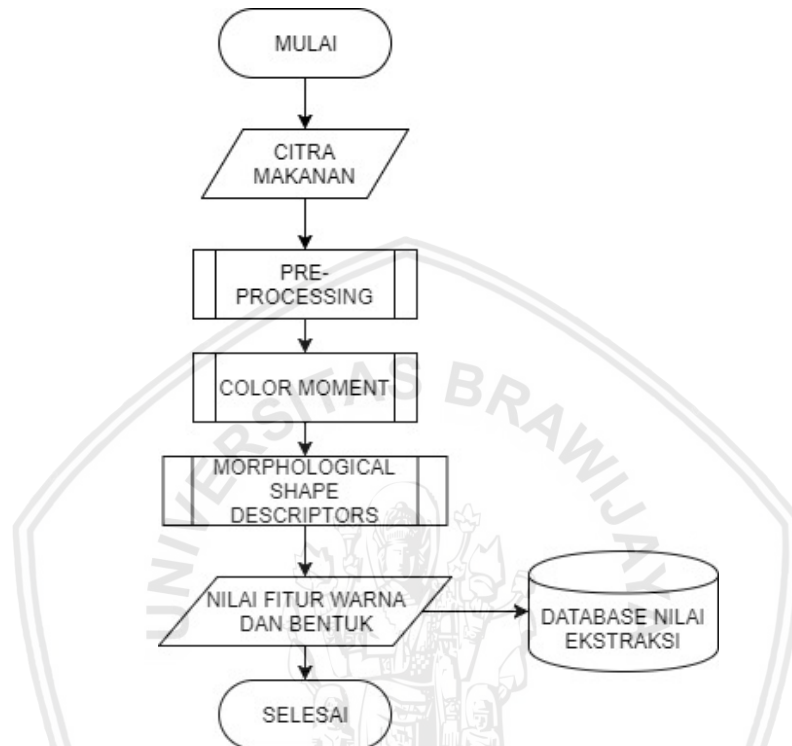
4.1 Perancangan Algoritme Pada Proses Implementasi

Proses yang ada di dalam proses pengimplementasian dibagi menjadi empat proses yaitu proses *preprocessing*, ekstraksi fitur HSI *color moment*, ekstraksi fitur *morphological shape descriptors*, dan klasifikasi *Naive Bayes*. Di dalam tahap *preprocessing*, citra dari makanan akan diolah sesuai dengan kebutuhan yang diperlukan untuk proses ekstraksi fitur warna dan bentuk. Pada tahap selanjutnya pada proses ekstraksi warna menggunakan HSI *Color Moment* di mana fitur yang akan dihasilkan adalah nilai *Mean*, Standar deviasi, dan *skewness* untuk masing-masing *channel* warna yaitu *Hue*, *Saturation*, dan *Intensity*. Pada tahap ekstraksi fitur bentuk akan menghasilkan fitur diameter, sumbu lebar, sumbu tinggi, rasio aspek dan *rectangularity*. Proses-proses tersebut digambarkan pada Gambar 4.1.



Gambar 4.1 Alur Proses Pengujian

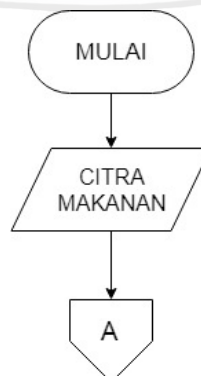
Proses yang akan dilakukan suatu proses pelatihan data citra yang akan digunakan untuk proses klasifikasi. Algoritme yang digunakan pada proses pelatihan adalah menyimpan nilai-nilai hasil ekstraksi fitur warna dan bentuk yang didapatkan dari metode *color moment* dan *morphological shape descriptors*. Selanjutnya hasil tersebut disimpan pada *microsoft excel* untuk diolah pada proses utama. Proses ini digambarkan pada Gambar 4.2.

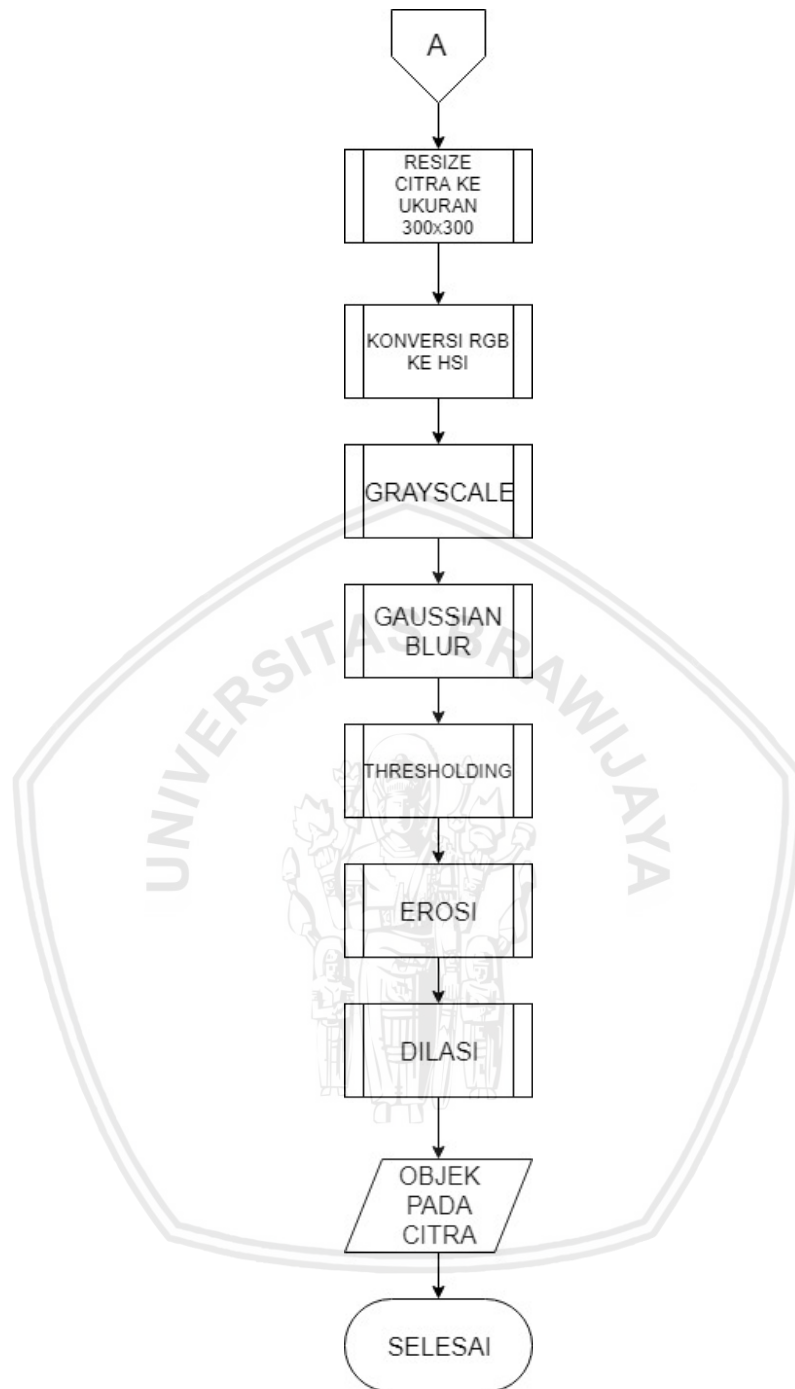


Gambar 4.2 Alur Proses Pengambilan Data Latih

4.1.1 Proses *Preprocessing*

Proses *preprocessing* adalah proses yang dilakukan untuk mengolah citra yang akan digunakan agar sesuai dengan kebutuhan yang diperlukan pada proses selanjutnya. Alur dari proses *preprocessing* digambarkan pada Gambar 4.3.

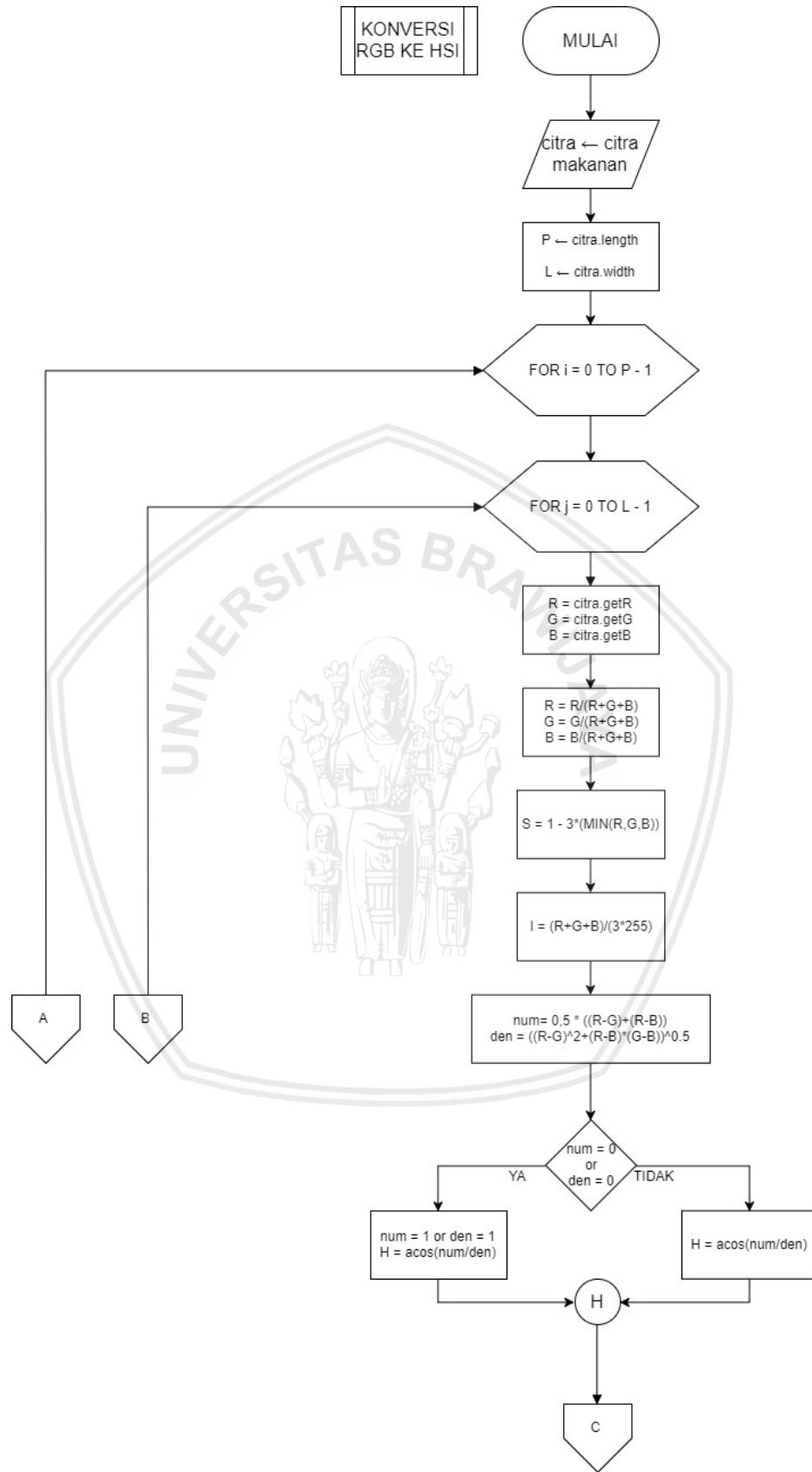


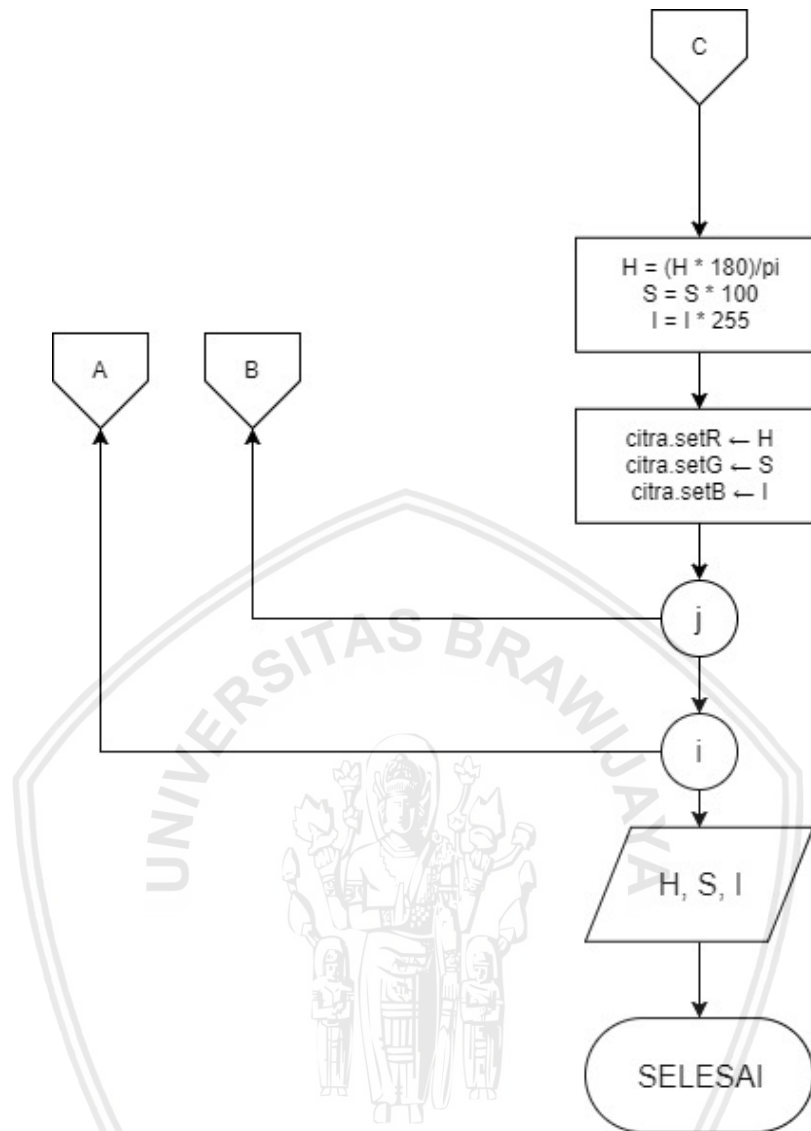


Gambar 4.3 Alur Proses *Preprocessing*

4.1.2 Proses Konversi RGB ke HSI

Proses konversi dilakukan untuk mengkonversi gambar dengan model warna *Red, Green, Blue* (RGB) menjadi *Hue, Saturation, Intensity* (HSI). Proses dari konversi RGB ke HSI digambarkan pada Gambar 4.5.

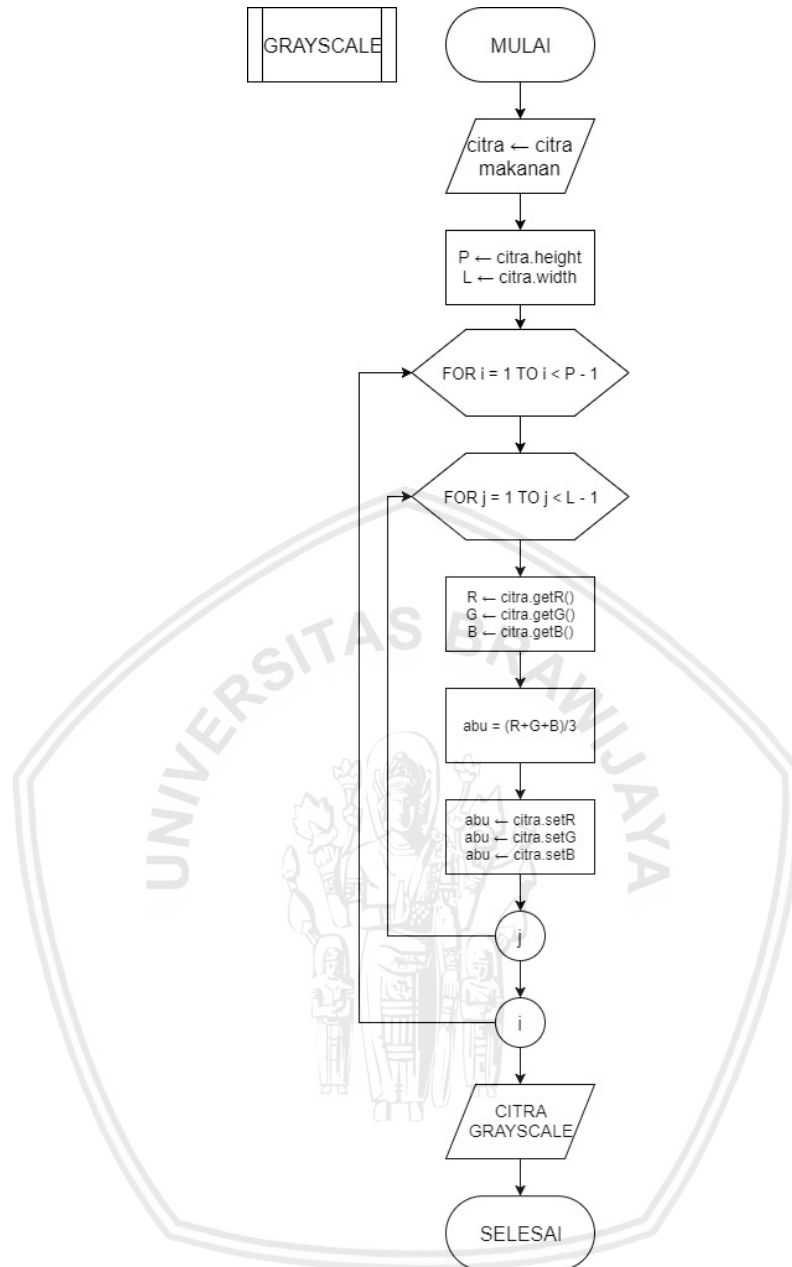




Gambar 4.4 Alur Proses Konversi RGB Ke HSI

4.1.3 Proses *Grayscale*

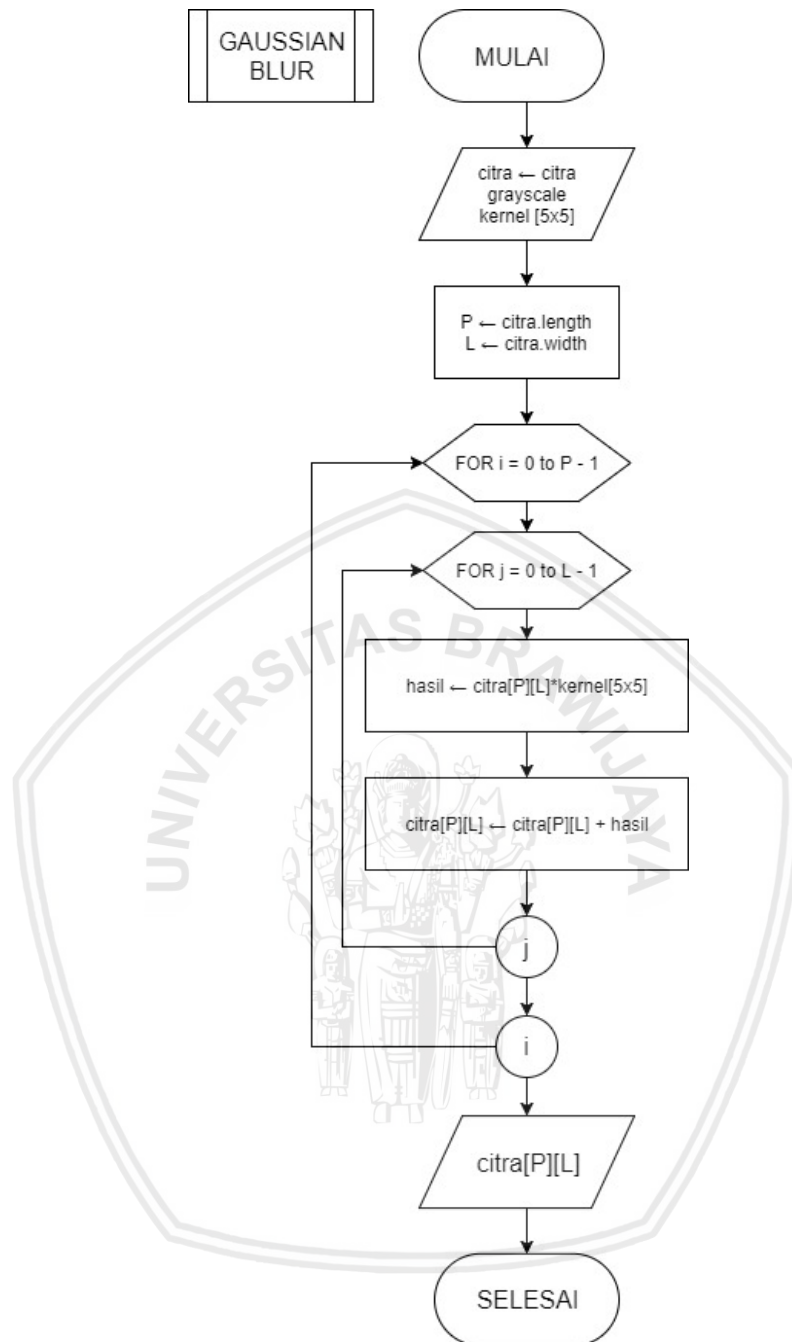
Proses *grayscale* adalah proses mengubah gambar yang berwarna merah, hijau, dan biru menjadi berwarna abu-abu. Proses mengubah warna ini dilakukan dengan menambahkan nilai piksel dari *channel* warna merah, hijau, dan biru lalu membagi hasilnya dengan tiga. Hasil dari pembagian ini akan menghasilkan *channel* berwarna abu-abu. Proses dari *grayscale* digambarkan pada Gambar 4.4.



Gambar 4.5 Alur Proses *Grayscale*

4.1.4 Proses *Gaussian Blur*

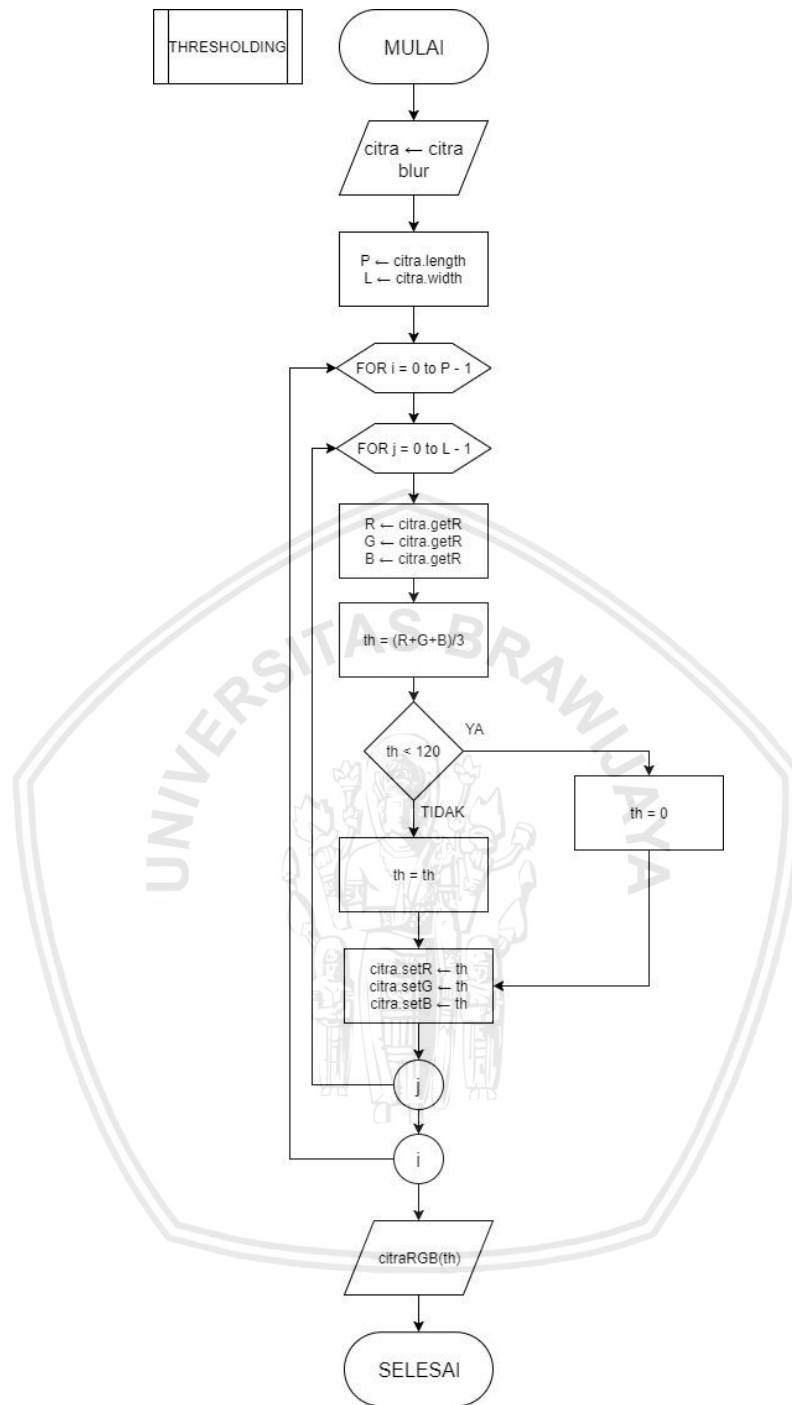
Proses *gaussian blur* dilakukan untuk memberikan efek *blur* atau keburaman pada citra. Tujuan mengubah citra menjadi citra yang mempunyai efek keburaman adalah agar pada saat proses *thresholding* nanti dapat menghasilkan citra yang lebih baik lagi. Proses *gaussian blur* akan digambarkan pada Gambar 4.6.



Gambar 4.6 Alur Proses *Gaussian Blur*

4.1.5 Proses *Thresholding*

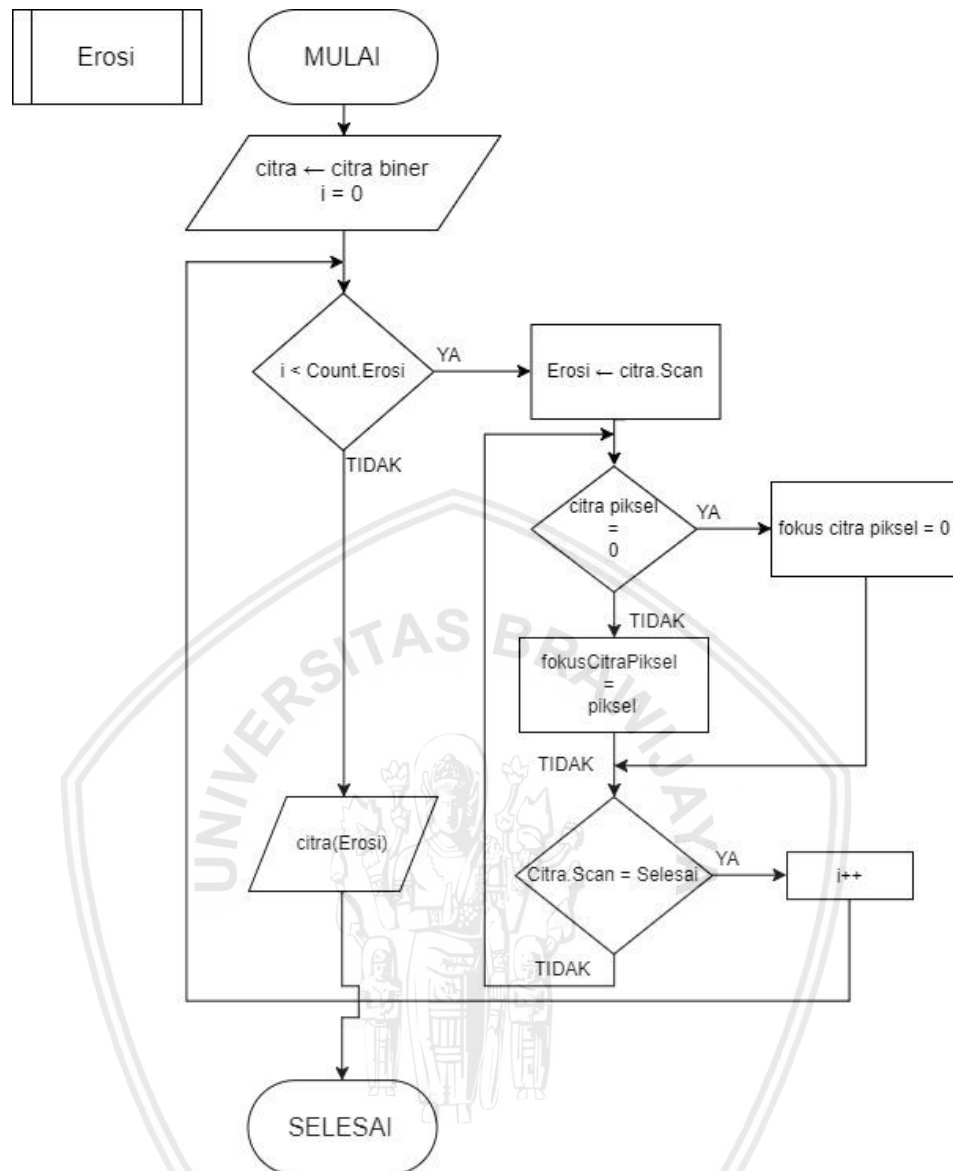
Proses *thresholding* dilakukan untuk mengubah citra yang memiliki nilai warna abu-abu menjadi citra dengan nilai biner. Tujuan mengubah nilai keabuan menjadi nilai biner adalah untuk membagi citra menjadi ke beberapa bagian seperti bagian objek dan latar belakang dari suatu citra. Proses *thresholding* akan digambarkan pada Gambar 4.7.



Gambar 4.7 Alur Proses *Thresholding*

4.1.6 Proses Erosi

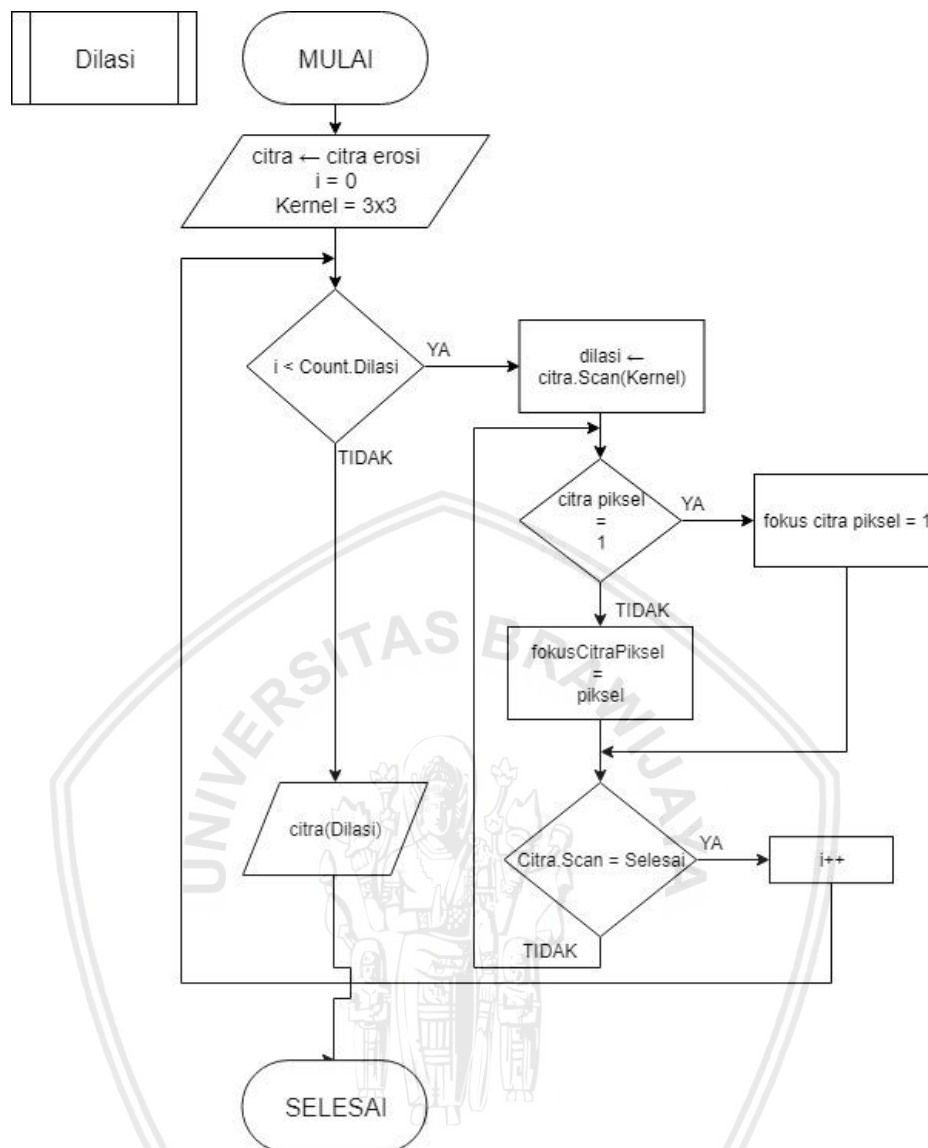
Proses erosi dilakukan untuk mengurangi nilai dan lapisan pada sekeliling objek yang ada pada citra sehingga membuat objek yang ada pada citra semakin kecil. Proses erosi akan digambarkan pada Gambar 4.8.



Gambar 4.8 Alur Proses Erosi

4.1.7 Proses Dilasi

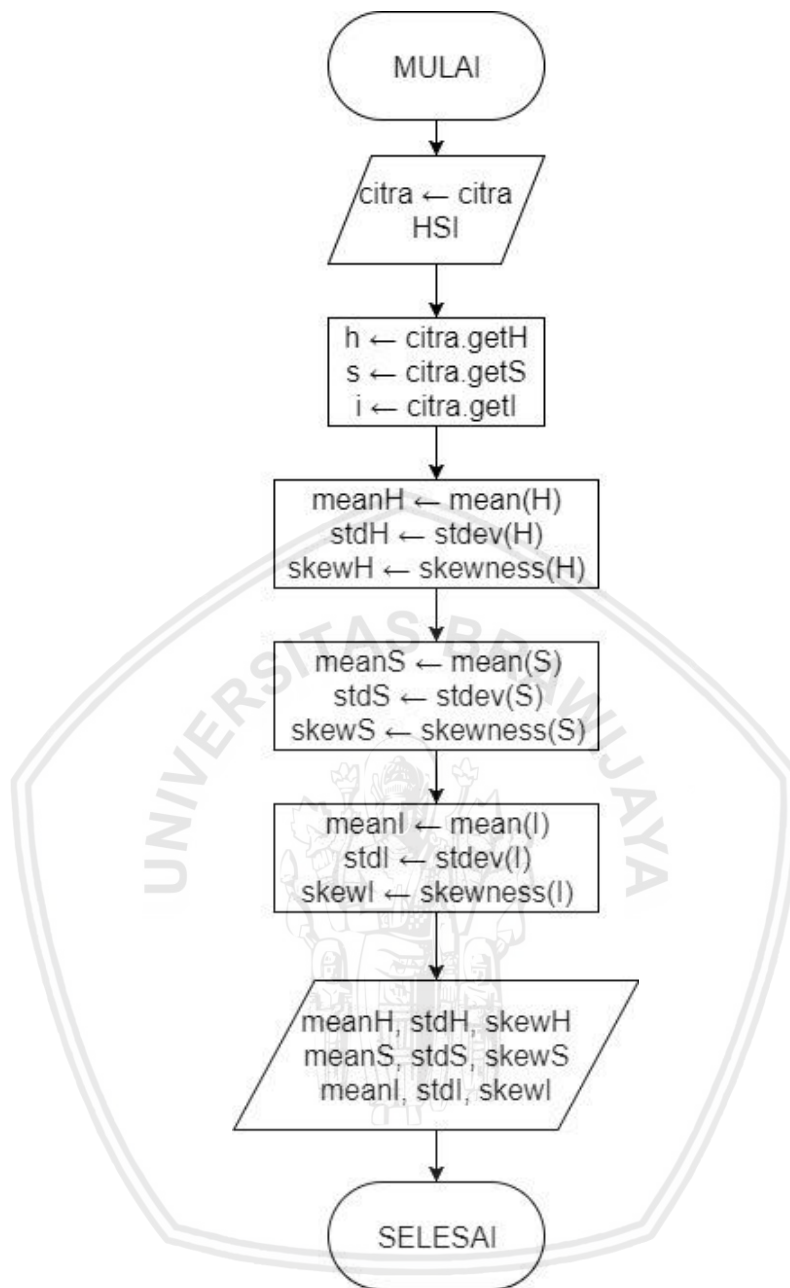
Proses dilasi dilakukan untuk menambahkan nilai dan lapisan pada sekeliling objek yang ada pada citra sehingga membuat objek yang ada pada citra semakin besar. Proses dilasi akan digambarkan pada Gambar 4.9.



Gambar 4.9 Alur Proses Dilasi

4.1.8 Proses *Color Moment*

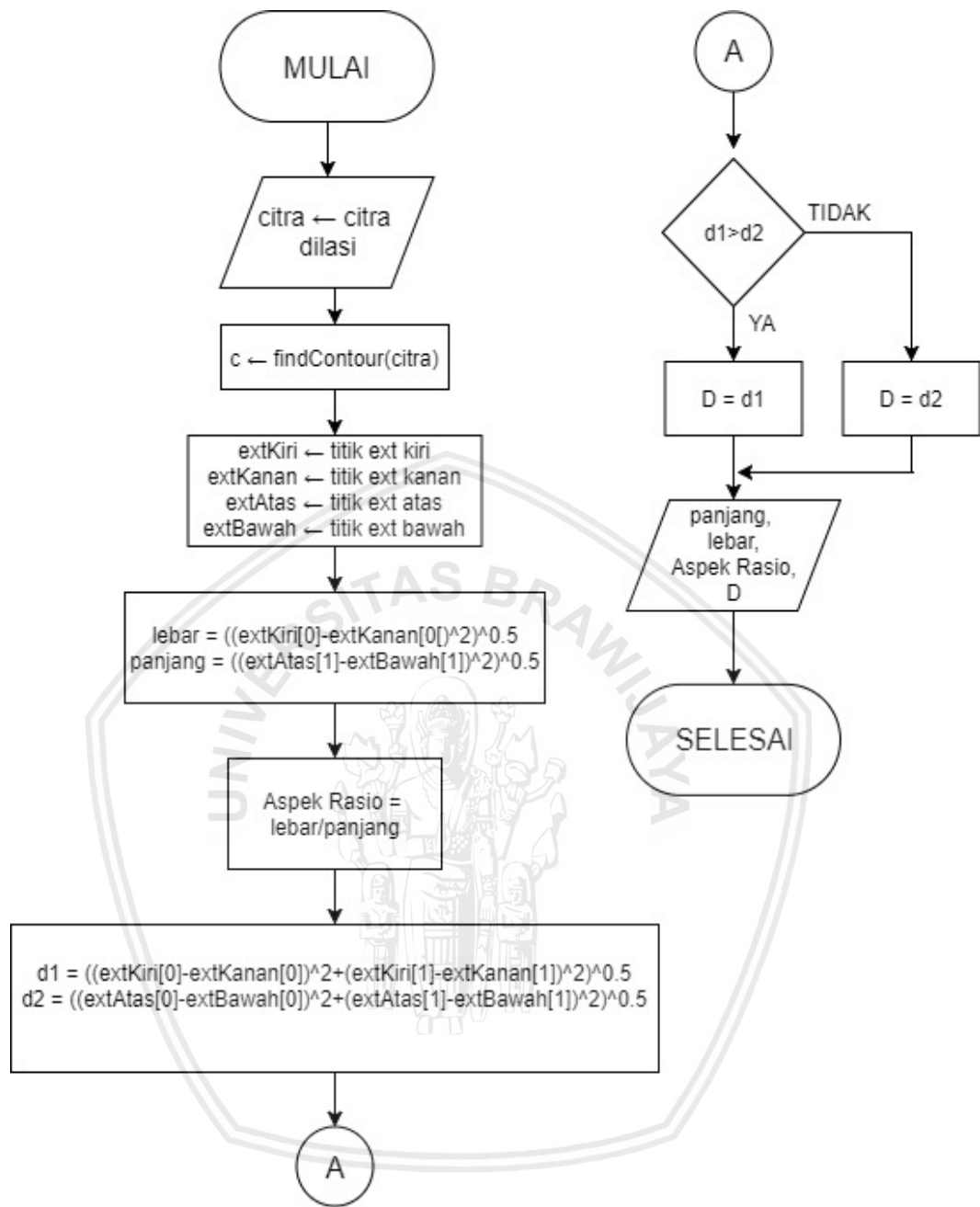
Proses penghitungan *color moment* dilakukan untuk menghasilkan fitur-fitur seperti nilai *mean*, nilai standar deviasi, dan nilai *skewness* dari masing-masing *channel* dari model warna H, S, dan I. Proses penghitungan *color moment* akan digambarkan pada Gambar 4.10.



Gambar 4.10 Alur Proses Klasifikasi Color Moment

4.1.9 Proses *Morphological Shape Descriptors*

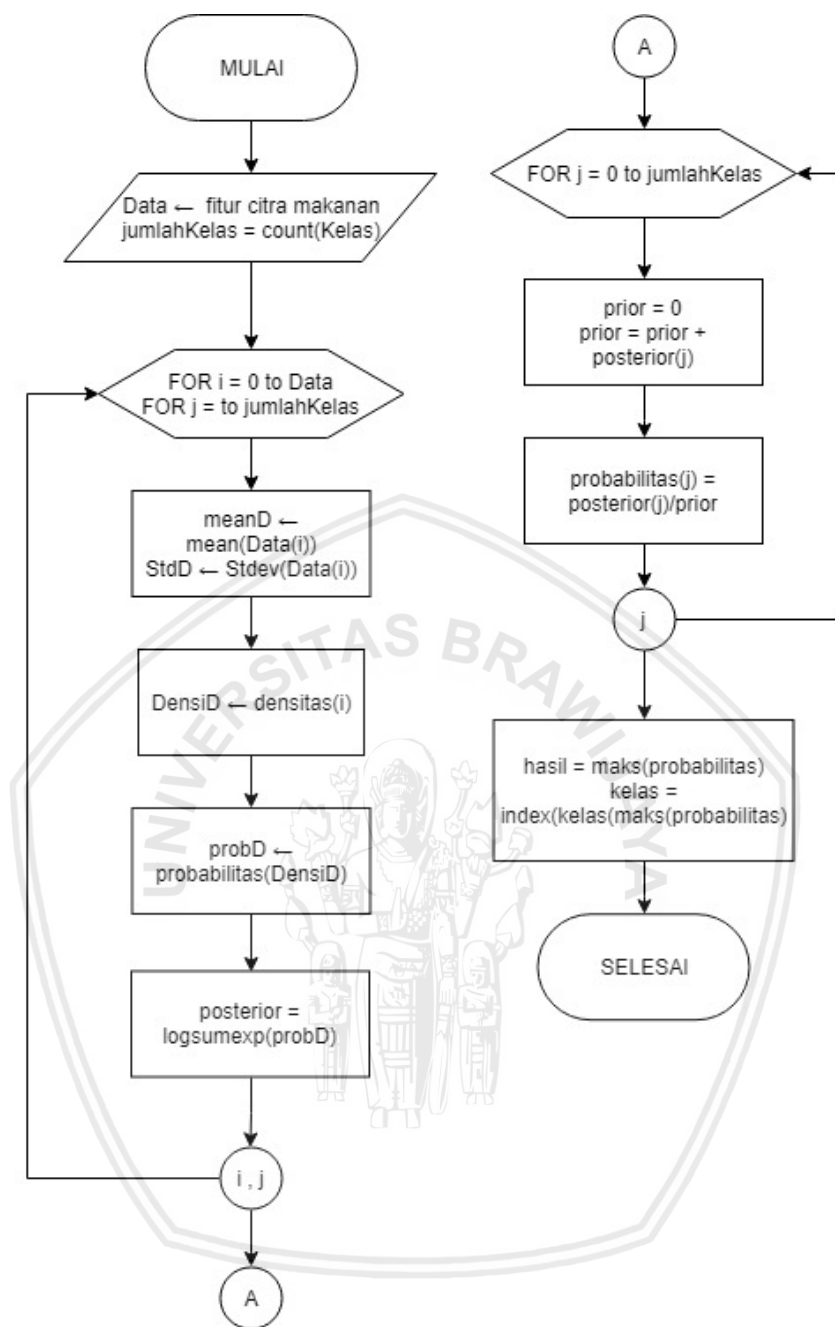
Proses *morphological shape descriptors* dilakukan untuk mengekstraksi bentuk dari citra. Fitur-fitur yang akan diekstraksi dari citra adalah fitur diameter, sumbu panjang, sumbu lebar, rasio aspek, dan *rectangularity*. Proses *morphological shape descriptors* akan digambarkan pada Gambar 4.11.



Gambar 4.11 Alur Proses *Morphological Shape Descriptors*

4.1.10 Proses Klasifikasi *Naive Bayes*

Proses klasifikasi *Naive Bayes* dilakukan untuk mengklasifikasikan fitur-fitur yang telah diekstraksi sebelumnya ke dalam suatu kelas. Proses klasifikasi *Naive Bayes* akan digambarkan pada Gambar 4.12.



Gambar 4.12 Alur Proses Klasifikasi Naive Bayes

4.2 Manualisasi Penghitungan

Proses manualisasi penghitungan dilakukan dengan menggunakan citra yang telah diterapkan proses *preprocessing* yaitu proses *grayscale*, *gaussian blur*, *thresholding*, erosi, dan dilasi. Proses manualisasi penghitungan digunakan untuk menjelaskan proses pengekstraksian fitur-fitur warna dan bentuk. Proses penghitungan pada bab ini menggunakan salah satu sampel data citra dari database citra. Citra yang akan digunakan memiliki ukuran 400 x 400 piksel seperti pada Gambar 4.13. Proses manualisasi penghitungan yang akan diproses hanya pada ukuran 3 x 3 piksel yang ada pada citra sampel.



Gambar 4.13 Sampel Data Citra Makanan

4.2.1 Penghitungan Konversi model warna RGB ke model warna HSI

Langkah 1:

Mengambil nilai R, G, dan B pada citra

Channel Warna	Nilai Piksel (3x3)		
R	38	78	60
	98	35	32
	49	25	51
G	94	168	185
	72	37	169
	10	96	70
B	13	14	163
	81	62	106
	168	201	193

Langkah 2:

Menghitung nilai rata-rata dan jumlah dari R, G, dan B sesuai dengan koordinat pikselnya

$$Jumlah = 38 + 94 + 13 = 145$$

$$Rata - rata = \frac{38 + 94 + 13}{3} = 48,33$$

Langkah 3:

Menormalisasi nilai dari R, G, dan B

$$R = \frac{38}{145} = 0,262 \quad G = \frac{94}{145} = 0,648 \quad B = \frac{13}{145} = 0,089$$

Langkah 4:

Menghitung nilai H

$$H = \cos^{-1} \left\{ \frac{\frac{1}{2} [(0,262 - 0,648) + (0,262 - 0,089)]}{[(0,262 - 0,648)^2 + (0,262 - 0,089)(0,648 - 0,089)]^{\frac{1}{2}}} \right\}$$

$$H = 1,787$$

Langkah 5:

Menghitung nilai S

$$S = 1 - \frac{3}{(0,262 + 0,648 + 0,089)} [\min(0,262, 0,648, 0,089)]$$

$$S = 0,733$$

Langkah 6:

Menghitung nilai I

$$I = \frac{1}{3} (0,262 + 0,648 + 0,089)$$

$$I = 0,332$$

Langkah 7:

Menormalisasikan kembali nilai H, S, dan I dan membulatkan hasilnya

$$H = 1,787 \times \frac{180}{\pi} = 102$$

$$S = 0,733 \times 100 = 73$$

$$I = 0,332 \times 255 = 84$$

Langkah 8:

Menginputkan kembali nilai H, S, dan I ke dalam citra

Channel Warna	Nilai Piksel (3x3)		
H	102		
S	73		
I	84		



4.2.2 Penghitungan *Color Moment*

Langkah 1:

Mengambil nilai H, S, dan I pada citra

Channel Warna	Nilai Piksel (3x3)		
H	25	253	0
	24	29	26
	0	28	30
S	78	100	100
	67	68	69
	100	58	73
I	55	5	1
	86	81	68
	0	97	76

Langkah 2:

Menghitung nilai *mean* dari masing-masing nilai H, S, dan I dengan fungsi yang ada pada 2.5

$$\text{Mean } H = \frac{25 + 253 + 0 + 24 + 29 + \dots + 30}{9} = 46,111$$

$$\text{Mean } S = \frac{78 + 100 + 100 + 67 + 68 + \dots + 73}{9} = 79,222$$

$$\text{Mean } I = \frac{55 + 5 + 1 + 86 + 81 + \dots + 76}{9} = 52,111$$

Langkah 3:

Menghitung nilai standar deviasi dari masing-masing nilai H, S, dan I dengan fungsi yang ada pada 2.6

Standar Deviasi H

$$\begin{aligned} &= \sqrt{(25 - 46,111)^2 + (253 - 46,111)^2 + (0 - 46,111)^2 + \dots + (30 - 46,111)^2} \\ &= 78,482 \end{aligned}$$

Standar Deviasi S

$$\begin{aligned} &= \sqrt{(78 - 79,222)^2 + (100 - 79,222)^2 + (100 - 79,222)^2 + \dots + (73 - 79,222)^2} \\ &= 16,452 \end{aligned}$$

Standar Deviasi I

$$\begin{aligned} &= \sqrt{(55 - 52,111)^2 + (5 - 52,111)^2 + (1 - 52,111)^2 + \dots + (76 - 52,111)^2} \\ &= 39,333 \end{aligned}$$

Langkah 4:

Menghitung nilai *skewness* dari masing-masing nilai H, S, dan I dengan fungsi yang ada pada 2.7

Skewness H

$$= \sqrt{(25 - 46,111)^3 + (253 - 46,111)^3 + (0 - 46,111)^3 + \dots + (30 - 46,111)^3}$$

$$= 2,864$$

Skewness S

$$= \sqrt{(78 - 79,222)^3 + (100 - 79,222)^3 + (100 - 79,222)^3 + \dots + (73 - 79,222)^3}$$

$$= 0,461$$

Skewness I

$$= \sqrt{(55 - 52,111)^3 + (5 - 52,111)^3 + (1 - 52,111)^3 + \dots + (76 - 52,111)^3}$$

$$= -0,551$$

Langkah 5:

Masukkan hasil perhitungan *mean*, standar deviasi, dan *skewness* dari masing H, S, dan I seperti Tabel 4.1.

Tabel 4.1 Tabel Fitur *Color Moment*

No	Mean H	Mean S	Mean I	St. Dev H	St. Dev S	St. Dev I	Skew H	Skew S	Skew I
1	46,111	79,222	52,111	78,482	16,452	39,333	2,864	0,461	-0,551

4.2.3 Penghitungan *Morphological Shape Descriptors*

Langkah 1:

Melakukan pencarian kontur pada objek yang ada di dalam citra makanan

Langkah 2:

Menentukan titik ekstrim pada bagian kiri, kanan, atas, dan bawah

Koordinat titik kanan $(x, y) = (335, 196)$

Koordinat titik kiri $(x, y) = (63, 141)$

Koordinat titik bawah $(x, y) = (82, 320)$

Koordinat titik atas $(x, y) = (281, 95)$

Langkah 3:

Melakukan penghitungan lebar dengan cara

Sumbu lebar $= 335 - 63 = 272$

Langkah 4:

Melakukan penghitungan panjang dengan cara mengurangi titik atas dengan bawah



$$\text{Sumbu panjang} = 320 - 95 = 225$$

Langkah 5:

Melakukan penghitungan rasio aspek

$$\text{Aspek rasio} = \frac{225}{272} = 0,827$$

Langkah 6:

Melakukan penghitungan nilai diameter dengan cara mencari nilai maksimum d1 dan d2

$$D1 = \sqrt{(335 - 63)^2 + (196 - 141)^2}$$

$$D1 = 277,504$$

$$D2 = \sqrt{(82 - 281)^2 + (320 - 95)^2}$$

$$D2 = 300,376$$

$$\text{Diameter} = \text{Max}(D1, D2)$$

$$\text{Diameter} = D1 = 300,376$$

4.2.4 Penghitungan Klasifikasi Naive Bayes

Langkah 1:

Memasukkan nilai-nilai hasil ekstraksi fitur warna dan bentuk beserta kelasnya ke dalam tabel seperti pada Tabel 4.2.

Tabel 4.2 Fitur dan Kelas

No	MH	SDH	SH	MS	SDS	SS	MI
1	191,47	9,392	10,515	0,511	9,392	10,515	230,528
2	198,456	10,285	11,338	0,789	9,785	12,342	228,508
3	160,376	4,338	6,497	2,367	12,329	5,856	187,446
4	162,462	3,961	7,237	2,894	13,478	5,256	185,235
5	185,378	6,497	3,866	1,443	10,728	8,384	196,352
6	187,361	6,856	4,743	0,923	11,137	7,996	197,128

Tabel 4.2 Fitur dan Kelas (Lanjutan)

No	SDI	SI	LEBAR	TINGGI	AR	DIAMETER	KELAS
1	2,666	0,774	321	350	0,917	361,842	1
2	2,934	0,678	241	320	0,753	320,189	1
3	1,232	1,976	272	208	1,307	278,976	2

4	1,145	1,623	273	225	1,213	300,376	2
5	1,842	0,793	336	351	0,957	363,616	3
6	1,579	0,985	337	352	0,957	362,663	3

Langkah 2:

Menghitung nilai rata-rata dari masing-masing nilai fitur dari masing-masing kelas

Mean MH Kelas 1:

$$\text{Mean MH 1} = \frac{191,47 + 198,45}{2}$$

$$\text{Mean MH 1} = 194,963$$

Mean SDH Kelas 1:

$$\text{Mean SDH1} = \frac{9,392 + 10,285}{2}$$

$$\text{Mean SDH1} = 9,8385$$

Mean SH Kelas 1:

$$\text{Mean SH1} = \frac{10,515 + 11,338}{2}$$

$$\text{Mean SH1} = 10,9265$$

Mean MS Kelas 1:

$$\text{Mean MS1} = \frac{0,511 + 0,789}{2}$$

$$\text{Mean MS1} = 0,65$$

Mean SDS Kelas 1:

$$\text{Mean SDS1} = \frac{9,392 + 9,785}{2}$$

$$\text{Mean SDS1} = 9,5885$$

Mean SS Kelas 1:

$$\text{Mean SS1} = \frac{10,515 + 12,342}{2}$$

$$\text{Mean SS1} = 11,285$$

Mean MI Kelas 1:

$$\text{Mean MI1} = \frac{230,528 + 228,508}{2}$$

$$\text{Mean MI1} = 229,518$$

Mean SDI Kelas 1:

$$\text{Mean SDI1} = \frac{2,666 + 2,934}{2}$$

$$\text{Mean SDI1} = 2,8$$

Mean SI Kelas 1:

$$\text{Mean SI1} = \frac{0,774 + 0,678}{2}$$

$$\text{Mean SI1} = 0,726$$

Mean Lebar Kelas 1:

$$\text{Mean L1} = \frac{321 + 241}{2}$$

$$\text{Mean L1} = 281$$

Mean Tinggi Kelas 1:

$$\text{Mean T1} = \frac{350 + 320}{2}$$

$$\text{Mean T1} = 335$$

Mean Rasio aspek Kelas 1:

$$\text{Mean AR1} = \frac{0,92 + 0,75}{2}$$

$$\text{Mean AR1} = 0,835$$

Mean Diameter Kelas 1:

$$\text{Mean D1} = \frac{361,84 + 320,19}{2}$$

$$\text{Mean D1} = 341,015$$

Kemudian lakukan penghitungan *mean* terhadap kelas 2 dan kelas 3 sehingga menghasilkan Tabel 4.3.

Tabel 4.3 Mean Fitur

Kelas	Mean MH	Mean SDH	Mean SH	Mean MS	Mean SDS	Mean SS	Mean MI
1	194,963	9,8385	10,9265	0,65	9,5885	11,4285	229,518
2	161,419	4,1495	6,867	2,6305	12,9035	5,556	186,3405
3	186,369	6,6765	4,3045	1,183	10,9325	8,19	196,74

Tabel 4.3 Mean Fitur (Lanjutan)

Kelas	Mean SDI	Mean SI	Mean L	Mean T	Mean AR	Mean D
1	2,8	0,726	281	335	0,835	341,0155
2	1,1885	1,7995	272,5	216,5	1,26	289,676
3	1,7105	0,889	336,5	351,5	0,957	363,1395

Langkah 3:

Menghitung nilai standar deviasi dari masing-masing nilai fitur dari masing-masing kelas

Standar Deviasi MH Kelas 1:

$$STD\ MH\ 1 = \sqrt{\frac{(191,47 - 194,963)^2 + (198,45 - 194,963)^2}{2 - 1}}$$

$$STD\ MH\ 1 = 4,939$$

Standar Deviasi SDH Kelas 1:

$$STD\ SDH1 = \sqrt{\frac{(9,392 - 9,8385)^2 + (10,285 - 9,8385)^2}{2 - 1}}$$

$$STD\ SDH1 = 0,6314$$

Standar Deviasi SH Kelas 1:

$$Mean\ SH1 = \sqrt{\frac{(10,515 - 10,9265)^2 + (11,338 - 10,9265)^2}{2 - 1}}$$

$$Mean\ SH1 = 0,5819$$

Standar Deviasi MS Kelas 1:

$$STD\ MS1 = \sqrt{\frac{(0,511 - 0,65)^2 + (0,789 - 0,65)^2}{2 - 1}}$$



$$STD MS1 = 0,1965$$

Standar Deviasi SDS Kelas 1:

$$STD SDS1 = \sqrt{\frac{(9,392 - 9,5885)^2 + (9,785 - 9,5885)^2}{2 - 1}}$$

$$STD SDS1 = 0,2778$$

Standar Deviasi SS Kelas 1:

$$STD SS1 = \sqrt{\frac{(10,515 - 11,285)^2 + (12,342 - 11,285)^2}{2 - 1}}$$

$$STD SS1 = 1,2918$$

Standar Deviasi MI Kelas 1:

$$STD MI1 = \sqrt{\frac{(230,528 - 229,518)^2 + (228,508 - 229,518)^2}{2 - 1}}$$

$$STD MI1 = 1,4283$$

Standar Deviasi SDI Kelas 1:

$$STD SDI1 = \sqrt{\frac{(2,666 - 2,8)^2 + (2,934 - 2,8)^2}{2 - 1}}$$

$$STD SDI1 = 0,8915$$

Standar Deviasi SI Kelas 1:

$$STD SI1 = \sqrt{\frac{(0,774 - 0,726)^2 + (0,678 - 0,726)^2}{2 - 1}}$$

$$STD SI1 = 0,0678$$

Standar Deviasi Lebar Kelas 1:

$$STD L1 = \sqrt{\frac{(321 - 281)^2 + (241 - 281)^2}{2 - 1}}$$

$$STD L1 = 57$$

Standar Deviasi Tinggi Kelas 1:

$$STD T1 = \sqrt{\frac{(350 - 335)^2 + (320 - 335)^2}{2 - 1}}$$

$$STD T1 = 21$$

Standar Deviasi Rasio aspek Kelas 1:

$$STD AR1 = \sqrt{\frac{(0,92 - 0,835)^2 + (0,75 - 0,835)^2}{2 - 1}}$$

$$STD AR1 = 0,116$$

Standar Deviasi Diameter Kelas 1:

$$STD D1 = \sqrt{\frac{(361,84 - 341,015)^2 + (320,19 - 341,015)^2}{2 - 1}}$$

$$STD D1 = 29,453$$

Kemudian lakukan penghitungan standar deviasi terhadap kelas 2 dan kelas 3 sehingga menghasilkan Tabel 4.4.

Tabel 4.4 Standar Deviasi Fitur

Kelas	STD MH	STD SDH	STD SH	STF MS	STD SDS	STD SS	STD MI
1	4,9398	0,6314	0,5819	0,1965	0,2778	1,2918	1,4283
2	1,4750	0,2665	0,5232	0,3726	0,8124	0,4242	1,5634
3	1,4021	0,2538	0,6201	0,3676	0,2892	0,2743	0,5487

Tabel 4.4 Standar Deviasi Fitur (Lanjutan)

Kelas	STD SDI	STD SI	STD L	STD T	STD AR	STD D
1	0,1895	0,0678	57	21	0,116	29,4531
2	0,0615	0,2496	0,7	12	0,066	15,1321
3	0,1859	0,1357	0,7	0,7	0	0,6738

Langkah 4:

Misalkan terdapat suatu citra makanan dengan nilai fitur-fitur:

Mean MH	Mean SDH	Mean SH	Mean MS	Mean SDS	Mean SS	Mean MI
170,25	6,783	8,544	1,335	11,951	8,952	215,439

Mean SDI	Mean SI	Mean L	Mean T	Mean AR	Mean D
2,156	1,568	289	341	1,15	350,765

Menghitung nilai antara nilai citra yang ingin diklasifikasi dengan nilai dari masing-masing kelas:

Probabilitas MH kelas 1:

$$MH(1|2,85) = \frac{1}{\sqrt{2 \times 3,14 \times 4,9398}} \times 2,7182 \frac{(170,25 - 194,963)^2}{2 \times 4,93984^2}$$

$$MH(1|2,85) = 6,5976E - 07$$

Probabilitas SDH kelas 1:

$$SDH(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 0,6314}} \times 2,7182 \frac{(6,783 - 9,8385)^2}{2 \times 0,6314^2}$$

$$SDH(1|0,563) = 4,1278E - 06$$

Probabilitas SH kelas 1:

$$SH(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 0,5819}} \times 2,7182 \frac{(8,544 - 10,9265)^2}{2 \times 0,5819^2}$$

$$SH(1|0,563) = 0,000119815$$

Probabilitas MS kelas 1:

$$MS(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 0,1965}} \times 2,7182 \frac{(0,65 - 1,335)^2}{2 \times 0,1965^2}$$

$$MS(1|0,563) = 0,002274757$$

Probabilitas SDS kelas 1:

$$SDS(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 0,2778}} \times 2,7182 \frac{(9,5885 - 11,951)^2}{2 \times 0,2778^2}$$

$$SDS(1|0,563) = 1,4955E - 16$$

Probabilitas SS kelas 1:

$$SS(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 1,2918}} \times 2,7182 \frac{(11,4285 - 8,952)^2}{2 \times 1,2918^2}$$

$$SS(1|0,563) = 0,055896665$$

Probabilitas MI kelas 1:

$$MI(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 1,4283}} \times 2,7182 \frac{(215,439 - 229,518)^2}{2 \times 1,4283^2}$$

$$MI(1|0,563) = 2,66231E - 22$$

Probabilitas SDI kelas 1:

$$SDI(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 0,1895}} \times 2,7182 \frac{(2,156 - 2,8)^2}{2 \times 0,1895^2}$$

$$SDI(1|0,563) = 0,0002845$$

Probabilitas SI kelas 1:

$$SI(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 0,0678}} \times 2,7182 \frac{(1,568 - 0,726)^2}{2 \times 0,0678^2}$$

$$SI(1|0,563) = 4,9671E - 34$$

Probabilitas L kelas 1:

$$L(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 57}} \times 2,7182 \frac{(289 - 281)^2}{2 \times 57^2}$$

$$L(1|0,563) = 0,52332$$

Probabilitas T kelas 1:

$$T(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 21}} \times 2,7182 \frac{(341 - 335)^2}{2 \times 21^2}$$

$$T(1|0,563) = 0,0835$$

Probabilitas AR kelas 1:

$$AR(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 0,116}} \times 2,7182 \frac{(1,15 - 0,835)^2}{2 \times 0,116^2}$$

$$AR(1|0,563) = 0,00021$$

Probabilitas D kelas 1:

$$D(1|0,563) = \frac{1}{\sqrt{2 \times 3,14 \times 29,4531}} \times 2,7182 \frac{(350,765 - 341,0155)^2}{2 \times 29,4531^2}$$

$$D(1|0,563) = 0,06958$$

Hitung juga probabilitas dari citra yang ingin diklasifikasi terhadap kelas 2 dan 3

Langkah 5:

Setelah mendapatkan semua probabilitas dari tiap fitur, masukkan nilai-nilai fitur tersebut ke dalam bentuk array

$$\begin{aligned} \text{fitur1} &= 6,5976E - 07; 4,1278E \\ &\quad - 06; 0,000119815; 0,002274757; 1,4955E \\ &\quad - 16; 0,055896665; 2,66231E - 22; 0,0002845; 4,9671E \\ &\quad - 34; 0,52332; 0,0835; 0,00021; 0,06958 \end{aligned}$$

Langkah 6:

Menghitung nilai maks dari array fitur1

$$\text{maks} = \text{maks}(\text{fitur1})$$

$$\text{maks} = 0,52332$$

Langkah 7:

Mengurangi semua nilai pada matriks fitur1 dengan nilai maks dan memasukkannya ke dalam matriks baru yang berukuran sama

$$\text{hasil ke } - 1 = (6,5976E - 07 - 0,52332)$$

$$\text{hasil ke } - 2 = (4,1278E - 06 - 0,52332)$$

$$\text{hasil ke } - 3 = (0,002274757 - 0,52332)$$

$$\text{hasil ke } - 4 = (1,495E - 16 - 0,52332)$$

sampai semua data telah dikurangi nilai maks

Langkah 8:

Mencari nilai eksponen dari masing-masing pada array hasil dan menjumlahkan semua nilai eksponennya

$$\text{Eks ke } - i = \text{Eksponen}(\text{hasil ke } - i)$$

$$\text{jumlahEks} = (\text{Eks ke } - 1 + \text{Eks ke } - 2 + \text{Eks ke } - 3 + \dots + \text{Eks ke } - 13)$$

$$\text{jumlahEks} = 7,648$$

Langkah 9:

Menambahkan nilai maks dengan nilai log dari jumlah eksponen

$$p1 = 0,52332 + \log(7,648)$$

$$p1 = 2,557$$

Dan dengan cara yang sama menghitung untuk posterior kelas 2 dan 3

$$p2 = 2,145$$

$$p3 = 2,987$$

Langkah 10:

Hitung nilai probabilitas akhir dari masing-masing kelas terhadap data citra yang ingin diklasifikasikan

$$P(X|1) = \frac{2,557}{(2,557 + 2,145 + 2,987)} = 0,3212$$

$$P(X|2) = \frac{2,145}{(2,557 + 2,145 + 2,987)} = 0,3034$$

$$P(X|3) = \frac{2,987}{(2,557 + 2,145 + 2,987)} = 0,3752$$

Karena nilai data citra yang akan diklasifikasikan terhadap kelas 2 memiliki nilai yang paling tinggi, maka data tersebut diklasifikasi menjadi kelas 2.

4.2.5 Penghitungan Evaluasi

Terdapat *confusion matrix* dengan nilai berikut:

$$K1 = 250 \text{ dan } K2 = 32$$

Langkah 1:

Menentukan hasil pengurangan antara K1 dengan K2

$$250 - 32 = 218$$

Langkah 2:

Membagi hasil pengurangan K1 dan K2 dengan nilai K1

$$\frac{218}{250} = 0,872$$

Langkah 3:

Mengalikan hasil pembagian dengan 100%

$$0,872 \times 100\% = 87,2\%$$

BAB 5 IMPLEMENTASI

5.1 Batasan Implementasi

Pada proses pengimplementasian dibutuhkan batasan-batasan agar menghasilkan hasil yang sama dengan hasil dari proses pengujian. Batasan-batasan dalam proses pengimplementasian program ini adalah:

1. Fitur yang akan diekstraksi dari citra makanan adalah fitur warna yang akan diekstraksi dengan metode *color moment* dan fitur bentuk yang akan diekstraksi dengan metode *morphological shape descriptors* dan diklasifikasikan dengan metode *Naive Bayes*.
2. Proses *preprocessing* citra makanan yang digunakan adalah proses konversi warna RGB menjadi warna HSI, *grayscale*, *gaussian-blur*, *thresholding*, erosi, dan dilasi. Semua proses yang akan diterapkan pada proses *preprocessing* akan menggunakan *library* yang ada pada *openCV*.
3. Citra makanan yang digunakan adalah data primer, dimana citra diambil menggunakan kamera dari *smartphone* Samsung Galaxy S8.
4. Pencahayaan yang digunakan pada saat pengambilan citra menggunakan cahaya matahari yang berlangsung pada pukul 10.00 sampai dengan 14.00 WIB

5.2 Implementasi Algoritme

Algoritme yang akan digunakan dan diterapkan ke dalam bentuk program adalah algoritme *preprocessing*, algoritme *color moment*, algoritme *morphological shape descriptors*, dan algoritme klasifikasi *Naive Bayes*.

5.2.1 Implementasi *Preprocessing* Citra Makanan

Proses *preprocessing* citra adalah proses untuk mengolah citra makanan agar dapat mendeteksi objek makanan yang ada pada citra makanan. Proses *preprocessing* citra yang digunakan adalah proses mengubah ukuran dimensi citra, proses mengubah warna RGB menjadi LAB, proses *grayscale*, proses *gaussian blur*, proses *thresholding*, proses erosi, dan proses dilasi.

Kode: Tahap <i>Preprocessing</i> citra makanan	
1	<code>import cv2</code>
2	<code>import os.path</code>
3	<code>from imutils import paths</code>
4	<code>import numpy as np</code>
5	
6	<code>jGambar = 0</code>
7	<code>for imagePath in paths.list_images("Folder image sebelum di</code>
8	<code>preprocessing"):</code>
9	<code> jGambar += 1</code>


```

10 img = cv2.imread(imagePath)
11 imgR = cv2.resize(img, (400,400))
12 hsv = cv2.cvtColor(imgR, cv2.COLOR_BGR2HSV)
13 gray = cv2.cvtColor(hsv, cv2.COLOR_BGR2GRAY)
14 gBlur = cv2.GaussianBlur(gray, (5,5),0)
15 ret, th =
16 cv2.threshold(gBlur,120,255,cv2.THRESH_BINARY+cv2.THRESH_OTSU
17 )
18 kernell = np.ones((3,3), np.uint8)
19 img_erosion1 = cv2.erode(th, kernell, iterations=1)
20 img_dilation1 = cv2.dilate(img_erosion1, kernell,
21 iterations=1)
22 imgOut1 = cv2.bitwise_and(imgR, imgR, mask =
23 img_dilation1)
24 rename = "File image sebelum di preprocessing" +
25 str(os.path.split(os.path.dirname(imagePath))[-1]) + "_SG8_"+
26 str(jGambar) + ".jpg"
27 print(rename)
28 cv2.imwrite(rename, imgR)

```

Penjelasan Kode:

1. Baris 1-4: Meng-*import* library `cv2`, `os.path`, `imutils`, dan `numpy` yang akan digunakan dalam pengimplementasian kode.
2. Baris 6: Deklarasi variabel `jGambar` dengan nilai 0.
3. Baris 7: Deklarasi perulangan dengan variabel `imagePath` untuk membaca semua gambar yang ada pada folder untuk diimplementasikan proses *preprocessing*.
4. Baris 9: Menambahkan nilai variabel `jGambar` dengan 1 pada setiap perulangan.
5. Baris 10: Mengambil gambar dengan lokasi gambar dalam variabel `imagePath` dan menyimpan gambar tersebut dalam variabel `img`.
6. Baris 11: Mengubah dimensi ukuran gambar yang ada variabel `img` menjadi ukuran 400x400 dan menyimpannya pada variabel `imgR`.
7. Baris 12: Mengubah citra yang sebelumnya memiliki model warna RGB menjadi HSV dan menyimpannya pada variabel `hsv`.
8. Baris 13: Mengubah warna citra menjadi keabuan dan menyimpannya pada variabel `gray`.
9. Baris 14: Mendeklarasikan variabel `gBlur` dengan library `GaussianBlur` pada variabel `gray` dengan ukuran kernel 5x5.
10. Baris 16: Melakukan proses *thresholding* binary dan *otsu* pada variabel gambar `gBlur` dengan ukuran nilai 120-255.
11. Baris 18: Deklarasi variabel `kernell` yang berisi array bernilai 0 dengan ukuran 3x3.
12. Baris 19: Mengimplementasi proses erosi pada gambar hasil *thresholding* dengan `kernell` dan menyimpannya pada variabel `img_erosion1`.

13. Baris 20: Mengimplementasi proses dilasi pada gambar hasil erosi dengan kernel1 dan menyimpannya pada variabel `img_dilation1`.
14. Baris 22: Mengimplementasi library `bitwise_and` dengan mask `img_dilation1` dan menyimpannya pada `imgOut1`.
15. Baris 24-26: Deklarasi variabel `rename` dengan nilai alamat folder gambar akan disimpan.
16. Baris 27: Proses mencetak nama file yang telah selesai dilakukan proses *preprocessing*.
17. Baris 28: Mencetak gambar hasil proses *preprocessing* pada alamat folder dalam variabel `rename`.

5.2.2 Implementasi Proses Ekstraksi Fitur Warna Dan Bentuk Pada Citra Makanan

Proses ekstraksi fitur adalah proses untuk mengambil nilai yang ada pada objek citra makanan agar nilai-nilai tersebut nanti dapat digunakan untuk menentukan kelas pada proses pengklasifikasian. Nilai-nilai fitur yang diambil dari citra makanan adalah *mean H*, *mean S*, *mean I*, standar deviasi H, standar deviasi S, standar deviasi I, *skewness H*, *skewness S*, *skewness I*, panjang citra, lebar citra, rasio aspek citra, dan diameter yang ada pada citra.

Kode: Tahap Training dan Ekstraksi Fitur Warna dan Bentuk	
1	<code>from PIL import Image</code>
2	<code>import cv2</code>
3	<code>import os.path</code>
4	<code>import math</code>
5	<code>from imutils import paths</code>
6	<code>import numpy as np</code>
7	<code>import pandas as pd</code>
8	
9	<code>jGambar = 0</code>
10	<code>arrmeanH=[]</code>
11	<code>arrstdH=[]</code>
12	<code>arrskewH=[]</code>
13	<code>arrmeanS=[]</code>
14	<code>arrstdS=[]</code>
15	<code>arrskewS=[]</code>
16	<code>arrmeanI=[]</code>
17	<code>arrstdI=[]</code>
18	<code>arrskewI=[]</code>
19	<code>namaFile=[]</code>
20	<code>l=[]</code>
21	<code>w=[]</code>
22	<code>D=[]</code>
23	<code>asr=[]</code>
24	
25	<code>def HSIColor(img):</code>
26	<code> hco = []</code>
27	<code> sco = []</code>
28	<code> ico = []</code>
29	<code> if isinstance(img, Image.Image):</code>
30	<code> r,g,b = img.split()</code>

```

31         for red, green, blue in zip(r.getdata(), g.getdata(),
32 b.getdata()):
33             jumlah = red+green+blue
34             if jumlah == 0:
35                 continue
36             red = red/jumlah
37             green = green/jumlah
38             blue = blue/jumlah
39
40             # HITUNG H
41             numer = 0.5 * ((red-green)+(red-blue))
42             denom = (((red-green)*(red-green))+ (red-
43 blue)*(green-blue))**(0.5)
44             if denom == 0:
45                 continue
46             theta = numer/denom
47             if blue<=green:
48                 h = math.acos(theta)
49             else:
50                 h = 2*math.pi-math.acos(theta)
51             # HITUNG S
52             s = 1 - 3*(min(red,min(green,blue)))
53
54
55             # HITUNG I
56             i = jumlah/(3*255)
57             hco.append(int(h*180/math.pi))
58             sco.append(int(s*100))
59             ico.append(int(i*255))
60             return hco,sco,ico
61
62 def cMoment(hco,sco,ico):
63     ##### H #####
64     # HITUNG MEAN H
65     meanH = 0
66     for i in hco:
67         meanH = meanH+i
68     meanH = meanH/len(hco)
69     # HITUNG STD H
70     stdH = 0
71     dak = 0
72     for i in hco:
73         dak = dak + (i-meanH)**(2)
74     stdH = dak/len(hco)
75     stdH = stdH**(0.5)
76
77     # HITUNG SKEW H
78     skewH = 0
79     das = 0
80     for i in hco:
81         das = das + (i-meanH)**(3)
82     skewH = das/len(hco)
83     skewH = skewH**(1/3)
84     skewH = skewH.real
85     ##### H #####
86
87     ##### S #####
88     # HITUNG MEAN S
89     meanS = 0

```

```

90     for i in sco:
91         means = means+i
92     means = means/len(sco)
93     # HITUNG STD S
94     stdS = 0
95     dat = 0
96     for i in sco:
97         dat = dat + (i-means)**(2)
98     stdS = dat/len(sco)
99     stdS = stdS**(0.5)
100
101     # HITUNG SKEW S
102     skewS = 0
103     dav = 0
104     for i in sco:
105         dav = dav + (i-means)**(3)
106     skewS = dav/len(sco)
107     skewS = skewS**(1/3)
108     skewS = skewS.real
109     ##### S #####
110
111     ##### I #####
112     # HITUNG MEAN I
113     meanI = 0
114     for i in ico:
115         meanI = meanI+i
116     meanI = meanI/len(ico)
117     # HITUNG STD I
118     stdI = 0
119     dal = 0
120     for i in ico:
121         dal = dal + (i-meanI)**(2)
122     stdI = dal/len(hco)
123     stdI = stdI**(0.5)
124
125     # HITUNG SKEW I
126     skewI = 0
127     dap = 0
128     for i in ico:
129         dap = dap + (i-meanI)**(3)
130     skewI = dap/len(ico)
131     skewI = skewI**(1/3)
132     skewI = skewI.real
133     return
134 meanH,meanS,meanI,stdH,stdS,stdI,skewH,skewS,skewI
135
136 for imagePath in paths.list_images("C:/Users/IAN-
137 LORD/Downloads/YanBaru/TESTING"):
138     jGambar += 1
139     img = cv2.imread(imagePath)
140     sizeGambar = 300,300
141     imgR = cv2.resize(img,(sizeGambar))
142     hsv = cv2.cvtColor(imgR, cv2.COLOR_BGR2HSV)
143     gray = cv2.cvtColor(hsv, cv2.COLOR_BGR2GRAY)
144     height, width = gray.shape
145     gBlur = cv2.GaussianBlur(gray, (5,5),0)
146     ret, th =
147     cv2.threshold(gBlur,120,255,cv2.THRESH_BINARY+cv2.THRESH_OTSU
148 )

```

```

149     kernell = np.ones((3,3), np.uint8)
150     img_erosion1 = cv2.erode(th, kernell, iterations=1)
151     img_dilation1 = cv2.dilate(img_erosion1, kernell,
152 iterations=1)
153     imgOut1 = cv2.bitwise_and(imgR, imgR, mask =
154 img_dilation1)
155     print(imagePath)
156     a = Image.open(imagePath)
157     b = a.resize((sizeGambar))
158     hco,sco,ico = HSIColor(b)
159     meanH,meanS,meanI, stdH, stdS, stdI, skewH, skewS, skewI =
160 cMoment(hco, sco, ico)
161     cnts = cv2.findContours(img_dilation1,
162 cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
163     cnts = cnts[1]
164     c = max(cnts, key=cv2.contourArea)
165     extLeft = tuple(c[c[:, :, 0].argmin()][0])
166     extRight = tuple(c[c[:, :, 0].argmax()][0])
167     extTop = tuple(c[c[:, :, 1].argmin()][0])
168     extBot = tuple(c[c[:, :, 1].argmax()][0])
169     # Nilai W
170     width = abs(extLeft[0]-extRight[0])
171     # Nilai H
172     length = abs(extTop[1]-extBot[1])
173     # Nilai Rasio aspek
174     ar = width/length
175     # Diameter
176     d1 = ((extRight[0] - extLeft[0])**2 + (extRight[1] -
177 extLeft[1])**2)**0.5
178     d2 = ((extTop[0] - extBot[0])**2 + (extTop[1] -
179 extBot[1])**2)**0.5
180     dmax = max(d1,d2)
181     arrmeanH.append(meanH)
182     arrstdH.append(stdH)
183     arrskewH.append(skewH)
184     arrmeanS.append(meanS)
185     arrstdS.append(stdS)
186     arrskewS.append(skewS)
187     arrmeanI.append(meanI)
188     arrstdI.append(stdI)
189     arrskewI.append(skewI)
190     l.append(length)
191     w.append(width)
192     D.append(dmax)
193     asr.append(ar)
194
195     namaFile.append(os.path.split(os.path.dirname(imagePath))[-
196 1])
197
198     dataHSI = {
199         'Mean H' : arrmeanH,
200         'Standar Deviansi H' : arrstdH,
201         'Skewness H' : arrskewH,
202         'Mean S' : arrmeanS,
203         'Standar Deviansi S' : arrstdS,
204         'Skewness S' : arrskewS,
205         'Mean I' : arrmeanI,
206         'Standar Deviansi I' : arrstdI,
207         'Skewness I' : arrskewI,

```

```
208         'Panjang' : l,  
209         'Lebar' : w,  
210         'Diameter' : D,  
211         'Rasio aspek' : asr,  
212         'Kelas' : namaFile  
213     }  
214     df = pd.DataFrame(dataHSI)  
215     df.to_excel("File akhir.xlsx", sheet_name="Sheet1",  
216     index=False)
```

Penjelasan Kode:

1. Baris 1-7: Mendeklarasikan library-library yang akan digunakan pada proses pengimplementasian kode.
2. Baris 9: Mendeklarasikan variabel `jGambar` dengan nilai 0.
3. Baris 10-23: Mendeklarasikan array-array yang akan digunakan untuk menampung nilai-nilai fitur yang akan diekstrak.
4. Baris 25: Mendeklarasikan metode `HSIColor` dengan parameter `img`.
5. Baris 26-28: Mendeklarasikan array-array untuk menampung nilai warna H, S, dan I.
6. Baris 29: Mendeklarasikan percabangan untuk mengecek gambar yang dimasukkan.
7. Baris 30: Memisahkan nilai warna r, g, b dengan metode `split` yang ada pada `img`.
8. Baris 31: Mendeklarasikan perulangan untuk nilai red, green, blue dengan parameter nilai r, g, b.
9. Baris 33: Mendeklarasikan variabel `jumlah` yang berisi nilai jumlah dari red, green, dan blue.
10. Baris 34-35: Mendeklarasikan percabangan di mana jika nilai `jumlah` bernilai 0 maka proses akan berlanjut ke citra berikutnya.
11. Baris 36-38: Menormalisasi nilai red, green, blue dengan membagi masing-masing tersebut dengan nilai `jumlah`.
12. Baris 40: Menandai dimulainya bagian untuk penghitungan nilai H.
13. Baris 41: Mendeklarasikan deklarasi variabel `numer` dengan nilai 0.5 dikali dengan hasil pengurangan red green dan red blue.
14. Baris 42: Mendeklarasikan variabel `denom` dengan nilai pangkat 2 dari pengurangan red green dikalikan dengan hasil pengurangan red blue dikali dengan green blue yang selanjutnya diakar kuadrat.
15. Baris 44: Mendeklarasikan percabangan jika nilai `denom` sama dengan 0 maka proses akan dilanjutkan ke proses berikutnya.
16. Baris 46: Mendeklarasikan variabel `theta` yang bernilai hasil pembagian dari `numer` dan `denom`.

17. Baris 47-48: Mendeklarasikan percabangan jika nilai blue lebih rendah sama dengan green maka nilai h adalah hasil acos dari theta.
18. Baris 49-50: Jika kondisi percabangan tidak terpenuhi maka nilai h adalah hasil dari 2 dikali dengan pi dikurang dengan hasil acos dari theta.
19. Baris 52: Mendeklarasikan variabel s yang mempunyai nilai hasil dari 1 dikurang dengan 3 dikali dengan nilai terendah diantara red, green, dan blue.
20. Baris 56: Mendeklarasikan variabel i yang mempunyai nilai hasil dari variabel jumlah dibagi dengan perkalian 3 dengan 255.
21. Baris 57-59: Memasukkan hasil penghitungan h, s, dan i ke dalam array hco, sco, dan ico yang dinormalisasikan dulu sebelumnya.
22. Baris 60: Mengembalikan nilai dari array hco, sco, ico.
23. Baris 62: Mendeklarasikan metode cMoment dengan parameter hco, sco, ico.
24. Baris 65-68: Mengimplementasikan penghitungan nilai *mean H*.
25. Baris 69-75: Mengimplementasikan penghitungan nilai standar deviasi H.
26. Baris 77-84: Mengimplementasikan penghitungan nilai *skewness H*.
27. Baris 88-92: Mengimplementasikan penghitungan nilai *mean S*.
28. Baris 93-99: Mengimplementasikan penghitungan nilai standar deviasi S.
29. Baris 101-108: Mengimplementasikan penghitungan nilai *skewness S*.
30. Baris 112-116: Mengimplementasikan penghitungan nilai *mean I*.
31. Baris 117-123: Mengimplementasikan penghitungan nilai standar deviasi I.
32. Baris 125-132: Mengimplementasikan penghitungan nilai *skewness I*.
33. Baris 133-134: Mengembalikan nilai hasil penghitungan *mean H*, standar deviasi H, *skewness H*, *mean S*, standar deviasi S, *skewness S*, *mean I*, standar deviasi I, *skewness I*.
34. Baris 136-137: Mendeklarasikan perulangan untuk membaca citra pada folder.
35. Baris 138: Menambahkan nilai 1 pada variabel jGambar.
36. Baris 139: Membaca gambar yang berada pada imagePath dan menyimpan nilainya pada variabel img.
37. Baris 140: Mendeklarasikan variabel sizeGambar yang berukuran 300x300.
38. Baris 141: Mendeklarasikan variabel imgR yang berisi hasil resize gambar dengan ukuran dari variabel sizeGambar.
39. Baris 142: Mengubah citra yang sebelumnya memiliki model warna RGB menjadi HSV dan menyimpannya pada variabel hsv.
40. Baris 143: Mengubah warna citra menjadi keabuan dan menyimpannya pada variabel gray.

41. Baris 144: Mendeklarasikan variabel height dan width yang digunakan untuk menyimpan hasil dari ukuran variabel gray
42. Baris 145: Mendeklarasikan variabel gBlur dengan library GaussianBlur pada variabel gray dengan ukuran kernel 5x5.
43. Baris 146-147: Melakukan proses thresholding binary dan otsu pada variabel gambar gBlur dengan ukuran nilai 120-255.
44. Baris 149: Deklarasi variabel kernel1 yang berisi array bernilai 0 dengan ukuran 3x3.
45. Baris 150: Mengimplementasi proses erosi pada gambar hasil thresholding dengan kernel1 dan menyimpannya pada variabel img_erosion1.
46. Baris 151: Mengimplementasi proses dilasi pada gambar hasil erosi dengan kernel1 dan menyimpannya pada variabel img_dilation1.
47. Baris 153: Mengimplementasi library bitwise_and dengan mask img_dilation1 dan menyimpannya pada imgOut1.
48. Baris 155: Mencetak isi dari variabel imagePath.
49. Baris 156: Mendeklarasikan variabel a yang berisi dari hasil membaca gambar imagePath.
50. Baris 157: Mendeklarasikan variabel b yang berisi hasil resize a dengan ukuran dari sizeGambar.
51. Baris 158: Menginisialisasikan variabel hco, sco, ico dengan hasil dari pengimplementasian metode HSIColor dengan parameter b
52. Baris 159: Menginisialisasikan variabel meanH, meanS, meanI, stdH, stdS, stdI, skewH, skewS, skewI dengan hasil dari pengimplementasian cMoment dengan parameter hco,sco,ico.
53. Baris 161: Mendeklarasikan variabel cnts yang berisi hasil dari findContours dengan parameter img_dilation1.
54. Baris 164: Mendeklarasikan variabel c yang berisi nilai max dengan parameter cnts dan contourArea.
55. Baris 165-168: Mencari titik-titik terjauh dari gambar dan inialisasikan pada variabel extLeft, extRight, extTop, dan extBot.
56. Baris 169-180: Menghitung nilai dari panjang, lebar, rasio aspek dan diameter pada gambar dan menginisialisasikannya pada variabel width, length, ar, dan dmax.
57. Baris 181-193: Menambahkan nilai dari hasil ekstraksi fitur ke masing-masing jenis variabelnya.
58. Baris 195: Menambahkan alamat direktori pada variabel namaFile.
59. Baris 198-213: Mendeklarasikan variabel dataHSI yang berisi hasil dari nilai yang sudah ditambahkan kepada masing-masing variabel fitur dan kelas.

60. Baris 214: Mendeklarasikan variabel df yang berisi dataframe dari dataHSI.
61. Baris 215: Menginputkan data yang ada pada df ke dalam file excel dengan parameter nama file, dan nama sheet.

5.2.3 Implementasi Pengujian Klasifikasi Citra Makanan

Proses pengujian klasifikasi adalah proses untuk menguji akurasi yang akan dihasilkan oleh pengimplentasian kode sehingga dapat menghasilkan bahan dukungan dalam proses analisis. Proses klasifikasi yang digunakan mengimplementasikan metode klasifikasi *Naive Bayes*.

```
Kode: Tahap Klasifikasi Citra
1 import math
2 import pandas as pd
3 from scipy.special import logsumexp
4
5 excelnya = pd.read_excel("Nama File Training", "Nama Lembar")
6 def separateByClass(dataset):
7     separated = {}
8     for i in range(len(dataset)):
9         vector = [dataset['Mean H'][i],
10                 dataset['Standar Deviansi H'][i],
11                 dataset['Skewness H'][i],
12                 dataset['Mean S'][i],
13                 dataset['Standar Deviansi S'][i],
14                 dataset['Skewness S'][i],
15                 dataset['Mean I'][i],
16                 dataset['Standar Deviansi I'][i],
17                 dataset['Skewness I'][i],
18                 dataset['Panjang'][i],
19                 dataset['Lebar'][i],
20                 dataset['Diameter'][i],
21                 dataset['Rasio aspek'][i]]
22         if (dataset['Kelas'][i] not in separated):
23             separated[dataset['Kelas'][i]] = []
24             separated[dataset['Kelas'][i]].append(vector)
25     return separated
26
27 def testing(training, dataTesting):
28     meanFitur = [[0 for i in range(len(excelnya.columns)-1)]
29 for j in range(len(training))]
30     index = 0
31     for i in training:
32         temp= [0 for i in range(len(excelnya.columns)-1)]
33         for j in range(len(training[i])):
34             for k in range(len(training[i][j])):
35                 temp[k]+=training[i][j][k]
36         for k in range(len(training[i][0])):
37             meanFitur[index][k] = temp[k] / len(training[i])
38         index+=1
39
40     index = 0
41     stdFitur = [[0 for i in range(len(excelnya.columns)-1)]
42 for j in range(len(training))]
43     for i in training:
44         temp= [0 for i in range(len(excelnya.columns)-1)]
```

```

45         for j in range(len(training[i])):
46             for k in range(len(training[i][j])):
47                 temp[k]+=(training[i][j][k]-
48 meanFitur[index][k])**2
49             for k in range(len(training[i][0])):
50                 stdFitur[index][k] = math.sqrt(temp[k] /
51 (len(training[i])-1))
52                 index+=1
53
54     pFiturKelas = [[0 for i in range(len(excelnya.columns)-
55 1)] for j in range(len(training))]
56     posKelas = [0 for i in range(len(training))]
57
58     index = 0
59     jumlah = 0
60     for i in training:
61         for k in range(len(training[i][0])):
62             exAtas = (dataTesting[k]-meanFitur[index][k])**2
63             exBawah = 2*(stdFitur[index][k]**2)
64             exBagi = exAtas/exBawah
65             fBawah = math.sqrt(2*math.pi*stdFitur[index][k])
66             pFiturKelas[index][k] = (1 / fBawah)*(math.e **
67 (-exBagi))
68             posKelas[index] = logsumexp(pFiturKelas[index])
69             jumlah += posKelas[index]
70             index+=1
71
72     probKelas = [0 for i in range(len(training))]
73     index = 0
74     for i in training:
75         probKelas[index] = posKelas[index]/jumlah
76         index+=1
77     tes = probKelas.index(max(probKelas))
78     kembalian = 0
79     index=0
80     for i in training:
81         if (index==tes):
82             kembalian = i
83         index+=1
84     return kembalian
85 training = separateByClass(excelnya)
86 excelTesting = pd.read_excel("Nama File Testing.xlsx", "Nama
87 Lembar")
88 arrData = []
89 arrPredict = []
90 for i in range(len(excelTesting)):
91     dataTesting = [excelTesting['Mean H'][i],
92                   excelTesting['Standar Deviansi H'][i],
93                   excelTesting['Skewness H'][i],
94                   excelTesting['Mean S'][i],
95                   excelTesting['Standar Deviansi S'][i],
96                   excelTesting['Skewness S'][i],
97                   excelTesting['Mean I'][i],
98                   excelTesting['Standar Deviansi I'][i],
99                   excelTesting['Skewness I'][i],
100                   excelTesting['Panjang'][i],
101                   excelTesting['Lebar'][i],
102                   excelTesting['Diameter'][i],
103                   excelTesting['Rasio aspek'][i]]

```

```

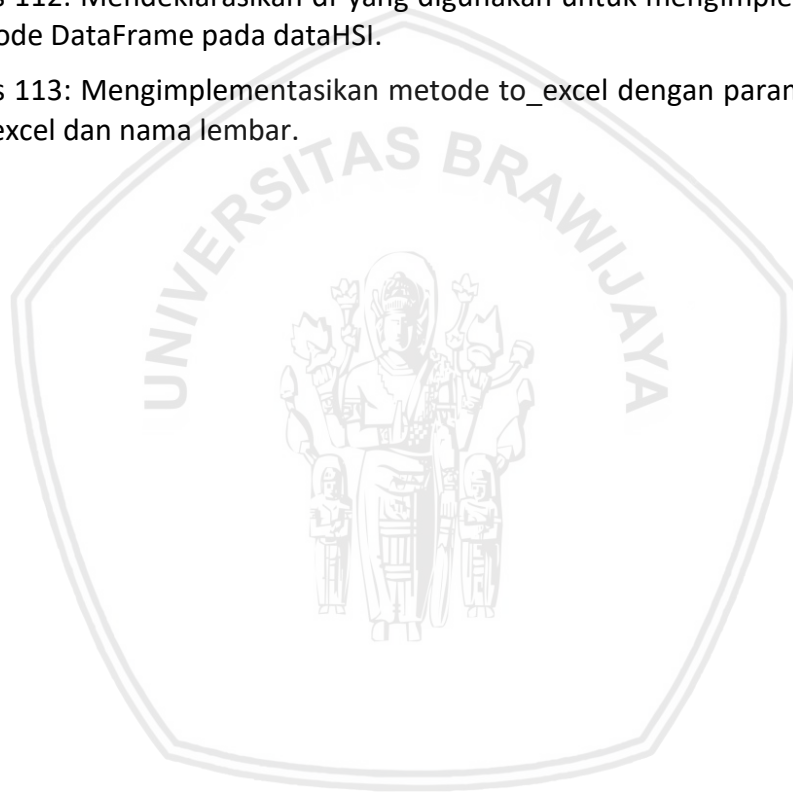
104     test = testing(training, dataTesting)
105     arrPredict.append(test)
106     arrData.append(excelTesting['Kelas'][i])
107
108     dataHSI = {
109         'Data' : arrData,
110         'Predict' : arrPredict
111     }
112     df = pd.DataFrame(dataHSI)
113     df.to_excel("Nama File Hasil.xlsx", sheet_name="Nama Lembar",
114     index=False)

```

Penjelasan Kode:

1. Baris 1-3: Mendeklarasikan library-library yang akan digunakan di dalam proses pengimplementasian kode.
2. Baris 5: Mendeklarasikan variabel excelnya yang berisi dari file data training.
3. Baris 6: Mendeklarasikan metode `separateByClass` dengan parameter dataset.
4. Baris 7: Mendeklarasikan variabel `separated`.
5. Baris 8-21: Mengambil nilai dari masing-masing fitur yang ada pada dataset sesuai dengan headernya.
6. Baris 22-24: Mendeklarasikan percabangan `if` untuk mengecek apakah kelas `if` sudah ada di dalam variabel `separated` atau belum.
7. Baris 25: Mengembalikan nilai `separated`.
8. Baris 27: Mendeklarasikan metode `testing` dengan parameter `training` dan `dataTesting`.
9. Baris 28-38: Melakukan penghitungan mean pada setiap fitur di dalam masing-masing kelas.
10. Baris 40-52: Melakukan penghitungan standar deviasi pada setiap fitur di dalam masing-masing kelas.
11. Baris 54: Mendeklarasikan variabel `pFiturKelas` yang akan digunakan sebagai penampung nilai dari probabilitas kelas dari masing-masing fitur dalam tiap kelas.
12. Baris 58-70: Melakukan penghitungan `pFiturKelas` dari masing-masing fitur di dalam kelas.
13. Baris 72-84: Melakukan penghitungan probabilitas dari masing-masing kelas dan mengambil nilai kelas yang memiliki nilai probabilitas tertinggi setelah itu mengembalikannya.
14. Baris 85: Mendeklarasikan variabel `training` yang berisi hasil dari pengimplementasian metode `separateByClass` dengan parameter excelnya.
15. Baris 86: Mendeklarasikan variabel `excelTesting` dan membuka file excel yang berisi data yang akan diuji.

16. Baris 87: Mendeklarasikan variabel arrData.
17. Baris 89: Mendeklarasikan variabel arrPredict.
18. Baris 90-104: Melakukan perulangan yang berisi proses pengujian dengan menggunakan metode testing dan dengan parameter training dan dataTesting.
19. Baris 105: Memasukan nilai dari test ke dalam arrPredict.
20. Baris 106: Memasukan nilai kelas ke dalam arrData.
21. Baris 108: Mendeklarasikan variabel dataHSI yang berisi data yang akan dimasukkan ke dalam file excel.
22. Baris 112: Mendeklarasikan df yang digunakan untuk mengimplementasikan metode DataFrame pada dataHSI.
23. Baris 113: Mengimplementasikan metode to_excel dengan parameter nama file excel dan nama lembar.



BAB 6

PENGUJIAN DAN ANALISIS

Pada bab 6 akan menjelaskan hasil dari pengujian dan juga analisis terhadap hasil dari pengimplementasian kode program. Proses pengujian yang dilakukan adalah menguji apakah kelas makanan yang diuji sama dengan kelas makanan sesungguhnya dari citra tersebut. Kelas makanan yang akan diuji pada pada proses pengujian adalah donat, roti gandum, mie goreng, telur dadar, rendang, selada, stroberi, pisang hijau, nasi kuning, dan mie soba.

Tabel 6.1 Tabel Kode Makanan dan Nama Makanan

Kode Makanan	Nama Makanan
1	Donat
2	Roti Gandum
4	Mie Goreng
7	Telur Dadar
9	Rendang
13	Selada
16	Stroberi
17	Pisang Hijau
21	Nasi Kuning
25	Mie Soba

6.1 Pengujian Hasil Akurasi Klasifikasi Citra Makanan

Pengujian ini memiliki tujuan untuk mencari tahu hasil akurasi dari proses klasifikasi citra makanan. Pengujian dilakukan dengan menggunakan citra makanan berukuran 400x400 piksel sebanyak 100 citra makanan dan data latih sebanyak 202 citra makanan. Fitur yang digunakan sebanyak 13 fitur yang berasal dari 9 fitur hasil metode *color moment* dan 4 fitur hasil metode *morphological shape descriptor*.

Pada Tabel 6. adalah hasil pengujian dari citra makanan sebanyak 10 kelas yang menggunakan fitur-fitur warna dan bentuk. Hasil akurasi yang didapatkan dari proses pengujian adalah sebesar 78%. Kelas yang paling banyak mengalami proses salah pengklasifikasian adalah kelas 9 (rendang) yaitu kelas rendang dengan banyaknya data citra yang salah diklasifikasikan sebanyak 7 data dan disalahklasifikasikan menjadi kelas 1 (donat).

Tabel 6.2 Pengujian Data Uji Klasifikasi Makanan

No	Data	Prediksi	Hasil
1	1	1	BENAR
2	1	1	BENAR
3	1	1	BENAR
4	1	25	SALAH
5	1	1	BENAR
6	1	1	BENAR
7	1	21	SALAH
8	1	21	SALAH
9	1	1	BENAR
10	1	1	BENAR
11	2	2	BENAR
12	2	7	SALAH
13	2	2	BENAR
14	2	2	BENAR
15	2	2	BENAR
16	2	2	BENAR
17	2	2	BENAR
18	2	2	BENAR
19	2	25	SALAH
20	2	2	BENAR
21	4	1	SALAH
22	4	4	BENAR
23	4	4	BENAR
24	4	4	BENAR
25	4	7	SALAH
26	4	1	SALAH
27	4	4	BENAR
28	4	4	BENAR
29	4	4	BENAR
30	4	7	SALAH
31	7	7	BENAR
32	7	7	BENAR
33	7	7	BENAR
34	7	7	BENAR
35	7	7	BENAR
36	7	7	BENAR
37	7	7	BENAR
38	7	7	BENAR
39	7	7	BENAR
40	7	7	BENAR
41	9	9	BENAR
42	9	1	SALAH

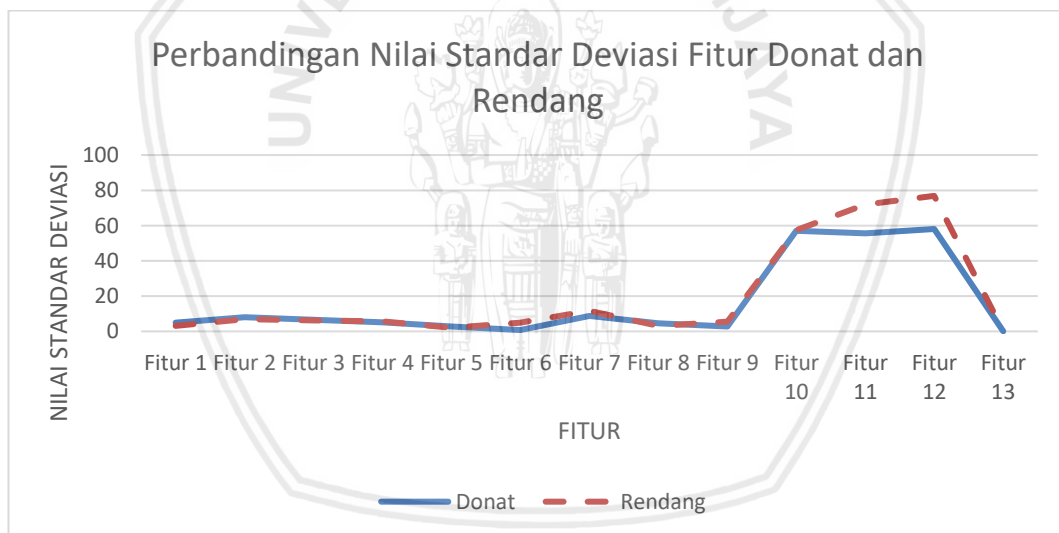
Tabel 6.2 (Lanjutan)

43	9	1	SALAH
44	9	25	SALAH
45	9	1	SALAH
46	9	7	SALAH
47	9	1	SALAH
48	9	9	BENAR
49	9	9	BENAR
50	9	1	SALAH
51	13	13	BENAR
52	13	13	BENAR
53	13	1	SALAH
54	13	13	BENAR
55	13	13	BENAR
56	13	13	BENAR
57	13	13	BENAR
58	13	13	BENAR
59	13	13	BENAR
60	13	1	SALAH
61	16	21	SALAH
62	16	16	BENAR
63	16	16	BENAR
64	16	16	BENAR
65	16	16	BENAR
66	16	16	BENAR
67	16	4	SALAH
68	16	16	BENAR
69	16	16	BENAR
70	16	7	SALAH
71	17	17	BENAR
72	17	17	BENAR
73	17	4	SALAH
74	17	17	BENAR
75	17	17	BENAR
76	17	17	BENAR
77	17	17	BENAR
78	17	17	BENAR
79	17	17	BENAR
80	17	17	BENAR
81	21	21	BENAR
82	21	21	BENAR
83	21	21	BENAR
84	21	21	BENAR
85	21	21	BENAR
86	21	21	BENAR



87	21	21	BENAR
88	21	21	BENAR
89	21	21	BENAR
90	21	21	BENAR
91	25	25	BENAR
92	25	25	BENAR
93	25	25	BENAR
94	25	25	BENAR
95	25	25	BENAR
96	25	25	BENAR
97	25	25	BENAR
98	25	25	BENAR
99	25	25	BENAR
100	25	25	BENAR

Keterangan: Klasifikasi Salah



Gambar 6.1 Perbandingan Nilai Standar Deviasi Donat dan Rendang

Banyaknya salah pengklasifikasian rendang menjadi donat disebabkan karena donat dan rendang memiliki nilai data latih yang tidak terlalu jauh pada setiap fitur yang ada pada tiap kelas. Hal tersebut dapat dilihat dari nilai standar deviasi dari kedua kelas yang berasal dari data latih memiliki nilai yang hampir berdekatan pada setiap fitur terutama pada fitur warna seperti yang digambarkan pada Gambar 6.1. Karena kedua makanan yaitu donat dan rendang memiliki warna yang sebagian besar warnanya adalah warna cokelat.

Pada proses pengujian ini juga terdapat salah pengklasifikasian pada mie goreng yang diklasifikasikan menjadi donat, stroberi menjadi telur dadar, dan rendang menjadi donat seperti yang ada pada Tabel 6.3

Tabel 6.3 Kesalahan Klasifikasi Pada Pengujian

No	Citra Makanan	Hasil Klasifikasi	Kelas Asli
1		Donat	Mie Goreng
2		Telur Dadar	Stroberi
3		Donat	Rendang

6.2 Pengujian Terhadap Dimensi Data Citra

Pengujian ini memiliki tujuan untuk mencari tahu pengaruh dari ukuran dimensi data uji terhadap tingkat akurasi. Pengujian dibagi menjadi 3 bagian yaitu pengujian dengan menggunakan data uji berukuran 300x300 piksel, pengujian dengan menggunakan data uji berukuran 200x200 piksel, pengujian dengan menggunakan data uji berukuran 100x100 piksel. Pengujian dilakukan dengan menggunakan citra makanan berjumlah 100 data uji untuk tiap kelas makanan dan data latih sebanyak 202 citra yang berukuran 400x400 piksel.

Tabel 6.4 Hasil Akurasi Pengujian Terhadap Dimensi Citra

No	Jumlah Data Latih	Jumlah Data Uji	Ukuran Piksel	Hasil Akurasi
1	202	100	300x300	81%
2	202	100	200x200	79%
3	202	100	100x100	72%

Berikut adalah contoh pengujian dengan menggunakan citra berukuran 300x300 piksel dan 100 data uji untuk masing-masing kelas.

Tabel 6.5 Pengujian Data Uji Berukuran 300x300 Piksel

No	Data	Prediksi	Hasil
1	1	1	BENAR
2	1	1	BENAR

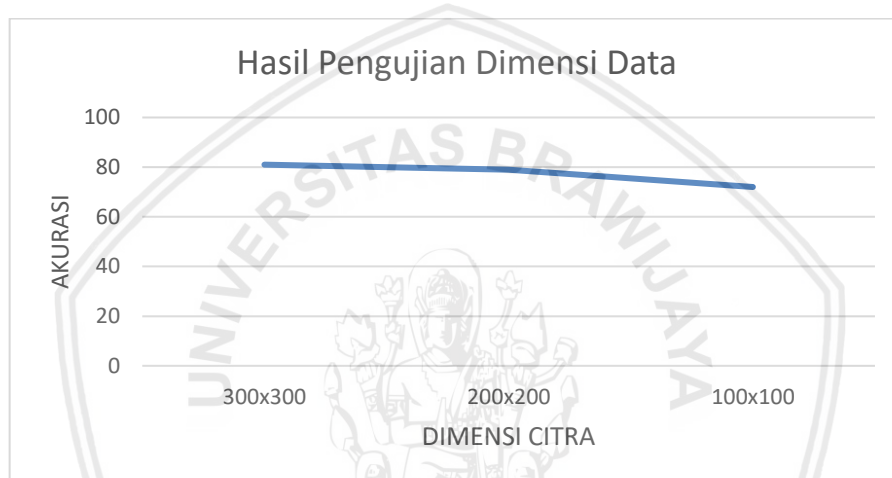
Tabel 6.5 (Lanjutan)

3	1	1	BENAR
4	1	1	BENAR
5	1	1	BENAR
6	1	1	BENAR
7	1	21	SALAH
8	1	21	SALAH
9	1	1	BENAR
10	1	1	BENAR
11	2	2	BENAR
12	2	7	SALAH
13	2	2	BENAR
14	2	2	BENAR
15	2	2	BENAR
16	2	2	BENAR
17	2	2	BENAR
18	2	2	BENAR
19	2	2	BENAR
20	2	2	BENAR
21	4	1	SALAH
22	4	4	BENAR
23	4	4	BENAR
24	4	4	BENAR
25	4	7	SALAH
26	4	1	SALAH
27	4	4	BENAR
28	4	4	BENAR
29	4	4	BENAR
30	4	7	SALAH
31	7	7	BENAR
32	7	7	BENAR
33	7	7	BENAR
34	7	7	BENAR
35	7	7	BENAR
36	7	7	BENAR
37	7	7	BENAR
38	7	7	BENAR
39	7	7	BENAR
40	7	7	BENAR
41	9	9	BENAR
42	9	1	SALAH
43	9	1	SALAH
44	9	25	SALAH
45	9	1	SALAH
46	9	7	SALAH



47	9	1	SALAH
48	9	9	BENAR
49	9	9	BENAR
50	9	1	SALAH
51	13	13	BENAR
52	13	13	BENAR
53	13	1	SALAH
54	13	13	BENAR
55	13	13	BENAR
56	13	13	BENAR
57	13	13	BENAR
58	13	13	BENAR
59	13	13	BENAR
60	13	1	SALAH
61	16	16	BENAR
62	16	16	BENAR
63	16	16	BENAR
64	16	16	BENAR
65	16	16	BENAR
66	16	16	BENAR
67	16	4	SALAH
68	16	16	BENAR
69	16	16	BENAR
70	16	7	SALAH
71	17	17	BENAR
72	17	17	BENAR
73	17	4	SALAH
74	17	17	BENAR
75	17	17	BENAR
76	17	17	BENAR
77	17	17	BENAR
78	17	17	BENAR
79	17	17	BENAR
80	17	17	BENAR
81	21	21	BENAR
82	21	21	BENAR
83	21	21	BENAR
84	21	21	BENAR
85	21	21	BENAR
86	21	21	BENAR
87	21	21	BENAR
88	21	21	BENAR
89	21	21	BENAR
90	21	21	BENAR

91	25	25	BENAR
92	25	25	BENAR
93	25	25	BENAR
94	25	25	BENAR
95	25	25	BENAR
96	25	25	BENAR
97	25	25	BENAR
98	25	25	BENAR
99	25	25	BENAR
100	25	25	BENAR



Gambar 6.2 Perbandingan Hasil Akurasi Pengujian Dengan Dimensi Data

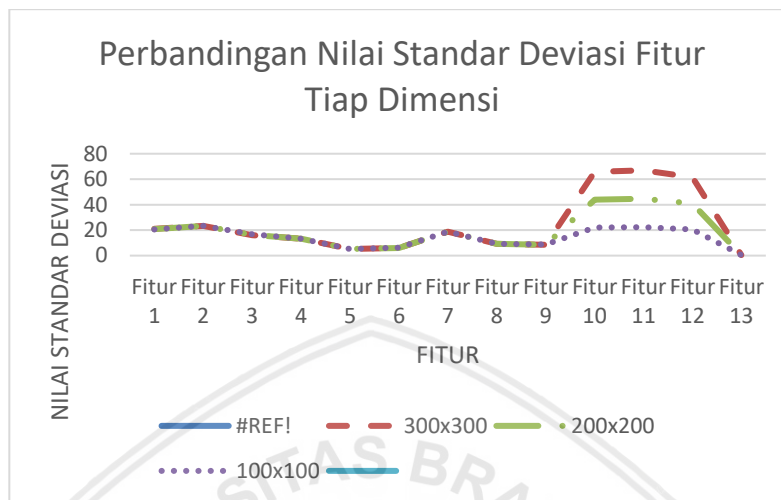
Pada pengujian yang telah dilakukan, jumlah data uji yang digunakan adalah 100 data uji dari setiap kelas yang dibagi menjadi 3 bagian yaitu 300x300 piksel, 200x200 piksel, dan 100x100 piksel dengan 202 data latih berukuran 400x400 piksel untuk masing-masing pengujian.

Dari proses pengujian terhadap dimensi data, hasil akurasi paling tinggi adalah hasil akurasi dari proses pengujian dengan dimensi data 300x300 piksel yaitu 81%. Sedangkan untuk pengujian dengan dimensi data 200x200 piksel menghasilkan akurasi sebesar 79% dan dimensi data 100x100 piksel menghasilkan akurasi paling rendah yaitu 72% seperti yang digambarkan pada Gambar 6.3.

Dari hasil proses pengujian tersebut disimpulkan bahwa dimensi dari data uji dapat mempengaruhi hasil akurasi dari proses klasifikasi citra makanan. Hal tersebut terlihat dimana nilai akurasi dari proses klasifikasi selalu menurun pada saat dimensi citra yang digunakan semakin kecil.

Penurunan nilai akurasi pada saat dimensi citra semakin kecil tersebut dipengaruhi karena, perbedaan nilai yang besar pada nilai fitur 10, fitur 11, dan fitur 12. Hal tersebut dapat dilihat pada nilai standar deviasi dari masing-masing fitur yang berubah sangat jauh pada fitur 10, fitur 11, fitur 12 di mana fitur tersebut adalah fitur yang berasal dari ekstraksi fitur bentuk. Hasil standar deviasi


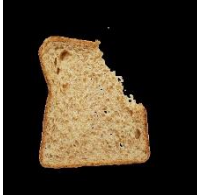
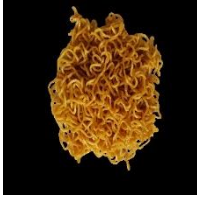
dari ekstraksi fitur warna tidak mengalami perubahan yang signifikan pada saat ukuran dimensi citra yang digunakan berbeda-beda, seperti yang terlihat pada nilai standar deviasi dari fitur 1 sampai dengan fitur 9 seperti yang digambarkan pada Gambar 6.4.



Gambar 6.3 Perbandingan Nilai Standar Deviasi Dimensi Citra

Pada proses pengujian dengan dimensi 300x300 terdapat kesalahan pengklasifikasian data uji yang berupa donat diklasifikasikan menjadi kelas telur nasi kuning. Lalu, pada pengujian dengan dimensi 200x200 terdapat kesalahan pengklasifikasian di mana citra roti gandum diklasifikasikan menjadi kelas telur dadar. Pada proses pengujian 100x100 terdapat kesalahan pengklasifikasian di mana citra mie goreng diklasifikasikan menjadi kelas donat.

Tabel 6.6 Kesalahan Klasifikasi Pada Pengujian Dimensi

No	Citra Makanan	Dimensi	Hasil Klasifikasi	Kelas Asli
1		300x300	Nasi Kuning	Donat
2		200x200	Telur Dadar	Roti Gandum
3		100x100	Donat	Mie Goreng

6.3 Pengujian Terhadap Jumlah Metode Ekstraksi Fitur

Pengujian ini memiliki tujuan untuk mencari tahu pengaruh dari banyaknya fitur yang digunakan dari citra terhadap tingkat akurasi. Pengujian dibagi menjadi 3 bagian yaitu pengujian dengan menggunakan fitur dari *color moment* dan *morphological shape descriptors*, pengujian dengan *color moment*, dan pengujian dengan *morphological shape descriptors*. Pengujian dilakukan dengan menggunakan citra makanan berukuran 400x400 piksel sebanyak 100 data dan data latih sebanyak 202 citra.

Tabel 6.7 Hasil Akurasi Pengujian Terhadap Jumlah Fitur

No	Jumlah Data Latih	Jumlah Data Uji	Metode	Hasil Akurasi
1	202	100	CM dan MSD	78%
2	202	100	CM	83%
3	202	100	MSD	32%

Berikut adalah contoh pengujian dengan menggunakan 1 metode ekstraksi fitur yaitu fitur warna.

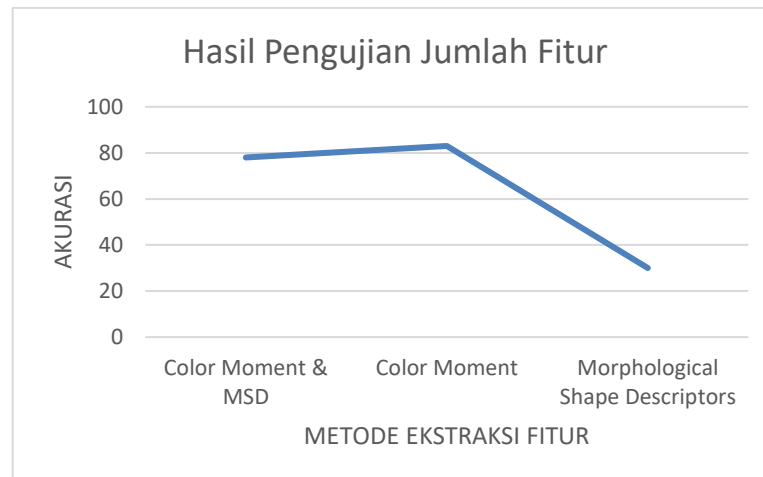
Tabel 6.8 Pengujian Dengan 1 Metode Ekstraksi Fitur Warna

No	Data	Prediksi	Hasil
1	1	1	BENAR
2	1	4	SALAH
3	1	4	SALAH
4	1	25	SALAH
5	1	1	BENAR
6	1	1	BENAR
7	1	21	SALAH
8	1	1	BENAR
9	1	1	BENAR
10	1	1	BENAR
11	2	2	BENAR
12	2	25	SALAH
13	2	2	BENAR
14	2	2	BENAR
15	2	2	BENAR
16	2	2	BENAR
17	2	2	BENAR
18	2	2	BENAR
19	2	2	BENAR
20	2	2	BENAR
21	4	4	BENAR
22	4	4	BENAR
23	4	4	BENAR

24	4	4	BENAR
25	4	4	BENAR
26	4	4	BENAR
27	4	4	BENAR
28	4	4	BENAR
29	4	4	BENAR
30	4	4	BENAR
31	7	7	BENAR
32	7	1	SALAH
33	7	17	SALAH
34	7	21	SALAH
35	7	7	BENAR
36	7	9	SALAH
37	7	1	SALAH
38	7	17	SALAH
39	7	21	SALAH
40	7	7	BENAR
41	9	9	BENAR
42	9	9	BENAR
43	9	25	SALAH
44	9	9	BENAR
45	9	9	BENAR
46	9	7	SALAH
47	9	9	BENAR
48	9	9	BENAR
49	9	7	SALAH
50	9	9	BENAR
51	13	13	BENAR
52	13	13	BENAR
53	13	13	BENAR
54	13	13	BENAR
55	13	13	BENAR
56	13	13	BENAR
57	13	13	BENAR
58	13	13	BENAR
59	13	13	BENAR
60	13	13	BENAR
61	16	21	SALAH
62	16	16	BENAR
63	16	16	BENAR
64	16	16	BENAR
65	16	16	BENAR
66	16	16	BENAR
67	16	16	BENAR

68	16	16	BENAR
69	16	16	BENAR
70	16	16	BENAR
71	17	17	BENAR
72	17	17	BENAR
73	17	4	SALAH
74	17	17	BENAR
75	17	17	BENAR
76	17	17	BENAR
77	17	17	BENAR
78	17	17	BENAR
79	17	17	BENAR
80	17	17	BENAR
81	21	21	BENAR
82	21	21	BENAR
83	21	21	BENAR
84	21	21	BENAR
85	21	21	BENAR
86	21	21	BENAR
87	21	21	BENAR
88	21	21	BENAR
89	21	21	BENAR
90	21	21	BENAR
91	25	25	BENAR
92	25	25	BENAR
93	25	25	BENAR
94	25	25	BENAR
95	25	25	BENAR
96	25	25	BENAR
97	25	25	BENAR
98	25	25	BENAR
99	25	25	BENAR
100	25	25	BENAR

Keterangan:  Klasifikasi Salah



Gambar 6.4 Perbandingan Hasil Akurasi Pengujian Dengan Jumlah Fitur

Pada pengujian yang telah dilakukan, jumlah data uji yang digunakan adalah 10 data uji dari setiap kelas yang dibagi menjadi 3 bagian pengujian yaitu pengujian dengan fitur *color moment* dan *morphological shape descriptors*, pengujian dengan fitur *color moment*, dan pengujian fitur *morphological shape descriptors* dengan 202 data latih berukuran 400x400 piksel untuk masing-masing pengujian.



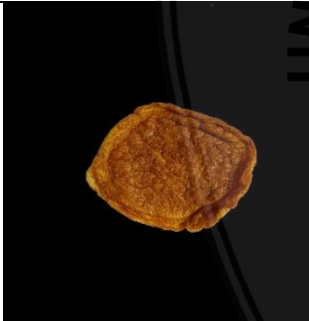
Dari proses pengujian terhadap dimensi data, hasil akurasi paling tinggi adalah hasil akurasi dari proses pengujian dengan fitur *color moment* yaitu 83%. Sedangkan untuk pengujian dengan fitur *morphological shape descriptors* menghasilkan nilai akurasi yang rendah yaitu 32%.

Dari hasil proses pengujian tersebut fitur *color moment* menghasilkan nilai akurasi yang paling tinggi jika dibandingkan dengan penggunaan 2 fitur dan fitur *morphological shape descriptors*. Dapat disimpulkan penambahan fitur tidak terlalu memberikan hasil yang signifikan terhadap nilai akurasi klasifikasi dan mengurangi nilai akurasi dari hasil klasifikasi.

Rendahnya nilai akurasi pada pengujian dengan menggunakan *morphological shape descriptors* terjadi jumlah fitur yang digunakan hanya sebanyak 4, sedangkan untuk metode ekstraksi fitur *color moment* banyaknya fitur yang digunakan adalah 9.

Kesalahan pengklasifiasian data uji pada pengujian jumlah fitur terdapat pada Tabel 6.9. Pada proses pengujian data yang diekstraksi fitur menggunakan metode *color moment* dan *morphological shape descriptors* terdapat data uji yang diklasifikasikan menjadi kelas mie soba padahal kelas seharusnya adalah donat. Lalu, pada pengujian dengan fitur *morphological shape descriptors* terdapat kesalahan pengklasifikasian pada citra selada yang diklasifikasikan menjadi kelas stroberi. Setelah itu, pada pengujian dengan *color moment* terjadi kesalahan pengklasifikasian pada citra telur dadar yang telah diekstraksi fitur dengan *color moment* yang diklasifikasi menjadi kelas donat.

Tabel 6.9 Kesalahan Klasifikasi Pada Pengujian Jumlah Metode Ekstraksi Fitur

No	Data Uji	Metode	Hasil Klasifikasi	Kelas Asli
1		CM dan MSD	Mie Soba	Donat
2		MSD	Stroberi	Selada
3		CM	Donat	Telur Dadar

BAB 7

KESIMPULAN

Pada bab ini akan membahas mengenai kesimpulan yang diambil berdasarkan hasil dari pengujian yang telah dilakukan pada bab sebelumnya dan berisi saran untuk perbaikan dalam pelaksanaan penelitian kedepannya.

7.1 Kesimpulan

Hasil dari proses perancangan, pengimplementasian, pengujian dan analisis yang dilakukan pada bab-bab sebelumnya disimpulkan bahwa:

1. Metode ekstraksi fitur *color moment* dan *morphological shape descriptors* dengan metode klasifikasi *Naive Bayes* dapat mengklasifikasikan jenis makanan seperti donat, roti gandum, mie goreng, telur dadar, rendang, selada, stroberi, pisang hijau, nasi kuning, dan mie soba dengan tingkat akurasi 83%. Hasil akurasi tersebut didapatkan dengan menggunakan pengujian 1 metode ekstraksi fitur yaitu metode ekstraksi fitur *color moment*.
2. Metode ekstraksi fitur *color moment* dan *morphological shape descriptors* dengan metode klasifikasi *Naive Bayes* dapat mengklasifikasikan jenis makanan seperti donat, roti gandum, mie goreng, telur dadar, rendang, selada, stroberi, pisang hijau, nasi kuning, dan mie soba dengan tingkat akurasi 78%. Hasil akurasi tersebut didapatkan dengan menggunakan pengujian 100 data uji dan 202 data latih.
3. Metode ekstraksi fitur *color moment* dan *morphological shape descriptors* dengan metode klasifikasi *Naive Bayes* dapat mengklasifikasikan jenis makanan seperti donat, roti gandum, mie goreng, telur dadar, rendang, selada, stroberi, pisang hijau, nasi kuning, dan mie soba dengan tingkat akurasi 81%. Hasil akurasi terbesar tersebut didapatkan dengan menggunakan pengujian 100 data uji yang memiliki dimensi citra yaitu 300x300 piksel dan 202 data latih yang memiliki dimensi citra yaitu 400x400 piksel.

7.2 Saran

Dalam penelitian yang telah dilakukan masih ada beberapa kekurangan, oleh karena itu terdapat saran yang dapat digunakan untuk menghasilkan hasil yang lebih baik dan maksimal untuk penelitian di masa yang akan datang.

1. Mempertahankan metode ekstraksi fitur warna karena memiliki nilai yang baik pada saat digunakan dalam keadaan dimensi citra yang berbeda-beda.
2. Menambahkan metode seleksi fitur agar fitur-fitur yang berperan penting terpilih pada saat proses pelatihan data dan hasil akurasi yang dihasilkan semakin meningkat.

DAFTAR REFERENSI

- Alpaydin, E. (2010). *Introduction to Machine Learning Second Edition*. Cambridge, Massachusetts, London: The MIT Press.
- Boodoo-Jahangeer, N. B., & Baichoo, S. (2013). Face Recognition Using Chain Codes. *Journal of Signal and Information Processing*, 154-157.
- Burges, C. J. (1998). A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*(2), 121-167.
- Ciocca, G., Napoletano, P., & Schettini, R. (2017). Food Recognition: A New Dataset, Experiments, and Results. *IEEE Journal Of Biomedical and Health Informatics*, 588-598.
- Dong-Chul, P. (2016). Image Classification Using Naive Bayes Classifier. *International Journal of Computer Science and Electronics Engineering*, 135-139.
- Farinella, G. M., Allegra, D., Moltisanti, M., & Stanco, F. (2016). Retrieval and Classification of Food Images. *Computers in Biology and Medicine*, 23-39.
- Hardinsyah, Riyaldi, H., & Napitupulu, V. (2013). Kecukupan Energi, Protein, Lemak, dan Karbohidrat. *Makalah WKNPG*, 1-27.
- Khatua, C. K., Nayak, S. K., & Panda, D. C. (2015). Content Based Image Retrieval Using Color Histogram. *International Journal Of Current Research*(7), 21692-21703.
- Kulkarni, A. A., Kurundkar, R. D., & Khare, S. V. (2015). A Review on Color Image Processing. *Internation Journal of Computer Science and Mobile Applications*(3), 5-10.
- Kusuma, S. F., Pawening, R. E., & Dijaya, R. (2016). Otomatisasi Klasifikasi Kematangan Buah Mengkudu Berdasarkan Warna dan Tekstur. *Jurnal Ilmiah Teknologi Sistem Informasi*, 17-23.
- Lukac, R. (2013). *Perceptual Digital Imaging: Methods And Applications* (Ke1 ed.). Boca Raton: CRC Press.
- Maruyama, Y., de Silva, G. C., Yamasaki, T., & Aizawa, K. (2010). Personalization of Food Image Analysis. *2010 16th International Conference on Virtual Systems and Multimedia* (pp. 75-78). Seoul: IEEE.
- Mezgec, S., & Seljak, B. K. (2017). NutriNet: A Deep Learning Food and Drink Image Recognition System for Dietary Assessment. *Nutrients*, 657.
- Murphy, K. P. (2006). *Naive Bayes Classifiers*. Vancouver: Departement of Computer Science, University of British Columbia.
- Nielsen, F., & Sun, K. (2016). Guaranteed Bounds on Information-Theoretic Measures of Univariate Mixtures Using Piecewise Log-Sum-Exp Inequalities. *Entropy*, 1-25.

- Riyanto. (2014). *Validasi & Verifikasi Metode Uji Sesuai dengan ISO/IEC 17025 Laboratorium Pengujian dan Kalibrasi*. Yogyakarta: Deepublish.
- Soman, S., Ghorpade, M., & Sonone, V. (2012). Content Based Image Retrieval Using Advanced Color and Texture Features. *International Journal of Computer Application (ICCIA)* 3(4), 1-5.
- Taheri, S., & Mammadov, M. (2013). Learning The Naive Bayes Classifier With Optimization Models. *International Journal Application Mathematic Computer Science*(23), 787-795.
- Vinayak, V., & Sonika, J. (2017). CBIR System using Color Moment and Color Auto-Correlogram with Block Truncation Coding. *International Journal of Computer Applications*.
- Wäldchen, J., & Mäder, P. (2017). Plant Species Identification Using Computer Vision Techniques: A Systematic Literature Review. *Archives of Computational Methods In Engineering*, 1-37.

