

**KLASIFIKASI CITRA MAKANAN MENGGUNAKAN *K-NEAREST
NEIGHBOR* DENGAN FITUR BENTUK *SIMPLE
MORPHOLOGICAL SHAPE DESCRIPTORS* DAN FITUR WARNA
*GRAYSCALE HISTOGRAM***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Muhammad Rizky Setiawan

NIM: 155150201111134



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

PENGESAHAN

KLASIFIKASI CITRA MAKANAN MENGGUNAKAN *K-NEAREST NEIGHBOR* DENGAN
FITUR BENTUK *SIMPLE MORPHOLOGICAL SHAPE DESCRIPTORS* DAN FITUR
WARNA *GRAYSCALE HISTOGRAM*

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh:
Muhammad Rizky Setiawan
NIM: 155150201111134

Skripsi ini telah diuji dan dinyatakan lulus pada
3 Januari 2019

Telah diperiksa dan disetujui oleh:

Pembimbing I

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

Pembimbing II

Putra Pandu Adikara, S.Kom., M.Kom
NIP: 19850725 200812 1 002

Mengetahui
Ketua Jurusan Teknik Informatika



Tri 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 pustaka.

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

Malang, 4 Januari 2019



Muhammad Rizky Setaiwan

NIM: 155150201111134

PRAKATA

Puji syukur penulis ucapkan kepada Allah SWT yang telah melimpahkan rahmat dan karunia-Nya penulis dapat menyelesaikan skripsi ini. Shalawat serta salam tercurahkan pada Nabi Muhammad SAW. Pada kesempatan ini penulis juga menyampaikan rasa terima kasih kepada pihak – pihak yang telah membantu penulis sehingga dapat menyelesaikan skripsi ini, diantaranya:

1. Ibu Yuita Arum Sari, S.Kom, M.Kom selaku dosen pembimbing pertama yang telah memberikan bimbingan, saran, arahan, dan motivasi kepada penulis selama penyusunan skripsi.
2. Bapak Putra Pandu Adikara, S.Kom, M.Kom selaku dosen pembimbing kedua yang telah memberikan bimbingan, saran, dan arahan dalam penulisan kepada penulis selama penyusunan skripsi.
3. Bapak Agus Wahyu Widodo. S.T, M.Cs selaku Ketua Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang.
4. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya Malang.
5. Ibu Lailil Muflikhah, S.Kom, M.Sc selaku dosen pendamping akademik yang telah memberikan bimbingan, saran, arahan, dan motivasi kepada penulis selama menempuh Pendidikan di Fakultas Ilmu Komputer Universitas Brawijaya.
6. Seluruh Dosen Fakultas Ilmu Komputer Universitas Brawijaya yang telah memberikan ilmu yang bermanfaat kepada penulis selama menempuh pendidikan.
7. Kedua orang tua serta keluarga penulis yang telah memberikan dukungan kepada penulis selama menempuh Pendidikan di Fakultas Ilmu Komputer Universitas Brawijaya.
8. Sahabat terdekat penulis selama menempuh pendidikan pada perkuliahan yaitu: Yunita Ayu, Ian Lord, Yosua Dwi Amerta, Gabriel Mulyawan dan Jimmy S. Dachi atas dukungan, dan bantuan yang diberikan selama masa Pendidikan kepada penulis.

Penulis menyadari dalam penyusunan skripsi ini masih banyak kesalahan dan kekurangan baik dalam teknik penyajian materi maupun pembahasan. Demi kemajuan penulis, saran dan kritik yang sifatnya membangun sangat penulis harapkan. Semoga skripsi ini dapat bermanfaat bagi pembaca secara umum dan penulis. Akhir kata penulis ucapkan banyak terima kasih.

Malang, 4 Januari 2019

Muhammad Rizky Setiawan

rizky03setiawan@student.ub.ac.id

ABSTRAK

Muhammad Rizky Setiawan, Klasifikasi Citra Makanan Menggunakan *K-Nearest Neighbor* dengan Fitur Bentuk *Simple Morphological Shape Descriptors* dan Fitur Warna *Grayscale Histogram*

Pembimbing: Yuita Arum Sari, S.Kom., M.Kom dan Putra Pandu Adikara, S.Kom., M.Kom.

Salah satu sumber energi yang diperlukan oleh masyarakat adalah makanan. Makanan yang dikonsumsi dapat berpengaruh terhadap daya tahan tubuh. Namun, masyarakat sulit mengenali jenis makanan yang dikonsumsi karena keanekaragaman makanan semakin banyak. Hal ini menyebabkan pola makan masyarakat menjadi tidak teratur. Diperlukannya suatu sistem yang dapat mengenali jenis makanan sehingga dapat mempermudah masyarakat dalam mengatur pola makan. Tahap pertama yang dilakukan adalah melakukan *preprocessing* untuk mendapatkan objek makanan pada citra. Selanjutnya dilakukan ekstraksi fitur bentuk *Simple Morphological Shape Descriptors* (SMSD) yang terdiri dari fitur *length*, *width*, *area*, *rectangularity N*, dan *aspect ratio*. Kemudian dilakukan ekstraksi fitur warna *Grayscale Histogram* yang menghasilkan fitur *mean*, *standard deviation*, dan *skewness*. Setelah didapatkan hasil ekstraksi fitur, dilakukan klasifikasi citra makanan menggunakan metode *K-Nearest Neighbor* (KNN). Berdasarkan pengujian yang telah dilakukan, ketika menggunakan rasio jumlah data 10:90 menghasilkan nilai akurasi sebesar 70,8%. Nilai *k* yang optimal pada penelitian ini yaitu *k*=1 dengan nilai akurasi sebesar 79,2%. Jika menggunakan metode *Grayscale Histogram* dan metode SMSD menghasilkan nilai akurasi sebesar 77,8%. Metode *Grayscale Histogram* dan metode SMSD dapat digunakan untuk klasifikasi citra makanan menggunakan metode klasifikasi KNN.

Kata kunci: Makanan, Klasifikasi, *Grayscale Histogram*, *Simple Morphological Shape Descriptors*, *K-Nearest Neighbor*

ABSTRACT

Muhammad Rizky Setiawan, Klasifikasi Citra Makanan Menggunakan *K-Nearest Neighbor* dengan Fitur Bentuk *Simple Morphological Shape Descriptors* dan Fitur Warna *Grayscale Histogram*

Pembimbing: Yuita Arum Sari, S.Kom., M.Kom dan Putra Pandu Adikara, S.Kom., M.Kom.

One of the energy needed by the community is food. Foods that can be used for endurance. However, people find it difficult to use food. This causes people's eating patterns to become irregular. The need for a system that can be used to help people manage their diet. The first step is to do preprocessing to get the food object in the image. Next is the feature extraction in the form of Simple Morphological Shape Descriptors (SMSD) which consists of features length, width, area, rectangularity N, and aspect ratio. Then Grayscale Histogram feature extraction is carried out which produces the mean, standard deviation, and skewness features. After obtaining the feature extraction results, food image classification was carried out using the K-Nearest Neighbor (KNN) method. Based on the tests that have been conducted, using a data ratio of 10:90 results in an accuracy value of 70.8%. The optimal k value in this study is $k = 1$ with an accuracy value of 79.2%. If using the Grayscale Histogram method and the SMSD method produces an accuracy value of 77.8%. The Grayscale Histogram method and the SMSD method can be used to process images using the KNN method.

Keywords: *Food, Classification, Grayscale Histogram, Simple Morphological Shape Descriptors, K-Nearest Neighbor*

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	Error! Bookmark not defined.
PRAKATA	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR	x
BAB 1 PENDAHULUAN	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah	3
1.6 Sistematika pembahasan	3
BAB 2 LANDASAN KEPUSTAKAAN	4
2.1 Kajian Pustaka	4
2.2 Citra Digital	4
2.2.2 Pengolahan Citra Digital.....	5
2.3 Ekstraksi Fitur.....	5
2.3.1 <i>Simple Morphological Shape Descriptors</i>	5
2.3.2 <i>Grayscale Histogram</i>	6
2.4 <i>K-Nearest Neighbor</i>	7
2.5 Evaluasi	8
BAB 3 METODOLOGI	10
3.1 Tipe Penelitian	10
3.2 Metode Penelitian	10
3.3 Lokasi Penelitian	11
3.4 Partisipan Penelitian	12
3.5 Peralatan Pendukung.....	12
3.6 Teknik Pengumpulan Data	12

BAB 4 PERANCANGAN	15
4.1 Perancangan Algoritme	15
4.1.1 <i>Preprocessing</i>	15
4.1.2 Ekstraksi Fitur <i>Grayscale Histogram</i>	15
4.1.3 Ekstraksi Fitur <i>Simple Morphological Shape Descriptors</i>	16
4.1.4 Klasifikasi <i>K-Nearest Neighbor</i>	17
4.2 Perhitungan Manualisasi	18
4.2.1 Perhitungan <i>Grayscale Histogram</i>	19
4.2.2 Perhitungan <i>Simple Morphological Shape Descriptors</i>	20
4.2.3 Perhitungan <i>K-Nearest Neighbor</i>	21
BAB 5 IMPLEMENTASI.....	23
5.1 Implementasi Algoritme	23
5.1.1 Implementasi <i>Preprocessing</i>	23
5.1.2 Implementasi Ekstraksi Fitur <i>Grayscale Histogram</i>	24
5.1.3 Implementasi Ekstraksi Fitur <i>Simple Morphological Shape Descriptors</i>	25
5.1.4 Implementasi <i>K-Nearest Neighbor</i>	26
BAB 6 HASIL PENGUJIAN DAN ANALISIS	28
6.1 Pengujian Rasio Jumlah Data	28
6.2 Pengujian Nilai <i>k</i>	29
6.3 Pengujian Fitur Warna <i>Grayscale Histogram</i>	32
6.4 Pengujian Fitur Bentuk <i>Simple Morphological Shape Descriptors</i>	34
6.5 Pengujian Fitur Warna <i>Grayscale Histogram</i> dan Fitur Bentuk <i>Simple Morphological Shape Descriptors</i>	37
BAB 7 KESIMPULAN	45
7.1 Kesimpulan	45
7.2 Saran	45
DAFTAR REFERENSI	46
LAMPIRAN A DATASET	47
A.1 DATA LATIH	47
A.2 DATA UJI	72

DAFTAR TABEL

Tabel 2.1 <i>Confusion Matrix Multiple Class</i>	8
Tabel 3.1 Jenis Makanan yang Digunakan	12
Tabel 4.1 Nilai Piksel 3x3	19
Tabel 4.2 Nilai <i>Histogram</i>	19
Tabel 4.3 Nilai <i>First Order Histogram</i>	19
Tabel 4.4 Nilai Fitur <i>Grayscale Histogram</i>	20
Tabel 4.5 Nilai <i>Euclidean Distance</i>	21
Tabel 4.6 Jumlah Nilai <i>Euclidean Distance</i>	22
Tabel 6.1 Daftar Kelas dan Kode Kelas	28
Tabel 6.2 Hasil Pengujian Rasio Jumlah Data	28
Tabel 6.3 Hasil Akurasi Pengujian Nilai <i>k</i> dan Perbandingan Jumlah Data	29
Tabel 6.4 Contoh Kesalahan Klasifikasi Karena Nilai <i>k</i>	31
Tabel 6.5 <i>Confusion Matrix</i> Metode <i>Grayscale Histogram</i>	32
Tabel 6.6 Evaluasi Pengujian Fitur Warna <i>Grayscale Histogram</i> Per Kelas	33
Tabel 6.7 Micro & Macro Averaging Fitur Warna <i>Grayscale Histogram</i>	33
Tabel 6.8 <i>Confusion Matrix</i> Metode SMSD	35
Tabel 6.9 Evaluasi Pengujian Fitur Bentuk SMSD Per Kelas	35
Tabel 6.10 Micro & Macro Averaging Fitur SMSD	36
Tabel 6.11 Hasil Pengujian Data Uji Fitur <i>Grayscale Histogram</i> dan Fitur SMSD .	37
Tabel 6.12 <i>Confusion Matrix</i> Hasil Pengujian Fitur <i>Grayscale Histogram</i> dan Fitur SMSD	40
Tabel 6.13 Evaluasi Pengujian Fitur Warna <i>Grayscale Histogram</i> dan SMSD Per Kelas	41
Tabel 6.14 Micro & Macro Averaging Fitur <i>Grayscale Histogram</i> dan SMSD	42
Tabel 6.15 Perbandingan Kombinasi Fitur	43

DAFTAR GAMBAR

Gambar 3.1 Diagram Alir Metode Pelatihan.....	10
Gambar 3.2 Diagram Alir Metode Pengujian	11
Gambar 4.1 Diagram Alir <i>preprocessing</i>	15
Gambar 4.2 Diagram Alir Ekstraksi Fitur <i>Grayscale Histogram</i>	16
Gambar 4.3 Diagram Alir ekstraksi Fitur <i>Simple Morphological Shape Descriptors</i>	17
Gambar 4.4 Diagram Alir Klasifikasi <i>K-Nearest Neighbor</i>	18
Gambar 5.1 Keluaran Program Hasil Klasifikasi	27
Gambar 6.1 Grafik Hasil Pengujian Rasio Jumlah Data	29
Gambar 6.2 Grafik Hasil Pengujian Nilai <i>k</i>	30
Gambar 6.3 Data Uji.....	31
Gambar 6.4 Grafik <i>Micro Averaging Grayscale Histogram</i>	34
Gambar 6.5 Grafik <i>Macro Averaging Grayscale Histogram</i>	34
Gambar 6.6 Grafik <i>Micro Averaging</i> Fitur SMSD	36
Gambar 6.7 Grafik <i>Macro Averaging</i> Fitur SMSD	37
Gambar 6.8 Grafik <i>Micro Averaging</i> Fitur <i>Grayscale Histogram</i> dan SMSD.....	42
Gambar 6.9 Grafik <i>Macro Averaging Grayscale Histogram</i> dan SMSD	42
Gambar 6.10 Grafik <i>Micro Averaging</i> Perbandingan Pengujian	43
Gambar 6.11 Grafik <i>Macro Averaging</i> Perbandingan Pengujian.....	43

BAB 1

PENDAHULUAN

1.1 Latar belakang

Salah satu sumber energi yang diperlukan oleh masyarakat adalah makanan (Saparinto & Hidayati, 2010). Makanan yang dikonsumsi oleh masyarakat berpengaruh dalam pemeliharaan jaringan dalam tubuh karena membutuhkan gizi. Kekurangan gizi dapat menyebabkan menurunnya daya tahan tubuh, menurunnya produktivitas kerja serta terjadinya gangguan kecerdasan. Makanan merupakan kebutuhan yang harus dipenuhi dengan baik agar tubuh tidak menjadi lelah dan juga lemas.

Keanekaragaman makanan yang ada di masyarakat menyebabkan masyarakat kesulitan dalam mengenali jenis makanan. Sulitnya masyarakat dalam mengenali makanan menyebabkan pilihan makanan yang dikonsumsi menjadi tidak seimbang kandungan gizinya. Seperti makanan *fast food* yang mudah didapatkan karena daya tarik dari teknik promosi (Mudjiyanto, et al., 1994). Jenis makanan yang dikonsumsi sangat penting karena dapat mempengaruhi daya tahan tubuh. Dengan mengetahui jenis makanan, masyarakat dapat mengatur asupan makanan yang sehat dan bergizi oleh tubuh.

Suatu cara untuk mengidentifikasi jenis makanan perlu dilakukan untuk mempermudah dan mempercepat waktu dalam pengenalan makanan. Perkembangan ilmu dalam bidang teknologi dapat diterapkan sebagai salah satu alat bantu klasifikasi jenis makanan tersebut. Klasifikasi merupakan suatu proses penggabungan atau pengelompokan dua atau lebih data sesuai dengan kesamaan suatu kriteria tertentu. Metode klasifikasi yang biasa digunakan antara lain metode *C4.5*, *K-Nearest Neighbor* (KNN), *ID3*, *Naïve Bayes* dan metode lainnya. *K-Nearest Neighbor* (KNN) termasuk metode klasifikasi dalam kelompok *Instance-Based Learning*. Berdasarkan jaraknya paling dekat dengan objek tersebut dari data pembelajaran, KNN dapat dilakukan. Penelitian yang dilakukan oleh Budianita tentang penentuan daging sapi dan babi menggunakan metode KNN menghasilkan akurasi sebesar 86,87% (2015). Metode KNN adalah suatu metode yang sederhana dan efektif, sehingga lebih akurat hasilnya ketika memiliki data *training* yang besar. Selain itu, ada juga klasifikasi citra menggunakan metode *Naïve Bayes* mengenai kematangan buah apel (Ciputra, et al., 2018).

Pada penelitian yang dilakukan oleh Wäldchen & Mäder tentang identifikasi spesies tanaman menggunakan *Simple Morphological Shape Descriptor* sebagai salah satu metode untuk melakukan ekstraksi fitur bentuk (2016). Penelitian ini menunjukkan penggunaan dan perhitungan *Morphological Descriptors* yang sering digunakan adalah rasio, karena perhitungan yang mudah dan invarian terhadap translasi, rotasi, dan perubahan skala. Hal itu membuat nilai rasio tetap bernilai baik walaupun dengan representasi objek yang berbeda (Wäldchen & Mäder, 2016).

Selain mengklasifikasikan berdasarkan bentuknya, bisa juga menggunakan ciri warna seperti yang dilakukan oleh Karmilasari & Sumarna tentang temu kembali citra dengan ekstraksi fitur menggunakan metode *Local Histogram Color* dan *Global Histogram Color* yang dilakukan oleh Karmilasari. Penelitian ini membandingkan hasil antara metode *Local Histogram Color* dan *Global Histogram Color*. Dari kedua metode tersebut, hasil akurasi tertinggi didapatkan ketika menggunakan metode *Local Histogram Color* informasi yang dihasilkan lebih detail berdasarkan blok-blok piksel (2011). Pada penelitian yang dilakukan oleh Kumar, et al. mengenai temu kembali citra menggunakan metode *Weighted Average* menghasilkan hasil yang bagus ketika menggunakan nilai *Grayscale* (2016).

Berdasarkan latar belakang yang sudah dijelaskan, metode yang digunakan dalam penelitian ini adalah metode *Simple Morphological Shape Descriptors* untuk ekstraksi fitur bentuk dan metode *Grayscale Histogram* untuk ekstraksi fitur warna serta metode *K-Nearest Neighbor* untuk klasifikasi pada citra makanan.

1.2 Rumusan masalah

1. Bagaimana merancang algoritme *Simple Morphological Shape Descriptors* dan *Grayscale Histogram* dalam mengekstraksi ciri bentuk dan warna pada citra makanan?
2. Bagaimana tingkat akurasi dari metode *K-Nearest Neighbor* dalam menentukan jenis makanan?
3. Bagaimana pengaruh ekstraksi fitur bentuk *Simple Morphological Shape Descriptors* dan ekstraksi fitur warna *Grayscale Histogram* terhadap hasil klasifikasi jenis makanan?

1.3 Tujuan

1. Dapat mengetahui rancangan algoritme *Simple Morphological Shape Descriptors* dan *Grayscale Histogram* dalam mengekstraksi ciri bentuk dan warna pada citra makanan.
2. Dapat mengetahui tingkat akurasi dari metode *K-Nearest Neighbor* dalam menentukan jenis makanan.
3. Dapat mengetahui pengaruh ekstraksi fitur bentuk *Simple Morphological Shape Descriptors* dan ekstraksi fitur warna *Grayscale Histogram* terhadap hasil klasifikasi jenis makanan.

1.4 Manfaat

Dari penelitian ini manfaatnya antara lain:

1. Dihasilkan sebuah sistem yang dapat mengklasifikasi jenis makanan berdasarkan bentuk dan warna dari citranya.
2. Diharapkan memperoleh pemahaman terhadap metode *K-Nearest Neighbor* yang digunakan dari hasil penelitian ini.

1.5 Batasan masalah

Dari penelitian ini batasan masalahnya sebagai berikut:

1. Makanan yang digunakan berupa makanan padat.
2. Data yang digunakan adalah data primer yang diambil menggunakan kamera pada *smartphone*.
3. Kelas yang digunakan berjumlah 12 yaitu donat, mie, ayam krispi, daging rendang, selada, tomat, pisang hijau, pisang kuning, nasi kuning, happy toss, biskuit, genjie pie.

1.6 Sistematika pembahasan

Sistematika yang digunakan untuk menyusun laporan penelitian ini adalah:

BAB 1 PENDAHULUAN

Bab ini membahas tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, dan sistematika pembahasan.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini membahas tentang teori-teori yang mendukung dalam penelitian.

BAB 3 METODOLOGI

Bab ini membahas tentang tahapan penelitian, metode penelitian, dan kebutuhan yang digunakan untuk penelitian.

BAB 4 PERANCANGAN

Bab ini berisi tentang perancangan metode yang digunakan pada penelitian

BAB 5 IMPLEMENTASI

Bab ini berisi tentang implementasi metode yang digunakan pada penelitian

BAB 6 PENGUJIAN

Bab ini berisi tentang pengujian dan analisis metode yang telah dirancang dan diimplementasikan.

BAB 7 KESIMPULAN

Bab ini membahas tentang kesimpulan dan saran dari penelitian.

BAB 2

LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Penelitian yang dilakukan oleh Wäldchen & Mäder mengenai identifikasi spesies tanaman menggunakan metode *Simple Morphological Shape Descriptors* (2016). Pada penelitian tersebut terdapat 19 *descriptor* yang digunakan untuk mengidentifikasi pada setiap tanaman (Wäldchen & Mäder, 2016). Berdasarkan analisisnya pada daun, 8 dari 10 daun yang dianalisis selalu menemukan deskriptor warna bersamaan dengan deskriptor bentuk pada daun. Selain itu, pencocokan fitur ekstraksi bentuk juga dapat dilakukan dengan cepat. Oleh karena itu banyak digunakan pada pengenalan objek, *content based image retrieval* (CBIR), kesamaan bentuk, dan klasifikasi daun (C, et al., 2014).

Penelitian berikutnya tidak mengklasifikasi citra berdasarkan bentuk melainkan warnanya, seperti yang dilakukan oleh Karmilasari (2011). Pada penelitian tersebut membandingkan antara metode *Local Histogram Color* dan metode *Global Histogram Color*. Hasil akurasi *Local Histogram Color* lebih tinggi karena informasi yang dihasilkan lebih *detail*. Keunggulan metode *Grayscale Histogram* ini sangat cepat dalam perbandingan komputasinya dan efisien. Selain itu, kecepatan dalam mengidentifikasi perbedaan citra yang berhubungan dengan warna sangat cepat (Mali & L., 2014). Selain itu, penggunaan channel *Grayscale* menghasilkan nilai akurasi yang lebih baik dibandingkan RGB karena dimensi vektor fitur yang digunakan telah dikurangi (Kumar, et al., 2016).

Kemudian penelitian yang dilakukan oleh Budianita tentang klasifikasi daging sapi dan babi menggunakan metode KNN dengan menggunakan ekstraksi fitur warna (2015). Kelebihan dari metode KNN adalah sederhana dan efektif dalam penggunaannya, sehingga lebih akurat hasilnya ketika memiliki data *training* yang besar. Pada penelitian tersebut dilakukan pengujian data yang digunakan yaitu dengan menggunakan *background* dan tanpa menggunakan *backgorund*. Pengujian data yang menghasilkan nilai akurasi tertinggi yaitu ketika tidak menggunakan *background* pada data dengan nilai akurasi sebesar 93,33%.

2.2 Citra Digital

Citra digital adalah kumpulan nilai yang terbatas dari representasi gambar dua dimensi yang disebut piksel. Nilai piksel biasanya merepresentasikan nilai tingkat keabuan, warna, tinggi, dan lainnya. Selain itu, *digitization* menyatakan bahwa citra digital adalah suatu perkiraan dari adegan nyata (Gonzalez, 2002).

Format gambar umumnya meliputi:

1. 1 sampel/titik (*B&W* atau *Grayscale*)
2. 3 sampel/titik (*Red, Green, Blue*)
3. 4 sampel/titik (*Red, Green, Blue, dan Alpha* atau Keabuan)

2.2.2 Pengolahan Citra Digital

Suatu proses pengolahan piksel pada citra digital untuk mencapai suatu tujuan tertentu disebut pengolahan citra digital. Tahapan yang umum digunakan dalam pengolahan citra digital antara lain (Gonzalez, 2002):

1. *Image Acquisition*
2. *Image Enhancement*
3. *Image Restoration*
4. *Morphological Processing*
5. *Segmentation*
6. *Object Recognition*
7. *Representation & Description*

2.3 Ekstraksi Fitur

Ekstraksi fitur adalah suatu pengambilan fitur atau ciri-ciri bagian penting dari citra yang memberikan pemahaman lebih rinci tentang citra. Setelah fitur diekstrak, maka analisis dapat dilakukan. Ekstraksi fitur dilakukan dengan pengecekan dari setiap piksel yang ada pada citra.

2.3.1 Simple Morphological Shape Descriptors

Simple Morphological Shape Descriptors merupakan metode ekstraksi fitur bentuk yang menganalisis berdasarkan geometris dasar pada objek seperti centroid, keliling, luas, dan minimal sumbu pada objek. Pada deskriptor dasar, terdapat fitur yang dapat digunakan yaitu *rectangularity*, *aspect ratio* dan *area*. *Rectangularity* mempresentasikan seberapa besar objek mengisi luas minimum persegi panjang. *Aspect ratio* menunjukkan lebar atau sempitnya sebuah objek. *Area* adalah jumlah piksel pada objek. Untuk menghitung nilai *rectangularity*, *aspect ratio* menggunakan Persamaan 2.1 dan Persamaan 2.2 (Wäldchen & Mäder, 2016).

$$rectangularity = \frac{A}{LW} \quad (2.1)$$

$$aspect\ ratio = \frac{L}{W} \quad (2.2)$$

Keterangan:

A: Area

L: Major axis length

W: Minoraxis length

2.3.2 Grayscale Histogram

Grayscale Histogram adalah deskripsi secara umum dari suatu citra. Histogram mempertimbangkan frekuensi relatif dari *gray level* dan informasi spasial dari citra tidak diperdulikan. *Grayscale Histogram* menghasilkan nilai piksel yang berbeda dengan nilai 8bit, mulai dari 0 sampai 255 (Kumar, et al., 2016). Citra yang berbeda dapat memiliki nilai *histogram* yang sama.

2.3.2.1 Fitur Histogram

Probabilitas *order* pertama *histogram* $p(g)$ ditunjukkan pada Persamaan 2.3 (Mali & L., 2014).

$$P(g) = \frac{N(g)}{M} \quad (2.3)$$

Keterangan:

g : *gray level*

$P(g)$: probabilitas *gray level* g pada citra

$N(g)$: nilai piksel *gray level* g pada citra

M : total piksel pada citra

Beberapa fitur berdasarkan probabilitas orde pertama histogram antara lain (Mali & L., 2014):

1. Mean

Citra yang cerah memiliki nilai rata-rata yang tinggi, sedangkan citra yang gelap memiliki nilai rata-rata yang rendah. Fungsi *mean* dapat didefinisikan menggunakan Persamaan 2.4.

$$Mean = \bar{g} = \sum_{g=0}^{L-1} gP(g) \quad (2.4)$$

Keterangan:

$\bar{g}$: *mean*

$P(g)$: probabilitas *gray level* g pada citra

g : *gray level*

L : Banyak nilai histogram

2. Standard Deviation

Standard deviation mendeskripsikan mengenai kontras suatu citra. Citra dengan kontras yang tinggi akan memiliki *standard deviation* yang tinggi.

Fungsi *standard deviation* dapat didefinisikan menggunakan Persamaan 2.5.

$$\sigma_g = \sqrt{\sum_{g=0}^{L-1} (g - \bar{g})^2 P(g)} \quad (2.5)$$

Keterangan:

σ : *standard deviation*

g : *gray level*

$\bar{g}$: *mean*

$P(g)$: probabilitas *gray level* g pada citra

L : Banyak nilai histogram

3. Skewness

Skewness mengukur keseimbangan rata-rata pada suatu *gray-level*. Fungsi *skew* dapat didefinisikan menggunakan Persamaan 2.6.

$$SKEW = \frac{1}{\sigma_g^3} \sum_{g=0}^{L-1} (g - \bar{g})^3 P(g) \quad (2.6)$$

Keterangan:

σ : *standard deviation*

g : *gray level*

$\bar{g}$: *mean*

$P(g)$: probabilitas *gray level* g pada citra

L : Banyak nilai histogram

2.4 K-Nearest Neighbor

Suatu metode klasifikasi yang digunakan pada sekumpulan data berdasarkan data yang sudah diklasifikasikan disebut *K-Nearest Neighbor (KNN)*. KNN ini juga termasuk *supervised* karena klasifikasi data yang baru akan diklasifikasikan sesuai dengan banyaknya kedekatan jarak yang mayoritas (Santoso, 2007).

Tahapan algoritme KNN:

1. Menentukan nilai k .
2. Menghitung jarak *Euclidean* data baru terhadap data yang sudah di-*training* menggunakan Persamaan 2.7 (Santoso, 2007).

$$D(a, b) = \sqrt{\sum_{k=1}^d (a_k - b_k)^2} \quad (2.7)$$

Keterangan:

$D(a, b)$: jarak antara vektor a dengan vektor b

3. Mengurutkan hasil tahap ke-2 secara *ascending*
4. Klasifikasi berdasarkan nilai k .
5. Kategori yang paling mayoritas dapat menjadi kategori data yang baru.

2.5 Evaluasi

Suatu metode yang digunakan dalam perhitungan akurasi pada konsep *data mining* disebut *confusion matrix*. Nilai *accuracy*, *precision*, *recall* dan *f-measure* dihasilkan dari evaluasi *confusion matrix*. Pertama, *Precision* merupakan tingkat ketepatan informasi yang diminta dengan hasil yang diberikan. *Recall* merupakan tingkat kesesuaian informasi dari seluruh dokumen yang didapatkan. *F-measure* merupakan nilai dari penyetaraan nilai *precision* dan *recall*. *Accuracy* merupakan tingkat kedekatan nilai prediksi dengan nilai yang sebenarnya. Untuk menghitung nilai *accuracy*, *precision*, *recall* dan *f-measure* menggunakan Persamaan 2.8, Persamaan 2.9, Persamaan 2.10 dan Persamaan 2.11. *Confusion Matrix* dijelaskan pada Tabel 2.1 (Han, et al., 2006).

Tabel 2.1 Confusion Matrix Multiple Class

Actual Class	Predicted Class		
	A	B	C
A	TP_A	E_{AB}	E_{AC}
B	E_{BA}	TP_B	E_{BC}
C	E_{CA}	E_{CB}	TP_C

$$Precision = \frac{TP}{(TP + FP)} \quad (2.8)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (2.9)$$

$$Accuracy = \frac{TP + TN}{(TP + FP + FN + TN)} \quad (2.10)$$

$$F - Measure = \frac{2 \times precision \times recall}{(precision + recall)} \quad (2.11)$$

Keterangan:

TP : data dengan prediksi positif dan terdeteksi positif

FP: data dengan prediksi negatif dan terdeteksi positif

FN: data dengan prediksi positif dan terdeteksi negatif

TN: data dengan prediksi negatif dan terdeteksi negatif

Nilai *FN* didapatkan dari jumlah dalam baris kelas kecuali nilai *TP*-nya. Nilai *FP* didapatkan dari jumlah dalam kolom kelas kecuali nilai *TP*-nya. Untuk nilai *TN* pada kelas tertentu didapatkan dengan cara menjumlahkan semua data pada baris dan kolomnya kecuali nilai *TP*-nya.

Pada *confusion matrix* terdapat dua metode yaitu *micro-averaging* dan *macro-averaging*. *Micro-averaging* merupakan metode yang digunakan untuk menghitung evaluasi berdasarkan data dari seluruh kelas yang digabungkan menjadi satu *confusion matrix*. *Macro-averaging* merupakan metode evaluasi yang digunakan berdasarkan data dari masing-masing kelasnya dan menghasilkan rata-rata dari hasil evaluasinya (Sokolova & Lapalme, 2009). Kedua metode evaluasi ini digunakan ketika terdapat banyak kelas yang digunakan dalam klasifikasi. Untuk menghitung nilai *accuracy*, *precision*, *recall* dan *f-measure* dengan *macro-averaging* menggunakan Persamaan 2.12, Persamaan 2.13, Persamaan 2.14 dan Persamaan 2.15. Untuk menghitung nilai *accuracy*, *precision*, *recall* dan *f-measure* dengan *micro-averaging* menggunakan Persamaan 2.16, Persamaan 2.17, Persamaan 2.18 dan Persamaan 2.19.

$$M \text{ Precision} = \frac{\sum_{i=1}^K \frac{tp_i}{tp_i + fp_i}}{K} \quad (2.12)$$

$$M \text{ Recall} = \frac{\sum_{i=1}^K \frac{tp_i}{tp_i + fn_i}}{K} \quad (2.13)$$

$$M \text{ F - Measure} = \frac{2 \times M \text{ Precision} \times M \text{ Recall}}{M \text{ Precision} + M \text{ Recall}} \quad (2.14)$$

$$M \text{ Akurasi} = \frac{\sum_{i=1}^K \frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i}}{K} \times 100\% \quad (2.15)$$

$$\mu \text{ Precision} = \frac{\sum_{i=1}^K tp_i}{\sum_{i=1}^K (tp_i + fp_i)} \quad (2.16)$$

$$\mu \text{ Recall} = \frac{\sum_{i=1}^K tp_i}{\sum_{i=1}^K (tp_i + fn_i)} \quad (2.17)$$

$$\mu \text{ F - Measure} = \frac{2 \times \mu \text{ Precision} \times \mu \text{ Recall}}{\mu \text{ Precision} + \mu \text{ Recall}} \quad (2.18)$$

$$\mu \text{ Akurasi} = \sum_i^K \frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i} \times 100\% \quad (2.19)$$

Keterangan:

M : Macro

μ : Micro

K : Jumlah Kelas data

TP : data dengan prediksi positif dan terdeteksi positif

FP : data dengan prediksi negatif dan terdeteksi positif

TN : data dengan prediksi positif dan terdeteksi negatif

FN : data dengan prediksi negatif dan terdeteksi negatif

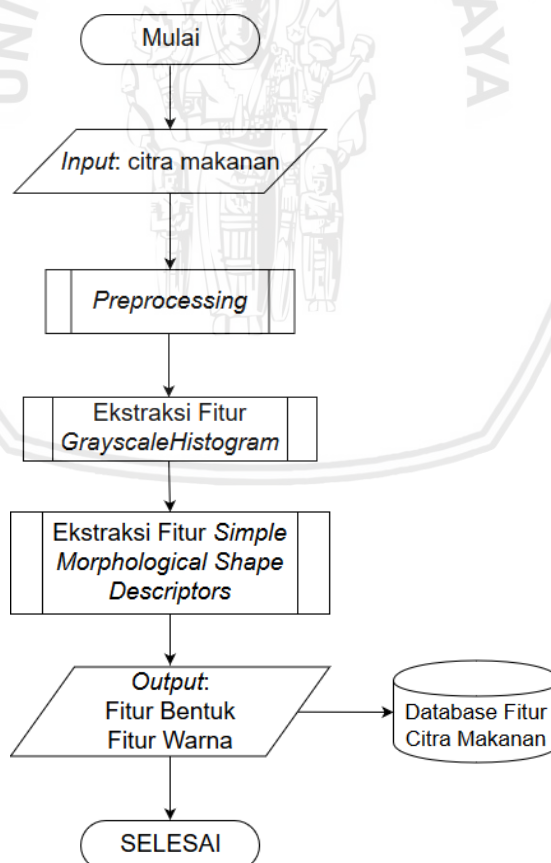
BAB 3 METODOLOGI

3.1 Tipe Penelitian

Tipe penelitian pada penelitian ini adalah non-implementatif analitik. Penelitian non-implementatif merupakan penelitian yang bertujuan untuk mengidentifikasi hubungan antar elemen dalam objek penelitian yang digunakan sebagai dasar untuk mengambil keputusan.

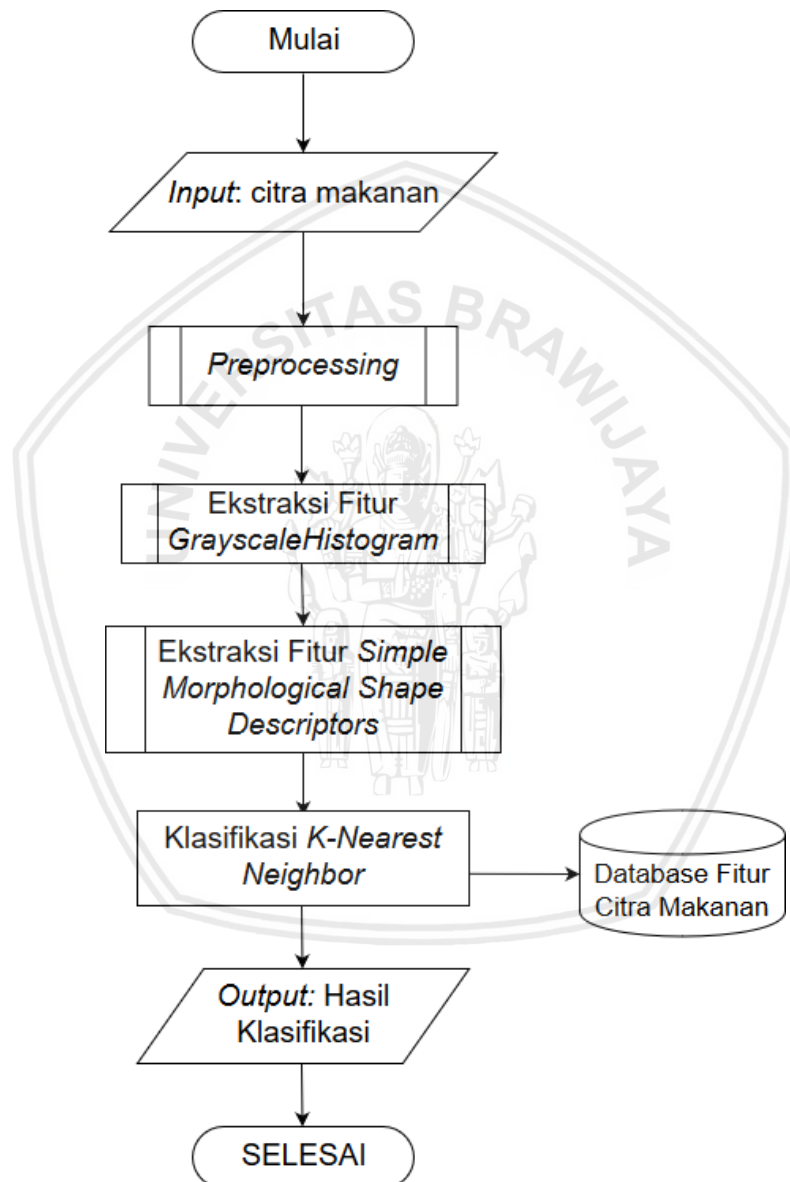
3.2 Metode Penelitian

Pada penelitian ini metode yang digunakan terbagi menjadi dua yaitu metode untuk pelatihan dan metode untuk pengujian. Pada pelatihan, dimulai dengan memasukkan data citra/gambar yang akan diekstraksi fitur bentuk dan warnanya. Sebelumnya dilakukan tahap *preprocessing* untuk mengambil objek makanan pada gambar saja. Kemudian dilakukan ekstraksi fitur menggunakan metode *Simple Morphological Shape Descriptors* dan *Grayscale Histogram*. Setelah sudah didapatkan nilai dari setiap fiturnya, maka akan dimasukkan ke dalam database berbentuk Microsoft Excel. Alur metode pelatihan ditunjukkan pada Gambar 3.1.



Gambar 3.1 Diagram Alir Metode Pelatihan

Pada pengujian, dimulai dengan memasukkan data citra/gambar yang akan diekstraksi fitur bentuk dan warnanya. Sebelumnya dilakukan tahap *preprocessing* untuk mengambil objek makanan pada gambar saja. Kemudian dilakukan ekstraksi fitur menggunakan metode *Simple Morphological Shape Descriptors* dan *Grayscale Histogram*. Kemudian keluaran berupa hasil dari klasifikasi *k-nearest neighbor* dengan menghitung jarak antar setiap fiturnya dengan data pelatihan yang sudah dimasukkan ke database. Alur metode pengujian ditunjukkan pada Gambar 3.2.



Gambar 3.2 Diagram Alir Metode Pengujian

3.3 Lokasi Penelitian

Lokasi dalam melakukan penelitian dilakukan di Gedung F Lantai 9 Ruang F9.3, Fakultas Ilmu Komputer, Universitas Brawijaya.

3.4 Partisipan Penelitian

Partisipan pada penelitian ini adalah mahasiswa dari kelas induksi riset F yang dibimbing oleh Ibu Yuita Arum Sari, S.Kom, M.Kom. Subjek yang digunakan adalah makanan. Makanan yang digunakan sebagai data penelitian terdapat 12 jenis makanan.

3.5 Peralatan Pendukung

Peralatan yang mendukung penelitian ini dalam pengumpulan data maupun pengembangan sistem antara lain:

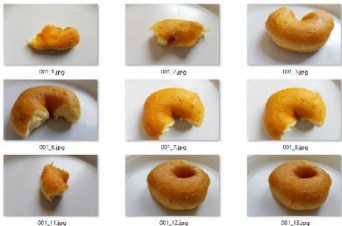
1. Kebutuhan pengambilan data
 - Samsung Galaxy S8.
2. Kebutuhan pengembangan sistem
 1. Spesifikasi Kebutuhan *hardware*
 - Laptop dengan processor Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz, Memori 4GB DDR3, Ruang penyimpanan dengan kapasitas 500GB.
 2. Spesifikasi kebutuhan *software*
 - Microsoft Windows 10.
 - Python 3.6.
 - Library OpenCV.
 - IDLE Python 3.6.

3.6 Teknik Pengumpulan Data

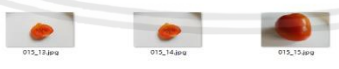
Pada penelitian ini data yang digunakan adalah data berupa citra yang di ambil secara manual menggunakan *smartphone*. Makanan tersusun rapi di atas piring berwarna putih. Citra makanan yang dikumpulkan terbagi menjadi 2 jenis, yaitu citra sebelum dimakan dan citra sesudah dimakan. Dari setiap jenisnya, pengambilan citra dilakukan sebanyak tiga kali. Pada citra sesudah dimakan, terbagi berdasarkan ukurannya yaitu $\frac{3}{4}$ bagian, $\frac{1}{2}$ bagian, dan $\frac{1}{4}$ bagian.

Jenis makanan yang digunakan pada penelitian ini ditunjukkan pada Tabel 3.1.

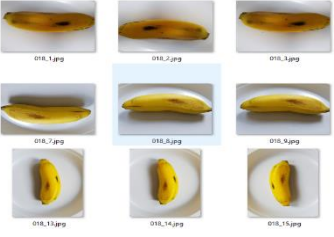
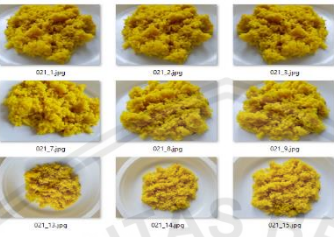
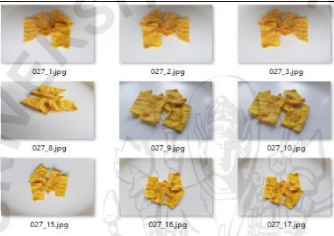
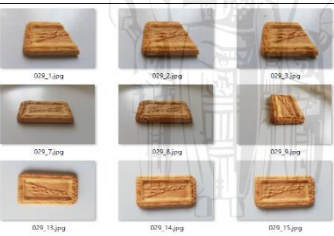
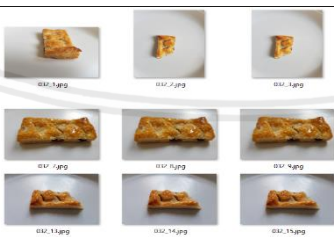
Tabel 3.1 Jenis Makanan yang Digunakan

No	Jenis Makanan	Data Training	Jumlah Data
1	Donat		21

Tabel 3.1 Jenis Makanan yang Digunakan (Lanjutan)

No	Jenis Makanan	Data Training	Jumlah Data
2	Mie	  	18
3	Ayam Crispi	  	17
4	Daging Rendang	  	21
5	Selada	  	21
6	Tomat	  	21
7	Pisang Hijau	  	21

Tabel 3.1 Jenis Makanan yang Digunakan (Lanjutan)

No	Jenis Makanan	Data Training	Jumlah Data
8	Pisang Kuning		19
9	Nasi Kuning		21
10	Happy Toss		21
11	Biskuat		21
12	Genjie Pie		21

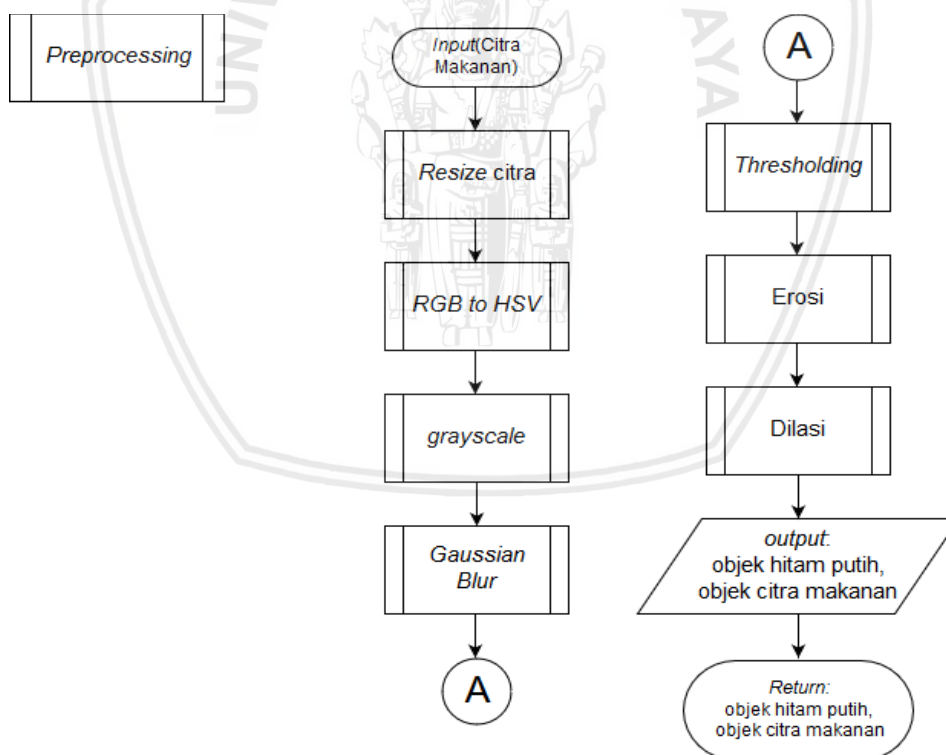
BAB 4 PERANCANGAN

4.1 Perancangan Algoritme

Secara garis besar terdapat empat proses utama yang diimplementasikan ke dalam program untuk penelitian ini, yaitu *Preprocessing*, ekstraksi fitur *Grayscale Histogram*, ekstraksi fitur *Simple Morphological Shape Descriptors*, dan klasifikasi *K-Nearest Neighbor*. Pada tahap *preprocessing*, citra makanan diolah hingga menghasilkan citra yang sesuai dengan kebutuhan yang diperlukan untuk proses ekstraksi fitur warna dan bentuk. Pada tahap selanjutnya pada ekstraksi fitur warna menggunakan metode *Grayscale Histogram* di mana fitur yang akan dihasilkan adalah nilai *mean*, *standard deviation*, dan *skewness*. Kemudian dilakukan ekstraksi fitur bentuk menggunakan metode *SMSD*.

4.1.1 Preprocessing

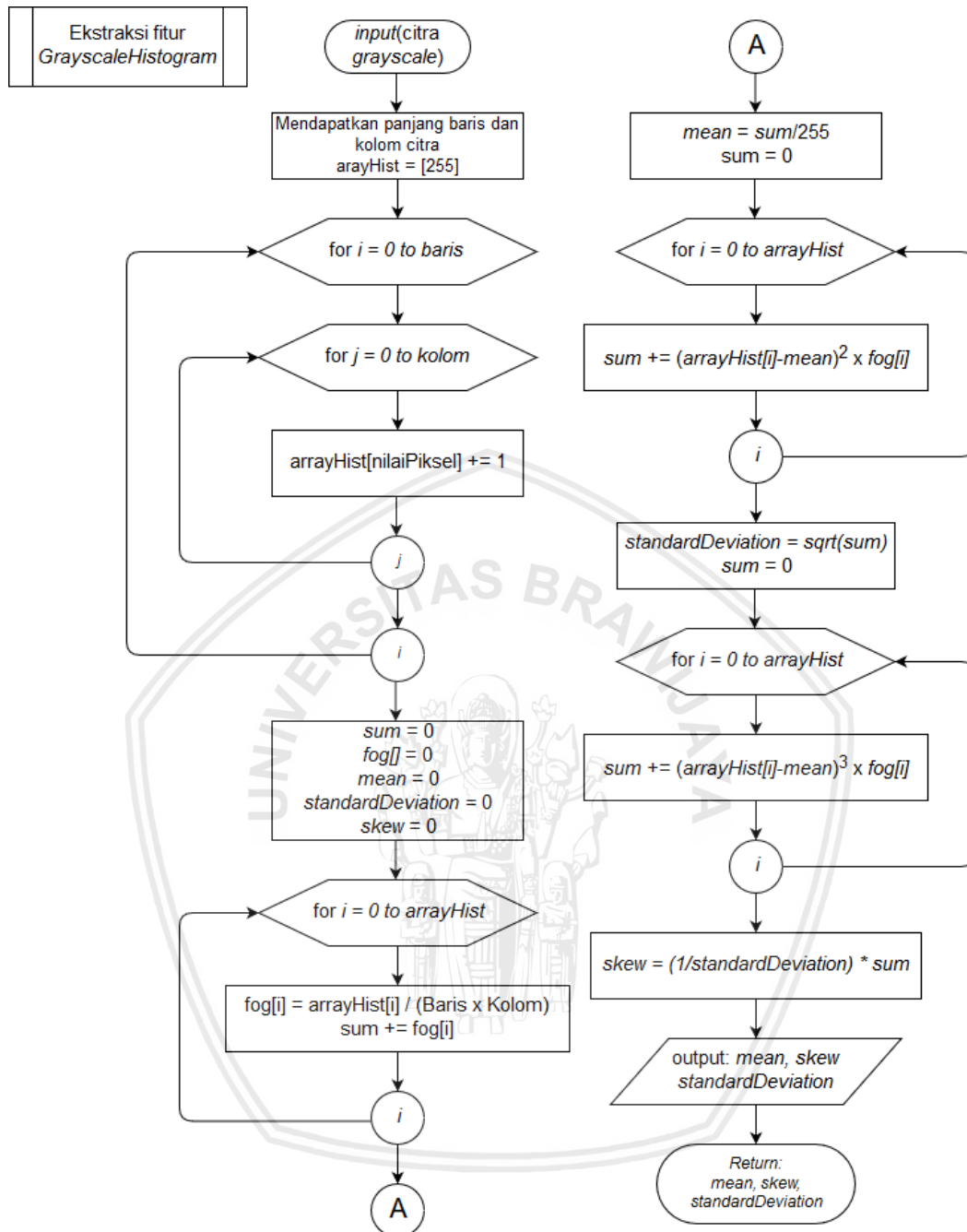
Proses *Preprocessing* dilakukan untuk memisahkan antara background pada citra dengan objek pada citra agar sesuai dengan kebutuhan yang dibutuhkan pada proses ekstraksi fitur. Alur *preprocessing* ditunjukkan pada Gambar 4.1.



Gambar 4.1 Diagram Alir *preprocessing*

4.1.2 Ekstraksi Fitur *Grayscale Histogram*

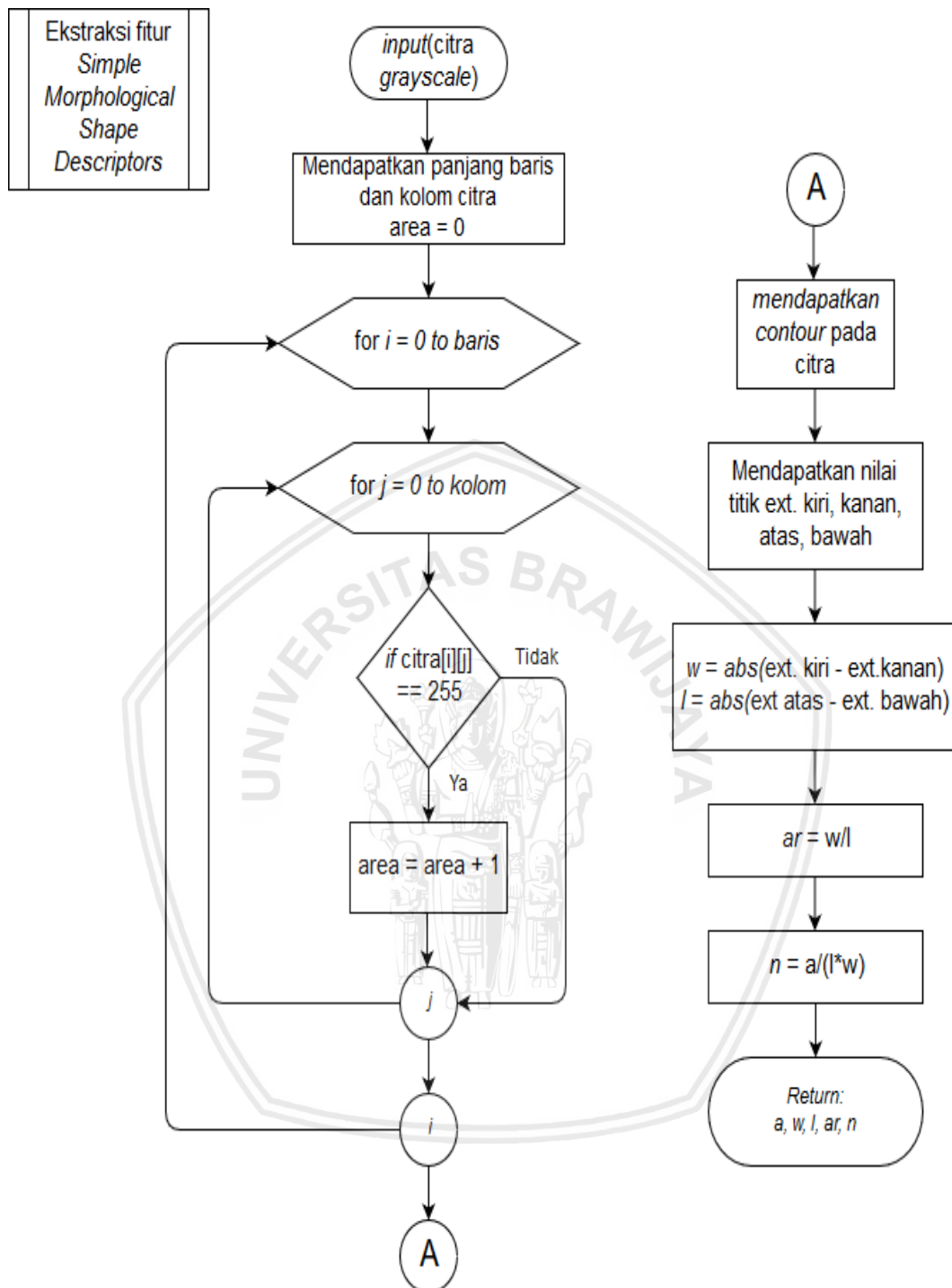
Proses ekstraksi fitur *Grayscale Histogram* dilakukan untuk menghasilkan fitur-fitur seperti nilai *mean*, nilai *standard deviation*, dan nilai *skew* dari *channel grayscale*. Alur ekstraksi fitur *Grayscale Histogram* ditunjukkan pada Gambar 4.2.



Gambar 4.2 Diagram Alir Ekstraksi Fitur *Grayscale Histogram*

4.1.3 Ekstraksi Fitur *Simple Morphological Shape Descriptors*

Proses ekstraksi fitur *Simple Morphological Shape Descriptors* dilakukan untuk menghasilkan fitur-fitur seperti nilai panjang (l), nilai lebar (w), nilai area (a), nilai *aspect ratio* (ar), dan nilai *rectangularity* N (n) dari input citra biner. Alur ekstraksi fitur *Grayscale Histogram* ditunjukkan pada Gambar 4.3.



Gambar 4.3 Diagram Alir ekstraksi Fitur *Simple Morphological Shape Descriptors*

4.1.4 Klasifikasi *K-Nearest Neighbor*

Proses klasifikasi *k-Nearest Neighbor* (KNN) dilakukan untuk mengklasifikasikan fitur yang sudah di ekstraksi dan menghasilkan kelas yang sudah ada dalam data *training*. Alur klasifikasi KNN ditunjukkan pada Gambar 4.4.



Gambar 4.4 Diagram Alir Klasifikasi *K-Nearest Neighbor*

4.2 Perhitungan Manualisasi

Pada penelitian ini, perhitungan manualisasi dilakukan pada citra yang telah melalui tahapan *preprocessing*.

4.2.1 Perhitungan *Grayscale Histogram*

Pada ekstraksi fitur *Grayscale Histogram*, masukkan berupa citra yang sudah dilakukan *preprocessing*. Pada Tabel 4.1 untuk data piksel berupa nilai keabuan atau *grayscale* pada citra.

Tabel 4.1 Nilai Piksel 3x3

Piksel		
171	166	175
178	168	175
169	157	158

Langkah 1:

Menghitung jumlah dari setiap nilai piksel

Tabel 4.2 Nilai *Histogram*

Histogram									
0	1	2	...	157	158	159	...	254	255
0	0	0	...	1	1	0	...	0	0

Langkah 2:

Menghitung *first order histogram* menggunakan Persamaan 2.3.

Tabel 4.3 Nilai *First Order Histogram*

First Order	Histogram									
	0	1	2	...	157	158	159	...	254	255
$P(g) = \frac{N(g)}{M}$	0	0	0	...	0,111111	0,111111	0	...	0	0

Langkah 3:

Menghitung fitur *Grayscale Histogram*.

1. *Mean*

Menghitung nilai *mean* menggunakan Persamaan 2.4.

$$\begin{aligned}
 &= \frac{0 + 0 + 0 + \dots + 0,111111 + 0,111111 + 0 + \dots + 0 + 0}{255} \\
 &= \frac{1}{255} \\
 &= 0,00396
 \end{aligned}$$

2. *Standard Deviation*

Menghitung nilai *standard deviation* menggunakan Persamaan 2.5.

	0	1	2	...	157	158	159	...	254	255
$(g - \bar{g})^2 P(g)$	0	0	0	...	0,110245	0,110245	0	...	0	0

$$\sigma_g = \sqrt{1,657133}$$

$$= 1,287297$$

3. Skewness

Menghitung nilai *skewness* menggunakan Persamaan 2.6.

	0	1	2	...	157	158	159	...	254	255
$(g - \bar{g})^3 P(g)$	0	0	0	...	0,109814	0,109814	0	...	0	0

$$SKEW = \frac{1}{1,287297^3} * 2,53608$$

$$= 1,188849$$

Hasil perhitungan fitur warna *Grayscale Histogram* untuk 1 citra

Tabel 4.4 Nilai Fitur *Grayscale Histogram*

Fitur	Nilai
<i>Mean</i>	0,300396
<i>Standard Deviation</i>	1,287297
<i>Skew</i>	1,188849

4.2.2 Perhitungan *Simple 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 $(x, y) = (305, 156)$

Koordinat titik kiri $(x, y) = (33, 101)$

Koordinat titik bawah $(x, y) = (52, 280)$

Koordinat titik atas $(x, y) = (251, 55)$

Langkah 3:

Menghitung lebar objek pada citra.

Sumbu lebar = titik kanan(x) – titik kiri(x)

$$\text{Sumbu lebar} = 305 - 33 = 272$$

Langkah 4:

Menghitung panjang objek pada citra.

$$\text{Sumbu panjang} = \text{titik bawah}(y) - \text{titik atas}(y)$$

$$\text{Sumbu panjang} = 280 - 55 = 225$$

Langkah 5:

Menghitung aspek rasio menggunakan Persamaan 2.2.

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

Langkah 6:

Menghitung Area dengan cara menjumlahkan piksel pada objek citra.

$$\text{Area} = 14897$$

Langkah 7:

Menghitung *rectangularity* N menggunakan Persamaan 2.1.

$$\text{Rectangularity} = \frac{14897}{225 * 272} = 0,243$$

4.2.3 Perhitungan *K-Nearest Neighbor*

Pada proses klasifikasi, menggunakan permissalan manualisasi dengan 6 data latih yang terdiri dari 3 kelas dan masing-masing dua data latih dari setiap kelasnya.

Langkah 1:

Menentukan nilai k terlebih dahulu. Nilai k yang digunakan pada proses ini adalah 2. Nilai k berfungsi untuk menentukan jumlah tetangga paling dekat.

Langkah 2:

Menghitung jarak antar data uji dengan data latih menggunakan perhitungan *Euclidean Distance* pada Persamaan 2.7.

Tabel 4.5 Nilai *Euclidean Distance*

Data	$(a_1 - b_1)^2$	$(a_2 - b_2)^2$	$(a_3 - b_3)^2$	$d(a, b)$	Kelas
1	1,456924	6,7E-05	205,2685	14,37795	Daging Rendang
2	116,3196	0,006084	3787,211	62,47829	Selada
3	1558,707	0,453303	2584,563	64,37176	Happy Toss
4	1420,937	0,497382	2118,626	59,4984	Happy Toss
5	18,19556	0,019336	6393,163	80,07108	Selada
6	4,940201	0,00017	371,9136	19,41273	Daging Rendang

Langkah 3:

Setelah jarak antara data uji dengan data latih didapatkan, selanjutnya data diurutkan berdasarkan jarak terkecil ke terbesar.

Tabel 4.6 Jumlah Nilai *Euclidean Distance*

Data	$d(a, b)$	Kelas
1	14,37795	Daging Rendang
6	19,41273	Daging Rendang
4	59,4984	Happy Toss
2	62,47829	Selada
3	64,37176	Happy Toss
5	80,07108	Selada

Langkah 4:

Langkah terakhir, menentukan kelas dari data uji sesuai dengan nilai k . Dari Tabel 4.6 didapatkan bahwa dengan nilai $k=3$, kelas yang paling sering muncul adalah daging rendang.

BAB 5 IMPLEMENTASI

5.1 Implementasi Algoritme

5.1.1 Implementasi Preprocessing

Kode Program 1: <i>Preprocessing</i>	
1	#import
2	import cv2
3	
4	#class
5	class preProcessing:
6	def proses(img):
7	#RGB to HSV
8	imgHSV = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)
9	
10	#HSV to Gray
11	imgGray = cv2.cvtColor(imgHSV, cv2.COLOR_BGR2GRAY)
12	
13	#Gaussian Blur
14	blur = cv2.GaussianBlur(imgGray, (5,5), 0)
15	
16	#Otsu + Biner
17	ret3, th4 =
18	cv2.threshold(blur, 120, 255, cv2.THRESH_BINARY+cv2.THRESH_OTSU)
19	
20	#Erosi + Dilasi
21	kernel = np.ones((5,5), np.uint8)
22	img_erosion = cv2.erode(th4, kernel, iterations=1)
23	img_dilation = cv2.dilate(img_erosion, kernel,
24	iterations=1)
25	#Fusion
26	imgFusion = cv2.bitwise_and(img, img, mask = img_dilation)
27	
28	return imgFusion, img_dilation

Kode Program 5.1 Preprocessing

Penjelasan Kode Program 1:

1. Baris 2 merupakan deklarasi openCV.
2. Baris 5 merupakan deklarasi kelas *preProcessing*.
3. Baris 6 merupakan deklarasi method proses dengan parameter *img*.
4. Baris 8 digunakan untuk konversi citra dari RGB ke HSV.
5. Baris 11 digunakan untuk konversi citra menjadi *grayscale*.
6. Baris 14 digunakan untuk konversi citra menjadi *gaussian blur*.
7. Baris 17-18 digunakan untuk konversi citra menjadi *biner* menggunakan metode otsu.
8. Baris 21-24 digunakan untuk konversi citra menjadi *erosion*.
9. Baris 26 digunakan untuk menggabungkan hasil segmentasi dengan gambar asli.
10. Baris 28 digunakan untuk mengembalikan nilai pada variabel *imgFusion* dan *img_dilation*.

5.1.2 Implementasi Ekstraksi Fitur *Grayscale Histogram*

Kode Program 2: <i>Grayscale Histogram</i>	
1	<code>class colorHistogram:</code>
2	<code>def __init__(self):</code>
3	<code>pass</code>
4	
5	<code>def cariFitur(self, img):</code>
6	<code>#Ukuran Gambar</code>
7	<code>height, width = img.shape</code>
8	<code>jumlahPiksel = height*width</code>
9	
10	<code>#hitung nilai histogram</code>
11	<code>numImg = np.array(img)</code>
12	<code>unique, count = np.unique(numImg, return_counts=True)</code>
13	<code>dictHistogram = dict(zip(unique, count))</code>
14	<code>histData = []</code>
15	<code>for i in range(256):</code>
16	<code>if i in dictHistogram:</code>
17	<code>histData.append(dictHistogram[i])</code>
18	<code>else:</code>
19	<code>histData.append(0)</code>
20	<code>histData[0] = 0</code>
21	<code>histData[255] = 0</code>
22	
23	<code>#fog</code>
24	<code>firstOrder = []</code>
25	<code>for i in range(len(histData)):</code>
26	<code>firstOrder.append(histData[i]/(jumlahPiksel))</code>
27	
28	<code>#mean</code>
29	<code>temp = 0</code>
30	<code>for i in range(len(histData)):</code>
31	<code>temp += histData[i]</code>
32	<code>mean = temp / len(histData)</code>
33	
34	<code>#standard Deviation</code>
35	<code>temp = 0</code>
36	<code>for i in range(len(histData)):</code>
37	<code>temp += ((histData[i]-mean)**2)*firstOrder[i]</code>
38	<code>standarDev = math.sqrt(temp)</code>
39	
40	<code>#skew</code>
41	<code>temp = 0</code>
42	<code>for i in range(len(histData)):</code>
43	<code>temp += ((histData[i]-mean)**3)*firstOrder[i]</code>
44	<code>skew = (1/(standarDev**3)) * temp</code>
45	
46	<code>return mean, standarDev, skew</code>

Kode Program 5.2 *Grayscale Histogram*

Penjelasan Kode Program 2:

1. Baris 1 merupakan deklarasi kelas *colorHistogram*.
2. Baris 2 merupakan inisialisasi ketika ada pemanggilan kelas.
3. Baris 5 merupakan deklarasi method dengan parameter *img* dan *self*.
4. Baris 7-8 digunakan untuk menghitung ukuran citra dan jumlah piksel pada citra.
5. Baris 11-21 digunakan untuk menghitung nilai histogram dari setiap piksel.
6. Baris 24-26 digunakan untuk menghitung nilai *first order histogram*.
7. Baris 29-32 digunakan untuk menghitung nilai *mean* dari *first order histogram*.

8. Baris 34-38 digunakan untuk menghitung nilai *standard deviation*.
9. Baris 41-44 digunakan untuk menghitung nilai *skewness*.
10. Baris 46 digunakan untuk mengembalikan nilai variabel *mean*, *standarDev*, dan *skew*.

5.1.3 Implementasi Ekstraksi Fitur *Simple Morphological Shape Descriptors*

Kode Program 3: <i>Simple Morphological Shape Descriptors</i>	
1	class simpleMorphological:
2	def __init__(self):
3	pass
4	
5	#Area
6	def cariFitur(self, img):
7	height, width = img.shape
8	area = [img[x][y] for x in range(height) for y in
9	range(width) if img[x][y] == 255]
10	a = len(area)
11	cnts = cv2.findContours(img,
12	cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
13	cnts = cnts[1]
14	c = max(cnts, key=cv2.contourArea)
15	kiri = tuple(c[c[:, :, 0].argmin()][0])
16	kanan = tuple(c[c[:, :, 0].argmax()][0])
17	atas = tuple(c[c[:, :, 1].argmin()][0])
18	bawah = tuple(c[c[:, :, 1].argmax()][0])
19	# Nilai w
20	w = abs(kiri[0]-kanan[0])
21	# Nilai l
22	l = abs(atas[1]-bawah[1])
23	# Nilai Aspek Rasio
24	ar = w/l
25	# Nilai Rectangularity N
26	n = a / (l*w)
27	return a, w, l, ar, n

Kode Program 5.3 *Simple Morphological Shape Descriptors*

Penjelasan Kode Program 3:

1. Baris 1 merupakan deklarasi kelas *simpleMorphological*.
2. Baris 2 merupakan inisialisasi ketika ada pemanggilan kelas.
3. Baris 6 merupakan deklarasi method dengan parameter *img* dan *self*.
4. Baris 7 digunakan untuk mendapatkan nilai panjang dan lebar citra.
5. Baris 8-10 digunakan untuk menghitung jumlah piksel objek pada citra.
6. Baris 11-14 digunakan untuk mencari titik-titik objek pada citra.
7. Baris 15-18 digunakan untuk mencari posisi matriks (kiri, kanan, atas, bawah) paling ekstrim pada citra.
8. Baris 20 digunakan untuk menghitung nilai *width*.
9. Baris 22 digunakan untuk menghitung nilai *height*.
10. Baris 24 digunakan untuk menghitung nilai *aspect ratio*.
11. Baris 26 digunakan untuk menghitung nilai *rectangularity N*.
12. Baris 27 digunakan untuk mengembalikan nilai variabel *a, w, l, ar*, dan *n*.

5.1.4 Implementasi K-Nearest Neighbor

Kode Program 4: K-Nearest Neighbor	
1	def hitungJarak(data1, data2):
2	tempJarak = 0
3	for i in range(len(data1)):
4	tempJarak += ((data1[i]-data2[i])**2)
5	return math.sqrt(tempJarak)
6	
7	def hitungKNN(training, labels, testing, k):
8	jarak = []
9	for i in range(len(training)):
10	temp = [training['Area'][i],training['Aspect
11	Ratio'][i],training['Length'][i],
12	training['Mean'][i],training['Rectangularity
13	N'][i],training['Skew'][i],
14	
15	training['StandardDeviation'][i],training['Width'][i]]
16	tempJarak = hitungJarak(testing, temp)
17	jarak.append((temp,tempJarak,labels[i]))
18	jarak.sort(key=lambda x: x[1])
19	neighbors = jarak[:k]
20	return neighbors
21	df = pd.read_excel("excel/testing-121118.xlsx", "Sheet1")
22	labels = []
23	for i in range(len(df)):
24	labels.append(df['ZFile'][i])
25	#KNN NIH
26	k=3
27	dataTestingg = [mean, skew, standarDev, a, w, l, ar, n]
28	knn = hitungKNN(df, labels, dataTestingg, k)
29	hasilKNN = {}
30	for i in knn:
31	print(i)
32	if i[2] not in hasilKNN:
33	hasilKNN[i[2]] = 0
34	hasilKNN[i[2]] += 1
35	hasilKNN = sorted(hasilKNN, key=hasilKNN.__getitem__,
36	reverse=True)[:1]
37	print("Termasuk Data Ke-",hasilKNN[0])

Kode Program 5.4 K-Nearest Neighbor

Penjelasan Kode Program 4:

1. Baris 1 merupakan deklarasi method dengan parameter *data1* dan *data2*.
2. Baris 2-5 digunakan untuk menghitung jarak antara *data1* dan *data2*.
3. Baris 7 merupakan deklarasi method dengan parameter *training*, *labels*, *testing* dan *k*.
4. Baris 8-20 digunakan untuk menghitung nilai KNN dari setiap fiturnya dan diurutkan mulai dari yang terkecil ke terbesar sebanyak *k*.
5. Baris 21 digunakan untuk memanggil file excel yang berisi *dataTraining*.
6. Baris 23-24 digunakan untuk menyimpan nilai kelas dari *dataTraining*.
7. Baris 26 merupakan deklarasi variabel *k*.
8. Baris 27 merupakan deklarasi variabel *dataTestingg*.
9. Baris 28 merupakan pemanggilan method *hitungKNN*.
10. Baris 29-34 digunakan untuk menghitung jumlah dari hasil KNN.
11. Baris 35-37 digunakan untuk mengurutkan jumlah dari hasil KNN dari yang terbesar dan menampilkannya dalam program.



Gambar 5.1 Keluaran Program Hasil Klasifikasi

BAB 6

HASIL PENGUJIAN DAN ANALISIS

Bab ini menjelaskan mengenai hasil pengujian dan analisis terhadap algoritme yang telah diimplementasikan pada bab sebelumnya. Pada pengujian, dilakukan pada masing-masing metode ekstraksi fitur dan penggabungan kedua metode ekstraksi fitur yang digunakan yaitu *Simple Morphological Shape Descriptors* (SMSD) dan *Grayscale Histogram*. Tabel 6.1 menunjukkan kelas yang digunakan dalam pengujian yang dilakukan beserta kode kelasnya.

Tabel 6.1 Daftar Kelas dan Kode Kelas

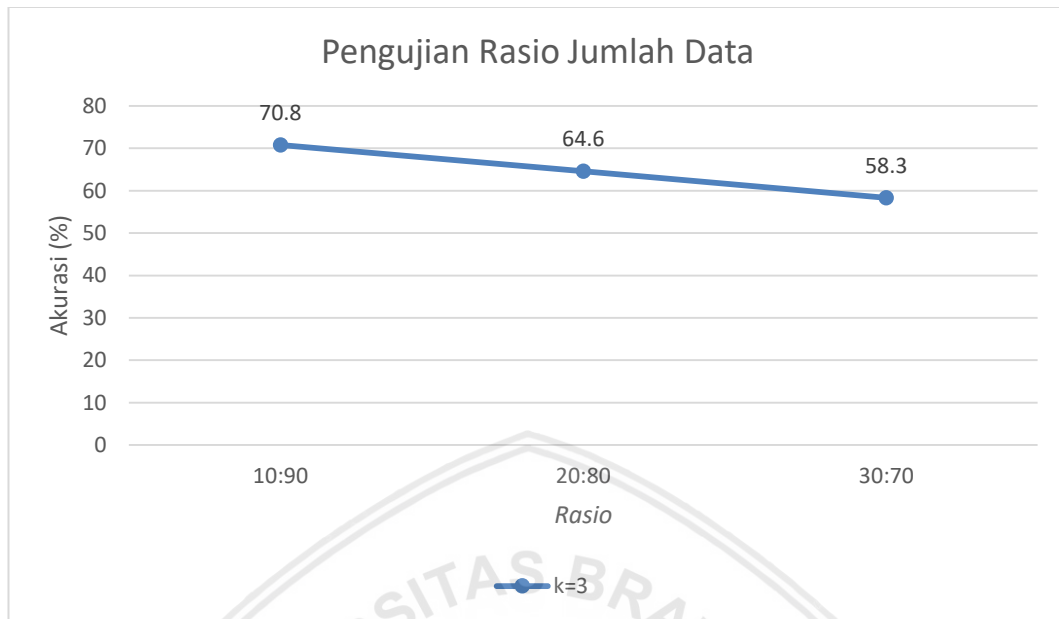
No	Kelas	Kode Kelas
1	Donat	001
2	Mie	004
3	Ayam Krispi	008
4	Daging Rendang	009
5	Selada	013
6	Tomat	015
7	Pisang Hijau	017
8	Pisang Kuning	018
9	Nasi Kuning	021
10	Happy Toss	027
11	Biskuat	029
12	Genjie Pie	032

6.1 Pengujian Rasio Jumlah Data

Pengujian rasio jumlah data dilakukan dengan menggunakan ekstraksi fitur warna *Grayscale Histogram*, fitur bentuk SMSD, dan nilai $k=3$. Pada pengujian rasio, data *testing* dan *training* yang digunakan adalah 10:90, 20:80, dan 30:70. Hasil pengujian rasio jumlah data ditunjukkan pada Tabel 6.2.

Tabel 6.2 Hasil Pengujian Rasio Jumlah Data

Nilai k	Perbandingan Data <i>Testing:Training</i>		
	10:90	20:80	30:70
3	70,8%	64,6%	58,3%



Gambar 6.1 Grafik Hasil Pengujian Rasio Jumlah Data

Pada grafik pengujian yang ditunjukkan Gambar 6.1, rasio jumlah data yang menghasilkan nilai akurasi tertinggi yaitu ketika data *testing* dan *training* memiliki rasio 10:90 dengan nilai akurasi 70,8%. Pada pengujian dengan rasio 20:80 dan 30:70, akurasi yang dihasilkan semakin menurun dengan nilai akurasi 64,6% dan 58,3%. Hal ini menunjukkan bahwa semakin besar rasio yang digunakan, maka semakin tinggi nilai akurasi yang dihasilkan. Hasil pengujian juga menunjukkan bahwa jumlah data latih yang digunakan memengaruhi hasil klasifikasi jenis makanan.

6.2 Pengujian Nilai k

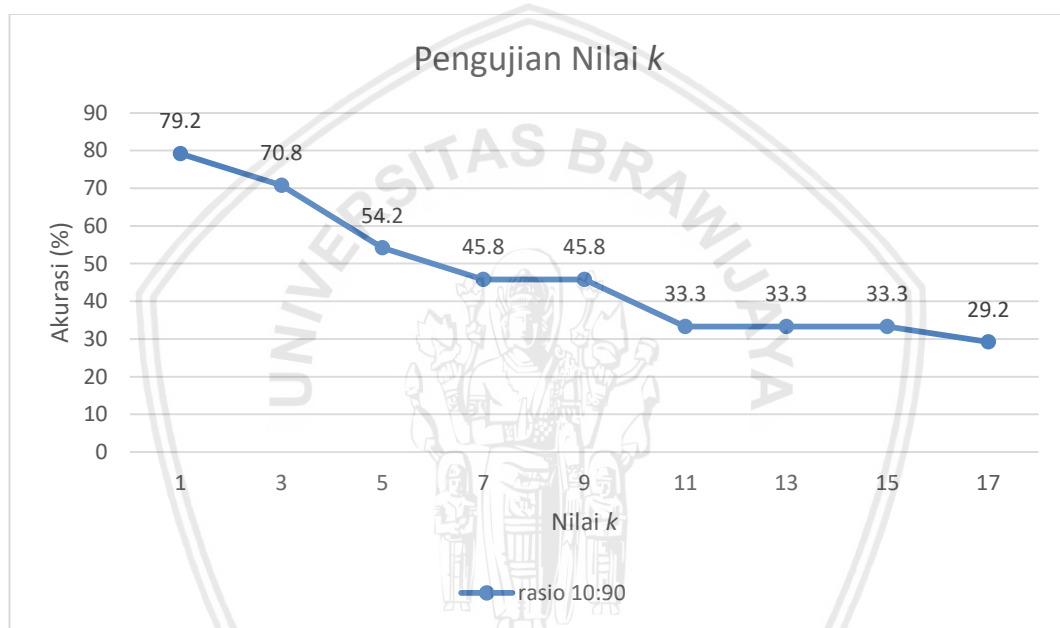
Pengujian nilai k dilakukan dengan menggunakan metode ekstraksi fitur warna *Grayscale Histogram*, fitur bentuk SMSD. Nilai k yang digunakan yaitu 1, 3, 5, 7, 9, 11, 13, 15, dan 17. Pemilihan nilai k ganjil dilakukan dengan tujuan agar didapatkan hanya satu kelas terbanyak dari ketetanggaan data uji untuk dijadikan kelas data uji. Pada pengujian nilai k menggunakan rasio jumlah data dengan akurasi tertinggi pada pengujian rasio jumlah data yaitu 10:90. Hasil pengujian nilai k ditunjukkan pada Tabel 6.3.

Tabel 6.3 Hasil Akurasi Pengujian Nilai k dan Perbandingan Jumlah Data

Nilai k	Akurasi
1	79,2%
3	70,8%
5	54,2%
7	45,8%

Tabel 6.3 Hasil Akurasi Pengujian Nilai k dan Perbandingan Jumlah Data (Lanjutan)

Nilai k	Akurasi
9	45,8%
11	33,3%
13	33,3%
15	33,3%
17	29,2%



Gambar 6.2 Grafik Hasil Pengujian Nilai k

Berdasarkan Gambar 6.2, Nilai maksimal k yang digunakan pada pengujian adalah 17 karena mendekati jumlah data *training* setiap kelas yang digunakan. Nilai k yang menghasilkan nilai akurasi tertinggi yaitu $k=1$ dengan nilai 79,2%. Hal ini disebabkan karena data uji mencari satu tetangga terdekat dari data latih ketika k bernilai 1. Sehingga data latih lainnya tidak diperhitungkan dalam penentuan kelas data uji karena hanya ada satu nilai ketetanggaan.

Hasil pengujian juga menunjukkan bahwa nilai k yang semakin mendekati jumlah data membuat hasil akurasi semakin menurun. Ketika nilai $k=3$, nilai akurasinya mengalami penurunan sebesar 8,4%. Ketika nilai $k=5$, nilai akurasinya mengalami penurunan yang jauh sebesar 16,6% sampai seterusnya ketika nilai k mendekati jumlah data *training* yang digunakan pada setiap kelas. Hal ini disebabkan karena ketika nilai k nya lebih dari satu, prediksi kelas ketetanggaan yang berbeda dianggap sebagai kelas terdekat dari data uji. Semakin mendekati nilai k dengan jumlah data setiap kelas, jumlah kelas ketetanggaan semakin bertambah dan menyebabkan terjadinya kesalahan klasifikasi. Contoh kesalahan

klasifikasi karena nilai k ditunjukkan pada Tabel 6.4. Citra data uji yang digunakan ditunjukkan pada Gambar 6.3.



027_2.jpg

Gambar 6.3 Data Uji

Tabel 6.4 Contoh Kesalahan Klasifikasi Karena Nilai k

k	Data Uji	Jarak	Hasil Sebenarnya	Hasil Klasifikasi
3	 027_14.jpg	35,02	027	008
	 008_1.jpg	104,42		
	 008_12.jpg	104,43		

Pada Tabel 6.4 menunjukkan hasil pengujian pada salah satu data uji yang digunakan dengan nilai $k=3$. Dapat dilihat bahwa kelas pertama yang terdekat dengan data uji adalah kelas 027. Kelas kedua terdekat adalah 008, dan ketiga adalah 008. Dapat dilihat walaupun kelas pertama yang terdekat adalah 027, hasil yang ditunjukkan pada Tabel 6.4 adalah kelas 008. Hal ini disebabkan metode KNN ketika jumlah kelas ketetanggaan semakin bertambah, maka hasil yang dikeluarkan adalah kelas yang paling dominan pada ketetanggaan data uji. Pada Tabel 6.4, kelas 008 berjumlah 2 data sedangkan kelas 027 hanya satu data.

6.3 Pengujian Fitur Warna *Grayscale Histogram*

Pengujian ketiga dilakukan dengan menggunakan metode ekstraksi fitur warna *Grayscale Histogram* dan *K-Nearest Neighbor (KNN)* dalam melakukan klasifikasi pada data uji. Pada pengujian fitur warna *Grayscale Histogram* menggunakan rasio jumlah data dengan akurasi tertinggi pada pengujian rasio jumlah data yaitu 10:90 dan menggunakan nilai k dengan akurasi tertinggi pada pengujian nilai k yaitu $k=1$. Metode *Grayscale Histogram* digunakan untuk fitur warna dan menghasilkan fitur warna yaitu *mean*, *standard deviation*, dan *skewness*. *Confusion matrix* dibuat dari hasil pengujian yang sudah dilakukan. *Confusion matrix* pengujian fitur warna *Grayscale Histogram* yang ditunjukkan pada Tabel 6.5. Pada Tabel 6.6 ditunjukkan hasil *confusion matrix* dari setiap kelasnya beserta nilai akurasi, *precision*, *recall*, dan *f-measure*.

Tabel 6.5 Confusion Matrix Metode *Grayscale Histogram*

Actual Class	Predicted Class											
	001	004	008	009	013	015	017	018	021	027	029	032
001	2	0	0	0	0	0	0	0	0	0	0	0
004	0	2	0	0	0	0	0	0	0	0	0	0
008	0	0	2	0	0	0	0	0	0	0	0	0
009	0	0	0	1	0	1	0	0	0	0	0	0
013	0	0	0	0	2	0	0	0	0	0	0	0
015	0	0	0	0	0	1	0	0	0	1	0	0
017	0	0	0	0	0	0	2	0	0	0	0	0
018	0	0	0	0	0	0	0	2	0	0	0	0
021	0	0	0	0	0	0	0	0	2	0	0	0
027	0	0	0	0	0	0	0	0	0	2	0	0
029	0	0	0	0	0	0	0	0	0	0	0	2
032	0	0	0	0	0	0	1	1	0	0	0	0

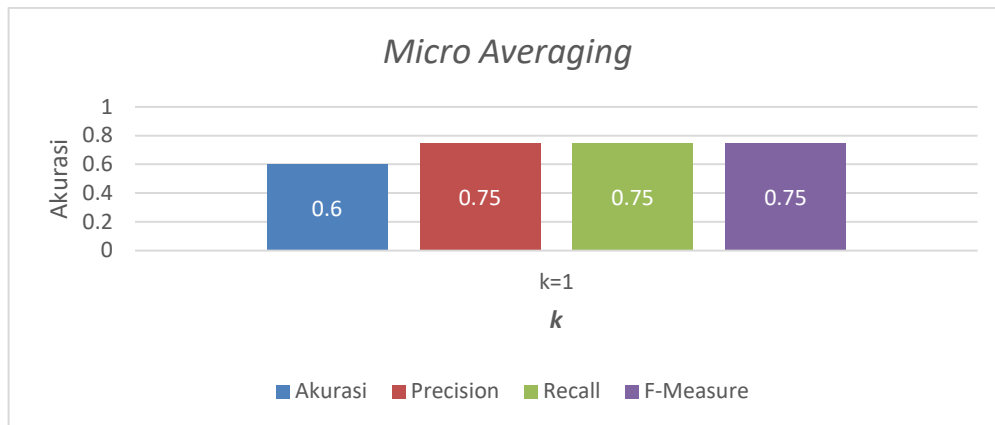
Tabel 6.6 Evaluasi Pengujian Fitur Warna *Grayscale Histogram* Per Kelas

No	Kelas	Confusion Matrix		Akurasi	Precision	Recall	F-Measure
		TP FN	FP TN				
1	001	2 0	0 0	1	1	1	1
2	004	2 0	0 0	1	1	1	1
3	008	2 0	0 0	1	1	1	1
4	009	1 1	0 0	0,5	1	0,5	0,667
5	013	2 0	0 0	1	1	1	1
6	015	1 1	0 0	0,333	0,5	0,5	0,5
7	017	2 0	1 0	0,667	0,667	1	0,8
8	018	2 0	1 0	0,667	0,667	1	0,8
9	021	2 0	0 0	1	1	1	1
10	027	2 0	1 0	0,667	0,667	1	0,8
11	029	0 2	0 0	0	0	0	0
12	032	0 2	0 0	0	0	0	0

Dari hasil *confusion matrix* pada Tabel 6.5 dapat dihasilkan nilai *micro averaging* dan *macro averaging*. Pada Tabel 6.6 dihasilkan nilai akurasi, *precision*, *recall*, *f-measure* dari masing-masing kelas yang digunakan untuk menghitung nilai *micro-averaging* dan *macro-averaging*. Pada Tabel 6.7, menunjukkan hasil *macro-averaging* dan *micro-averaging* pada pengujian fitur warna *Grayscale Histogram* dengan nilai $k=1$.

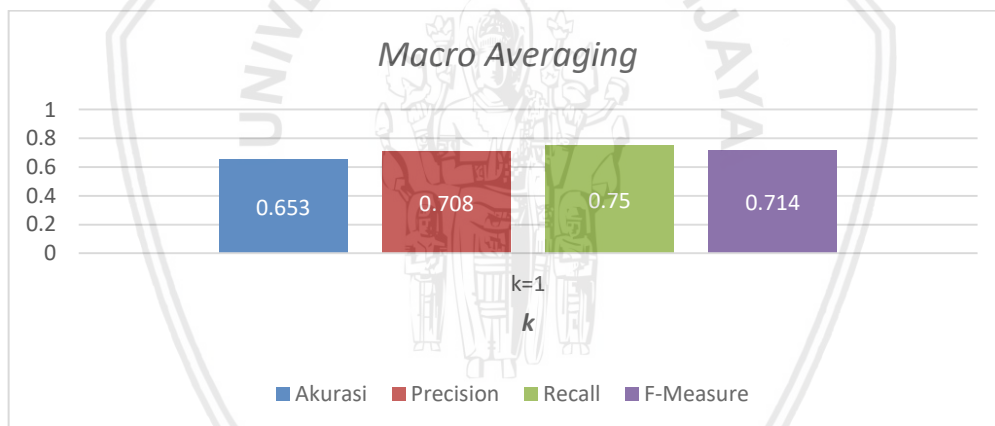
Tabel 6.7 Micro & Macro Averaging Fitur Warna *Grayscale Histogram*

No	k	Micro Averaging				Macro Averaging			
		Akurasi	Precision	Recall	F-Measure	Akurasi	Precision	Recall	F-Measure
1	1	0,6	0,75	0,75	0,75	0,653	0,708	0,75	0,714



Gambar 6.4 Grafik Micro Averaging Grayscale Histogram

Pada pengujian fitur warna *Grayscale Histogram* menggunakan *micro-averaging* dihasilkan akurasi sebesar 0,6 atau 60%. Hal ini disebabkan karena pada *dataset* yang digunakan banyak yang memiliki kesamaan warna. Selain itu, nilai dari *f-measure* yang dihasilkan memiliki nilai yang sama seperti nilai *precision* dan *recall*. Dari kesamaan tersebut, membuktikan bahwa nilai dari *precision* dan *recall* terdapat keseimbangan antar nilai.



Gambar 6.5 Grafik Macro Averaging Grayscale Histogram

Pada Gambar 6.5 menunjukkan bahwa nilai dari akurasi, *precision*, *recall*, *f-measure* berbeda dengan nilai yang ada pada Gambar 6.4. Pada *micro-averaging* menghasilkan rata-rata nilai akurasi sebesar 0,653 atau 65,3%. Hal ini disebabkan karena perhitungan pada *macro averaging* hanya berdasarkan per kelasnya saja.

6.4 Pengujian Fitur Bentuk *Simple Morphological Shape Descriptors*

Pengujian ketiga dilakukan dengan menggunakan metode ekstraksi fitur bentuk SMSD dan *K-Nearest Neighbor (KNN)* dalam melakukan klasifikasi pada data uji. Pada pengujian fitur bentuk SMSD menggunakan rasio jumlah data dengan akurasi tertinggi pada pengujian rasio jumlah data yaitu 10:90 dan menggunakan nilai *k* dengan akurasi tertinggi pada pengujian nilai *k* yaitu $k=1$. Metode SMSD digunakan untuk fitur bentuk dan menghasilkan fitur bentuk yaitu *area*, *length*, *width*, *rectangularity* *N* dan *aspect ratio*. *Confusion matrix* dibuat dari

hasil pengujian yang sudah dilakukan. *Confusion matrix* metode SMSD ditunjukkan pada Tabel 6.8.

Tabel 6.8 Confusion Matrix Metode SMSD

Actual Class	Predicted Class											
	001	004	008	009	013	015	017	018	021	027	029	032
001	2	0	0	0	0	0	0	0	0	0	0	0
004	0	1	1	0	0	0	0	0	0	0	0	0
008	0	1	1	0	0	0	0	0	0	0	0	0
009	0	0	0	2	0	0	0	0	0	0	0	0
013	0	0	0	0	2	0	0	0	0	0	0	0
015	0	0	0	0	0	0	0	0	1	1	0	0
017	0	0	0	0	0	0	2	0	0	0	0	0
018	0	0	0	0	0	0	0	2	0	0	0	0
021	0	0	0	0	0	0	0	0	2	0	0	0
027	0	0	1	0	0	0	0	0	0	1	0	0
029	0	0	0	0	0	0	0	0	0	0	2	0
032	0	0	1	0	0	0	0	1	0	0	0	0

Tabel 6.9 Evaluasi Pengujian Fitur Bentuk SMSD Per Kelas

No	Kelas	Confusion Matrix		Akurasi	Precision	Recall	F-Measure
		TP FN	FP TN				
1	001	2 0	0 0	1	1	1	1
2	004	1 1	0 0	0,333	0,5	0,5	0,5
3	008	1 0	1 0	0,2	0,25	0,5	0,333
4	009	2 0	0 0	1	1	1	1
5	013	2 0	0 0	1	1	1	1
6	015	1 1	0 0	0	0	0	0
7	017	2 0	0 0	1	1	1	1
8	018	2 0	1 0	0,667	0,667	1	0,8

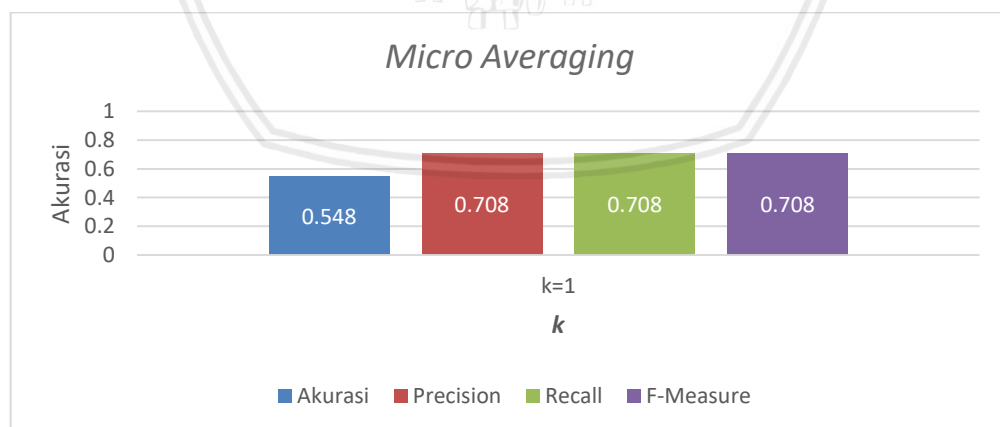
Tabel 6.9 Evaluasi Pengujian Fitur Bentuk SMSD Per Kelas (Lanjutan)

No	Kelas	Confusion Matrix		Akurasi	Precision	Recall	F-Measure
		TP FN	FP TN				
9	021	2 0	1 0	0,667	0,667	1	0,8
10	027	2 0	0 0	0,333	0,5	0,5	0,5
11	029	2 0	0 0	1	1	1	1
12	032	1 1	0 0	0	0	0	0

Dari hasil *confusion matrix* pada Tabel 6.8 dapat dihasilkan nilai *micro averaging* dan *macro averaging*. Pada Tabel 6.9 dihasilkan nilai akurasi, *precision*, *recall*, *f-measure* dari masing-masing kelas yang digunakan untuk menghitung nilai *micro-averaging* dan *macro-averaging*. Pada Tabel 6.10, menunjukkan hasil *macro-averaging* dan *micro-averaging* pada pengujian fitur SMSD dengan nilai $k=1$.

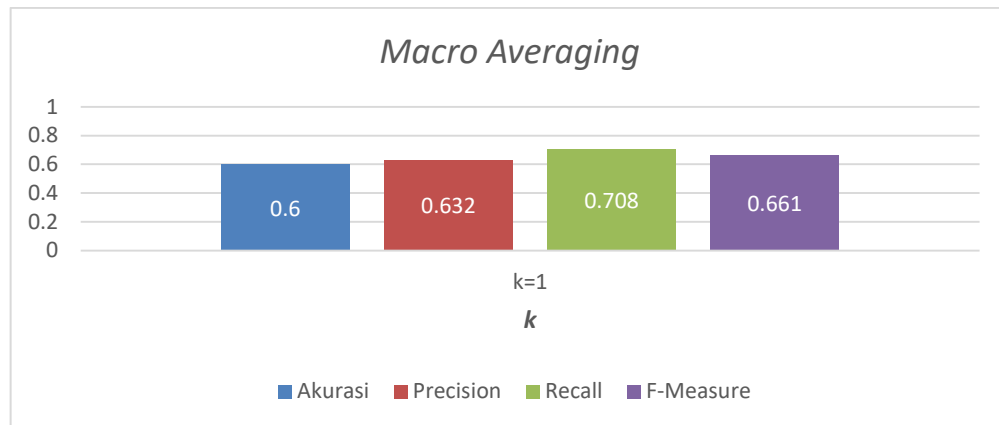
Tabel 6.10 Micro & Macro Averaging Fitur SMSD

No	k	Micro Averaging				Macro Averaging			
		Akurasi	Precision	Recall	F-Measure	Akurasi	Precision	Recall	F-Measure
1	1	0,548	0,708	0,708	0,708	0,6	0,632	0,708	0,661



Gambar 6.6 Grafik Micro Averaging Fitur SMSD

Pada pengujian fitur bentuk SMSD dihasilkan nilai akurasi sebesar 0,6 atau 60%. Hal ini disebabkan karena pada *dataset* yang digunakan yang memiliki kesamaan pada fitur bentuk mulai dari panjang dan lebar objeknya sampai ke jumlah piksel pada objeknya. Namun, nilai dari *precision* dan *recall* nya memiliki nilai yang tetap.



Gambar 6.7 Grafik *Macro Averaging* Fitur SMSD

Pada Gambar 6.7 menunjukkan bahwa nilai dari akurasi, *precision*, *recall*, *f-measure* berbeda dengan nilai yang ada pada Gambar 6.6. Pada *micro-averaging* menghasilkan rata-rata nilai akurasi sebesar 0,6 atau 60%. Namun pada nilai *precision*, *recall*, dan *f-measure* menghasilkan nilai yang tidak tetap atau tidak seimbang. Hal ini disebabkan karena jumlah data *training* pada setiap kelas tidak sama.

6.5 Pengujian Fitur Warna *Grayscale Histogram* dan Fitur Bentuk *Simple Morphological Shape Descriptors*

Pengujian ketiga dilakukan dengan menggunakan metode ekstraksi fitur warna *Grayscale Histogram*, fitur bentuk *Simple Morphological Shape Descriptors* dan *K-Nearest Neighbor (KNN)* dalam melakukan klasifikasi pada data uji. Pada pengujian fitur warna *Grayscale Histogram* menggunakan rasio jumlah data dengan akurasi tertinggi pada pengujian rasio jumlah data yaitu 10:90 dan menggunakan nilai k dengan akurasi tertinggi pada pengujian nilai k yaitu $k=1$. Metode *Grayscale Histogram* digunakan untuk fitur warna dan menghasilkan fitur warna yaitu *mean*, *standard deviation*, dan *skewness*. Metode SMSD digunakan untuk fitur bentuk dan menghasilkan fitur warna yaitu *area*, *length*, *width*, *rectangularity* N dan *aspect ratio*. Hasil pengujian fitur *Grayscale Histogram* dan fitur SMSD ditunjukkan pada Tabel 6.11. *Confusion matrix* dari hasil pengujian fitur warna *Grayscale Histogram* dan fitur bentuk SMSD ditunjukkan pada Tabel 6.12.

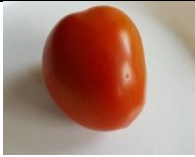
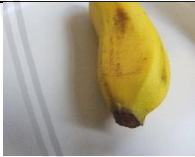
Tabel 6.11 Hasil Pengujian Data Uji Fitur *Grayscale Histogram* dan Fitur SMSD

No	Data Uji	Hasil Sebenarnya	Hasil Klasifikasi
1		001	001

Tabel 6.11 Hasil Pengujian Data Uji Fitur *Grayscale Histogram* dan Fitur SMSD (Lanjutan)

No	Data Uji	Hasil Sebenarnya	Hasil Klasifikasi
2		001	001
3		004	008
4		004	004
5		008	008
6		008	008
7		009	009
8		009	009
9		013	013
10		013	013

Tabel 6.11 Hasil Pengujian Data Uji Fitur *Grayscale Histogram* dan Fitur SMSD (Lanjutan)

No	Data Uji	Hasil Sebenarnya	Hasil Klasifikasi
11		015	027
12		015	015
13		017	017
14		017	017
15		018	018
16		018	018
17		021	021
18		021	021
19		027	027

Tabel 6.11 Hasil Pengujian Data Uji Fitur *Grayscale Histogram* dan Fitur SMSD (Lanjutan)

No	Data Uji	Hasil Sebenarnya	Hasil Klasifikasi
20		027	027
21		029	029
22		029	029
23		032	032
24		032	018

Tabel 6.12 *Confusion Matrix* Hasil Pengujian Fitur *Grayscale Histogram* dan Fitur SMSD

Actual Class	Predicted Class											
	001	004	008	009	013	015	017	018	021	027	029	032
001	2	0	0	0	0	0	0	0	0	0	0	0
004	0	1	1	0	0	0	0	0	0	0	0	0
008	0	0	2	0	0	0	0	0	0	0	0	0
009	0	0	0	2	0	0	0	0	0	0	0	0
013	0	0	0	0	2	0	0	0	0	0	0	0
015	0	0	0	0	0	1	0	0	1	0	0	0
017	0	0	0	0	0	0	2	0	0	0	0	0
018	0	0	0	0	0	0	0	2	0	0	0	0

Tabel 6.12 Confusion Matrix Hasil Pengujian Fitur *Grayscale Histogram* dan Fitur SMSD (Lanjutan)

Actual Class	Predicted Class											
	001	004	008	009	013	015	017	018	021	027	029	032
021	0	0	0	0	0	0	0	0	2	0	0	0
027	0	0	0	0	0	0	0	0	0	2	0	0
029	0	0	0	0	0	0	0	0	0	0	2	0
032	0	0	0	0	0	0	0	1	0	0	0	1

Tabel 6.13 Evaluasi Pengujian Fitur Warna *Grayscale Histogram* dan SMSD Per Kelas

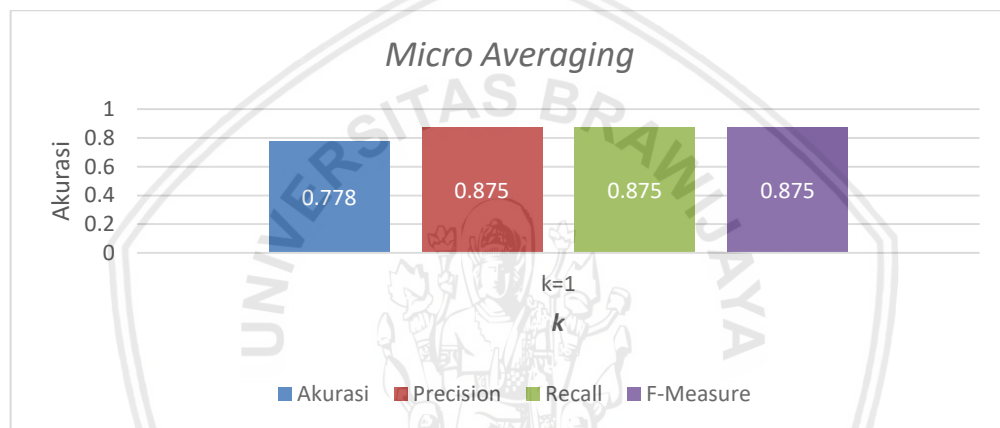
No	Kelas	Confusion Matrix		Akurasi	Precision	Recall	F-Measure
		TP FN	FP TN				
1	001	2 0	0 0	1	1	1	1
2	004	1 1	0 0	0,5	1	0,5	0,667
3	008	1 0	1 0	0,667	0,667	1	0,8
4	009	2 0	0 0	1	1	1	1
5	013	2 0	0 0	1	1	1	1
6	015	1 1	0 0	0,5	1	0,5	0,667
7	017	2 0	0 0	1	1	1	1
8	018	2 0	1 0	0,667	0,667	1	0,8
9	021	2 0	1 0	0,667	0,667	1	0,8
10	027	2 0	0 0	1	1	1	1
11	029	2 0	0 0	1	1	1	1
12	032	1 1	0 0	0,5	1	0,5	0,667

Dari hasil *confusion matrix* pada Tabel 6.11, dapat dihasilkan nilai *micro averaging* dan *macro averaging*. Pada Tabel 6.12 dihasilkan nilai akurasi, *precision*, *recall*, *f-measure* dari masing-masing kelas yang digunakan untuk menghitung nilai

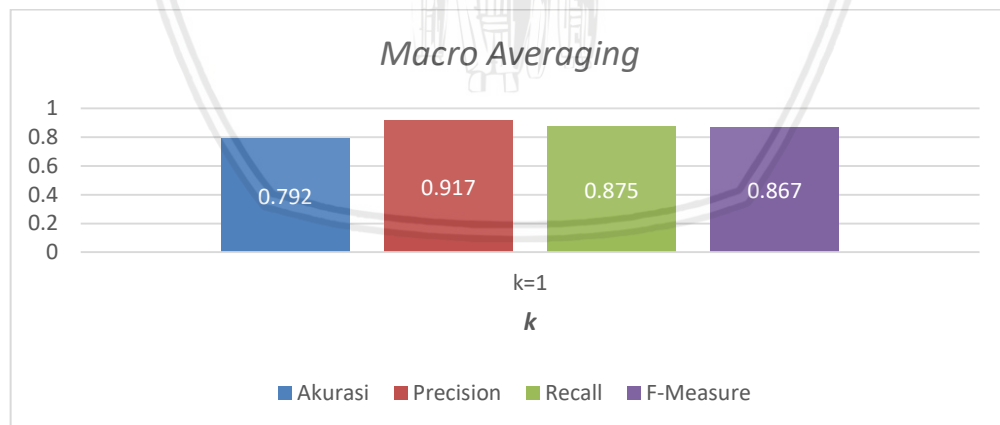
micro-averaging dan *macro-averaging*. Pada Tabel 6.13, menunjukkan hasil *macro-averaging* dan *micro-averaging* pada pengujian fitur warna *Grayscale Histogram* dan fitur bentuk SMSD dengan nilai $k=1$ dan rasio jumlah data 10:90. Pada Gambar 6.7 dan Gambar 6.8 menunjukkan grafik dari hasil hasil *macro-averaging* dan *micro-averaging*.

Tabel 6.14 Micro & Macro Averaging Fitur *Grayscale Histogram* dan SMSD

No	k	Micro Averaging				Macro Averaging			
		Akurasi	Precision	Recall	F-Measure	Akurasi	Precision	Recall	F-Measure
1	1	0,778	0,875	0,875	0,875	0,792	0,917	0,875	0,867



Gambar 6.8 Grafik *Micro Averaging* Fitur *Grayscale Histogram* dan SMSD



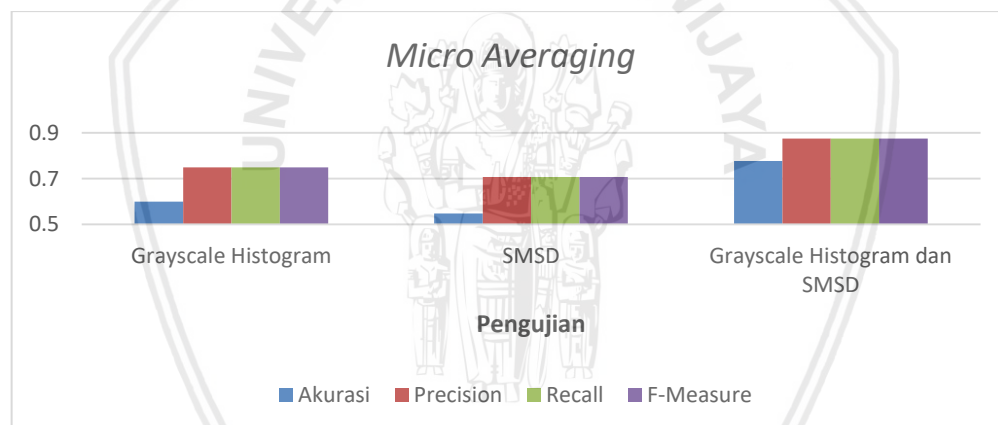
Gambar 6.9 Grafik *Macro Averaging* *Grayscale Histogram* dan SMSD

Pada pengujian fitur warna *Grayscale Histogram* dan fitur bentuk SMSD menggunakan *micro-averaging* yang ditunjukkan pada Gambar 6.8 dihasilkan akurasi sebesar 0,778 atau 77,8%. Selain itu, nilai dari *f-measure* yang dihasilkan memiliki nilai yang sama seperti nilai *precision* dan *recall*. Dari kesamaan tersebut, membuktikan bahwa nilai dari *precision* dan *recall* terdapat keseimbangan antar nilai.

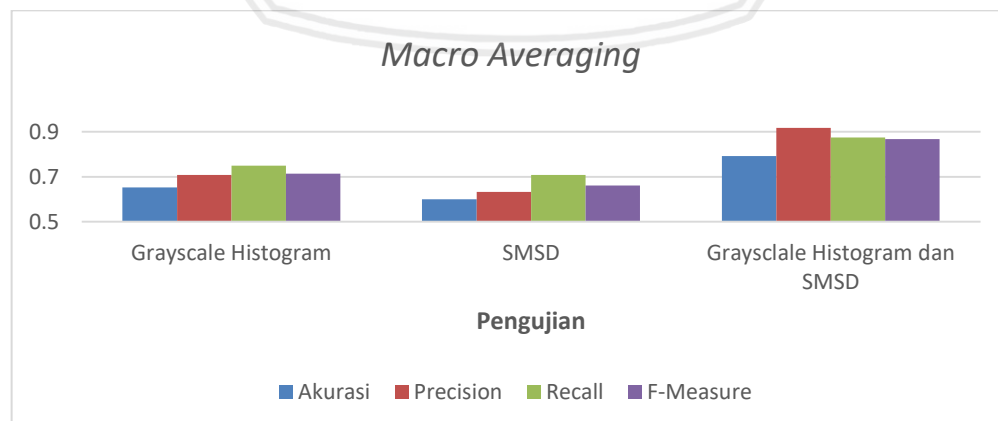
Pada Gambar 6.9 menunjukkan bahwa nilai dari akurasi, *precision*, *recall*, *f-measure* berbeda dengan nilai yang ada pada Gambar 6.6. Pada *micro-averaging* menghasilkan rata-rata nilai akurasi sebesar 0,792 atau 79,2%. Namun pada nilai *precision*, *recall*, dan *f-measure* menghasilkan nilai yang tidak seimbang. Hal ini disebabkan karena jumlah data *training* pada setiap kelas tidak sama.

Tabel 6.15 Perbandingan Kombinasi Fitur

Metode	Akurasi		<i>Precision</i>		<i>Recall</i>		<i>F-Measure</i>	
	<i>micro</i>	<i>macro</i>	<i>micro</i>	<i>macro</i>	<i>micro</i>	<i>macro</i>	<i>micro</i>	<i>macro</i>
<i>Grayscale Histogram</i>	0,6	0,653	0,75	0,708	0,75	0,75	0,75	0,714
SMSD	0,548	0,6	0,708	0,632	0,708	0,708	0,708	0,661
<i>Grayscale Histogram dan SMSD</i>	0,778	0,792	0,875	0,917	0,875	0,875	0,875	0,867



Gambar 6.10 Grafik *Micro Averaging* Perbandingan Pengujian



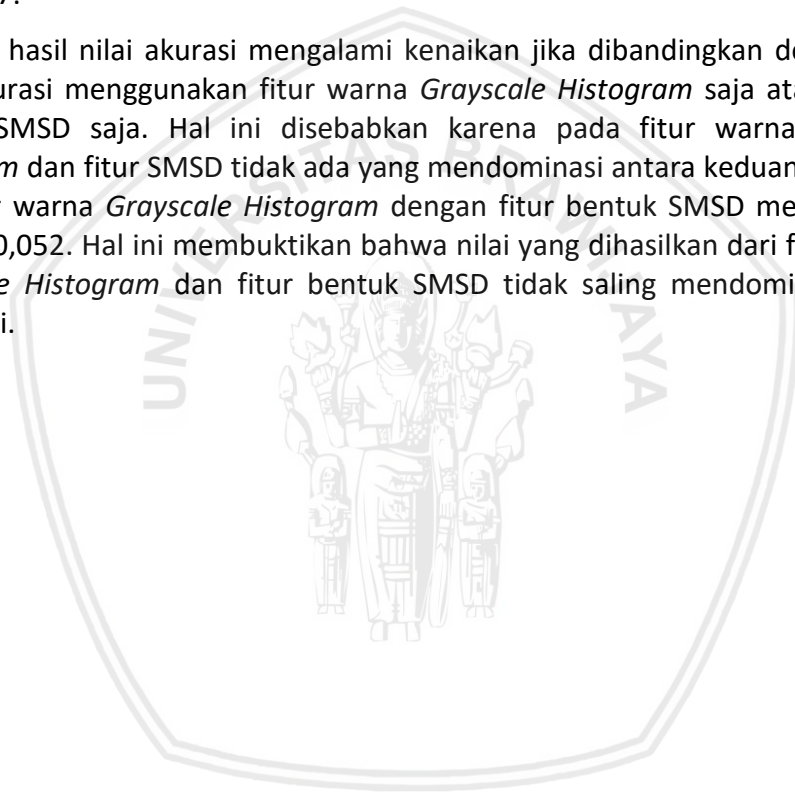
Gambar 6.11 Grafik *Macro Averaging* Perbandingan Pengujian

Dari hasil pada Gambar 6.10, dapat dilihat bahwa nilai dari akurasi, *precision*, *recall* dan *f-measure* dengan menggunakan *micro-averaging* memiliki jarak yang

dekat. Hasil dari pengujian SMSD lebih kecil dibandingkan hasil dari pengujian *Grayscale Histogram*. Namun hasil dari kombinasi lebih baik dari pengujian setiap fitur bentuk ataupun fitur warna yang digunakan. Hal ini disebabkan karena seimbangannya nilai *precision*, *recall*, dan *f-measure* dari setiap metode ekstraksi fiturnya sendiri. Nilai tertinggi akurasi, *precision*, *recall*, dan *f-measure* menggunakan *micro-averaging* dihasilkan oleh kombinasi antara fitur *Grayscale Histogram* dan SMSD yaitu 0,778, 0,875, 0,875, dan 0,875.

Nilai dari akurasi, *precision*, *recall*, dan *f-measure* dengan menggunakan *macro-averaging* yang ditunjukkan pada Gambar 6.1. Nilai tertinggi akurasi, *precision*, *recall*, dan *f-measure* menggunakan *macro-averaging* dihasilkan oleh kombinasi antara fitur *Grayscale Histogram* dan SMSD yaitu 0,792, 0,917, 0,875, dan 0,867.

Dari hasil nilai akurasi mengalami kenaikan jika dibandingkan dengan hasil nilai akurasi menggunakan fitur warna *Grayscale Histogram* saja ataupun fitur bentuk SMSD saja. Hal ini disebabkan karena pada fitur warna *Grayscale Histogram* dan fitur SMSD tidak ada yang mendominasi antara keduanya. Akurasi dari fitur warna *Grayscale Histogram* dengan fitur bentuk SMSD memiliki jarak sebesar 0,052. Hal ini membuktikan bahwa nilai yang dihasilkan dari fitur bentuk *Grayscale Histogram* dan fitur bentuk SMSD tidak saling mendominasi dalam klasifikasi.



BAB 7

KESIMPULAN

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

7.1 Kesimpulan

Kesimpulan yang didapat dalam penelitian ini yaitu sebagai berikut.

1. Metode ekstraksi fitur warna *Grayscale Histogram* dan ekstraksi fitur bentuk *Simple Morphological Shape Descriptors* dapat digunakan dalam melakukan klasifikasi citra makanan.
2. Metode yang digunakan dalam melakukan klasifikasi citra menggunakan KNN dengan membandingkan nilai k yang digunakan memengaruhi nilai akurasi. Jika menggunakan nilai $k=1$ dalam melakukan klasifikasi maka nilai akurasi yang dihasilkan sebesar 79,2%. Jika menggunakan nilai $k=3$ dalam melakukan klasifikasi maka nilai akurasi yang dihasilkan sebesar 70,8%. Serta, jika menggunakan nilai $k=5$ dalam melakukan klasifikasi maka nilai akurasi yang dihasilkan sebesar 54,2%.
3. Dari penelitian ini didapatkan hasil rata-rata nilai akurasi menggunakan *micro-averaging* dan *macro-averaging* yaitu 60% dan 65,3% jika hanya menggunakan fitur warna *Grayscale Histogram*. Pada pengujian fitur bentuk *Simple Morphological Shape Descriptors* dihasilkan nilai akurasi 54,8% dan 60%. Pada kombinasi fitur bentuk dan warna dihasilkan nilai akurasi 77,8% dan 79,2%. Hal ini disebabkan oleh banyak faktor, yaitu kesamaan dominan warna dan bentuk pada *dataset* yang digunakan dan juga banyaknya data yang digunakan pada data *training*.

7.2 Saran

Saran yang didapatkan dari hasil penelitian ini yaitu sebagai berikut.

1. Diperlukan penambahan data latih yang digunakan agar menghasilkan klasifikasi yang lebih tinggi.
2. Diperlukan penambahan ekstraksi fitur yang digunakan agar fitur yang digunakan untuk klasifikasi makanan bertambah.

DAFTAR REFERENSI

- Budianita, E., 2015. Implementasi Pengolahan Citra dan Klasifikasi K-Nearest Neighbour Untuk Membangun Aplikasi Pembeda Daging Sapi dan Babi. *Jurnal Sains, Teknologi dan Industri*, Volume 12, pp. 242-247.
- Ciputra, A., Moses Sediadi, D. R. I., Rahmawanto, E. H. & Susanto, A., 2018. KLASIFIKASI TINGKAT KEMATANGAN BUAH APEL MANALAGI DENGAN ALGORITMA NAIVE BAYES DAN EKSTRAKSI FITUR CITRA DIGITAL. *SIMETRIS*, 9(1).
- C, S. A., M, T. & P, A., 2014. A Study on Curvature Scale Space. *International Journal of Innovative Research in Computer and Communication Engineering*, 2(3), pp. 168-173.
- Gonzalez, R. C., 2002. *Digital Image Processing*. 2nd ed. Upper Saddle River New Jersey 07458: Prentice Hall.
- Han, J., Kamber, M. & Pei, J., 2006. *Data Mining Concepts & Techniques*. 2nd ed. San Fransisco: Elsevier.
- Karmilasari & Sumarna, A., 2011. TEMU KEMBALI CITRA BERBASIS CITRA WARNA. *Seminar Nasional Aplikasi Teknologi Informasi*.
- Kumar, K., Li, J.-P., Z.-u.-a. & Shaikh, R. A., 2016. Content Based Image Retrieval Using Gray Scale Weighted Average Method. *International Journal of Advanced Computer Science and Applications*, 7(1).
- Mali, S. N. & L., T. M., 2014. Color Histogram Features for Image Retrieval Systems. *International Journal of Innovative Research in Science, Engineering and Technology*, 3(4).
- Mudjiyanto, T. T., Susanto, D. & Luciasari, E., 1994. KEBIASAAN MAKAN GOLONGAN REMAJA DI ENAM KOTA BESAR DI INDONESIA. In: *Penelitian Gizi dan Makanan*. s.l.:National Institute of Health Research and Development, Ministry of Health of Republic of Indonesia.
- Santoso, B., 2007. *Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis*. 1st ed. Yogyakarta: Graha Ilmu.
- Saparinto, C. & Hidayati, 2010. *Bahan Tambahan Pangan*. Yogyakarta: Kanisius.
- Sokolova, M. & Lapalme, G., 2009. A Systematic Analysis of Performance Measures For Classification Tasks. *Information Processing and Management*, 45(4), pp. 427-437.
- Wäldchen, J. & Mäder, P., 2016. Plant Species Identification Using Computer Vision Techiques: A Systematic Literature Review. *Computational Methods In Engineering*.