



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KI141502

SISTEM KLASIFIKASI CITRA MAKANAN MENGUNAKAN REPRESENTASI ANTI TEXTONS DAN K-NEAREST NEIGHBOUR

NUNING SEPTIANA
5113100143

Dosen Pembimbing
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.
Dini Adni Navastara, S.Kom., M.Sc.

JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2017



[Halaman ini sengaja dikosongkan]



TUGAS AKHIR - KI141502

SISTEM KLASIFIKASI CITRA MAKANAN MENGUNAKAN REPRESENTASI ANTI TEXTONS DAN K-NEAREST NEIGHBOUR

NUNING SEPTIANA
5113100143

Dosen Pembimbing I
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.

Dosen Pembimbing II
Dini Adni Navastara, S.Kom., M.Sc.

JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya, 2017



[Halaman ini sengaja dikosongkan]



FINAL PROJECT - KI141502

FOOD CLASSIFICATION SYSTEMS USING REPRESENTATION ANTI TEXTONS AND K-NEAREST NEIGHBOR

NUNING SEPTIANA
5113100143

Supervisor I
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.

Supervisor II
Dini Adni Navastara, S.Kom., M.Sc.

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
Sepuluh Nopember Institute of Technology
Surabaya, 2017



[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

SISTEM KLASIFIKASI CITRA MAKANAN MENGUNAKAN REPRESENTASI ANTI TEXTONS DAN K-NEAREST NEIGHBOUR

TUGAS AKHIR

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Rumpun Mata Kuliah Komputasi Cerdas dan Visi
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

NUNING SEPTIANA

NRP: 5113 100 143

Disetujui oleh Dosen Pembimbing Tugas Akhir:

Dr. Eng. Nanik Suciati, S.Kom., M.Kom.
NIP. 197104281994122001

Dini Adni Navastara, S.Kom., M.Sc.
NIP. 198510172015042001



(Pembimbing 1)

(Pembimbing 2)

SURABAYA

JULI, 2017

[Halaman ini sengaja dikosongkan]

SISTEM KLASIFIKASI CITRA MAKANAN MENGUNAKAN REPRESENTASI ANTI TEXTONS DAN K-NEAREST NEIGHBOUR

Nama Mahasiswa : Nuning Septiana
NRP : 5113 100 143
Jurusan : Teknik Informatika, FTIf ITS
Dosen Pembimbing 1 : Dr. Eng. Nanik Suciati, S.Kom., M.Kom.
Dosen Pembimbing 2 : Dini Adni Navastara, S.Kom., M.Sc.

Abstrak

Saat ini, masyarakat mudah mendapatkan citra. Citra yang didapat bisa diambil sendiri lewat media seperti handphone atau dari internet. Kemudahan dalam mendapatkan citra menimbulkan kebutuhan untuk mengetahui informasi citra berdasarkan kontennya. Salah satu kebutuhannya yaitu klasifikasi citra. Klasifikasi citra bisa diterapkan pada dietary assesment, evaluasi komprehensif tentang makanan yang dikonsumsi oleh manusia.

Pada tugas akhir ini dibangun sistem klasifikasi citra menggunakan fitur anti-texton dan klasifikasi K-NN. Anti-Texton dihitung dengan cara mengukur jarak spasial dari texton. Texton sendiri berasal dari Texton Library yang didapatkan dengan menggunakan K-Means clustering pada data training. Input dari sistem ini adalah citra makanan dan output-nya adalah kelas makanan.

Berdasarkan uji coba, akurasi tertinggi mencapai 77,4%. Hasil ini dipengaruhi oleh kondisi dataset dimana terdapat beberapa data citra yang punya kemiripan meskipun kelasnya berbeda.

Kata kunci: Anti Texton, K-Means clustering, Texton library, K-NN, citra makanan

[Halaman ini sengaja dikosongkan]

FOOD CLASSIFICATION SYSTEMS USING REPRESENTATION ANTI TEXTONS AND K-NEAREST NEIGHBOR

Student Name : Nuning Septiana
Registration Number : 5113 100 143
Department : Informatics Engineering, FTIf ITS
First Supervisor : Dr. Eng. Nanik Suciati, S.Kom.,
M.Kom.
Second Supervisor : Dini Adni Navastara, S.Kom., M.Sc

Abstract

In this era, it is easy to get an image. People can get it by media like phone or from internet. The easiness in getting image make people become need about the information of image based on content. One of the exampe is image classification. Image classification can be implemented in Dietary assesment, a comprehensive evaluation of a person's food intake.

In this final project the system is being constructed using the Anti-Texton representation and K-NN. Anti-Texton is a value of spatial distance between texton. The texton itself is gained from Texton Library which is obtained by K-Means clustering of training data. The system need food image as the input and will decide the class that suit to the food image as the output

Based on the test results, the maximum average accuracy of this classification is 60.2%. This result is caused by the condition of the dataset which have some similar image eventhough it is different class.

Keywords: Anti Texton, Dietary assesment, K-Means clustering, Texton library, K-NN, food images

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji bagi Allah SWT yang telah melimpahkan rahmat dan anugerah-Nya sehingga penulis dapat menyelesaikan tugas akhir yang berjudul **“SISTEM KLASIFIKASI CITRA MAKANAN MENGGUNAKAN REPRESENTASI ANTI TEXTONS DAN K-NEAREST NEIGHBOUR”**

Buku tugas akhir ini disusun dengan harapan dapat memberikan manfaat dalam penelitian klasifikasi citra makanan lebih lanjut. Selain itu, penulis berharap dapat memberikan kontribusi positif bagi kampus Teknik Informatika ITS. Dalam perancangan, pengerjaan, dan penyusunan tugas akhir ini, penulis banyak mendapatkan bantuan dari berbagai pihak. Penulis ingin mengucapkan terima kasih kepada:

1. Ibu Dr.Eng. Nanik Suciati, S.Kom.,M.Kom. dan Ibu Dini Adni Navastara, S.Kom., M.Sc. selaku dosen pembimbing penulis yang telah memberi ide, nasihat dan arahan sehingga penulis dapat menyelesaikan tugas akhir dengan tepat waktu.
2. Orang tua penulis Ibu Chairunnisak yang telah memberikan dukungan moral, spiritual dan material serta senantiasa memberikan doa demi kelancaran dan kemudahan penulis dalam mengerjakan tugas akhir.
3. Kakak saya mas Leo dan keluarga (Mbak Icha, Pakde Didik, Bude Lina) serta seluruh keluarga besar yang telah memberikan dukungan yang besar baik secara langsung maupun secara implisit.
4. Ilham sebagai pembimbing waktu implementasi pengerjaan dan teman-teman satu bimbingan: Sani, Ine, Ayu, Lophita, Asri, dan Nela.

5. Teman –teman seperti Rere, Eriko, Lusi, Nindy dan , dan mahasiswa angkatan 2013 lain yang sama-sama mengarungi empat tahun masa perkuliahan bersama penulis.
6. Raga Krilido yang telah membantu dalam pengerjaan buku dan memberi semangat.
7. Pihak-pihak lain yang tidak bisa penulis sebutkan satu-persatu.

Penulis menyadari masih ada kekurangan dalam penyusunan tugas akhir ini. Penulis mohon maaf atas kesalahan, kelalaian maupun kekurangan dalam penyusunan tugas akhir ini. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan ke depan.

Surabaya, Januari 2017

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	v
Abstrak	vii
<i>Abstract</i>	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvi
DAFTAR KODE SUMBER	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Tugas Akhir	3
1.5 Manfaat Tugas Akhir	3
1.6 Metodologi.....	3
1.7 Sistematika Laporan.....	4
BAB II DASAR TEORI	7
2.1 Kelas Makanan.....	7
2.2 LAB <i>Colorspace</i>	10
2.3 <i>Schmidt Filter Bank</i>	11
2.4 <i>K-means Clustering</i>	12
2.5 <i>Distance Transform</i>	13
2.6 Normalisasi	14
2.6.1 Normalisasi <i>Filtering</i>	14
2.6.2 Normalisasi Ekstraksi <i>Anti-Texton</i>	14
2.7 <i>Texton</i>	15
2.7.1 <i>Bag-of-texton</i>	15
2.7.2 Representasi <i>Anti-Textons</i>	15
2.8 <i>Euclidean Distance</i>	17
2.9 <i>Chi-Square Distance</i>	17
2.10 Algoritma k-Nearest Neighbour	17
2.11 <i>Confusion Matrix</i>	18

2.12	<i>K-fold Cross validation</i>	18
	BAB III PERANCANGAN SISTEM	21
3.1	Perancangan Sistem	21
3.2	Perancangan Data	22
3.3	Perancangan Proses	22
3.3.1	<i>Preprocessing</i>	24
3.3.2	<i>Filtering</i>	25
3.3.3	Pembentukan <i>Texton Library</i>	26
3.3.4	Ekstraksi <i>Texton map</i>	28
3.3.5	Ekstraksi <i>Anti-Texton</i>	28
3.3.6	Klasifikasi K-NN	30
	BAB IV IMPLEMENTASI	33
4.1	Lingkungan Implementasi	33
4.2	Implementasi Tahap <i>Preprocessing</i>	33
4.3	Implementasi <i>Filtering</i>	34
4.4	Implementasi Pembentukan <i>Texton Library</i>	34
4.5	Implementasi Ekstraksi <i>Texton map</i>	35
4.6	Implementasi Ekstraksi <i>Anti-Texton</i>	36
4.7	Implementasi Proses Klasifikasi	37
4.8	Implementasi K-Fold <i>Cross validation</i>	37
	BAB V UJI COBA DAN EVALUASI	41
5.1	Lingkungan Uji Coba	41
5.2	Data Uji Coba	41
5.3	Skenario Uji Coba	42
5.4	Uji Coba Penentuan <i>Ukuran Texton Library</i> dengan Menggunakan Parameter <i>k</i> pada <i>k-means clustering</i>	42
5.5	Uji Coba Penentuan <i>k</i> pada K-NN	43
5.6	Evaluasi	43
	BAB VI KESIMPULAN DAN SARAN	47
6.1	Kesimpulan	47
6.2	Saran	48
	BIODATA PENULIS	90

DAFTAR GAMBAR

Gambar 2.1 LAB Colorspace	11
Gambar 2.2 Schmidt Filter Bank[7]	12
Gambar 2.3 Ilustrasi K-Means <i>Clustering</i> (a) Data Awal, (b) Inisialisasi <i>Centroid</i> ,(c) Mengelompokkan Data berdasarkan Jarak,(d) <i>Centroid</i> baru dipilih jika mengelompokkan salah ,(e) Pengelompokkan ulang dengan <i>Centroid</i> baru,(f) Pengelompokkan selesai.....	13
Gambar 2.4 <i>Chessboard Distance transform</i> [10].....	14
Gambar 2.5 K-Fold <i>Cross validation</i> [20].....	19
Gambar 3.1 Diagram Alir Klasifikasi Citra Makanan menggunakan Anti-Texton.....	23
Gambar 3.2 Diagram Alir Preprocessing.....	24
Gambar 3.3 Citra RGB (asli).....	25
Gambar 3.4 Citra LAB <i>Colorspace</i>	25
Gambar 3.5 Diagram Alir <i>Training Texton Library</i>	27
Gambar 3.6 Diagram Alir Ekstraksi <i>Texton map</i>	28
Gambar 3.7 <i>Texton Map</i> dari citra masukan.....	29
Gambar 3.8 <i>Binary Map</i> dari citra masukan.....	29
Gambar 3.9 Contoh <i>map Ei</i>	30
Gambar 3.10 Histogram <i>Anti-Texton</i>	30
Gambar 3.11 Diagram Alir K-NN.....	31
Gambar 3.12 Diagram Alir Ekstraksi <i>Anti-Texton</i>	32
Gambar 5.1 Contoh Citra dari UNICT-FD1200	42

DAFTAR TABEL

Tabel 2.1 Contoh Citra Makanan	8
Tabel 2.2 Confusion Matrix 8 class	19
Tabel 3.1 <i>Filter Response</i>	25
Tabel 4.1 Spesifikasi Perangkat Keras yang Digunakan	33
Tabel 5.1 Confusion Matrix 8 class	45
Tabel 5.2 Akurasi berdasarkan <i>texton library</i>	45
Tabel 5.3 Akurasi berdasarkan ukuran K-NN	46

DAFTAR KODE SUMBER

Kode Sumber 4.1 Implementasi perubahan tipe colorspace.....	34
Kode Sumber 4.2 Implementasi <i>Filtering</i>	34
Kode Sumber 4.3 Implementasi Training Texton <i>Library</i>	35
Kode Sumber 4.4 implementasi dari K-NN dengan k=1	39
Kode Sumber 4.6 K-fold <i>Cross validation</i>	40

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

Pada bab ini dibahas hal-hal yang mendasari tugas akhir. Bahasan meliputi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, metodologi, dan sistematika laporan tugas akhir.

1.1 Latar Belakang

Dengan perkembangan teknologi saat ini manusia sudah tidak mengalami kesulitan lagi dalam mengambil dan mengolah citra secara digital. Hal ini menjadikan jumlah data citra yang tersebar meningkat serta lebih. Banyaknya citra yang tersedia menimbulkan keingintahuan kesulitan tersendiri untuk mendapatkan citra yang diinginkan. Ada pula kasus lain dimana manusia ingin mengetahui informasi dari citra yang didapatkan secara otomatis. Oleh karena itu muncul kebutuhan akan sistem yang bisa mengenal objek khususnya klasifikasi citra.

Di bidang kesehatan diet menjadi salah satu faktor penting dalam kesembuhan pasien. Diet adalah pola makan yang dilakukan manusia untuk menjaga kesehatan. Namun, Diet yang salah bisa menyebabkan masalah kesehatan seperti diabetes, obesitas, serta bisa menimbulkan resiko alergi. Oleh karena itu diperlukan *dietary assesment*, evaluasi komprehensif tentang makanan yang dikonsumsi oleh manusia. Namun *dietary assesment* masih dilakukan secara konvensional yang menyebabkan hasil kurang akurat.[1] Teknologi klasifikasi citra memungkinkan pengguna untuk mengetahui secara otomatis apakah makanan yang disajikan sesuai dengan kebutuhan dietnya dengan mengambil citra makanan tersebut.

Penelitian tentang sistem klasifikasi citra makanan sudah pernah dilakukan dengan menggunakan banyak metode. Taichi Joutou dan Keiji Yanai menggunakan *Multiple Kernel Learning* (MKL) untuk mengintegrasikan fitur citra seperti warna, tekstur, dan SIFT[2]. Metode ini mendapatkan nilai akurasi rata-rata sebesar

61.34%. Farinella et al. menggunakan *Bag of Textons* untuk ekstraksi fiturnya. Metode *Bag of Textons* menunjukkan hasil yang lebih akurat yaitu 79.6 % namun sebenarnya akurasi ini masih bisa ditingkatkan lagi dengan memperhitungkan jarak geometri ruang antar *Texton* yaitu *Anti-Texton*. [3]

Oleh karena itu pada tugas akhir ini dilakukan pengembangan sistem klasifikasi makanan menggunakan metode *Anti-Textons* dan *k-Nearest Neighbour* atau K-NN. Dalam proses pengembangan sistem ini diperlukan beberapa tahapan yaitu mendapatkan *Texton vocabulary* menggunakan *K-Means Clustering*, menghitung *Anti-Texton*, dan klasifikasi dengan K-NN. Data yang digunakan adalah data citra makanan dari UNICT-FD1200[4]. Diharapkan nilai akurasi dari sistem ini lebih baik daripada penelitian yang sudah dilakukan sehingga bisa memberi hasil yang tepat dan bisa dijadikan langkah awal dalam membentuk sistem *dietary assesment* otomatis.

1.2 Rumusan Masalah

Berikut adalah beberapa rumusan masalah yang diangkat dalam pembuatan Tugas Akhir ini :

1. Bagaimana melakukan ekstraksi fitur data citra makanan dengan menggunakan *Anti-Texton*?
2. Bagaimana melakukan klasifikasi data citra makanan dengan menggunakan K-NN?
3. Bagaimana melakukan uji performa perangkat lunak yang akan dibangun?

1.3 Batasan Masalah

Batasan masalah yang ditetapkan dalam pembuatan Tugas Akhir ini adalah sebagai berikut :

1. Dataset yang digunakan adalah dataset UNICT-FD1200 yang diambil dari *University of Catania*.
2. Pembangunan perangkat lunak menggunakan Matlab R2015.

3. Ada delapan kelas, yaitu *Appetizer, Main Course, Second Course, Single Course, Side Dish, Dessert, Breakfast, Fruit*.
4. Setiap data hanya memiliki satu label atau kelas.

1.4 Tujuan Tugas Akhir

Tujuan dari pembuatan tugas akhir ini adalah untuk mengembangkan sistem klasifikasi citra makanan menggunakan representasi *Anti-Texton* dan K-NN.

1.5 Manfaat Tugas Akhir

Manfaat yang diperoleh dari pembuatan tugas akhir ini adalah untuk memudahkan pasien melakukan *dietary assesment* dengan menerapkan sistem klasifikasi citra makanan agar hasilnya lebih akurat..

1.6 Metodologi

Tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir ini adalah sebagai berikut:

1. Studi Literatur

Pada studi literatur, dilakukan pengumpulan data dan studi terhadap sejumlah referensi yang diperlukan dalam pengerjaan tugas akhir. Referensi tersebut didapatkan dari beberapa artikel yang dipublikasikan oleh jurnal. Selain dari artikel, studi literatur juga dilakukan melalui pencarian referensi dari internet yang membahas mengenai informasi yang dibutuhkan, seperti *texton, image filtering, K-Means clustering, K-NN classification, dan visual codebook*.

2. Analisis dan Desain Perangkat Lunak

Untuk membangun perangkat lunak sistem klasifikasi citra makanan ini harus melalui beberapa tahap yaitu mendapatkan *Texton* vocabulary terlebih dahulu. Setelah itu mendapatkan representasi *Anti Textons* dari setiap data citra. Kemudian membuat model classifier dengan k-Means. Terakhir adalah

melakukan klasifikasi dengan metode K-NN untuk mencocokkan input dengan model.

3. Implementasi Perangkat Lunak

Perangkat lunak ini akan dibangun dengan menggunakan bahasa pemrograman dan kaskas bantu Matlab R2015a dengan fungsi yang sudah tersedia di dalamnya.

4. Uji Coba dan Evaluasi

Dalam tahap ini, dilakukan pengujian parameter-parameter yang dibutuhkan pada proses ekstraksi fitur, *clustering* dan klasifikasi. *Performance matrix* akan digunakan untuk mengukur performa klasifikasi.

1.7 Sistematika Laporan

Buku tugas akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan tugas akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri atas beberapa bagian seperti berikut:

Bab I Pendahuluan

Bab yang berisi mengenai latar belakang, tujuan, dan manfaat dari pembuatan tugas akhir. Selain itu permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penulisan juga merupakan bagian dari bab ini.

Bab II Dasar Teori

Bab ini berisi penjelasan secara detail mengenai dasar-dasar penunjang dan teori-teori yang digunakan untuk mendukung pembuatan tugas akhir ini.

Bab III Analisis dan Perancangan

Bab ini berisi tentang analisis dan perancangan desain sistem klasifikasi citra makanan.

Bab IV Implementasi

Bab ini membahas implementasi dari desain yang telah dibuat pada bab sebelumnya. Penjelasan berupa kode yang digunakan untuk proses implementasi.

Bab V Uji Coba dan Evaluasi

Bab ini membahas tahap-tahap uji coba. Kemudian hasil uji coba dievaluasi untuk kinerja dari aplikasi yang dibangun.

Bab VI Kesimpulan dan Saran

Bab ini merupakan bab terakhir yang menyampaikan kesimpulan dari hasil uji coba yang dilakukan dan saran untuk pengembangan aplikasi ke depannya.

[Halaman ini sengaja dikosongkan]

BAB II

DASAR TEORI

Pada bab ini diuraikan mengenai dasar-dasar teori yang digunakan dalam pengerjaan tugas akhir dengan tujuan untuk memberikan gambaran secara umum terhadap penelitian yang dikerjakan. Bab ini berisi penjelasan mengenai citra makanan sebagai ruang lingkup dan studi kasus, metode *bag of texton* untuk ekstraksi fitur *texton*, K-Means *clustering* dalam proses *learning stage* membentuk *texton library*, metode K-NN untuk klasifikasi.

2.1 Kelas Makanan

Secara garis besar, makanan bisa dikelompokkan kedalam 8 kelas yaitu: *Main Course*, *Single Course*, *Second Course*, *Side Dish*, *Breakfast*, *Appetizer*, *Dessert*, dan *Fruit*. [5]

Main Course adalah kelas makanan yang disajikan sebagai makanan utama. Di beberapa negara seperti U.S , makanan tipe *Main Course* biasa disajikan waktu makan malam. Makanan kelas *Main Course* biasanya berupa makanan berat dan komposisinya paling kompleks. Contohnya ikan, daging, dan makanan lain yang kaya protein. *Second Course* dan *Single Course* juga termasuk dalam golongan makanan berat namun tidak menjadi makanan utama seperti *Main Course*.

Side Dish adalah kelas makanan yang disajikan sebagai pendamping hidangan utama. *Side Dish* mempunyai porsi makanan yang lebih kecil daripada *Main Course*. Contoh makanan tipe *Side Dish* adalah salad.

Dessert adalah kelas makanan yang disajikan sebagai makanan penutup dari hidangan utama. Makanan yang masuk kelas *Dessert* biasanya adalah makanan manis seperti kue atau biskuit



Appetizer adalah kelas makanan yang disajikan sebelum hidangan utama. Makanan pada kelas ini memiliki porsi yang lebih kecil daripada hidangan utama. Contoh makanan tipe *Appetizer* adalah bruchetta.





Fruit adalah kelas makanan yang terdiri dari buah-buahan. makanan ini biasanya dihidangkan sebagai makanan pendamping. Contoh makanan dari kelas *Fruit* adalah jeruk, apel, dan anggur.



Breakfast atau sarapan adalah kelas makanan yang disajikan waktu pagi hari. Contoh makanan dari kelas *Breakfast* adalah croissant.

Tabel 2.1 memberikan contoh citra makanan dari masing-masing kelas makanan yang digunakan pada tugas akhir ini.

Tabel 2.1 Contoh Citra Makanan

Kategori	Contoh Citra
Main Course	
Single Course	

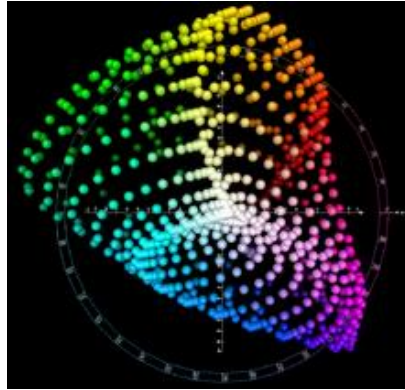
Kategori	Contoh Citra
Second Course	
Side Dish	
Breakfast	
Appetizer	

Kategori	Contoh Citra
Fruit	
Dessert	

2.2 LAB Colorspace

LAB *Colorspace* adalah sistem pewarnaan 3-axis dengan dimensi L untuk pencahayaan (*lightness*), dan a serta b untuk dimensi warna. LAB *colorspace* bekerja pada semua warna termasuk spektrum dan warna yang berada diluar persepsi manusia. *Colorspace* ini memiliki representasi warna paling akurat dan independen. Akurasi dan sifat independennya membuat LAB *colorspace* cocok digunakan untuk beberapa bidang seperti *printing*, otomotif, tekstil, dan plastik. [6]

LAB *colorspace* jarang digunakan oleh umum meski ia memiliki akurasi representasi warna paling tinggi. Biasanya citra LAB akan diubah ke mode *colorspace* yang kurang akurat seperti RGB dan CYMK, karena monitor komputer dan printer menggunakan tiga atau empat warna untuk merepresentasikan citra. **Gambar 2.1** adalah ilustrasi LAB *Colorspace* dari atas.



Gambar 2.1 LAB Colorspace

2.3 Schmidt Filter Bank

Filter citra digunakan untuk mereduksi frekuensi tinggi pada citra seperti *smoothing* atau frekuensi rendah seperti deteksi tepi. Beberapa contoh dari filter citra adalah *mean filter*, *median filter*, *gaussian filter* dan *gabor filter*. [7]

Filter terbagi menjadi filter frekuensi dan filter spasial. Pada filter frekuensi citra dirubah kedalam domain frekuensi dan dikalikan dengan filter frekuensi sebelum dikembalikan ke domain spasial. Sedangkan pada filter spasial, pada citra dilakukan konvolusi dengan sebuah *kernel*.

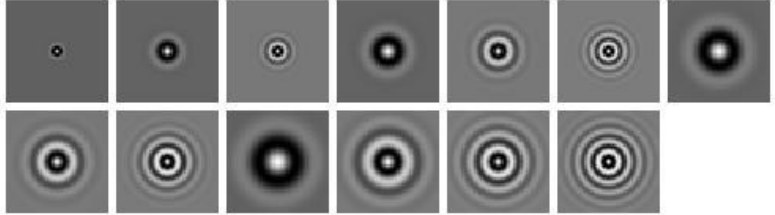
Schmidt Filter Bank adalah filter yang terdiri dari 13 filter isotropic [8]. Filter ini membutuhkan tiga parameter yaitu r , σ , dan τ . Filter ini lalu didefinisikan dengan rumus sebagai berikut:

$$F(r, \sigma, \tau) = F_0(\sigma, \tau) + \cos\left(\frac{\pi\tau r}{\sigma}\right) e^{-\frac{r^2}{2\sigma^2}} \quad (2.1)$$

Dimana r adalah radius, σ adalah filter scale dalam piksel dan τ adalah nilai proporsional dari angka concentric ring dalam kernel. $F_0(\sigma, \tau)$ ditambahkan untuk mendapat zero DC component dari pasangan (σ, τ) yang memiliki nilai (2,1), (4,1), (4,2), (6,1),

(6,2), (6,3), (8,1), (8,2), (8,3), (10,1), (10,2), (10,3) dan (10,4).

Gambar 2.2 menunjukkan filter dari Schmidt Filter Bank.



Gambar 2.2 Schmidt Filter Bank[7]

2.4 K-means Clustering

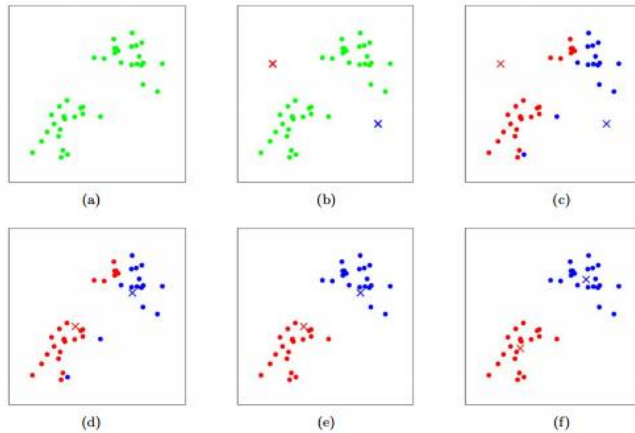
K-Means *clustering* adalah salah satu metode *unsupervised learning*, yaitu algoritma *machine learning* yang bertujuan untuk membentuk kelas-kelas atau kluster pada data yang belum atau tidak diketahui kelasnya. K-Means *clustering* adalah metode *clustering* untuk mempartisi sebuah data menjadi K kluster. [9]

K-Means *clustering* mengambil beberapa titik acak c_k sebagai titik pusat tiap kluster. Tiap titik pada data lalu dievaluasi jaraknya dengan masing-masing titik pada c_k hingga dapat ditemukan tiap titik adalah anggota kluster apa. Dalam perhitungan matematis J adalah jarak antara titik $x_i^{(j)}$ ke titik pusat c_j [26]. Nilai J dapat dihitung dengan dengan formula (2.4) berikut:

$$J = \sum_{j=1}^k \sum_{i=1}^n \left\| x_i^{(j)} - c_j \right\|^2 \quad (2.2)$$

Langkah selanjutnya, data yang sudah dikluster dilakukan penghitungan kembali titik pusat kluster untuk mendapat titik pusat baru menggantikan c_k . Titik pusat baru didapatkan dari rata-rata atribut seluruh anggota kluster. Setelah titik pusat baru didapat, dilakukan perulangan kembali menghitung jarak pada tiap titik ke titik-titik pusat kluster yang baru. K-Means *clustering* berhenti

ketika titik pusat tiap kluster tidak berubah nilainya ketika dilakukan penghitungan kembali titik pusat kluster. Proses K-Means *clustering* diilustrasikan pada **Gambar 2.3**.



Gambar 2.3 Ilustrasi K-Means Clustering
(a) Data Awal, (b) Inisialisasi Centroid, (c) Mengelompokkan Data berdasarkan Jarak, (d) Centroid baru dipilih jika pengelompokkan salah, (e) Pengelompokkan ulang dengan Centroid baru, (f) Pengelompokkan selesai

2.5 Distance Transform

Distance transform adalah metode yang biasa diaplikasikan pada *Binary Image* untuk mendapatkan representasi baru. Setiap piksel dengan jarak terdekat dari boundary piksel nya akan diberi label. Hasil dari transformasi ini adalah graylevel image yang mirip dengan *Binary image*. [10] **Gambar 2.4** adalah contoh *chessboard distance transform* dari *Binary image*.

0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0

⇒

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

Gambar 2.4 Chessboard Distance transform[10]

2.6 Normalisasi

Normalisasi adalah teknik untuk menstandarkan atau menyamakan rentang data sehingga tidak ada satu atribut yang terlalu dominan atas atribut yang lain.[11] Pada tugas akhir ini, normalisasi dilakukan dua kali, yaitu saat melakukan *filtering* dan proses ekstraksi *Anti-Texton*. Keduanya memakai metode normalisasi yang berbeda.

2.6.1 Normalisasi *Filtering*

Normalisasi yang dilakukan pada proses *filtering* menggunakan metode normalisasi Z-score. Z-score merupakan metode normalisasi yang berdasarkan mean (nilai rata-rata) dan standard deviation (deviasi standar) dari data[12]. Rumus normalisasi Z-score bisa dilihat pada persamaan 2.3. Dimana Z_i adalah nilai ke- i yang sudah dinormalisasi. X_i adalah nilai ke- i yang awal. \bar{X} adalah rata-rata dari semua nilai data yang belum dinormalisasi. S adalah standard deviasi dari data tersebut.

$$Z_i = (X_i - \bar{X})/s \quad (2.3)$$

2.6.2 Normalisasi Ekstraksi *Anti-Texton*

Pada proses ekstraksi Anti-texton, normalisasi dilakukan ketika *map* dari *distance transform* berhasil didapatkan. Langkah melakukan normalisasi dengan membagi setiap nilai di *map*

dengan nilai yang terbesar pada *map* tersebut. Rumus normalisasi ini bisa dilihat pada persamaan 2.4. D_i adalah nilai ke- i yang sudah dinormalisasi. T_i adalah nilai ke- i awal dan $\max(T)$ adalah nilai tertinggi pada *map*.

$$D_i = T_i/\max(T) \quad (2.4)$$

2.7 *Texton*

Texton adalah struktur mikro paling dasar pada citra alami. Sebuah *Texton* bisa terdiri dari beberapa pixel. Istilah ini sudah dikenalkan oleh Julesz sebagai unit yang dipertimbangkan untuk persepsi visual selama processing[13]. Model komputasi dari *texton* bisa didapat dengan mendapatkan *filter response* dari citra yang sudah mengalami konvolusi dengan filter bank. *Filter response* ini selanjutnya digunakan untuk melakukan ekstraksi *texton* dengan cara melakukan pembobotan vektor dengan *texton library* atau bag-of-texton.

2.7.1 *Bag-of-texton*

Bag-of-words bisa juga disebut *visual codebook*. *Visual codebook* adalah basis elemen untuk mendeskripsikan sebuah citra. Pada tugas akhir ini, elemen atau fitur yang digunakan adalah *texton*. Sehingga *visual codebook* yang dibentuk dinamakan *Bag-of-texton* atau *Texton Library*. [14] Langkah pembentukan *texton library* adalah dengan melakukan kuantisasi vektor *filter response* milik data training lewat *clustering*. Sehingga setiap centroid pada cluster tersebut bisa dianggap sebagai *texton* dan set yang terdiri dari *texton-texton* tadi disebut *visual codebook* atau lebih tepatnya *bag-of-texton*. Untuk merepresentasikan citra setiap piksel yang sudah difilter diasosiasikan dengan salah satu dari *texton* pada *Texton Library* yang dianggap paling dekat. Penghitungan asosiasi disini menggunakan *Euclidian distance*.

2.7.2 *Representasi Anti-Textons*

Texton adalah struktur mikro paling dasar pada citra alami. Sebuah *Texton* bisa terdiri dari beberapa piksel. Penelitian menunjukkan bahwa penggunaan *Texton* dengan metode statistika

sederhana bisa menghasilkan persepsi visual untuk membedakan tekstur secara signifikan. Sayangnya, representasi *Texton* tidak memperhitungkan hubungan spasial antar *visual word*, oleh karena itu pada tugas akhir ini kami menggunakan fitur *Anti-Texton* dimana geometri ruang antar *Texton* juga diperhitungkan agar mendapatkan performa yang lebih bagus pada klasifikasi nanti. [3]

Sebelum menghitung *Anti-Texton* dari citra kita harus mendapatkan *Texton vocabulary* terlebih dahulu. *Texton vocabulary* bisa didapat dengan mengolah data training. Berikut ini adalah langkah mendapatkan representasi *Anti-Texton*:

1. Mendapatkan *Texton map* dari masukan citra C . Setiap piksel dari *Texton map* diberi *Texton ID* i ($i=1, \dots, N$).
2. Mendapatkan *Binary Map* dari setiap *Texton*. *Binary Map* B_i dari *Texton* i mengandung nilai 1 dimana *Texton* i muncul dan 0 di posisi lainnya. Pada fase ini ada N *Binary Map* yang dihasilkan.
3. Menghitung *distance transform* dari setiap B_i . Hasil dari perhitungan ini adalah *saliency Map* dimana titik yang dekat dengan *Texton* i kurang menonjol daripada titik yang posisinya lebih jauh. *Saliency Map* ini digunakan untuk menentukan berapa banyak *Texton* dalam *Texton map* bisa dipertimbangkan sebagai *Anti-Texton* dari *Texton* i .
4. Setiap *saliency Map* dinormalisasi dengan membaginya menggunakan nilai yang paling besar. *Map* dari *Texton* i yang telah dinormalisasi dianotasikan sebagai D_i .
5. *Map* D_i di-invert menggunakan rumus:

$$E_i = 1 - D_i. \quad (2.5)$$

Ini adalah cara menghitung jarak ruang antara dua *Texton* yang berada pada kelas i yang sama. E_i adalah hasil invert map dari D_i .

6. Setiap E_i digunakan untuk menghitung bobot asli dari *Texton map* untuk mendapatkan distribusi final (histogram) dari *Anti-Texton* untuk *Texton* i .
7. Hasil rata-rata dari N buah histogram adalah representasi *Anti-Texton* dari citra C .

2.8 Euclidean Distance

Euclidean distance adalah metode penghitungan jarak antar dua titik yang paling umum digunakan. Pada tugas akhir ini, *Euclidian distance* digunakan untuk menghitung kedekatan antara nilai piksel data masukan dengan *Texton Library* untuk menentukan *texton*-nya. Rumus penghitungan jarak ini menggunakan teorema pitagoras yang bisa dilihat pada persamaan 2.6[15]. Variabel $d(p, q)$ adalah jarak antara titik p dan q .

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (2.6)$$

2.9 Chi-Square Distance

Chi-square distance adalah salah satu metode pengukuran yang digunakan untuk menganalisis dan teknik ordinas yang berhubungan. Pada tugas akhir ini, *Chi-square distance* digunakan untuk menghitung jarak antar histogram *anti-texton*. Rumus *chi-square distance* bisa dilihat pada persamaan 2.7.[16]. Variabel O adalah histogram O sedangkan E_i adalah rata-rata dari nilai bin ke- i pada histogram O dengan nilai bin ke- i pada histogram P .

$$x^2(O, P) = \sum_{i=1}^N \frac{(O_i - E_i)^2}{E_i} \quad (2.7)$$

2.10 Algoritma k-Nearest Neighbour

Algoritma K-NN (k-Nearest Neighbour) adalah salah satu metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang ada. K-NN termasuk algoritma *case-based learning*, artinya hasil dari sampel uji yang baru diklasifikan berdasarkan mayoritas dari kategori pada K-NN[5]. Tujuan dari klasifikasi ini adalah untuk mengklasifikan data masukan

berdasarkan atribut dan sampel dari data training. Langkah-langkah dalam melakukan klasifikasi menggunakan algoritma K-NN adalah sebagai berikut:

1. Tentukan nilai K
2. Hitung jarak antara nilai training data dengan input data
3. Urutkan jaraknya
4. Ambil K tetangga yang terdekat
5. Dapatkan nilai mayoritas sebagai hasil prediksi

2.11 *Confusion Matrix*

Confusion Matrix adalah tabel yang digunakan untuk menjelaskan performa dari model klasifikasi (classifier) pada data testing berdasarkan nilai yang didapat. Setiap kolom pada *confusion matrix* menunjukkan kelas prediksi untuk *query* klasifikasi, sedangkan tiap barisnya menunjukkan kelas aktual dari *query*. [18] Pada Tugas Akhir ini ada 8 kelas untuk klasifikasi. Tabel 2.2 adalah tabel *Confusion Matrix* untuk 8 kelas. $Tp(X)$ adalah nilai *true positif* untuk kelas X .

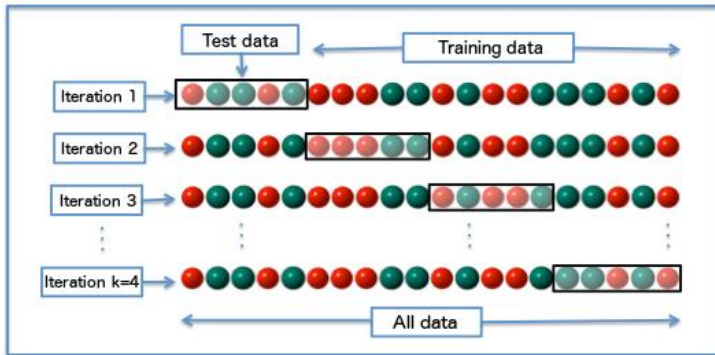
2.12 *K-fold Cross validation*

Cross validation adalah salah satu metode untuk evaluasi model prediksi. Metode ini bisa mengindikasikan seberapa baik model ini memprediksi data yang belum pernah dilihat sehingga pengujian ini membuat model tidak *overfitting*. [19]

Langkah pertama dalam melakukan *cross validation* adalah melakukan iterasi sebanyak k , dimana selama iterasi berlangsung data testing dan data training tidak pernah sama karena data testing pada iterasi sebelumnya menjadi data training dan pada iterasi selanjutnya diambil beberapa data dari data training di iterasi sebelumnya untuk dijadikan sebagai dataset. **Gambar 2.5** merupakan contoh diagram *K-fold cross validation* dimana $k=4$.

Tabel 2.2 Confusion Matrix 8 class

		Predicted Class							
		A	B	C	D	E	F	G	H
Actual class	A	Tp(A)	e(AB)	e(AC)	e(AD)	e(AE)	e(AF)	e(AG)	e(AH)
	B	e(BA)	Tp(B)	e(BC)	e(BD)	e(BE)	e(BF)	e(BG)	e(BH)
	C	e(CA)	e(CB)	Tp(C)	e(CD)	e(CE)	e(CF)	e(CG)	e(CH)
	D	e(DA)	e(DB)	e(DC)	Tp(D)	e(DE)	e(DF)	e(DG)	e(DH)
	E	e(EA)	e(EB)	e(EC)	e(ED)	Tp(E)	e(EF)	e(EG)	e(EH)
	F	e(FA)	e(FB)	e(FC)	e(FD)	e(FE)	Tp(F)	e(FG)	e(FH)
	G	e(GA)	e(GB)	e(GC)	e(GD)	e(GE)	e(GF)	Tp(G)	e(GH)
	H	e(HA)	e(HB)	e(HC)	e(HD)	e(HE)	e(HF)	e(HG)	Tp(H)



Gambar 2.5 K-Fold Cross validation[20]

[Halaman ini sengaja dikosongkan]



BAB III

PERANCANGAN SISTEM

Pada bab ini akan dijelaskan mengenai rancangan sistem perangkat lunak yang akan dibuat. Perancangan yang dijelaskan meliputi data dan proses. Data yang dimaksud adalah data yang akan diolah dalam perangkat lunak kemudian digunakan sebagai pembelajaran maupun pengujian sehingga tujuan Tugas Akhir ini bisa tercapai. Proses yaitu tahap-tahap yang ada dalam sistem sebagai pengolah data meliputi sistem klasifikasi citra makanan menggunakan representasi *Anti-Texton*.

3.1 Perancangan Sistem

Pada tugas akhir ini dibangun aplikasi aplikasi klasifikasi citra makanan menggunakan representasi *Anti-Texton*. Data masukan yang digunakan adalah data citra makanan yang berasal dari UNICT-FD1200. Data keluaran dari aplikasi hasil klasifikasi makanan.

Aplikasi ini diharapkan dapat digunakan untuk menunjang di bidang kesehatan khususnya pada *food monitoring* untuk *dietary assesment* pada pasien. Hasil klasifikasi citra pada tahap selanjutnya dapat digunakan sebagai informasi berapa banyak gizi yang dikonsumsi pasien.

Perancangan sistem dilakukan untuk menggambarkan proses secara keseluruhan dari klasifikasi citra makanan. Proses ini dimulai dengan menggunakan citra makanan sebagai data masukan. Citra makanan ini akan mengalami tahap *preprocessing* dulu sebelum diambil representasi *Anti-Texton*-nya. Tahapan *preprocessing* diantaranya adalah melakukan *filtering* dengan Schmidt Filter Bank. Tahap selanjutnya adalah fase pembelajaran untuk mendapatkan *Texton Library* yang biasa dikenal dengan *bag-of-textons*. *Texton Library* didapatkan dengan melakukan kuantisasi vektor pada data citra training menggunakan *k-means clustering*. *Texton Library* akan digunakan untuk mendapatkan *Texton map* dari setiap citra dengan menghitung kedekatan antara

nilai piksel dan *Texton* id dari *Texton Library*. Penghitungan kedekatan ini menggunakan *Euclidian distance* sebagai metodenya. Setelah mendapatkan *Texton map* selanjutnya adalah fase ekstraksi representasi *Anti-Texton* dalam bentuk histogram. *Anti-Texton* didapatkan dengan melakukan beberapa langkah komputasi. Diagram alir sistem bisa dilihat pada

Gambar 3.1.

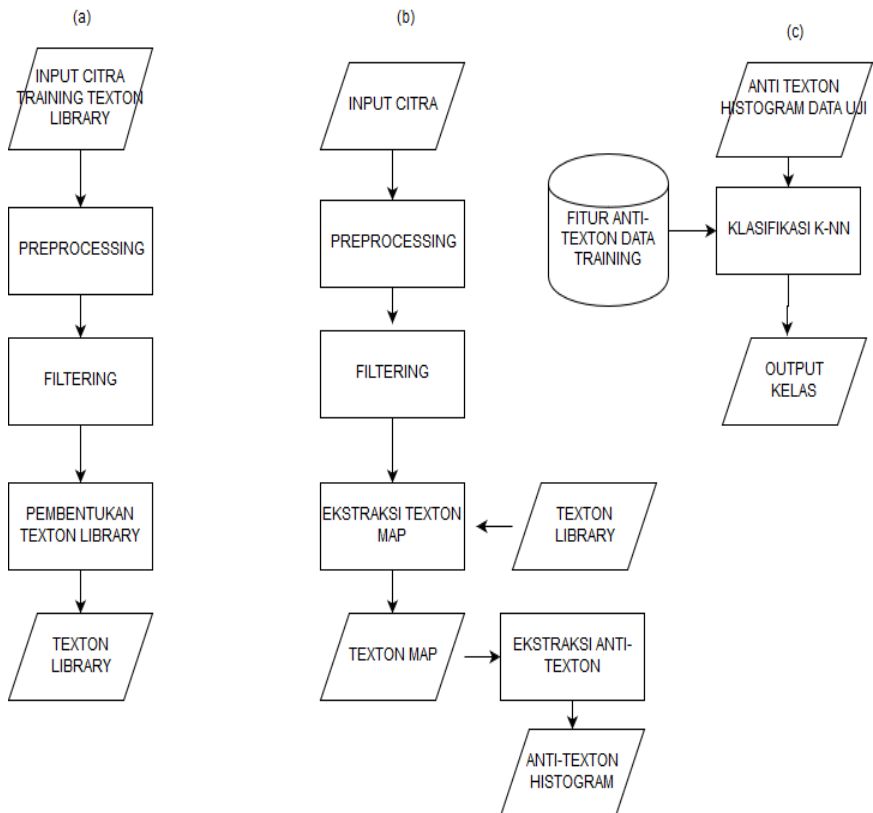
3.2 Perancangan Data

Pada sub-bab ini akan dijelaskan data yang digunakan di sistem klasifikasi citra makanan. Data yang digunakan untuk sistem klasifikasi citra makanan berasal dari dataset UNICT-FD1200, dataset yang terdiri dari 1200 citra yang diambil menggunakan *smartphone*. Data yang berupa citra makanan akan diproses untuk mendapatkan histogram yang merupakan representasi *Anti-Texton*. Berdasarkan histogram tersebut, sistem akan mengklasifikasi citra makanan kedalam kelas yang sudah ditentukan sebagai data keluaran.

Dataset untuk perancangan sistem klasifikasi citra makanan terdiri dari 465 citra, data ini akan di uji coba menggunakan *K-fold Cross validation* dimana $k=15$ yang artinya uji coba dilakukan sebanyak 15 kali. Setiap iterasi akan diambil 31 citra sebagai dataset dan sisanya sebagai data training untuk uji klasifikasi menggunakan *K-NN*. Sedangkan data training untuk pembentukan *Texton Library* berjumlah 80 citra, dimana setiap kelas terdiri dari 10 citra.

3.3 Perancangan Proses

Perancangan proses dilakukan untuk memberikan gambaran mengenai setiap proses yang terdapat pada sistem klasifikasi citra makanan lebih detail. Ada beberapa proses yang harus dilakukan yaitu: *Preprocessing*, *filtering*, *training texton library*, ekstraksi *texton map*, ekstraksi *anti-texton*, dan klasifikasi *K-NN*.



Gambar 3.1 Diagram Alir Klasifikasi Citra Makanan menggunakan Anti-Texton

3.3.1 Preprocessing

Sebelum mendapatkan ekstraksi *texton map*, citra masukan diubah dari RGB menjadi LAB *Colorspace* karena LAB lebih perseptual linear yang artinya perubahan nilai warna seharusnya menghasilkan perubahan pada kepentingan visual. Hal ini bertujuan agar mendapatkan hasil yang lebih baik mengingat informasi warna adalah hal yang penting dalam klasifikasi makanan. **Gambar 3.3** adalah contoh citra asli yang menjadi data masukan dan **Gambar 3.4** menunjukkan citra yang sudah diubah menjadi LAB *colorspace*. Diagram alir *preprocessing* secara umum bisa dilihat pada **Gambar 3.2**.

Tahap pertama pada *preprocessing* ini adalah mengonversi RGB ke XYZ. Konversi ini menggunakan rumus pada persamaan 3.1.

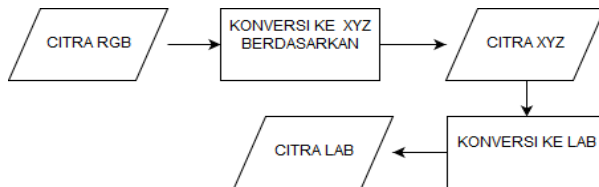
$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad (3.1)$$

Setelah diubah menjadi XYZ langkah selanjutnya adalah mengonversinya menjadi LAB. Konversi ini menggunakan rumus pada persamaan 3.2, 3.3, dan 3.4.

$$L^* = 116 f \left(\frac{Y}{Y_n} \right) - 16 \quad (3.2)$$

$$a^* = 500 \left(f \left(\frac{X}{X_n} \right) - f \left(\frac{Y}{Y_n} \right) \right) \quad (3.3)$$

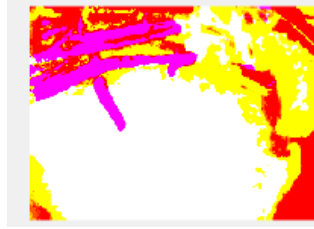
$$b^* = 200 \left(f \left(\frac{Y}{Y_n} \right) - f \left(\frac{Z}{Z_n} \right) \right) \quad (3.4)$$



Gambar 3.2 Diagram Alir Preprocessing




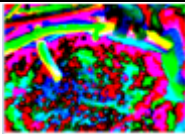
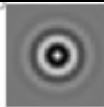
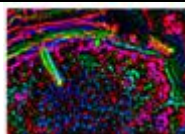
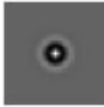
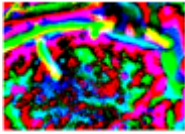

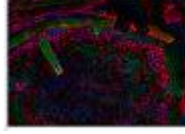
Gambar 3.3 Citra RGB (asli)

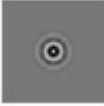
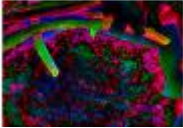
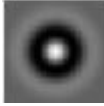
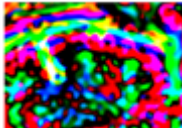
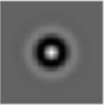
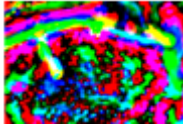
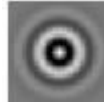
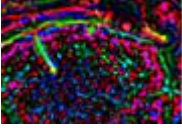

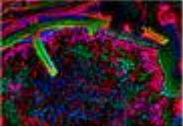
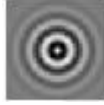
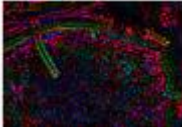
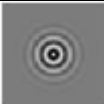
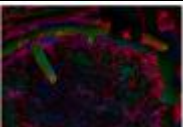

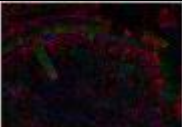
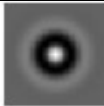
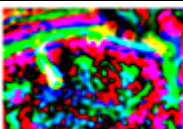
Gambar 3.4 Citra LAB
Colorspace

3.3.2 Filtering

Tahapan selanjutnya adalah melakukan *filtering* menggunakan *Schmidt Filter Bank*. Proses *filtering* ini dijabarkan pada sub-bab 2.3 tentang *Schmidt Filter Bank*. Setiap *filter response* dinormalisasi terlebih dahulu menggunakan metode z-index. Dari proses ini akan didapatkan 13 *filter response* yang nantinya akan diolah lagi untuk mendapatkan *Texton Library* dan *Texton map*. Contoh *filter response* bisa dilihat pada **Tabel 3.1**.

Tabel 3.1 *Filter Response*

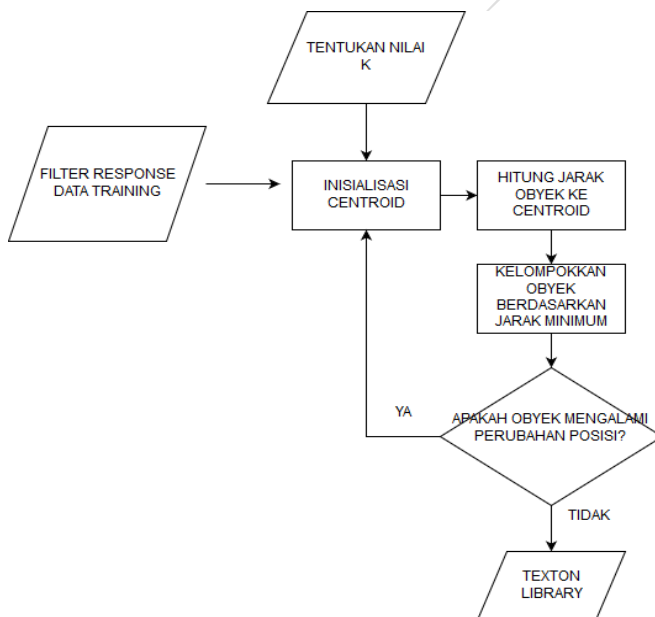
Filter (σ, τ)	Filter Response	Filter (σ, τ)	Filter Response
 (2,1)		 (8,2)	
 (4,1)		 (8,3)	

Filter (σ, τ)	Filter Response	Filter (σ, τ)	Filter Response
 (4,2)		 (10,1)	
 (6,1)		 (10,2)	
 (6,2)		 (10,3)	
 (6,3)		 (10,4)	
 (8,1)			

3.3.3 Pembentukan *Texton Library*

Pada tahap ini akan dijelaskan langkah dalam membentuk *texton library*. **Gambar 3.5** menunjukkan diagram alir pembentukan *texton library*. Setiap citra dari data training akan melalui tahap preprocessing dan *filtering* terlebih dahulu. Hasil *filtering* dari semua citra kemudian dikumpulkan menjadi satu matriks besar. Tidak semua nilai piksel dari data training masuk pada matriks tersebut. Hal ini bertujuan agar resource yang digunakan untuk proses komputasi tidak menghabiskan terlalu banyak. Pengambilan piksel dilakukan secara acak menggunakan metode random permutasi.

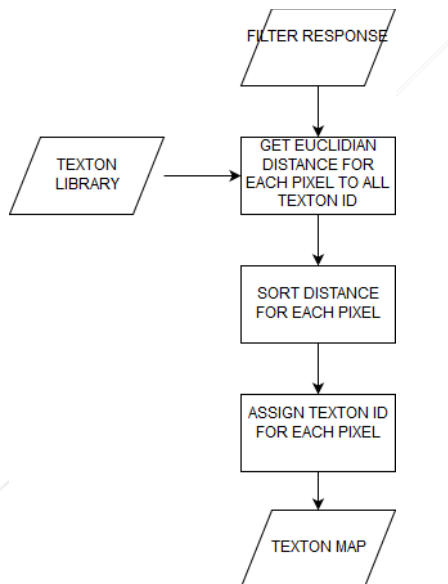
Selanjutnya matriks tersebut akan masuk dalam proses kuantisasi vektor menggunakan *k-means clustering* untuk mendapatkan *texton library*. Nilai k dari *k-means clustering* menunjukkan besar ukuran dari *texton library* yang dibentuk. Semakin besar k maka semakin lama pula membentuk *texton library*. Langkah pertama proses *clustering* setelah menentukan nilai k adalah menginisialisasi *centroid* sesuai jumlah k . Kemudian setiap obyek dihitung jaraknya ke *centroid* yang tadi sudah ditentukan. Selanjutnya obyek-obyek tersebut dikelompokkan berdasarkan jarak minimum. Jika obyek mengalami perubahan posisi maka proses kembali pada inisialisasi *centroid* hingga obyek tidak mengalami perubahan posisi. Hasil pada proses *clustering* ini disebut *texton library*.



Gambar 3.5 Diagram Alir Training Texton Library

3.3.4 Ekstraksi *Texton map*

Setelah mendapatkan *texton library* langkah selanjutnya adalah melakukan ekstraksi *texton map* dari *filter response* milik input data citra. Semua *filter response* yang menjadi data masukan akan dijadikan kedalam satu matriks. Matriks tersebut kemudian akan dihitung nilai *euclidian distance*-nya dengan *texton library*. Nilai *euclidian distance* akan dijadikan sebagai acuan kedekatan setiap piksel dengan id *texton* yang ada di *texton library*. Proses ekstraksi *texton map* bisa dilihat pada **Gambar 3.6**



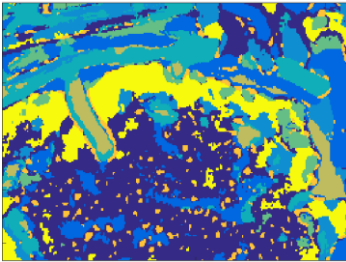
Gambar 3.6 Diagram Alir Ekstraksi *Texton map*

3.3.5 Ekstraksi *Anti-Texton*

Tahap selanjutnya adalah mendapatkan *Anti-Texton*. *Texton library* harus sudah terbentuk sebelum membuat *Anti Texton*. Hal ini disebabkan karena *Anti-Texton* adalah jarak spasial dari *texton*

map. Sedangkan *texton map* sendiri harus didapat menggunakan *texton library*.

Langkah pertama data citra yang sudah melewati fase *preprocessing* dan *filtering* akan dihitung nilai pikselnya. Cara menghitung nilai piksel ini adalah dengan mengukur kedekatan nilai piksel dengan *texton-texton* yang ada pada *texton library* menggunakan *texton library*. *texton* id yang paling dekat dengan nilai piksel tersebut akan menjadi *texton* pada *texton map* dari citra tersebut. Metode penghitungan yang digunakan adalah *Euclidian distance*. Setiap piksel pada *texton map* menyimpan nilai berupa *texton ID* yang tersimpan pada *texton library*. Contoh *texton map* bisa dilihat pada **Gambar 3.7**.



Gambar 3.7 *Texton Map* dari citra masukan



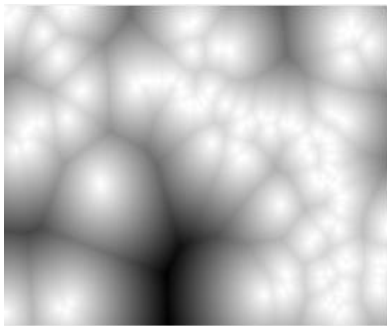
Gambar 3.8 *Binary Map* dari citra masukan

Untuk setiap *texton* dengan ID i ($i=1, \dots, N$) akan dihasilkan binary map sebanyak N . *Binary map* B_i didapat dengan cara memberi nilai 1 untuk piksel yang bernilai *texton ID* = i dan nilai 0 untuk selain *texton ID* tersebut. Contoh *binary map* bisa dilihat pada **Gambar 3.8**.

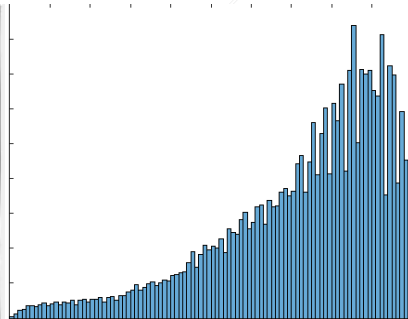
Setiap *binary map* B_i akan dihitung *distance transform*-nya. Hasilnya adalah *saliency map*, dimana titik yang dexton dengan *texton ID* i kurang menonjol daripada titik-titik yang lain. Setiap *saliency map* akan dinormalisasi dengan cara membaginya nilai tertinggi dari map tersebut. Map yang sudah dinormalisasi

dinotasikan sebagai D_i . Setiap D_i kemudian diinvert yang akan menghasilkan $map E_i$. Pada tahap inilah jarak spasial antar *texton* dihitung. Contoh $map E_i$ bisa dilihat pada **Gambar 3.9**

Setelah mendapatkan E_i langkah selanjutnya adalah mengambil nilai histogram pada E_i . **Gambar 3.10** adalah Histogram yang merupakan representasi dari *Anti-Texton*. Histogram dari E_i ($i=1, \dots, N$) kemudian di rata-rata untuk mendapatkan histogram keseluruhan dari *Anti-Texton* pada gambar tersebut. Langkah ekstraksi *Anti-Texton* bisa dilihat pada **Gambar 3.12**.



Gambar 3.9 Contoh $map E_i$

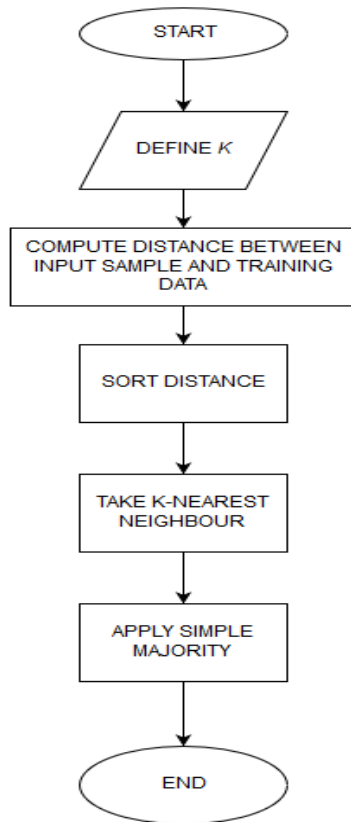


Gambar 3.10 Histogram *Anti-Texton*

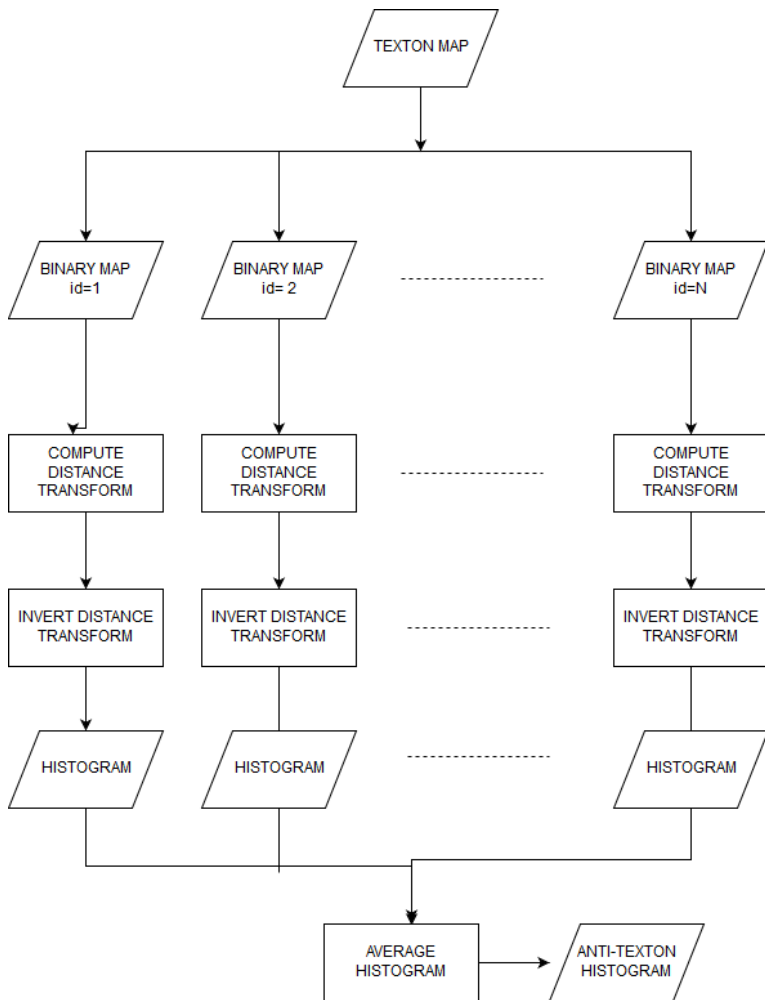
3.3.6 Klasifikasi K-NN

Proses klasifikasi merupakan tahapan akhir untuk menentukan kelas dari citra input makanan. Metode klasifikasi yang digunakan adalah K-NN dan *Chi-Square Distance*. histogram *Anti Texton* yang didapat dari proses ekstraksi *Anti Texton*, akan dibandingkan dengan histogram *Anti Texton* milik data training. Proses perbandingan menggunakan *Chi-Square Distance* dan dilakukan proses *sorting* berdasarkan nilai *Chi-Square Distance* yang paling kecil menuju paling besar. Pengklasifikasian berdasarkan data training yang memiliki *Chi-Square Distance*

terkecil dari data masukkan. Diagram alir K-NN secara umum bisa dilihat pada **Gambar 3.11**.



Gambar 3.11 Diagram Alir K-NN



Gambar 3.12 Diagram Alir Ekstraksi *Anti-Texton*

BAB IV IMPLEMENTASI

Pada bab ini diuraikan mengenai implementasi perangkat lunak dari rancangan metode yang telah dibahas pada Bab III meliputi kode program dalam perangkat lunak. Selain itu, implementasi dari tiap proses, parameter masukan, keluaran, dan beberapa keterangan yang berhubungan dengan program juga dijelaskan.

3.4 Lingkungan Implementasi

Objek citra yang akan diolah pada implementasi tugas akhir ini adalah citra makanan dari dataset yang diambil dari UNICT-FD1200. Dataset tersebut diambil menggunakan kamera Iphone 3s dan memiliki format gambar berformat JPG.

Lingkungan implementasi pada tugas akhir ini menggunakan sebuah *personal computer* (PC). Detail perangkat dan spesifikasi dapat dilihat pada **Tabel 0.1**.

Tabel 0.1 Spesifikasi Perangkat Keras yang Digunakan

Perangkat	Spesifikasi
Merk dan Tipe	Lenovo 80E4
Prosesor	Core i5-5200U @ 2.20GHz
RAM	6 GB
HDD	500 GB
OS	Microsoft Windows 10
Software	MATLAB R2015a

3.5 Implementasi Tahap *Preprocessing*

Tahap *preprocessing* pada tugas akhir ini berupa pengubahan tipe *colorspace* gambar dari RGB menjadi LAB. Implementasi pengubahan tipe *colorspace* bisa dilihat pada **Kode Sumber 0.1**.

```
citra =RGB2Lab(citra);
```

Kode Sumber 0.1 Implementasi perubahan tipe *colorspace*

3.6 Implementasi *Filtering*

Filtering digunakan untuk mendapat filter respon yang selanjutnya akan digunakan sebagai data input untuk mendapat data *texton map*. Implementasi *filtering* ditunjukkan pada **Kode Sumber 0.2**.

Baris 1 dan 2 adalah inisiasi *Schmidt Filter Bank*. Proses *filtering* ditunjukkan pada line 6. Setiap *filter response* sebelumnya dinormalisasi dengan metode z-index yang ditunjukkan pada baris 9 dan 10.

1	<code>a=makeSfilters();</code>
2	<code>a=double(a);</code>
3	<code>aaa=zeros(192*272, numFeatures);</code>
4	<code>for f=1:13</code>
5	<code>for i=1:3</code>
6	<code>testing(:, :, i) =conv2(citra(:, :, i), a(:, :, f), 'valid');</code>
7	<code>testing2 = reshape(testing(:, :, i), [192*272, 1]);</code>
8	<code> pikselFeatureVec = testing2;</code>
9	<code> pikselFeatureVec = pikselFeatureVec - mean(pikselFeatureVec);</code>
10	<code>pikselFeatureVec=pikselFeatureVec/ std(pikselFeatureVec);</code>
11	<code> aaa(:, (f-1)*3+i) = pikselFeatureVec; %hasil semua filter respon per gambar</code>

Kode Sumber 0.2 Implementasi *Filtering*

3.7 Implementasi Pembentukan *Texton Library*

Tahap *training texton library* adalah proses untuk mendapatkan *texton library*. semua *filter response* dari data

training dijadikan kedalam satu matriks besar yang selanjutnya akan dikuantisasi vektor menggunakan *k-means clustering*. Implementasi pada *training texton library* ditunjukkan pada **kode sumber 4.3**.

Tidak semua nilai piksel dari data training masuk pada matriks tersebut. Hal ini bertujuan agar resource yang digunakan untuk proses komputasi tidak menghabiskan terlalu banyak. Pengambilan piksel dilakukan secara acak menggunakan metode random permutasi. Baris 1 menunjukkan proses *clustering* untuk mendapatkan *Texton Library*. Variabel *X* adalah matriks besar yang terdiri dari kumpulan *filter response* data *training*. Variabel *SizeTL* menentukan ukuran *texton library* yang diinginkan. Setelah proses *clustering* selesai *texton library* disimpan untuk digunakan pada proses ekstraksi *texton map*.

1	<code>[def,TextonLibrary]=kmeans(X,SizeTL, 'EmptyAction','drop');</code>
2	<code>save TextonLibrary.mat TextonLibrary</code>

Kode Sumber 0.3 Implementasi Training Texton Library

3.8 Implementasi Ekstraksi *Texton map*

Tahap ekstraksi *texton map* adalah proses untuk mendapatkan *texton map* dari citra masukan. *Filter reponse* dari citra masukan akan dikuantisasi vektor dengan *texton library* menggunakan *euclidian distance*. Implementasi ekstraksi *texton map* bisa dilihat pada **Kode Sumber 0.4**.

Baris 12 menunjukkan proses kuantisasi vektor antara *filter response* dengan *texton library* menggunakan *euclidian distance*. Baris 13 menunjukkan setiap nilai pada piksel mendapatkan *texton ID* berdasarkan jarak terdekat piksel dengan *texton* yang ada pada *texton library*. Output dari proses ini adalah *texton map* dimana setiap nilai pada *map* tersebut berisi ID *texton*. Variabel *textonMap* adalah *texton map* dari citra masukan.

1	<code>load TextonLibrary375.mat;</code>
2	<code>TextonLibrary=rY;</code>
3	<code>citra =RGB2Lab(citra);</code>
4	<code>citra=double(citra);</code>
5	<code>TextonLibrary=rY;</code>
6	<code>numFeatures = 39;</code>
7	<code>a=makeSfilters();</code>
8	<code>a=double(a);</code>
9	<code>textonMap=zeros(192,272);</code>
10	<code>FilterResponse=zeros(192*272, numFeatures);</code>
11	<code>FilterResponse=SchmidtFilter(citra);</code>
12	<code>temp=pdist2(aaa,TextonLibrary);</code>
13	<code>[aaa, index]=min(temp');</code>
14	<code>textonMap=reshape(index,192,272);</code>
15	<code>imagesc(textonMap);</code>

Kode Sumber 0.4 Implementasi Ekstraksi *Texton Map*

3.9 Implementasi Ekstraksi *Anti-Texton*

Tahap ekstraksi *anti-texton* adalah proses untuk mendapatkan representasi *anti-texton* berupa histogram dari citra masukan. Data masukan dari tahap ini adalah *texton map* dari citra. Implementasi ekstraksi *anti-texton* bisa dilihat pada **Kode Sumber 0.5**

Variabel *textonMap* merupakan *texton map* dari citra masukan. *Texton map* selanjutnya dirubah menjadi beberapa *binary map*. Untuk setiap *texton* dengan *ID* ($i=1,\dots,N$) akan dihasilkan *binary map* sebanyak N . *Binary map* B_i didapat dengan cara memberi nilai 1 untuk piksel yang bernilai *texton ID* = i dan nilai 0 untuk selain *texton ID* tersebut seperti yang ditunjukkan pada baris 7 hingga 15. Variabel biner adalah *binary map* dari citra. Setiap *binary map* kemudian dihitung *distance transform* dan dinormalisasi. Proses ini ditunjukkan pada baris 17 hingga 19.

Distance transform yang sudah didapatkan kemudian di-*invert* untuk menghitung jarak spasial antar *texton*. Proses ini ditunjukkan pada baris 21. Hasilnya adalah histogram anti-*texton* untuk satu id. Setelah semua id *texton* memiliki histogram anti-*texton* maka akan dilakukan penghitungan rata-rata untuk semua histogram tersebut seperti yang ditunjukkan pada baris 32. Histogram rata-rata adalah representasi anti-*texton* dari citra masukan. Variabel *Anti_Texton_Histogram* adalah output dari proses ini.

3.10 Implementasi Proses Klasifikasi

Tahap klasifikasi pada tugas akhir ini menggunakan K-NN. Setelah melakukan ekstrasi *Anti-Texton* pada data masukan, histogram *Anti-Texton* tersebut akan dibandingkan dengan histogram *Anti-Texton* milik data training. Perbandingan dilakukan dengan menghitung *chi-square distance*. Hasil dari *chi-square distance* lalu diurutkan dari yang terkecil. Semakin kecil nilai *chi-square* berarti semakin mirip dengan data masukan. **Kode Sumber 0.6** adalah implementasi dari K-NN dimana nilai k adalah 1.

3.11 Implementasi K-Fold Cross validation

K-Fold *Cross validation* adalah metode uji coba yang diterapkan pada tugas akhir ini. Dataset yang digunakan dibagi k , dataset yang sudah dibagi kemudian diambil beberapa untuk menjadi data set, dan sisanya menjadi data training. Implementasi K-Fold Cross Validation bisa dilihat pada **Kode Sumber 0.7**.

Variabel *data_all* adalah dataset yang nantinya akan dibagi menjadi data testing dan data training untuk pengujian nanti. Baris 10 menunjukkan pengujian akan dilakukan sebanyak k kali. Baris 13 hingga 16 adalah proses pembagian data testing dan data training untuk pengujian pada iterasi tersebut. Data testing yang digunakan untuk setiap iterasi tidak sama dan diambil secara acak.

2	size_bin=100;
3	H_sumall=zeros(1,size_bin);
4	ping=zeros(1,size_bin);
5	for textonID=1:sizeTL
6	nol=0;
7	for k=1:192 for l=1:272
9	if textonMap(k,l)==textonID
10	biner(k,l)=1;
11	nol=nol+1;
12	else biner(k,l)=0;
13	End End
16	if(nol~=0)
17	DT=bwdist(biner);
18	max_value=max(DT(:));
19	DT2=DT/max_value;
21	Ei=1-DT2;
22	hn = histogram(Ei,size_bin);
23	ping=hn.Values;
24	else
25	Ei=biner;
26	for g=1:size_bin
27	ping(g)=0;
28	end
30	H_sumall=H_sumall+ping;
32	Anti Texton Histogram=H_sumall/sizeTL;

Kode Sumber 0.5 Implementasi Ekstraksi Anti-Texton

1	for j=1:max
2	input= dataset(j+20).histogram;
3	for i=1:51
4	distance=dist_chisquare (input,data_training(i).histogram);
5	mean_distance=mean(distance);
6	result(i).nilai=mean distance;
7	result(i).label=data_training(i).label;
8	result(i).filename=baca2{1}{i};
9	end
10	[ss,sx]=sort([result.nilai]);
11	ss=result(sx);
12	total_result(j).filename = baca{1}{j};
13	total_result(j).actual = dataset(j).label;
14	total_result(j).predicted=ss(1).label;
15	total_result(j).filename2=ss(1).filename;
16	total_result(j).nilai=ss(1).nilai;
17	end

Kode Sumber 0.6 implementasi dari K-NN dengan k=1

1	<code>sizey=size(data_all);</code>
2	<code>nomor=randperm(dataz);</code>
3	<code>mykfold = 15; %tentukan k</code>
4	<code>k=dataz/mykfold;</code>
5	<code>total_iterasi= mykfold;</code>
6	<code>hasil_knn=struct();</code>
7	<code>keepnomor=[];</code>
8	<code>data_testing=struct();</code>
9	<code>addition=0;</code>
10	<code>for iterasi=1:total_iterasi</code>
12	<code>for test_number=1:k</code>
13	<code>ambil=nomor(test_number+addition);</code>
14	<code>keepnomor(test_number)=ambil;</code>
15	<code>...%pengambilan data_testing</code>
16	<code>End</code>
17	<code>addition=addition+k;</code>
18	<code>data_training=struct();</code>
19	<code>train_number=1;</code>
20	<code>for all_number=1:dataz</code>
21	<code>if ismember(all_number,keepnomor) == 0</code>
22	<code>...%pengambilan data_training</code>
23	<code>train_number=train_number+1;</code>
24	<code>end</code>
25	<code>end</code>
26	<code>... %klasifikasi menggunakan K-NN</code>
27	<code>rata2=sum(cat(1,hasil_knn.akurasi))</code> <code>/total_iterasi;</code>
28	<code>disp(rata2);</code>

Kode Sumber 0.7 K-fold Cross validation

BAB V

UJI COBA DAN EVALUASI

Dalam bab ini dibahas mengenai hasil uji coba sistem yang telah dirancang dan dibuat. Uji coba dilakukan untuk mengetahui kinerja sistem dengan lingkungan uji coba yang telah ditentukan.

3.12 Lingkungan Uji Coba

Lingkungan uji coba pada tugas akhir ini adalah sebuah *personal computer* (PC). Spesifikasi PC dari sisi perangkat keras adalah memiliki prosesor Intel Core i3 3240 dengan kecepatan 3,4 GHz dan memori untuk proses sebesar 4,00 GB. PC yang digunakan memiliki sistem operasi Windows 8.

Pada sisi perangkat lunak, uji coba pada tugas akhir ini dilakukan dengan menggunakan *software* MATLAB R2015a. Penggunaan MATLAB didukung dengan dua *toolbox* utama yaitu *statistical and machine learning toolbox* dan *image processing toolbox*. Dokumentasi hasil uji coba dilakukan dengan menggunakan *Microsoft Paint*.

3.13 Data Uji Coba

Data uji coba yang digunakan sebagai masukan adalah 465 citra makanan berukuran 172 dan 192 piksel yang dari UNICT-FD1200. Contoh citra ditunjukkan pada **Gambar 5.1**. Dari citra makanan dilakukan proses perubahan tipe *colorspace* menjadi LAB. Setelah diubah menjadi LAB langkah selanjutnya adalah melakukan *filtering* menggunakan *Schmidt Filter Bank* untuk mendapatkan 13 *filter response*. *Filter response* ini kemudian akan diolah lebih lanjut untuk mendapatkan *Texton Library* dan *Anti-Texton*. Hasil ekstraksi fitur *Anti-Texton* bisa dilihat pada Lampiran A.



Gambar 0.1 Contoh Citra dari UNICT-FD1200

3.14 Skenario Uji Coba

Uji coba dilakukan untuk mengetahui nilai-nilai parameter yang tepat untuk digunakan pada masing-masing proses. Nilai parameter yang tepat penting untuk diketahui karena penggunaan parameter yang tepat akan memberikan hasil yang terbaik pada keluaran tiap proses. Metode pengujian menggunakan k-fold *cross validation* dengan nilai $k = 15$.

Skenario pengujian terdiri dari tiga macam yaitu:

1. Uji coba penentuan ukuran *Texton Library* dengan menggunakan parameter k pada k-means *clustering*.
2. Uji coba penentuan parameter k pada K-NN.
3. Uji coba akurasi setiap kelas.

3.15 Uji Coba Penentuan Ukuran *Texton Library* dengan Menggunakan Parameter k pada k-means *clustering*

Uji coba penentuan ukuran *Texton Library* digunakan untuk mengetahui nilai optimal pada fase ekstraksi fitur *Anti-Texton*. *Texton Library* diperlukan untuk mendapatkan *texton map* dari citra masukan. Pada tugas akhir ini digunakan 80 citra dimana setiap kelas terdiri dari 10 citra untuk membentuk *texton library*. Pada uji coba ini ukuran *texton* yang dipakai adalah 50,150, dan

375. Metode klasifikasi yang dipakai adalah K-NN dengan nilai $k=1$. Hasil uji coba bisa dilihat pada **Tabel 0.2**. Berdasarkan tabel tersebut, *texton library* dengan ukuran 375 memiliki rata-rata akurasi paling tinggi yaitu sebesar 60.2%. Sedangkan *texton library* dengan ukuran 50 dan 150 masing-masing mendapatkan nilai rata-rata akurasi sebesar 56,9% dan 57,8%.

3.16 Uji Coba Penentuan k pada K-NN

Uji coba penentuan ukuran k pada K-NN digunakan untuk mengetahui nilai k yang optimal pada fase klasifikasi. Pada uji coba ini k yang dipakai menggunakan nilai 1, 3, dan 5. Sedangkan *texton library* yang digunakan mempunyai ukuran sebesar 375.

Tabel 0.3 menunjukkan hasil uji coba penentuan k pada K-NN. Pada tabel tersebut bisa kita lihat bahwa akurasi terbesar didapatkan saat $k=1$.

3.17 Evaluasi

Dari hasil uji coba yang dilakukan sebelumnya, didapatkan bahwa akurasi tertinggi jatuh pada uji coba klasifikasi makanan yang menggunakan *texton library* berukuran 375. Metode klasifikasi yang paling optimal adalah 1-NN. Hasil uji coba menunjukkan akurasi sebesar 60,2%. Sedangkan uji coba yang lain dimana ukuran *texton library* 50 dan 150 hanya menunjukkan akurasi antara 56-57% saja. Hal ini menunjukkan bahwa ukuran *texton library* dan K-NN memiliki pengaruh pada akurasi klasifikasi.

Pada uji coba *texton library* dengan ukuran TL=50 diperoleh hasil rata-rata yakni 56,9% dengan akurasi tertinggi didapatkan saat melakukan uji coba ke-3 dan 13 dimana nilai akurasi mencapai 74,1%. Sedangkan akurasi terendah berada pada uji coba ke-14 dengan nilai akurasi 45,1%.

Selanjutnya untuk TL=175 diperoleh hasil rata-rata yakni 57,8% dengan akurasi tertinggi didapatkan saat melakukan uji coba ke-3 dan 13 dimana nilai akurasi mencapai 74,1%. Sedangkan

akurasi terendah berada pada uji coba ke-14 dengan nilai akurasi 45,1%.

Terakhir untuk TL=375 diperoleh hasil rata-rata yakni 60,2% dengan akurasi tertinggi didapatkan saat melakukan uji coba ke-7 dan 12 dimana nilai akurasi mencapai 70,9%. Sedangkan akurasi terendah berada pada uji coba ke-14 dengan nilai akurasi 41,9%.

Dari ketiga ukuran *texton library* nilai rata-rata cenderung naik sejajar dengan ukuran *texton library*. Hal ini dikarenakan semakin banyak ukuran *texton library*, semakin banyak *clustering* yang dilakukan membuat *texton* pada tiap citra lebih bervariasi.

Pada uji coba penentuan K-NN didapatkan akurasi tertinggi di 1-NN dimana rata-rata akurasinya mencapai 60,2%. Sedangkan rata-rata akurasi dari 3-NN dan 5-NN hanya mencapai masing-masing 56,1% dan 57,6%. Hal ini menunjukkan bahwa nilai optimal untuk klasifikasi ini menggunakan 1-NN.

Uji coba terakhir adalah mencari akurasi untuk setiap kelas makanan. Berdasarkan **Tabel 0.1** diketahui bahwa kelas *breakfast* memiliki akurasi tertinggi sebesar 65% dan *main_course* mendapat akurasi terendah sebesar 48%. Rendahnya akurasi ini karena beberapa data *main course* memiliki kemiripan dengan data dari kelas *second course*. Penyebab lain dataset difoto dalam keadaan pencahayaan yang berbeda-beda dan hal itu mempengaruhi proses klasifikasi.

Berdasarkan hasil dua uji coba sebelumnya didapatkan bahwa rata-rata akurasi tertinggi berasal dari hasil uji coba dengan TL=375 dan klasifikasi 1-NN. Selanjutnya, dibuat *confusion matrix* untuk mengetahui akurasi setiap kelas makanan. **Tabel 0.1** menunjukkan *confusion matrix* dari hasil uji coba.

Berdasarkan tabel tersebut, kelas *breakfast* memiliki akurasi tertinggi yaitu 65% sedangkan kelas *main course* mendapatkan akurasi terendah yaitu sebesar 48,33%. Lebih lanjut, kelas *main course* dan *second course* mendapat akurasi dibawah 60% yaitu 48,33% dan 53,33%.

Tabel 0.1 Confusion Matrix 8 class

Actual Class	Predicted Class							
	Appetizer	Breakfast	Dessert	Fruit	Main_Course	Second_Course	Side_Dish	Single_Course
Appetizer	62,90	3,33	5,45	9,43	10,00	1,67	3,64	3,33
Breakfast	4,84	65,00	9,09	11,32	10,00	10,00	5,45	3,33
Dessert	4,84	5,00	60,00	0,00	1,67	6,67	3,64	10,00
Fruit	6,45	0,00	0,00	62,26	1,67	3,33	1,82	5,00
Main_Course	9,68	6,67	1,82	5,66	48,33	11,67	10,91	6,67
Second_Course	4,84	11,67	10,91	0,00	13,33	53,33	7,27	10,00
Side_Dish	3,23	3,33	1,82	3,77	8,33	6,67	61,82	0,00
Single_Course	3,23	5,00	10,91	7,55	6,67	6,67	5,45	61,67

Tabel 0.2 Akurasi berdasarkan *texton library*

Iterasi ke-	TL=50(%)	TL=175(%)	TL=375(%)
1	58,0	51,6	61,2
2	54,8	64,5	41,9
3	51,6	74,1	51,6
4	64,5	54,8	64,5
5	58,0	61,2	54,8
6	38,7	51,6	61,2
7	74,1	54,8	70,9

Iterasi ke-	TL=50(%)	TL=175(%)	TL=375(%)
8	58,0	61,2	61,2
9	58,0	54,8	58
10	64,5	51,6	70,9
11	51,6	51,6	67,7
12	51,6	61,2	70,9
13	54,8	74,1	41,9
14	54,8	45,1	58
15	61,2	54,8	67,7
rata-rata	56,9	57,8	60,2

Tabel 0.3 Akurasi berdasarkan ukuran K-NN

Iterasi ke-	K=1(%)	K=3(%)	K=5(%)
1	61,2	51,6	58
2	41,9	48,3	64,5
3	51,6	61,2	58
4	64,5	77,4	35,4
5	54,8	51,6	64,5
6	61,2	38,7	38,7
7	70,9	51,6	61,2
8	61,2	61,2	64,5
9	58	64,5	61,2
10	70,9	58	67,7
11	67,7	61,2	61,2
12	70,9	51,6	54,8
13	41,9	48,3	54,8
14	58	54,8	51,6
15	67,7	61,2	67,7
rata-rata	60,2	56,1	57,6

BAB VI

KESIMPULAN DAN SARAN

Bab ini membahas mengenai kesimpulan yang dapat diambil dari hasil uji coba yang telah dilakukan sebagai jawaban dari rumusan masalah. Selain itu juga terdapat saran yang ditujukan untuk pengembangan penelitian lebih lanjut.

4.1 Kesimpulan

Kesimpulan yang diperoleh dari uji coba dan evaluasi adalah sebagai berikut:

1. Sistem klasifikasi citra makanan dengan metode representasi *Anti-Texton* dengan *kombinasi K-Means Clustering* dan *K-NN Classification* mendapat akurasi tertinggi sebesar 77,4%.
2. Rata-rata akurasi tertinggi berdasarkan ukuran *texton library* adalah 60,2% dimana ukuran *texton library*-nya adalah 375.
3. Rata-rata akurasi tertinggi berdasarkan K-NN adalah 60,2% pada 1-NN.
4. Rata-rata akurasi cenderung naik jika nilai dari ukuran *texton library* semakin besar dimana ukuran *texton library* 50 memiliki rata-rata akurasi 56,9% sedangkan ukuran *texton library* 375 rata-rata memiliki akurasi 60,2%.
5. Kelas *breakfast* mencapai akurasi tertinggi sebesar 65% sedangkan kelas *main course* mendapat akurasi terendah sebesar 48%.



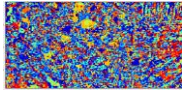
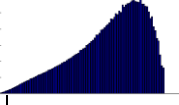


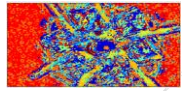



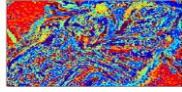
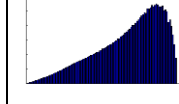


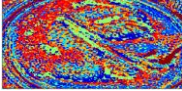
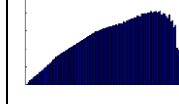

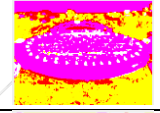
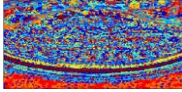
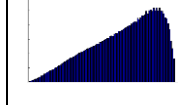

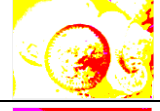
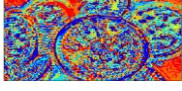
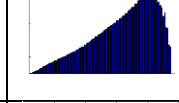


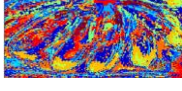
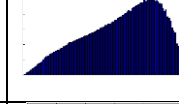


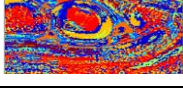
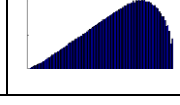
4.2 Saran



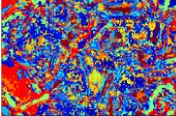
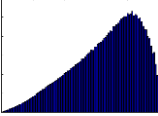


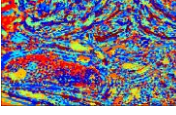
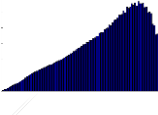

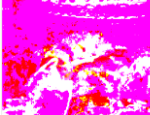
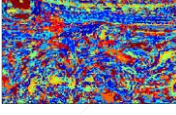
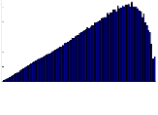


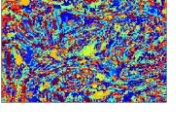
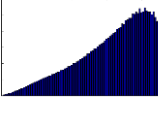

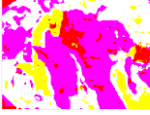
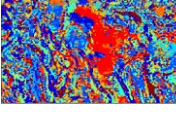
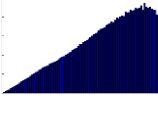


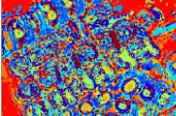
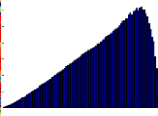

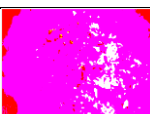
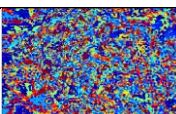
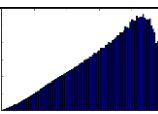

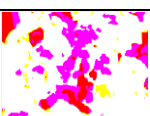
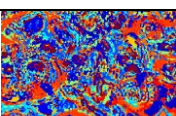
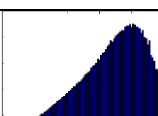
Saran yang dapat diberikan dalam pengujian klasifikasi citra makanan adalah sebagai berikut:

1. Melakukan perbandingan jumlah fitur yang diambil dari setiap data training yang akan digunakan untuk membentuk *texton library* untuk mengetahui hasil klasifikasi yang optimal
2. Mencari metode lain untuk melakukan kuantisasi vektor saat membentuk *Texton Library*. Metode *k-means Clustering* memerlukan komputasi yang tinggi. Meskipun hal ini sudah bisa diatasi dengan cara mengambil jumlah fitur tertentu secara acak dikhawatirkan *Texton Library* yang dihasilkan kurang mewakili.
3. Menambahkan langkah untuk *cropping* pada bagian sentral citra masukan di tahap *preprocessing* untuk mendapatkan hasil yang lebih optimal.

LAMPIRAN

A. Hasil Proses Ekstraksi Anti-Texton

Citra Asli	Citra LAB	<i>Texton Map</i>	Histogram <i>Anti-Texton</i>
			
			
			
			
			
			
			
			

Citra Asli	Citra LAB	<i>Texton Map</i>	Histogram <i>Anti-Texton</i>
			
			
			
			
			
			
			
			

B. Hasil Uji Coba Klasifikasi TL=375 dengan 1-NN

iterasi 1			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0061_0001.jpg'	'Second_Course'	'Second_Course'	19,42908
'_Sample_0298_0001.jpg'	'Dessert'	'Second_Course'	21,2907
'_Sample_0112_0002.jpg'	'Appetizer'	'Appetizer'	47,68915
'_Sample_0254_0002.jpg'	'Main_Course'	'Main_Course'	17,5749
'_Sample_0455_0005.jpg'	'Side_Dish'	'Side_Dish'	20,1432
'_Sample_0085_0001.jpg'	'Appetizer'	'Main_Course'	21,9323
'_Sample_0135_0001.jpg'	'Single_Course'	'Main_Course'	24,839
'_Sample_0122_0001.jpg'	'Side_Dish'	'Side_Dish'	9,63605
'_Sample_0758_0003.jpg'	'Dessert'	'Dessert'	35,48936
'_Sample_0107_0002.jpg'	'Second_Course'	'Second_Course'	28,07618
'_Sample_0093_0001.jpg'	'Appetizer'	'Breakfast'	15,87256
'_Sample_0039_0001.jpg'	'Main_Course'	'Second_Course'	14,00626
'_Sample_0151_0001.jpg'	'Single_Course'	'Dessert'	211,9698

iterasi 1			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0233_0001.jpg'	'Side_Dish'	'Side_Dish'	25,80563
'_Sample_0005_0001.jpg'	'Second_Course'	'Second_Course'	24,19742
'_Sample_0068_0001.jpg'	'Appetizer'	'Dessert'	28,39646
'_Sample_0051_0001.jpg'	'Second_Course'	'Second_Course'	15,82985
'_Sample_0042_0001.jpg'	'Main_Course'	'Main_Course'	20,25627
'_Sample_0940_0001.jpg'	'Fruit'	'Fruit'	12,70977
'_Sample_0423_0003.jpg'	'Breakfast'	'Breakfast'	13,143
'_Sample_0873_0001.jpg'	'Fruit'	'Breakfast'	24,37304
'_Sample_0010_0001.jpg'	'Dessert'	'Main_Course'	101,7143
'_Sample_0073_0001.jpg'	'Main_Course'	'Appetizer'	28,20281
'_Sample_0182_0004.jpg'	'Appetizer'	'Appetizer'	7,860741
'_Sample_0113_0001.jpg'	'Single_Course'	'Appetizer'	20,83985
'_Sample_0866_0001.jpg'	'Breakfast'	'Dessert'	26,62781
'_Sample_0034_0001.jpg'	'Breakfast'	'Breakfast'	17,06121
'_Sample_0580_0004.jpg'	'Breakfast'	'Breakfast'	15,226

iterasi 1			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0265_0005.jpg'	'Breakfast'	'Breakfast'	13,43448
'_Sample_0225_0001.jpg'	'Appetizer'	'Dessert'	62,14742
'_Sample_0534_0004.jpg'	'Single_Course'	'Main_Course'	30,14444

iterasi 2			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0081_0001.jpg'	'Side_Dish'	'Side_Dish'	60,7002
'_Sample_0281_0004.jpg'	'Main_Course'	'Main_Course'	9,676931
'_Sample_0804_0004.jpg'	'Dessert'	'Breakfast'	64,35609
'_Sample_0580_0003.jpg'	'Breakfast'	'Breakfast'	44,57455
'_Sample_0599_0001.jpg'	'Breakfast'	'Breakfast'	22,55915
'_Sample_0235_0001.jpg'	'Single_Course'	'Main_Course'	47,68194
'_Sample_0114_0001.jpg'	'Single_Course'	'Single_Course'	39,67562
'_Sample_0335_0005.jpg'	'Main_Course'	'Main_Course'	22,29854
'_Sample_0423_0003.jpg'	'Breakfast'	'Breakfast'	52,57455

iterasi 2			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0066_0001.jpg'	'Appetizer'	'Main_Course'	16,32682
'_Sample_0292_0004.jpg'	'Main_Course'	'Main_Course'	18,86517
'_Sample_0019_0001.jpg'	'Single_Course'	'Single_Course'	20,7814
'_Sample_0636_0003.jpg'	'Breakfast'	'Main_Course'	41,63706
'_Sample_0393_0002.jpg'	'Appetizer'	'Second_Course'	22,90885
'_Sample_0180_0002.jpg'	'Fruit'	'Fruit'	17,88811
'_Sample_0323_0001.jpg'	'Side_Dish'	'Dessert'	26,80574
'_Sample_0705_0002.jpg'	'Breakfast'	'Second_Course'	62,75908
'_Sample_0101_0001.jpg'	'Side_Dish'	'Side_Dish'	30,26242
'_Sample_0024_0001.jpg'	'Dessert'	'Dessert'	32,32073
'_Sample_0380_0002.jpg'	'Side_Dish'	'Side_Dish'	32,61176
'_Sample_0344_0002.jpg'	'Main_Course'	'Breakfast'	16,84337
'_Sample_0708_0001.jpg'	'Breakfast'	'Breakfast'	16,92518
'_Sample_0186_0003.jpg'	'Fruit'	'Fruit'	33,66235

iterasi 2			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0234_0001.jpg'	'Appetizer'	'Side_Dish'	30,10073
'_Sample_0273_0004.jpg'	'Single_Course'	'Single_Course'	15,56932
'_Sample_0456_0005.jpg'	'Single_Course'	'Single_Course'	20,08984
'_Sample_0180_0001.jpg'	'Fruit'	'Fruit'	21,38889
'_Sample_0579_0001.jpg'	'Breakfast'	'Side_Dish'	37,22311
'_Sample_0086_0002.jpg'	'Appetizer'	'Appetizer'	46,38585
'_Sample_0567_0002.jpg'	'Single_Course'	'Single_Course'	12,70746
'_Sample_0062_0001.jpg'	'Second_Course'	'Side_Dish'	17,54952

iterasi 3			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0580_0001.jpg'	'Breakfast'	'Breakfast'	12,25813
'_Sample_0288_0001.jpg'	'Main_Course'	'Main_Course'	18,86517
'_Sample_1143_0001.jpg'	'Breakfast'	'Appetizer'	48,54921
'_Sample_0759_0004.jpg'	'Dessert'	'Dessert'	33,12657

iterasi 3			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0088_0001.jpg'	'Second_Course'	'Side_Dish'	12,60708
'_Sample_0182_0006.jpg'	'Appetizer'	'Appetizer'	48,91526
'_Sample_0117_0002.jpg'	'Fruit'	'Fruit'	34,84123
'_Sample_0580_0005.jpg'	'Breakfast'	'Breakfast'	45,5939
'_Sample_0152_0001.jpg'	'Side_Dish'	'Main_Course'	17,21939
'_Sample_0220_0002.jpg'	'Fruit'	'Fruit'	30,79808
'_Sample_0027_0001.jpg'	'Fruit'	'Appetizer'	25,75541
'_Sample_0580_0003.jpg'	'Breakfast'	'Breakfast'	44,34967
'_Sample_0403_0001.jpg'	'Dessert'	'Dessert'	14,27682
'_Sample_0014_0001.jpg'	'Main_Course'	'Main_Course'	13,82294
'_Sample_0468_0002.jpg'	'Appetizer'	'Appetizer'	25,86301
'_Sample_0021_0001.jpg'	'Second_Course'	'Second_Course'	13,06801
'_Sample_0239_0001.jpg'	'Dessert'	'Single_Course'	19,79029
'_Sample_0668_0003.jpg'	'Breakfast'	'Single_Course'	26,50271

iterasi 3			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0228_0002.jpg'	'Appetizer'	'Appetizer'	47,7594
'_Sample_0134_0003.jpg'	'Appetizer'	'Appetizer'	15,22749
'_Sample_0256_0001.jpg'	'Fruit'	'Single_Course'	17,67113
'_Sample_0112_0001.jpg'	'Appetizer'	'Appetizer'	36,06667
'_Sample_0079_0001.jpg'	'Second_Course'	'Main_Course'	14,00626
'_Sample_0147_0001.jpg'	'Single_Course'	'Dessert'	19,24074
'_Sample_0812_0003.jpg'	'Dessert'	'Second_Course'	34,67648
'_Sample_0012_0001.jpg'	'Second_Course'	'Fruit'	31,40515
'_Sample_0086_0001.jpg'	'Appetizer'	'Appetizer'	16,77352
'_Sample_0923_0001.jpg'	'Fruit'	'Main_Course'	40,71486
'_Sample_0167_0001.jpg'	'Dessert'	'Dessert'	29,12757
'_Sample_0031_0001.jpg'	'Main_Course'	'Breakfast'	19,97216
'_Sample_0036_0001.jpg'	'Single_Course'	'Appetizer'	27,57852

iterasi 4			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0439_0002.jpg'	'Appetizer'	'Appetizer'	42,48224
'_Sample_0268_0003.jpg'	'Side_Dish'	'Side_Dish'	19,00651
'_Sample_0131_0001.jpg'	'Side_Dish'	'Fruit'	23,41108
'_Sample_0017_0001.jpg'	'Second_Course'	'Second_Course'	22,64817
'_Sample_0064_0001.jpg'	'Second_Course'	'Breakfast'	26,78823
'_Sample_0044_0001.jpg'	'Second_Course'	'Single_Course'	21,62408
'_Sample_0293_0002.jpg'	'Side_Dish'	'Single_Course'	41,9304
'_Sample_0065_0001.jpg'	'Main_Course'	'Main_Course'	31,51547
'_Sample_0591_0001.jpg'	'Fruit'	'Single_Course'	17,97169
'_Sample_0896_0005.jpg'	'Appetizer'	'Appetizer'	301,792
'_Sample_0305_0001.jpg'	'Main_Course'	'Main_Course'	75,30498
'_Sample_0507_0007.jpg'	'Side_Dish'	'Side_Dish'	16,24398
'_Sample_0185_0002.jpg'	'Fruit'	'Fruit'	16,46194
'_Sample_0052_0001.jpg'	'Second_Course'	'Second_Course'	9,593938

iterasi 4			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0224_0002.jpg'	'Appetizer'	'Appetizer'	45,83714
'_Sample_0285_0002.jpg'	'Appetizer'	'Appetizer'	11,47764
'_Sample_0108_0001.jpg'	'Second_Course'	'Second_Course'	17,45819
'_Sample_1024_0001.jpg'	'Dessert'	'Dessert'	8,666968
'_Sample_0246_0001.jpg'	'Dessert'	'Appetizer'	62,14742
'_Sample_0375_0001.jpg'	'Dessert'	'Dessert'	20,55836
'_Sample_0532_0002.jpg'	'Single_Course'	'Breakfast'	17,9836
'_Sample_0200_0001.jpg'	'Single_Course'	'Single_Course'	17,68244
'_Sample_0153_0001.jpg'	'Main_Course'	'Main_Course'	15,2896
'_Sample_0265_0001.jpg'	'Breakfast'	'Breakfast'	22,87124
'_Sample_0941_0004.jpg'	'Fruit'	'Breakfast'	25,67055
'_Sample_0739_0005.jpg'	'Dessert'	'Dessert'	39,39118
'_Sample_0582_0003.jpg'	'Single_Course'	'Second_Course'	36,74413
'_Sample_0290_0001.jpg'	'Single_Course'	'Single_Course'	11,86199

iterasi 4			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0966_0001.jpg'	'Fruit'	'Fruit'	12,70977
'_Sample_0782_0001.jpg'	'Breakfast'	'Appetizer'	19,44972
'_Sample_0599_0002.jpg'	'Breakfast'	'Breakfast'	68,01383

iterasi 5			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0238_0001.jpg'	'Breakfast'	'Second_Course'	26,25254
'_Sample_0423_0004.jpg'	'Breakfast'	'Breakfast'	56,57938
'_Sample_0074_0001.jpg'	'Second_Course'	'Main_Course'	24,51512
'_Sample_0423_0003.jpg'	'Breakfast'	'Breakfast'	56,57938
'_Sample_0285_0001.jpg'	'Appetizer'	'Appetizer'	11,47764
'_Sample_0253_0001.jpg'	'Appetizer'	'Appetizer'	11,76879
'_Sample_0306_0002.jpg'	'Main_Course'	'Second_Course'	20,70272
'_Sample_0176_0001.jpg'	'Side_Dish'	'Breakfast'	49,18223
'_Sample_0218_0002.jpg'	'Fruit'	'Fruit'	54,66103

iterasi 5			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_1023_0001.jpg'	'Breakfast'	'Breakfast'	26,00174
'_Sample_0424_0002.jpg'	'Side_Dish'	'Side_Dish'	11,79055
'_Sample_0599_0004.jpg'	'Breakfast'	'Breakfast'	59,57938
'_Sample_0423_0003.jpg'	'Breakfast'	'Breakfast'	61,57938
'_Sample_0207_0002.jpg'	'Second_Course'	'Main_Course'	20,7644
'_Sample_0784_0004.jpg'	'Dessert'	'Single_Course'	24,32054
'_Sample_0586_0001.jpg'	'Side_Dish'	'Second_Course'	12,60708
'_Sample_0758_0002.jpg'	'Dessert'	'Dessert'	12,41543
'_Sample_0110_0001.jpg'	'Side_Dish'	'Side_Dish'	25,80563
'_Sample_0192_0001.jpg'	'Side_Dish'	'Main_Course'	24,29936
'_Sample_0912_0003.jpg'	'Dessert'	'Dessert'	12,53568
'_Sample_0077_0003.jpg'	'Second_Course'	'Single_Course'	24,18508
'_Sample_0154_0001.jpg'	'Appetizer'	'Breakfast'	35,58705
'_Sample_0329_0001.jpg'	'Single_Course'	'Fruit'	17,97169

iterasi 5			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0040_0001.jpg'	'Single_Course'	'Second_Course'	24,49377
'_Sample_0941_0003.jpg'	'Fruit'	'Breakfast'	15,05652
'_Sample_0092_0001.jpg'	'Side_Dish'	'Side_Dish'	10,24165
'_Sample_0255_0004.jpg'	'Main_Course'	'Main_Course'	13,82294
'_Sample_1066_0001.jpg'	'Dessert'	'Dessert'	34,27069
'_Sample_0374_0003.jpg'	'Single_Course'	'Single_Course'	16,11282
'_Sample_0395_0003.jpg'	'Side_Dish'	'Side_Dish'	30,26242
'_Sample_0194_0001.jpg'	'Dessert'	'Single_Course'	48,18495

iterasi 6			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0071_0001.jpg'	'Second_Course'	'Second_Course'	32,78845
'_Sample_0204_0003.jpg'	'Second_Course'	'Second_Course'	21,87586
'_Sample_0182_0002.jpg'	'Appetizer'	'Appetizer'	10,05508
'_Sample_0263_0002.jpg'	'Appetizer'	'Appetizer'	12,40273

iterasi 6			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0436_0001.jpg'	'Dessert'	'Breakfast'	22,122
'_Sample_0427_0001.jpg'	'Dessert'	'Single_Course'	17,89689
'_Sample_0215_0002.jpg'	'Fruit'	'Fruit'	45,22508
'_Sample_0238_0003.jpg'	'Breakfast'	'Second_Course'	18,74348
'_Sample_0197_0003.jpg'	'Side_Dish'	'Main_Course'	25,91483
'_Sample_0255_0003.jpg'	'Main_Course'	'Main_Course'	27,09141
'_Sample_0193_0002.jpg'	'Appetizer'	'Appetizer'	13,72018
'_Sample_0604_0005.jpg'	'Side_Dish'	'Side_Dish'	11,96763
'_Sample_0023_0001.jpg'	'Main_Course'	'Side_Dish'	17,21939
'_Sample_0107_0006.jpg'	'Second_Course'	'Second_Course'	24,19742
'_Sample_0532_0004.jpg'	'Single_Course'	'Single_Course'	10,59264
'_Sample_0241_0001.jpg'	'Appetizer'	'Appetizer'	18,46132
'_Sample_0179_0001.jpg'	'Main_Course'	'Breakfast'	15,72494
'_Sample_0759_0002.jpg'	'Dessert'	'Dessert'	22,09112

iterasi 6			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0002_0001.jpg'	'Single_Course'	'Single_Course'	25,85469
'_Sample_0173_0001.jpg'	'Side_Dish'	'Second_Course'	25,48763
'_Sample_0219_0003.jpg'	'Fruit'	'Fruit'	138,6577
'_Sample_0047_0001.jpg'	'Single_Course'	'Dessert'	41,69858
'_Sample_0144_0001.jpg'	'Main_Course'	'Dessert'	71,99134
'_Sample_0580_0004.jpg'	'Breakfast'	'Breakfast'	28,10145
'_Sample_0481_0001.jpg'	'Side_Dish'	'Side_Dish'	18,85717
'_Sample_0177_0001.jpg'	'Main_Course'	'Appetizer'	43,79803
'_Sample_0028_0001.jpg'	'Side_Dish'	'Second_Course'	39,7131
'_Sample_0361_0002.jpg'	'Appetizer'	'Appetizer'	32,90878
'_Sample_0138_0001.jpg'	'Fruit'	'Appetizer'	83,83912
'_Sample_0336_0002.jpg'	'Second_Course'	'Second_Course'	38,02209
'_Sample_0049_0001.jpg'	'Second_Course'	'Side_Dish'	43,13677

iterasi 7			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0076_0001.jpg'	'Side_Dish'	'Appetizer'	30,10073
'_Sample_0439_0004.jpg'	'Appetizer'	'Appetizer'	11,76879
'_Sample_0598_0002.jpg'	'Single_Course'	'Single_Course'	24,29031
'_Sample_0069_0001.jpg'	'Second_Course'	'Dessert'	24,04934
'_Sample_0784_0003.jpg'	'Dessert'	'Dessert'	12,41543
'_Sample_0158_0001.jpg'	'Main_Course'	'Breakfast'	41,70734
'_Sample_0733_0003.jpg'	'Dessert'	'Dessert'	29,12757
'_Sample_0301_0002.jpg'	'Main_Course'	'Second_Course'	32,74946
'_Sample_0337_0004.jpg'	'Main_Course'	'Breakfast'	29,60931
'_Sample_0319_0001.jpg'	'Dessert'	'Dessert'	21,48164
'_Sample_0847_0001.jpg'	'Fruit'	'Breakfast'	37,21854
'_Sample_0442_0001.jpg'	'Dessert'	'Dessert'	59,35445
'_Sample_0022_0001.jpg'	'Side_Dish'	'Main_Course'	33,31019

iterasi 7			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0095_0001.jpg'	'Side_Dish'	'Side_Dish'	11,96763
'_Sample_0708_0002.jpg'	'Breakfast'	'Breakfast'	16,92518
'_Sample_0535_0005.jpg'	'Single_Course'	'Single_Course'	26,16803
'_Sample_0784_0001.jpg'	'Dessert'	'Dessert'	32,99039
'_Sample_0009_0001.jpg'	'Second_Course'	'Breakfast'	18,74348
'_Sample_0007_0001.jpg'	'Breakfast'	'Main_Course'	15,72494
'_Sample_0654_0001.jpg'	'Fruit'	'Fruit'	87,48677
'_Sample_0599_0004.jpg'	'Breakfast'	'Breakfast'	55,35186
'_Sample_0828_0001.jpg'	'Dessert'	'Dessert'	196,3567
'_Sample_0308_0008.jpg'	'Main_Course'	'Second_Course'	20,36985
'_Sample_0273_0001.jpg'	'Single_Course'	'Single_Course'	22,16407
'_Sample_0004_0001.jpg'	'Main_Course'	'Side_Dish'	80,14916
'_Sample_0146_0001.jpg'	'Appetizer'	'Second_Course'	30,48539
'_Sample_0078_0001.jpg'	'Second_Course'	'Appetizer'	33,53876

iterasi 7			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0211_0003.jpg'	'Second_Course'	'Second_Course'	14,70139
'_Sample_0219_0002.jpg'	'Fruit'	'Fruit'	54,59976
'_Sample_0013_0001.jpg'	'Side_Dish'	'Main_Course'	22,54543
'_Sample_0491_0001.jpg'	'Side_Dish'	'Side_Dish'	23,00204

iterasi 8			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0271_0001.jpg'	'Appetizer'	'Single_Course'	20,61269
'_Sample_0654_0002.jpg'	'Fruit'	'Breakfast'	50,78916
'_Sample_0599_0003.jpg'	'Breakfast'	'Breakfast'	18,41217
'_Sample_0082_0001.jpg'	'Side_Dish'	'Breakfast'	58,68648
'_Sample_0037_0001.jpg'	'Side_Dish'	'Side_Dish'	16,19271
'_Sample_0286_0003.jpg'	'Fruit'	'Fruit'	45,22508
'_Sample_0579_0002.jpg'	'Breakfast'	'Breakfast'	28,41217
'_Sample_0566_0004.jpg'	'Single_Course'	'Single_Course'	38,28558

iterasi 8			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0224_0003.jpg'	'Appetizer'	'Appetizer'	45,83714
'_Sample_0228_0001.jpg'	'Appetizer'	'Appetizer'	40,50531
'_Sample_0337_0002.jpg'	'Main_Course'	'Main_Course'	36,74458
'_Sample_0579_0004.jpg'	'Breakfast'	'Breakfast'	19,52289
'_Sample_0161_0001.jpg'	'Single_Course'	'Dessert'	17,89689
'_Sample_0030_0001.jpg'	'Dessert'	'Breakfast'	27,20557
'_Sample_0580_0002.jpg'	'Breakfast'	'Breakfast'	12,25813
'_Sample_0109_0001.jpg'	'Side_Dish'	'Appetizer'	32,21135
'_Sample_0182_0010.jpg'	'Appetizer'	'Appetizer'	20,64317
'_Sample_0223_0001.jpg'	'Appetizer'	'Main_Course'	16,29869
'_Sample_0579_0003.jpg'	'Breakfast'	'Breakfast'	64,41217
'_Sample_0224_0001.jpg'	'Appetizer'	'Appetizer'	68,18361
'_Sample_0668_0002.jpg'	'Breakfast'	'Breakfast'	17,06121
'_Sample_0229_0001.jpg'	'Single_Course'	'Second_Course'	27,74921

iterasi 8			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0322_0001.jpg'	'Appetizer'	'Side_Dish'	29,16543
'_Sample_0423_0004.jpg'	'Breakfast'	'Breakfast'	61,41217
'_Sample_0333_0004.jpg'	'Single_Course'	'Single_Course'	27,13117
'_Sample_0169_0001.jpg'	'Side_Dish'	'Side_Dish'	19,00651
'_Sample_0160_0001.jpg'	'Side_Dish'	'Breakfast'	24,55453
'_Sample_0262_0003.jpg'	'Second_Course'	'Second_Course'	34,42252
'_Sample_0317_0001.jpg'	'Single_Course'	'Second_Course'	17,35714
'_Sample_0327_0001.jpg'	'Single_Course'	'Single_Course'	20,30148
'_Sample_0573_0003.jpg'	'Single_Course'	'Single_Course'	36,35606

iterasi 9			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0468_0004.jpg'	'Appetizer'	'Appetizer'	24,77483
'_Sample_0393_0001.jpg'	'Appetizer'	'Appetizer'	26,74273

iterasi 9			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0579_0003.jpg'	'Breakfast'	'Breakfast'	25,74807
'_Sample_0455_0005.jpg'	'Side_Dish'	'Side_Dish'	49,74807
'_Sample_0008_0001.jpg'	'Dessert'	'Second_Course'	36,0755
'_Sample_0174_0001.jpg'	'Side_Dish'	'Side_Dish'	16,24398
'_Sample_0050_0001.jpg'	'Main_Course'	'Second_Course'	24,51512
'_Sample_0240_0001.jpg'	'Single_Course'	'Dessert'	19,79029
'_Sample_0597_0003.jpg'	'Single_Course'	'Single_Course'	11,86199
'_Sample_0217_0001.jpg'	'Fruit'	'Appetizer'	19,01961
'_Sample_0532_0005.jpg'	'Single_Course'	'Single_Course'	26,75601
'_Sample_0334_0007.jpg'	'Main_Course'	'Appetizer'	32,1288
'_Sample_0190_0001.jpg'	'Fruit'	'Fruit'	32,75978
'_Sample_0220_0003.jpg'	'Fruit'	'Fruit'	13,2681
'_Sample_0668_0001.jpg'	'Breakfast'	'Breakfast'	27,81865
'_Sample_0060_0001.jpg'	'Second_Course'	'Second_Course'	28,07618

iterasi 9			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0075_0001.jpg'	'Main_Course'	'Main_Course'	21,5289
'_Sample_0084_0001.jpg'	'Appetizer'	'Main_Course'	23,13401
'_Sample_0187_0001.jpg'	'Fruit'	'Main_Course'	19,86087
'_Sample_0335_0011.jpg'	'Main_Course'	'Main_Course'	20,77675
'_Sample_0456_0007.jpg'	'Single_Course'	'Second_Course'	55,39351
'_Sample_0263_0001.jpg'	'Appetizer'	'Appetizer'	12,40273
'_Sample_0265_0002.jpg'	'Breakfast'	'Side_Dish'	35,62409
'_Sample_0057_0001.jpg'	'Second_Course'	'Dessert'	20,56067
'_Sample_0733_0001.jpg'	'Dessert'	'Dessert'	34,47585
'_Sample_0243_0001.jpg'	'Second_Course'	'Second_Course'	34,64752
'_Sample_0282_0001.jpg'	'Dessert'	'Dessert'	43,40275
'_Sample_0011_0001.jpg'	'Second_Course'	'Main_Course'	16,25062
'_Sample_0247_0001.jpg'	'Dessert'	'Single_Course'	25,13231
'_Sample_0232_0001.jpg'	'Appetizer'	'Fruit'	47,23917

iterasi 9			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0335_0003.jpg'	'Main_Course'	'Main_Course'	19,40494

iterasi 10			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0392_0002.jpg'	'Single_Course'	'Single_Course'	24,14337
'_Sample_0056_0001.jpg'	'Main_Course'	'Fruit'	28,45185
'_Sample_0137_0001.jpg'	'Dessert'	'Second_Course'	20,56067
'_Sample_0055_0001.jpg'	'Main_Course'	'Appetizer'	21,9323
'_Sample_0739_0004.jpg'	'Dessert'	'Dessert'	130,0174
'_Sample_0310_0003.jpg'	'Second_Course'	'Dessert'	34,67648
'_Sample_0087_0001.jpg'	'Side_Dish'	'Side_Dish'	28,73177
'_Sample_0171_0001.jpg'	'Side_Dish'	'Single_Course'	18,64564
'_Sample_0267_0001.jpg'	'Single_Course'	'Fruit'	36,13688
'_Sample_0033_0001.jpg'	'Second_Course'	'Breakfast'	31,12284
'_Sample_0183_0001.jpg'	'Dessert'	'Appetizer'	31,36367

iterasi 10			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0354_0002.jpg'	'Side_Dish'	'Side_Dish'	36,03309
'_Sample_0190_0002.jpg'	'Fruit'	'Fruit'	45,2165
'_Sample_0220_0001.jpg'	'Fruit'	'Single_Course'	32,48607
'_Sample_0559_0001.jpg'	'Single_Course'	'Single_Course'	20,30148
'_Sample_0575_0004.jpg'	'Dessert'	'Dessert'	69,22588
'_Sample_0313_0001.jpg'	'Second_Course'	'Dessert'	53,54636
'_Sample_0941_0001.jpg'	'Fruit'	'Breakfast'	17,02025
'_Sample_0257_0002.jpg'	'Second_Course'	'Second_Course'	17,34233
'_Sample_0124_0001.jpg'	'Fruit'	'Fruit'	29,02475
'_Sample_0058_0001.jpg'	'Main_Course'	'Side_Dish'	18,78901
'_Sample_0261_0001.jpg'	'Main_Course'	'Side_Dish'	25,91483
'_Sample_0344_0003.jpg'	'Main_Course'	'Main_Course'	22,26915
'_Sample_0226_0002.jpg'	'Second_Course'	'Single_Course'	36,74413
'_Sample_0957_0001.jpg'	'Breakfast'	'Second_Course'	25,8528

iterasi 10			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0100_0001.jpg'	'Main_Course'	'Single_Course'	29,99911
'_Sample_0423_0001.jpg'	'Breakfast'	'Second_Course'	26,94024
'_Sample_0444_0003.jpg'	'Appetizer'	'Appetizer'	20,1866
'_Sample_0043_0001.jpg'	'Second_Course'	'Second_Course'	26,80742
'_Sample_0187_0002.jpg'	'Fruit'	'Fruit'	34,58477
'_Sample_0207_0001.jpg'	'Second_Course'	'Second_Course'	22,34189

iterasi 11			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0245_0001.jpg'	'Single_Course'	'Single_Course'	20,08984
'_Sample_0215_0001.jpg'	'Fruit'	'Appetizer'	33,73824
'_Sample_0607_0002.jpg'	'Single_Course'	'Single_Course'	18,93599
'_Sample_0362_0001.jpg'	'Dessert'	'Side_Dish'	38,11872
'_Sample_0579_0002.jpg'	'Breakfast'	'Breakfast'	58,27654
'_Sample_0211_0001.jpg'	'Second_Course'	'Second_Course'	11,21808

iterasi 11			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0636_0002.jpg'	'Breakfast'	'Breakfast'	65,27654
'_Sample_0182_0001.jpg'	'Appetizer'	'Appetizer'	10,05508
'_Sample_0143_0001.jpg'	'Side_Dish'	'Main_Course'	34,87746
'_Sample_0128_0001.jpg'	'Main_Course'	'Second_Course'	16,25062
'_Sample_0218_0001.jpg'	'Fruit'	'Fruit'	81,60062
'_Sample_0294_0001.jpg'	'Main_Course'	'Main_Course'	15,2896
'_Sample_0507_0001.jpg'	'Side_Dish'	'Side_Dish'	9,63605
'_Sample_0117_0001.jpg'	'Fruit'	'Fruit'	16,40039
'_Sample_0504_0002.jpg'	'Single_Course'	'Single_Course'	12,70746
'_Sample_0244_0002.jpg'	'Side_Dish'	'Side_Dish'	24,25759
'_Sample_0238_0002.jpg'	'Breakfast'	'Main_Course'	19,65846
'_Sample_0255_0001.jpg'	'Main_Course'	'Main_Course'	15,78852
'_Sample_0184_0001.jpg'	'Dessert'	'Second_Course'	48,10773
'_Sample_0912_0004.jpg'	'Dessert'	'Dessert'	12,53568

iterasi 11			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0318_0001.jpg'	'Second_Course'	'Fruit'	21,15474
'_Sample_0178_0001.jpg'	'Main_Course'	'Main_Course'	36,75041
'_Sample_0499_0004.jpg'	'Appetizer'	'Main_Course'	32,1288
'_Sample_0270_0001.jpg'	'Breakfast'	'Dessert'	25,0962
'_Sample_1041_0001.jpg'	'Breakfast'	'Second_Course'	24,00345
'_Sample_0059_0001.jpg'	'Dessert'	'Breakfast'	47,73166
'_Sample_0339_0003.jpg'	'Second_Course'	'Second_Course'	14,70139
'_Sample_0186_0001.jpg'	'Fruit'	'Fruit'	13,2681
'_Sample_0216_0001.jpg'	'Single_Course'	'Single_Course'	24,14337
'_Sample_0582_0001.jpg'	'Single_Course'	'Single_Course'	30,17209
'_Sample_0164_0001.jpg'	'Side_Dish'	'Side_Dish'	24,07173

iterasi 12			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0072_0001.jpg'	'Single_Course'	'Single_Course'	10,59264

iterasi 12			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0182_0007.jpg'	'Appetizer'	'Appetizer'	7,860741
'_Sample_0270_0002.jpg'	'Breakfast'	'Breakfast'	22,40873
'_Sample_0636_0001.jpg'	'Breakfast'	'Breakfast'	19,52289
'_Sample_0374_0002.jpg'	'Single_Course'	'Single_Course'	19,97438
'_Sample_0136_0002.jpg'	'Fruit'	'Fruit'	16,40039
'_Sample_0188_0001.jpg'	'Fruit'	'Fruit'	18,41699
'_Sample_0206_0002.jpg'	'Second_Course'	'Second_Course'	72,63439
'_Sample_0599_0003.jpg'	'Breakfast'	'Breakfast'	54,52374
'_Sample_0252_0004.jpg'	'Second_Course'	'Second_Course'	11,21808
'_Sample_0718_0002.jpg'	'Dessert'	'Dessert'	32,32073
'_Sample_0155_0001.jpg'	'Main_Course'	'Single_Course'	34,75977
'_Sample_0003_0001.jpg'	'Side_Dish'	'Side_Dish'	58,19907
'_Sample_0286_0001.jpg'	'Fruit'	'Fruit'	15,98615
'_Sample_0618_0003.jpg'	'Dessert'	'Dessert'	32,7587

iterasi 12			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0189_0001.jpg'	'Fruit'	'Fruit'	50,27254
'_Sample_0444_0005.jpg'	'Appetizer'	'Appetizer'	24,77483
'_Sample_0443_0001.jpg'	'Dessert'	'Dessert'	13,03705
'_Sample_0129_0001.jpg'	'Appetizer'	'Dessert'	26,7052
'_Sample_0733_0002.jpg'	'Dessert'	'Breakfast'	13,72983
'_Sample_0530_0007.jpg'	'Dessert'	'Dessert'	32,69346
'_Sample_0196_0002.jpg'	'Second_Course'	'Main_Course'	20,36985
'_Sample_0400_0002.jpg'	'Single_Course'	'Main_Course'	45,26624
'_Sample_0538_0006.jpg'	'Single_Course'	'Single_Course'	22,89721
'_Sample_0286_0002.jpg'	'Fruit'	'Fruit'	16,92284
'_Sample_0778_0001.jpg'	'Fruit'	'Main_Course'	28,45185
'_Sample_0096_0001.jpg'	'Appetizer'	'Single_Course'	20,83985
'_Sample_0302_0001.jpg'	'Dessert'	'Dessert'	64,48588
'_Sample_0130_0001.jpg'	'Second_Course'	'Main_Course'	38,2823

iterasi 12			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0542_0001.jpg'	'Single_Course'	'Single_Course'	17,68244
'_Sample_0035_0001.jpg'	'Main_Course'	'Main_Course'	19,40494

iterasi 13			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0015_0001.jpg'	'Second_Course'	'Breakfast'	26,25254
'_Sample_0175_0001.jpg'	'Side_Dish'	'Single_Course'	19,10552
'_Sample_0385_0003.jpg'	'Side_Dish'	'Side_Dish'	28,73177
'_Sample_0191_0001.jpg'	'Fruit'	'Fruit'	18,41699
'_Sample_0606_0005.jpg'	'Single_Course'	'Single_Course'	39,67562
'_Sample_0326_0001.jpg'	'Single_Course'	'Breakfast'	27,13334
'_Sample_0237_0001.jpg'	'Dessert'	'Dessert'	25,62107
'_Sample_0349_0005.jpg'	'Appetizer'	'Appetizer'	39,61765
'_Sample_0566_0003.jpg'	'Single_Course'	'Dessert'	24,32054
'_Sample_0001_0001.jpg'	'Second_Course'	'Second_Course'	35,82397

iterasi 13			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0189_0002.jpg'	'Fruit'	'Fruit'	54,59976
'_Sample_0134_0001.jpg'	'Appetizer'	'Appetizer'	11,76431
'_Sample_0289_0001.jpg'	'Single_Course'	'Single_Course'	34,76499
'_Sample_0274_0001.jpg'	'Dessert'	'Dessert'	22,09112
'_Sample_0242_0001.jpg'	'Fruit'	'Side_Dish'	46,12467
'_Sample_0180_0004.jpg'	'Fruit'	'Single_Course'	38,60395
'_Sample_0136_0001.jpg'	'Fruit'	'Appetizer'	29,00841
'_Sample_0269_0003.jpg'	'Side_Dish'	'Side_Dish'	25,02926
'_Sample_0067_0001.jpg'	'Second_Course'	'Side_Dish'	16,52688
'_Sample_0124_0002.jpg'	'Fruit'	'Fruit'	29,02475
'_Sample_0034_0002.jpg'	'Breakfast'	'Dessert'	13,72983
'_Sample_0566_0005.jpg'	'Single_Course'	'Fruit'	35,97499
'_Sample_0705_0001.jpg'	'Breakfast'	'Second_Course'	34,30691
'_Sample_0204_0001.jpg'	'Second_Course'	'Second_Course'	26,13937

iterasi 13			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0214_0001.jpg'	'Appetizer'	'Fruit'	38,24563
'_Sample_0165_0001.jpg'	'Main_Course'	'Side_Dish'	22,54543
'_Sample_0127_0005.jpg'	'Second_Course'	'Second_Course'	19,42908
'_Sample_0193_0001.jpg'	'Appetizer'	'Appetizer'	13,72018
'_Sample_0185_0001.jpg'	'Fruit'	'Side_Dish'	23,41108
'_Sample_0328_0004.jpg'	'Second_Course'	'Second_Course'	9,593938
'_Sample_0417_0001.jpg'	'Dessert'	'Dessert'	32,99039

iterasi 14			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0265_0003.jpg'	'Breakfast'	'Breakfast'	22,87124
'_Sample_0248_0001.jpg'	'Dessert'	'Appetizer'	29,32884
'_Sample_0054_0003.jpg'	'Second_Course'	'Breakfast'	25,8528
'_Sample_0495_0007.jpg'	'Single_Course'	'Single_Course'	22,16407
'_Sample_0199_0001.jpg'	'Appetizer'	'Appetizer'	31,00068

iterasi 14			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0346_0001.jpg'	'Dessert'	'Dessert'	25,62107
'_Sample_0018_0001.jpg'	'Single_Course'	'Single_Course'	18,93599
'_Sample_0636_0004.jpg'	'Breakfast'	'Main_Course'	16,84337
'_Sample_0181_0001.jpg'	'Main_Course'	'Single_Course'	32,29804
'_Sample_0203_0003.jpg'	'Main_Course'	'Main_Course'	21,5289
'_Sample_0599_0002.jpg'	'Breakfast'	'Breakfast'	54,69482
'_Sample_0094_0001.jpg'	'Second_Course'	'Second_Course'	17,45819
'_Sample_0126_0001.jpg'	'Side_Dish'	'Side_Dish'	54,02955
'_Sample_0016_0001.jpg'	'Main_Course'	'Second_Course'	29,55845
'_Sample_0070_0001.jpg'	'Main_Course'	'Single_Course'	24,839
'_Sample_0091_0001.jpg'	'Single_Course'	'Second_Course'	23,58473
'_Sample_0163_0001.jpg'	'Side_Dish'	'Dessert'	33,6016
'_Sample_0272_0002.jpg'	'Main_Course'	'Breakfast'	19,65846
'_Sample_0208_0001.jpg'	'Appetizer'	'Fruit'	19,01961

iterasi 14			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0337_0001.jpg'	'Main_Course'	'Main_Course'	17,5749
'_Sample_0132_0001.jpg'	'Second_Course'	'Main_Course'	32,74946
'_Sample_0270_0004.jpg'	'Breakfast'	'Breakfast'	13,43448
'_Sample_0395_0002.jpg'	'Side_Dish'	'Side_Dish'	16,19271
'_Sample_0026_0001.jpg'	'Appetizer'	'Second_Course'	29,31073
'_Sample_0054_0001.jpg'	'Second_Course'	'Breakfast'	26,94024
'_Sample_0636_0002.jpg'	'Breakfast'	'Breakfast'	52,69482
'_Sample_0209_0003.jpg'	'Side_Dish'	'Side_Dish'	38,4499
'_Sample_0397_0006.jpg'	'Appetizer'	'Appetizer'	16,77352
'_Sample_0334_0004.jpg'	'Main_Course'	'Appetizer'	29,67471
'_Sample_0020_0001.jpg'	'Appetizer'	'Breakfast'	19,44972
'_Sample_0048_0001.jpg'	'Second_Course'	'Second_Course'	17,34233

iterasi 15			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0212_0001.jpg'	'Single_Course'	'Single_Course'	30,17209
'_Sample_0266_0001.jpg'	'Main_Course'	'Main_Course'	23,56181
'_Sample_0063_0001.jpg'	'Main_Course'	'Second_Course'	25,42657
'_Sample_0093_0003.jpg'	'Appetizer'	'Appetizer'	11,76431
'_Sample_0294_0004.jpg'	'Main_Course'	'Main_Course'	20,89694
'_Sample_0053_0001.jpg'	'Main_Course'	'Appetizer'	16,29869
'_Sample_0276_0001.jpg'	'Second_Course'	'Second_Course'	15,63126
'_Sample_0038_0001.jpg'	'Second_Course'	'Single_Course'	17,35714
'_Sample_0300_0001.jpg'	'Dessert'	'Single_Course'	39,47056
'_Sample_0668_0004.jpg'	'Breakfast'	'Single_Course'	29,97176
'_Sample_0524_0001.jpg'	'Dessert'	'Second_Course'	53,54636
'_Sample_0118_0001.jpg'	'Appetizer'	'Fruit'	25,75541
'_Sample_0187_0003.jpg'	'Fruit'	'Fruit'	33,66235
'_Sample_0219_0001.jpg'	'Fruit'	'Fruit'	32,75978

iterasi 15			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0148_0001.jpg'	'Main_Course'	'Main_Course'	27,09141
'_Sample_1049_0002.jpg'	'Dessert'	'Dessert'	8,666968
'_Sample_0819_0002.jpg'	'Appetizer'	'Appetizer'	18,46132
'_Sample_0599_0005.jpg'	'Breakfast'	'Single_Course'	17,9836
'_Sample_0157_0001.jpg'	'Side_Dish'	'Side_Dish'	11,79055
'_Sample_0546_0001.jpg'	'Side_Dish'	'Second_Course'	18,24105
'_Sample_0531_0004.jpg'	'Side_Dish'	'Side_Dish'	36,03309
'_Sample_0292_0003.jpg'	'Main_Course'	'Main_Course'	9,676931
'_Sample_0612_0002.jpg'	'Single_Course'	'Single_Course'	25,85469
'_Sample_0108_0002.jpg'	'Second_Course'	'Second_Course'	14,65238
'_Sample_0374_0001.jpg'	'Single_Course'	'Single_Course'	15,56932
'_Sample_0207_0003.jpg'	'Second_Course'	'Second_Course'	22,34189
'_Sample_0499_0002.jpg'	'Appetizer'	'Main_Course'	29,67471
'_Sample_0286_0005.jpg'	'Fruit'	'Fruit'	15,98615

iterasi 15			
Nama File	Actual Class	Predicted Class	Distance
'_Sample_0620_0002.jpg'	'Side_Dish'	'Side_Dish'	10,24165
'_Sample_0272_0004.jpg'	'Main_Course'	'Main_Course'	27,27034
'_Sample_0334_0003.jpg'	'Main_Course'	'Main_Course'	31,51547

DAFTAR PUSTAKA

- [1] R. S. McPherson, D. M. Hoelscher, M. Alexander, K. S. Scanlon, and M. K. Serdula, "Dietary Assessment Methods among School-Aged Children: Validity and Reliability," *Prev. Med.*, vol. 31, no. 2, pp. S11–S33, Aug. 2000.
- [2] T. Joutou and K. Yanai, "A food image recognition system with multiple kernel learning," in *2009 16th IEEE International Conference on Image Processing (ICIP)*, 2009, pp. 285–288.
- [3] G. M. Farinella, D. Allegra, M. Moltisanti, F. Stanco, and S. Battiato, "Retrieval and classification of food images," *Comput. Biol. Med.*, vol. 77, pp. 23–39, Oct. 2016.
- [4] "UNICT-FD1200." [Online]. Available: <http://iplab.dmi.unict.it/UNICT-FD1200/>. [Accessed: 12-Dec-2016].
- [5] "Meal," *Wikipedia*. 07-Jun-2017.
- [6] "Lab Color - MATLAB & Simulink." [Online]. Available: <https://www.mathworks.com/discovery/lab-color.html>. [Accessed: 14-Jun-2017].
- [7] "Visual Geometry Group - Texture Classification." [Online]. Available: <http://www.robots.ox.ac.uk/~vgg/research/texclass/filters.html>. [Accessed: 12-Dec-2016].
- [8] C. Schmid, "Constructing models for content-based image retrieval," 2001, vol. 2, p. II-39-II-45.
- [9] "k-means clustering - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/K-means_clustering. [Accessed: 15-Dec-2016].
- [10] "Morphology - Distance Transform." [Online]. Available: <http://homepages.inf.ed.ac.uk/rbf/HIPR2/distance.htm>. [Accessed: 12-Dec-2016].
- [11] S. Patro and K. K. Sahu, "Normalization: A Preprocessing Stage," *ArXiv Prepr. ArXiv150306462*, 2015.

- [12]L. Al Shalabi, Z. Shaaban, and B. Kasasbeh, “Data mining: A preprocessing engine,” *J. Comput. Sci.*, vol. 2, no. 9, pp. 735–739, 2006.
- [13]B. Julesz, “Textons, the elements of texture perception, and their interactions,” *Nature*, vol. 290, no. 5802, pp. 91–97, 1981.
- [14]G. M. Farinella, M. Moltisanti, and S. Battiato, “Classifying food images represented as bag of textons,” in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 5212–5216.
- [15]C. R. Maurer, R. Qi, and V. Raghavan, “A linear time algorithm for computing exact Euclidean distance transforms of binary images in arbitrary dimensions,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 265–270, 2003.
- [16]“Chi-Square Distance,” in *The Concise Encyclopedia of Statistics*, Springer New York, 2008, pp. 68–70.
- [17]L. E. Peterson, “K-nearest neighbor,” *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.
- [18]M. Story and R. G. Congalton, “Accuracy assessment: a user’s perspective,” *Photogramm. Eng. Remote Sens.*, vol. 52, no. 3, pp. 397–399, 1986.
- [19]P. Refaeilzadeh, L. Tang, and H. Liu, “Cross-validation,” in *Encyclopedia of database systems*, Springer, 2009, pp. 532–538.
- [20]“Cross-validation (statistics),” *Wikipedia*. 06-Jun-2017.



BIODATA PENULIS



Nuning Septiana, lahir di Mojokerto, pada tanggal 25 September 1994. Penulis menempuh pendidikan mulai dari SDN Kranggan I Mojokerto (2001-2007), SMPN I Mojokerto (2007-2010), SMAN 10 Malang Sampoerna Academy (2010-2013) hingga terakhir Institut Teknologi Sepuluh Nopember Surabaya (2013-2017) di jurusan Teknik Informatika, Fakultas Teknologi Informasi angkatan tahun 2013.

Selama belajar di kampus Teknik Informatika, penulis berkesempatan menjadi asisten dosen Sistem dan Teknologi Informasi pada tahun 2014 serta Kecerdasan Buatan pada tahun 2015. Selain mengikuti kegiatan akademik, penulis mengikuti kegiatan organisasi dan kepanitiaan dengan berpartisipasi sebagai anggota Departemen Riset Teknologi HMTTC (2014-2015), Staff Ahli Dana ITS EXPO 2015 (2015-2016) dan bendahara HMTTC (2015-2016).

Penulis memiliki bidang minat Komputasi Cerdas Visi (KCV) dengan fokus studi pada bidang *image processing*, komputasi biomedik. Komunikasi dengan penulis dapat melalui email: **nuning.s123@gmail.com**.

