

SEGMENTASI CITRA MAKANAN MENGGUNAKAN CLUSTERING IMPROVED K-MEANS UNTUK ESTIMASI SISA MAKANAN

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Alip Setiawan

NIM: 175150218113062



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2021

PENGESAHAN

SEGMENTASI CITRA MAKANAN MENGGUNAKAN *CLUSTERING IMPROVED K-MEANS* UNTUK ESTIMASI SISA MAKANAN

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh:

Alip Setiawan

NIM: 175150218113062

Skrripsi ini telah diuji dan dinyatakan lulus pada
26 Juli 2021

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing 1

Dosen Pembimbing 2

Yuita Arum Sari, S.Kom., M.Kom.
NIK: 2016098807152001

Bayu Rahayudi, M.T., M.M.
NIP: 197407122006041001

Mengetahui

Ketua Jurusan **Teknik Informatika**



Achmad Basuki, S.T., M.MG., Ph.D.
NIP: 197411182003121002

PERNYATAAN ORISINALITAS

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

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

Malang, 2 Februari 2021



Alip Setiawan

NIM: 175150218113062



ABSTRAK

Alip Setiawan, Segmentasi citra makanan menggunakan *clustering Improved K-Means* untuk estimasi sisa makanan

Pembimbing: Yuita Arum Sari, S.Kom., M.Kom. dan Bayu Rahayudi, M.T., M.M.

Setiap makhluk hidup membutuhkan makanan untuk hidup, tumbuh dan bereproduksi. Makanan merupakan suatu kebutuhan pokok yang diperlukan oleh manusia sebagai sumber energi untuk melakukan aktivitas dalam kehidupan sehari-hari. Dengan memakan makanan yang bergizi dapat membawa seseorang ke dalam pola makan yang sehat. Pola makan yang tidak sehat dapat menyebabkan seseorang terdampak penyakit seperti gizi buruk atau bahkan kematian. Studi di beberapa negara menunjukkan bahwa anak bergizi baik memiliki hasil tes mental yang lebih baik dibandingkan dengan anak yang pernah menderita gizi salah. Masalah gizi kurang sudah ada sejak negara Indonesia berdiri, namun masalah gizi hingga sekarang masih tetap ada walaupun jumlahnya sudah berkurang. Mengingat pentingnya mengatur jumlah konsumsi makanan untuk memenuhi dan menjaga kesehatan tubuh maka dibutuhkan suatu sistem yang dapat melakukan estimasi bobot dari makanan yang telah dikonsumsi. Untuk melakukan estimasi bobot makanan yang telah dikonsumsi oleh seseorang diperlukan citra dari makanan sebelum dan sesudah dikonsumsi.

Pada penelitian ini data yang digunakan adalah data sekunder berupa citra makanan di *tray box* berjumlah 31 citra yang terdiri dari 124 citra kompartemen. Metode *Improved K-Means* dipilih untuk segmentasi citra makanan pada *tray box*. Dengan penerapan metode tersebut diharapkan dapat memberikan hasil dengan tingkat kesalahan yang kecil dan memiliki akurasi yang baik pada estimasi sisa makanan pada *tray box*. Berdasarkan hasil evaluasi didapatkan rata-rata akurasi hasil segmentasi tertinggi sebesar 97% dengan *RMSE* estimasi bobot sisa makanan terkecil dicapai sebesar 2,19. Hasil pengujian tersebut menunjukkan bahwa metode *clustering* dengan algoritme *Improved K-Means* dapat digunakan untuk estimasi bobot citra makanan.

Kata kunci: segmentasi citra, *clustering*, *Improved K-Means*, citra makanan

ABSTRACT

Alip Setiawan, Food image segmentation using *Improved K-Means clustering* for leftover food estimation

Supervisors: Yuita Arum Sari, S.Kom., M.Kom. and Bayu Rahayudi, M.T., M.M.

Every living thing needs food to live, grow and reproduce. Food is a basic need that is needed by humans as a source of energy to carry out activities in daily life. By eating nutritious food can bring a person into a healthy diet. An unhealthy diet can cause a person to be affected by diseases such as malnutrition or even death. Studies in several countries show that well-nourished children have better mental test results than children who have suffered from malnutrition. The problem of malnutrition has existed since the state of Indonesia was founded, but nutritional problems still exist even though the number has decreased. Given the importance of regulating the amount of food consumption to meet and maintain a healthy body, a system is needed that can estimate the weight of the food that has been consumed. To estimate the weight of food that has been consumed by a person, an image of the food before and after consumption is needed.

In this study, the data used is secondary data in the form of food images in the tray box totaling 31 images consisting of 124 compartment images. The Improved K-Means method was chosen for segmenting food images on the tray box. With the application of this method, it is expected to give results with a small error rate and have good accuracy in the estimation of food waste in the tray box. Based on the evaluation results, the highest average accuracy of segmentation results was 97% with the smallest estimated RMSE of food waste weight achieved at 2.19. The test results indicate that the Clustering method with the Improved K-Means algorithm can be used to estimate the weight of food images.

Keyword: image segmentation, clustering, Improved K-Means, food image

DAFTAR ISI

PENGESAHAN.....	Error! Bookmark not defined.
PERNYATAAN ORISINALITAS.....	Error! Bookmark not defined.
PRAKATA.....	iv
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan.....	2
1.4 Manfaat.....	2
1.5 Batasan Masalah.....	2
1.6 Sistematika Pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN.....	4
2.1 Kajian Pustaka.....	4
2.2 Tray box.....	5
2.3 Ruang Warna CIELAB.....	5
2.4 Median Filter.....	6
2.5 Segmentasi Citra.....	7
2.6 Automatic Cropping Tray Box Image into Compartments.....	7
2.7 Clustering.....	8
2.8 K-Means.....	8
2.9 Head & Tail.....	9
2.10 Improved K-Means.....	10
2.11 Intersection Over Union.....	10
2.12 Elementary leftover Food Estimation.....	10
2.13 Evaluasi.....	11
BAB 3 METODOLOGI PENELITIAN.....	12
3.1 Tipe Penelitian.....	12

3.2 Lokasi Penelitian	12
3.3 Subjek Penelitian	12
3.4 Metode Pengumpulan Data.....	12
3.5 Strategi Penelitian.....	13
3.6 Spesifikasi Piranti Keras dan Piranti Lunak.....	14
3.6.1 Piranti Keras	14
3.6.2 Piranti Lunak.....	15
BAB 4 Perancangan	16
4.1 Perancangan Algoritme	16
4.1.1 <i>Preprocessing</i> citra	17
4.1.2 Segmentasi citra dengan <i>Improved K-Means</i>	18
4.1.3 Estimasi bobot makanan.....	25
4.2 Perancangan Pengujian	27
4.3 Perhitungan Manual	28
BAB 5 PEMBAHASAN.....	33
5.1 Implementasi <i>preprocessing</i> citra.....	33
5.2 Implementasi segmentasi dengan <i>Improved K-Means</i>	34
5.2.1 Implementasi menentukan jumlah <i>K cluster</i>	34
5.2.2 Implementasi menentukan <i>centroids</i>	34
5.2.3 Implementasi hitung jarak antara titik data dengan titik pusat.....	35
5.2.4 Implementasi hitung <i>centroids</i> baru.....	36
5.2.5 Implementasi kondisi proses berhenti.....	36
5.3 Implementasi estimasi bobot makanan	37
BAB 6 Pengujian dan analisis	39
6.1 Pengujian hasil segmentasi citra.....	39
6.2 Pengujian hasil estimasi bobot sisa makanan	42
BAB 7 PENUTUP	46
7.1 Kesimpulan.....	46
7.2 Saran	46
DAFTAR REFERENSI	47
Lampiran-Lampiran.....	49

DAFTAR TABEL

Tabel 4.1 Contoh hasil pengujian segmentasi citra menggunakan <i>IoU</i>	27
Tabel 4.2 Contoh pengujian estimasi bobot makanan menggunakan <i>EFLE</i>	28
Tabel 4.3 Contoh data penelitian.....	28
Tabel 4.4 Tahap mengurutkan data dari kecil ke besar.....	29
Tabel 4.5 Tahap pembagian data untuk $K = 1$	29
Tabel 4.6 Tahap pembagian data untuk $K = 2$	30
Tabel 4.7 Tahap mengambil 5 data pertama untuk $K = 1$	30
Tabel 4.8 Tahap mengambil 5 data terakhir untuk $K = 2$	31
Tabel 4.9 Tahap melakukan rata-rata.....	31
Tabel 4.10 Hasil perhitungan jarak dan penentuan <i>cluster</i>	31
Tabel 4.11 Hasil <i>centroids</i> baru.....	32
Tabel 6.1 Akurasi hasil segmentasi (<i>IKM</i>) terhadap <i>groundtruth</i>	40
Tabel 6.2 Akurasi hasil segmentasi (<i>IKM5</i>) terhadap <i>groundtruth</i>	41
Tabel 6.3 <i>RMSE</i> estimasi bobot sisa makanan pada <i>IKM</i>	42
Tabel 6.4 <i>RMSE</i> estimasi bobot sisa makanan pada <i>IKM5</i>	43
Tabel 6.5 Perbedaan <i>RMSE</i> antara <i>IKM</i> dengan <i>IKM5</i>	44

DAFTAR GAMBAR

Gambar 2.1 Diagram kromasitas <i>CIE</i>	5
Gambar 2.2 Ilustrasi <i>Median Filter</i>	7
Gambar 2.3 Pemisahan objek koin dengan latar belakang	7
Gambar 2.4 Contoh <i>automatic cropping</i> menggunakan <i>benchmark</i>	8
Gambar 2.5 Contoh <i>clustering</i> data	8
Gambar 2.6 <i>Ground-truth bounding box</i> dan <i>predicted bounding box</i>	10
Gambar 2.7 Penerapan <i>Clustering</i> pada segmentasi citra.....	11
Gambar 3.1 Contoh <i>dataset</i>	12
Gambar 3.2 Strategi penelitian	13
Gambar 4.1 Diagram alir perancangan algoritme	16
Gambar 4.2 Diagram alir <i>preprocessing</i> citra	17
Gambar 4.3 Diagram alir segmentasi dengan <i>Improved K-Means</i>	18
Gambar 4.4 Diagram alir inisialisasi <i>centroids</i>	19
Gambar 4.5 Diagram alir hitung jarak titik data dengan <i>centroids</i>	21
Gambar 4.6 Diagram alir tandai setiap titik data yang memiliki jarak terdekat dengan <i>centroids</i>	22
Gambar 4.7 Diagram alir hitung <i>centroids</i> baru	23
Gambar 4.8 Diagram alir <i>masking</i> citra	24
Gambar 4.9 Diagram alir penambahan dimensi	25
Gambar 4.10 Diagram alir estimasi bobot makanan	26
Gambar 6.1 <i>Dataset</i> citra makanan	39
Gambar 6.2 Contoh citra dan <i>groundtruth</i>	39
Gambar 6.3 Contoh ilustrasi pengurutan data	42
Gambar 6.4 Contoh masalah dari hasil segmentasi citra	43

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Setiap makhluk hidup membutuhkan makanan untuk hidup, tumbuh, dan bereproduksi. Makanan merupakan suatu kebutuhan pokok yang diperlukan oleh manusia sebagai sumber energi untuk melakukan aktivitas dalam kehidupan sehari-hari. Setiap orang seharusnya mendapat makanan yang dibutuhkan untuk hidup sehat dan aktif karena makanan juga termasuk hak asasi manusia yang paling dasar. Makanan mengandung berbagai macam nutrisi seperti karbohidrat, lemak, protein, vitamin, mineral yang berperan dalam menjaga kesehatan tubuh. Dengan memakan makanan yang bergizi dapat membawa seseorang ke dalam pola makan sehat. Pola makan sehat adalah dengan mengonsumsi jumlah makanan yang berimbang dari beberapa jenis makanan yang diperlukan oleh tubuh. Seseorang dengan pola makan yang tidak sehat dapat terdampak penyakit seperti gizi buruk atau bahkan kematian, hal ini dapat terjadi ketika manusia tidak memakan buah, sayuran, dan kacang-kacangan yang cukup, dan sebaliknya memakan terlalu banyak makanan dengan gula, lemak, dan garam yang banyak (FAO, 2019). Studi di beberapa negara menunjukkan bahwa anak bergizi baik memiliki hasil tes mental yang lebih baik dibandingkan dengan anak yang pernah menderita gizi salah. Gizi salah merupakan suatu kondisi dengan seseorang memiliki gizi kurang maupun lebih (Anisa *et al.*, 2017). Pada penelitian yang ditulis oleh (Mardalena and Suyani, 2016), masalah gizi kurang sudah ada sejak negara Indonesia berdiri, walaupun pembangunan disertai perbaikan distribusi pangan, perbaikan ekonomi, dan peningkatan daya beli masyarakat telah banyak memperbaiki keadaan gizi masyarakat, namun masalah gizi hingga sekarang masih tetap ada walaupun jumlahnya sudah berkurang. Mengingat pentingnya mengatur jumlah konsumsi makanan untuk memenuhi dan menjaga kesehatan tubuh maka dibutuhkan suatu sistem yang dapat melakukan estimasi bobot dari makanan yang telah dikonsumsi. Untuk melakukan estimasi bobot makanan yang telah dikonsumsi oleh seseorang diperlukan citra dari makanan sebelum dan sesudah dikonsumsi.

Adapun penelitian terdahulu yang telah dilakukan sebelumnya sebagai pembelajaran dalam penelitian ini. Penelitian yang dilakukan oleh (Sari *et al.*, 2019) menggunakan metode segmentasi citra untuk melakukan estimasi makanan menunjukkan hasil bahwa *tray box* dengan *background* hitam dengan konstanta (cd) terbaik untuk memproyeksikan sisa adalah 2,37 *Root Mean Square Error (RMSE)*. Penelitian sebelumnya seperti melakukan identifikasi manual buah yang cacat menggunakan metode *K-Means clustering* menunjukkan hasil percobaan meningkatkan kualitas segmentasi cacat dalam aspek presisi dan waktu komputasi (Dubey *et al.*, 2013). Pada penelitian lain, melakukan peningkatan metode *K-Means clustering* pada segmentasi citra untuk efisiensi waktu dan performa menghasilkan algoritme yang diusulkan memiliki kinerja lebih baik dibandingkan algoritme *K-Means* tradisional dengan perbedaan waktu rata-rata 1,73 (Sona & Rajiv, 2016). Penelitian yang ditulis oleh (Sinaga, 2019)

menjelaskan ruang warna *RGB*, *CMYK*, dan *HSV* merupakan warna citra digital hasil intensitas cahaya sehingga perlu ditransformasi menjadi beberapa nilai *value* warna sebelum dilakukan segmentasi. Ruang warna *LAB* dipilih karena merupakan ruang homogen untuk persepsi visual. *K-Means* dipilih sebagai metode untuk segmentasi karena algoritme *K-Means* memiliki beberapa keunggulan seperti ketelitian yang cukup tinggi terhadap ukuran objek, tidak terpengaruh terhadap urutan objek, dan dapat menyesuaikan pembagian *cluster* berdasarkan kemiripan anggotanya (Ediyanto, Mara and Intisari, 2013).

Berdasarkan pemaparan tersebut, maka metode *Improved K-Means* dipilih untuk estimasi bobot sisa makanan berdasarkan hasil segmentasi citra makanan pada *tray box*. Penelitian ini diharapkan dapat memberikan hasil dengan tingkat akurasi yang tinggi dan tingkat kesalahan yang kecil pada estimasi sisa makanan pada *tray box*.

1.2 Rumusan Masalah

Dari pemaparan latar belakang, didapatkan rumusan permasalahan berikut:

1. Bagaimana pengaruh *Improved K-Means clustering* terhadap akurasi hasil segmentasi?
2. Bagaimana pengaruh *Improved K-Means clustering* terhadap hasil estimasi sisa makanan?

1.3 Tujuan

Berikut adalah tujuan yang ingin dicapai dalam penelitian ini:

1. Mengetahui pengaruh *Improved K-Means* terhadap akurasi hasil segmentasi citra makanan dengan *groundtruth*
2. Mengetahui pengaruh *Improved K-Means* terhadap hasil estimasi sisa makanan

1.4 Manfaat

Manfaat yang diharapkan dalam penelitian ini, antara lain:

1. Mampu meningkatkan kesadaran masyarakat dalam memperhatikan nilai gizi yang terkandung di dalam makanan
2. Mampu memberikan kontribusi dalam dunia pengetahuan
3. Dapat diimplementasikan terhadap kasus yang berkaitan dengan segmentasi citra dan estimasi bobot sisa makanan

1.5 Batasan Masalah

Adapun dilakukan pembatasan permasalahan untuk memfokuskan penelitian yang akan dilakukan:

1. Citra yang digunakan pada penelitian ini adalah citra makanan pada *tray box*
2. Citra yang digunakan berformat *JPG* dengan latar belakang putih
3. Citra memiliki empat kompartemen dengan jenis makanan yang berbeda

4. *Clustering Improved K-Means* menggunakan nilai $k=2$
5. Bobot makanan sebelum dimakan telah diketahui
6. Estimasi bobot sisa makanan dan segmentasi citra menggunakan metode *clustering Improved K-Means* dikembangkan menggunakan bahasa pemrograman python
7. Terdapat perbedaan dimensi antara citra dengan objek sesungguhnya

1.6 Sistematika Pembahasan

Sistematika pembahasan yang disusun dalam laporan ini adalah sebagai berikut:

BAB I PENDAHULUAN

Pendahuluan berisi tentang latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika pembahasan.

BAB II LANDASAN KEPUSTAKAAN

Landasan kepustakaan menjelaskan tentang kajian pustaka terkait penelitian terdahulu dan dasar teori yang diperlukan untuk mendukung penelitian ini.

BAB III METODOLOGI

Metodologi menjelaskan tentang lokasi penelitian, subjek dan metode pengumpulan data yang digunakan, dan metode penelitian apa saja yang akan digunakan.

BAB IV PERANCANGAN SISTEM

Perancangan sistem menjelaskan mengenai bagaimana perancangan dari algoritme yang digunakan beserta dengan perhitungan manual.

BAB V IMPLEMENTASI

Implementasi menjelaskan mengenai bagaimana implementasi algoritme *Improved K-Means* ke dalam aplikasi berdasarkan perancangan yang telah dibuat.

BAB VI PENGUJIAN DAN PEMBAHASAN

Pada bab ini akan dijelaskan mengenai hasil dari pengujian beserta analisisnya terhadap hasil segmentasi dan estimasi bobot sisa makanan menggunakan *clustering Improved K-Means* untuk menyelesaikan masalah pada penelitian ini.

BAB VII PENUTUP

Pada bab ini akan dijelaskan mengenai kesimpulan dan saran dari penelitian yang telah dilakukan.

BAB 2 LANDASAN KEPUSTAKAAN

Landasan kepastakaan berisi pembahasan mengenai kajian pustaka terkait penelitian terdahulu dan dasar teori yang mendukung pengerjaan penelitian ini.

2.1 Kajian Pustaka

Pada tahap ini akan dijelaskan tentang penelitian terdahulu yang mendukung pengerjaan penelitian ini. Kajian pustaka yang akan dibahas adalah sebagai berikut:

1. Penelitian yang dilakukan oleh (Sari *et al.*, 2019) yang membahas estimasi sisa makanan pada *tray box* menggunakan segmentasi citra.
2. Penelitian yang dilakukan oleh (Sari, Maligan and Prakoso, 2020) yang membahas estimasi sisa makanan dengan *Elementary leftover Food Estimation* menggunakan *clustering* segmentasi citra.
3. Penelitian yang dilakukan oleh (Dubey *et al.*, 2013) yang membahas perihal deteksi bagian buah yang terinfeksi menggunakan segmentasi *K-Means clustering*.
4. Penelitian yang dilakukan oleh (Sona & Rajiv, 2016) yang membahas mengenai efisiensi *Improved K-Means* pada segmentasi citra.

Pada penelitian yang dilakukan oleh (Sari *et al.*, 2019) bertujuan untuk melakukan estimasi sisa makanan pada *tray box* menggunakan segmentasi citra. Dari penelitian ini menunjukkan hasil bahwa *tray box* dengan *background* hitam dengan konstanta (cd) terbaik untuk memproyeksikan sisa adalah 2,37 *Root Mean Square Error (RMSE)*.

Pada penelitian yang dilakukan oleh (Sari, Maligan and Prakoso, 2020) bertujuan untuk menghitung estimasi bobot beserta nutrisi yang terkandung pada citra makanan. Dari penelitian ini menunjukkan hasil bahwa algoritme yang di usulkan memiliki akurasi *RMSE* 95.81% lebih baik dibandingkan algoritme *Thresholding*.

Pada penelitian yang dilakukan oleh (Dubey *et al.*, 2013) bertujuan untuk deteksi bagian buah yang terinfeksi menggunakan segmentasi *K-Means clustering*. Dari penelitian ini menunjukkan hasil bahwa Hasil percobaan meningkatkan kualitas segmentasi dalam aspek presisi dan waktu komputasi.

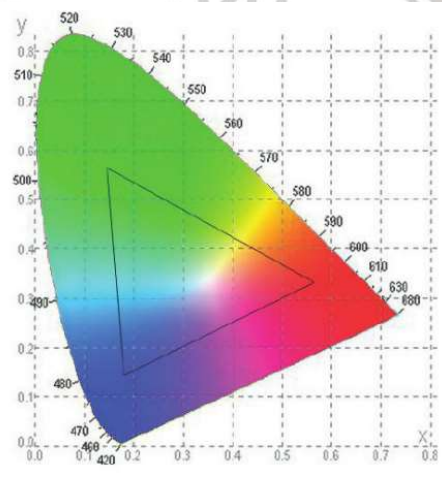
Pada penelitian yang dilakukan oleh (Sona & Rajiv, 2016) bertujuan untuk efisiensi *Improved K-Means* pada segmentasi citra. Dari penelitian ini menunjukkan hasil bahwa algoritme yang diusulkan memiliki kinerja lebih baik dibandingkan algoritme *K-Means* tradisional dengan perbedaan waktu rata-rata 1,73.

2.2 Tray box

Tray box merupakan jenis box untuk menyimpan bekal makanan berupa nasi dan lauk-pauk yang mudah dibawa dan dapat dimakan di tempat lain. *Tray box* terbuat dari bahan plastik bersifat ringan dan tahan lama. Untuk warna *tray box* memiliki 2 macam warna yaitu hitam dan abu-abu (Sari *et al.*, 2019).

2.3 Ruang Warna CIELAB

CIELAB adalah nama lain dari *CIE L*a*b**. Diagram kromasitas *Commission Internationale de L'Eclairage (CIE)* ditunjukkan pada Gambar 2.1. Setiap koordinat x dan y pada diagram tersebut menyatakan suatu warna. Warna pada area tapal kuda merepresentasikan warna yang dapat terlihat dan warna yang berada dalam area segitiga merupakan warna umum yang dihasilkan oleh komponen warna merah, hijau, dan biru. Angka pada tepi tapal kuda menyatakan panjang gelombang cahaya (Kadir and Susanto, 2012).



Gambar 2.1 Diagram kromasitas CIE

Sumber: (Kadir and Susanto, 2012)

Untuk mendapatkan nilai *LAB* digunakan rumus sebagai berikut:

Langkah pertama adalah melakukan perhitungan untuk transformasi *RGB* ke *CIELAB* dengan persamaan (2.1)-(2.7):

$$X = 0,412454R + 0,357580G + 0,180423B \quad (2.1)$$

$$Y = 0,212671R + 0,715160G + 0,072169B \quad (2.2)$$

$$Z = 0,019334R + 0,119193G + 0,950227B \quad (2.3)$$

Keterangan:

x, Y, Z : nilai koefisien dari RGB

Langkah kedua yaitu dengan mendefinisikan $L^*a^*b^*$:

$$L^* = 116f\left(\frac{Y}{Y_n}\right) - 16 \quad (2.4)$$

$$a^* = 500 \left[f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right] \quad (2.5)$$

$$b^* = 200 \left[f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right) \right] \quad (2.6)$$

Keterangan:

L^*, a^*, b^* : komponen warna CIELAB

$f\left(\frac{X}{X_n}\right)$: fungsi komponen warna R

$f\left(\frac{Y}{Y_n}\right)$: fungsi komponen warna G

$f\left(\frac{Z}{Z_n}\right)$: fungsi komponen warna B

X_n, Y_n, Z_n : nilai ketika $R=G=B=1$ dengan skala R,G,B adalah $[0,1]$

Selanjutnya adalah menghitung $f(q)$:

$$f(q) = \begin{cases} q^{1/3}, & \text{jika } q > 0,008856 \\ 7,787q + \frac{16}{116}, & \text{untuk yang lain} \end{cases} \quad (2.7)$$

Keterangan:

$f(q)$: fungsi hasil

2.4 Median Filter

Median Filter adalah metode yang digunakan untuk menghilangkan derau bintik-bintik pada citra. Nilai suatu piksel ditentukan berdasarkan nilai *median* setelah nilai piksel terurut dari rendah ke tinggi (Kadir and Susanto, 2012). Pada Gambar 2.2 ditunjukkan ilustrasi dari penggunaan *Median Filter* pada matriks dengan 8-ketetanggaan.

3	2	7
6	0	9
7	8	5

⇒ Matriks 3x3

3	2	7	6	0	9	7	8	5
---	---	---	---	---	---	---	---	---

⇒ Sebelum terurut

0	2	3	5	6	7	7	8	9
---	---	---	---	---	---	---	---	---

⇒ Sesudah terurut

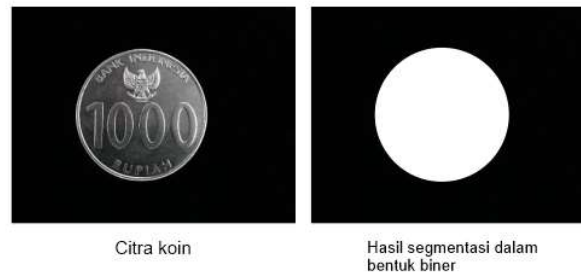
3	2	7
6	6	9
7	8	5

⇒ Hasil nilai tengah baru dengan nilai median

Gambar 2.2 Ilustrasi *Median Filter*

2.5 Segmentasi Citra

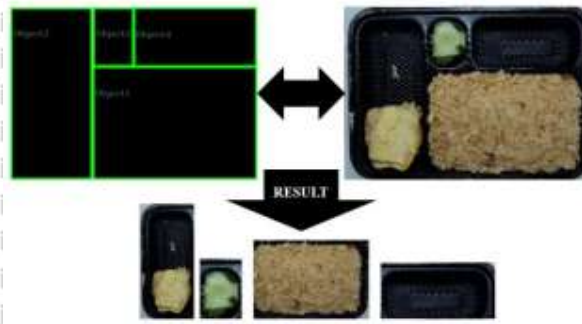
Segmentasi citra adalah suatu proses untuk mendapatkan objek-objek yang terdapat pada suatu citra (Kadir and Susanto, 2012). Segmentasi citra bertujuan untuk memisahkan antara objek dengan *background*. Proses segmentasi dilakukan dengan mengelompokkan piksel-piksel bertetangga berdasarkan kemiripan nilai piksel. Pada umumnya segmentasi citra berupa citra biner. Pada Gambar 2.3 ditunjukkan contoh dari segmentasi pada citra koin.



Gambar 2.3 Pemisahan objek koin dengan latar belakang

2.6 Automatic Cropping Tray Box Image into Compartments

Automatic Cropping Tray Box Image into Compartments merupakan proses memotong citra *tray box* menjadi beberapa kompartemen secara otomatis. Pemotongan citra digunakan untuk meningkatkan kualitas dari hasil segmentasi citra dan untuk memprediksi nilai rasio piksel antara objek yang terdeteksi pada seluruh area kompartemen. Untuk dapat melakukan *crop* secara otomatis dibutuhkan *benchmark*. Citra *benchmark* dibuat secara manual dengan menggunakan garis berbentuk kotak yang menentukan posisi koordinat x dan y dan mengembalikan nilai lebar dan ukuran setiap area persegi panjang yang terdeteksi (Sari, Maligan and Bihanda, 2020). Contoh dari *automatic crop* menggunakan *benchmark* ditunjukkan pada Gambar 2.4.

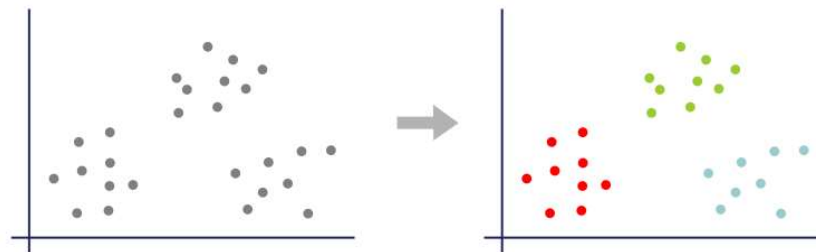


Gambar 2.4 Contoh *automatic cropping* menggunakan *benchmark*

Sumber: (Sari, Maligan and Bihanda, 2020)

2.7 Clustering

Clustering merupakan salah satu proses untuk mengelompokkan data menjadi dua atau lebih subkelompok, sedemikian hingga titik data subkelompok memiliki kesamaan ciri yang sangat mirip dan subkelompok yang berbeda memiliki kesamaan ciri yang beda (Gustientiedina, Adiya and Desnelita, 2019). Pada Gambar 2.5 ditunjukkan contoh dari *clustering* data.



Gambar 2.5 Contoh *clustering* data

2.8 K-Means

K-Means merupakan salah satu metode pengelompokan yang mempartisi data menjadi dua kelompok atau lebih (*cluster*) dan setiap titik data hanya dimiliki oleh satu kelompok. Berikut adalah langkah-langkah dari algoritme *K-Means* (Wakhidah, 2010):

- Menentukan jumlah *K cluster*
- Melakukan inisialisasi *centroids* secara acak
- Menghitung jarak titik data dengan *centroids*

Untuk menghitung jarak menggunakan *Euclidean distance* pada persamaan (2.8):

$$D_e = \sqrt{\sum_{j=1}^J (x_j - y_j)^2} \quad (2.8)$$

Keterangan:

D_e : *Euclidean distance*

J : jumlah data

j : indeks dari objek

x : titik data

y : *centroids*

- Tandai setiap titik data yang memiliki jarak terdekat dengan *centroids* menggunakan persamaan (2.9):

$$w_{ik} = \begin{cases} 1, & d = \min((x_j, y_j)) \\ 0, & \text{lainnya} \end{cases} \quad (2.9)$$

Keterangan:

w_{ik} : pengelompokan data dalam bentuk biner *cluster* ke- i untuk variabel ke- j

d : jarak

$\min((x_j, y_j))$: jarak terpendek

- Hitung *centroids* dengan mengambil rata-rata dari semua titik data yang dimiliki setiap *cluster* dapat dilihat pada persamaan (2.10):

$$v_{ij} = \frac{1}{N_i} \sum_{k=1}^{N_i} x_{kj} \quad (2.10)$$

Keterangan:

v_{ij} : centroid atau rata-rata *cluster* ke- i untuk variable ke- j

N_i : jumlah data yang menjadi anggota *cluster* ke- i

i, k : indeks dari *cluster*

j : indeks dari variabel

Kembali ke langkah 3, sampai telah mencapai iterasi maksimum atau nilai *centroids* yang dihasilkan tetap.

2.9 Head & Tail

Head & Tail merupakan sebuah fungsi pada *pandas* yang digunakan untuk mengembalikan nilai pada sebuah *dataframe* secara default 5 data pertama dan 5 data terakhir (*pandas.DataFrame.head* — *pandas 1.3.0 documentation*, no date). Fungsi ini digunakan untuk melihat rupa dari seluruh data secara cepat dan tepat.

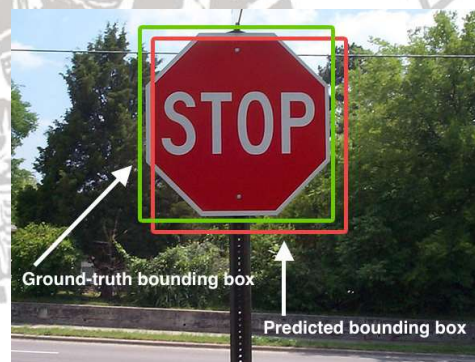
2.10 Improved K-Means

Improved K-Means merupakan salah satu metode *clustering* paling populer yang digunakan untuk meningkatkan performa dari metode *K-Means* tradisional. *Improved K-Means* bertujuan untuk mengubah proses inisialisasi *centroids* yang pada algoritme *K-Means* dilakukan secara acak menjadi suatu nilai tetap. Berikut langkah dari algoritme *Improved K-Means* (Sona & Rajiv, 2016):

- Urutkan data secara *ascending*
- Bagi data yang sudah terurut menjadi K bagian
- Hitung rata-rata setiap bagian dan jadikan hasil tersebut sebagai *centroids*

2.11 Intersection Over Union

Intersection Over Union adalah matrik evaluasi sederhana yang digunakan untuk mengukur keakuratan objek detektor pada kumpulan data tertentu. Untuk melakukan evaluasi objek detector menggunakan *Intersection Over Union* membutuhkan dua *bounding box* yaitu *ground-truth bounding box* dan *predicted bounding box*. *Ground-truth* diperoleh dengan memberi label pada sebuah objek citra, sedangkan *predicted box* diperoleh dari hasil pemodelan (Rosebrock, 2016). Contoh *Intersection Over Union* ditunjukkan pada Gambar 2.6.



Gambar 2.6 Ground-truth bounding box dan predicted bounding box

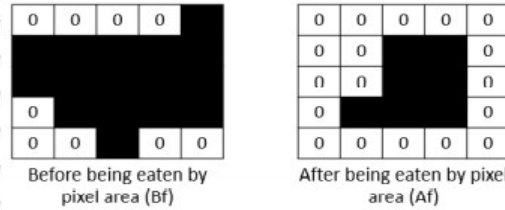
Sumber: (Rosebrock, 2016)

Untuk melakukan perhitungan dengan *Intersection Over Union* dapat dilihat pada persamaan (2.11) (Maulana, Bihanda and Sari, 2020):

$$IoU = \frac{\text{Intersection}}{\text{Union}} \times 100\% \quad (2.11)$$

2.12 Elementary leftover Food Estimation

Elementary leftover Food Estimation (EFLE) merupakan algoritme yang digunakan untuk menghitung bobot sisa makanan pada *tray box* setelah dimakan berdasarkan data bobot dari makanan sebelum dimakan. Pada Gambar 2.7 mengilustrasikan piksel berwarna hitam merupakan area makanan pada segmentasi citra (Sari, Maligan and Prakoso, 2020).



Gambar 2.7 Penerapan Clustering pada segmentasi citra

Sumber: (Sari, Maligan and Prakoso, 2020)

Untuk memperoleh bobot dari sisa makanan dengan *EFLE* dapat dilihat pada persamaan (2.12) (Sari, Maligan and Prakoso, 2020):

$$EFLE = \frac{OBf}{Bf} \times Af \quad (2.12)$$

Dimana:

OBf : berat sesungguhnya dari makanan sebelum dimakan

Bf : area piksel dari makanan sebelum dimakan

Af : area piksel dari makanan sesudah dimakan

2.13 Evaluasi

Untuk memvalidasi antara berat yang sesungguhnya dengan berat estimasi maka dibutuhkan suatu evaluasi. Pada tahap ini akan menggunakan *Root Mean Square Error (RMSE)* sebagai evaluasi. *RMSE* digunakan untuk menentukan tingkat kesalahan dengan satuan ukuran yang sama. Untuk menghitung *Mean Square Error (MSE)* dan *RMSE* ditunjukkan pada Persamaan (2.13) dan Persamaan (2.14) (Sari, Maligan and Prakoso, 2020).

$$MSE = \sum_{k=1}^n (Oaf_k - EFLE)^2 / n \quad (2.13)$$

Dimana:

Oaf : berat sesungguhnya

EFLE : berat estimasi

k : kompartemen pada *tray box*

n : jumlah kompartemen *tray box*

$$RMSE = \sqrt{MSE} \quad (2.14)$$

Perhitungan *RMSE* diterapkan pada *EFLE*. Untuk menghitung besar perbedaan *RMSE* dijelaskan pada Persamaan (2.15). *Diff* merepresentasikan perbedaan *RMSE* antara *EFLE* pada metode segmentasi citra pertama dengan metode segmentasi citra kedua. *RMSE₁* merepresentasikan *RMSE* dari metode segmentasi citra pertama dan *RMSE₂* merepresentasikan *RMSE* dari metode segmentasi citra kedua (Sari, Maligan and Prakoso, 2020).

$$Diff = \frac{RMSE_1 - RMSE_2}{RMSE_1} \quad (2.15)$$

BAB 3 METODOLOGI PENELITIAN

Pada metodologi penelitian akan dijelaskan tentang tipe penelitian, lokasi penelitian, subjek dan metode pengumpulan data yang digunakan, dan strategi penelitian yang digunakan.

3.1 Tipe Penelitian

Tipe penelitian ini adalah non-implementatif analitik. Tipe ini bertujuan untuk menggali informasi dengan mengimplementasikan pengembangan piranti lunak atau piranti keras untuk menghasilkan sebuah keluaran yang dapat dimanfaatkan dalam proses analisis terhadap suatu permasalahan.

3.2 Lokasi Penelitian

Lokasi penelitian segmentasi citra makanan menggunakan *clustering Improved K-Means* untuk estimasi sisa makanan ini bertempat di Fakultas Ilmu Komputer Universitas Brawijaya.

3.3 Subjek Penelitian

Subjek penelitian dapat berupa orang, benda, hal ataupun tempat yang diamati sebagai sumber informasi. Pada penelitian ini subjek yang digunakan adalah makanan pada *tray box* yang diambil citra nya kemudian di proses lebih lanjut.

3.4 Metode Pengumpulan Data

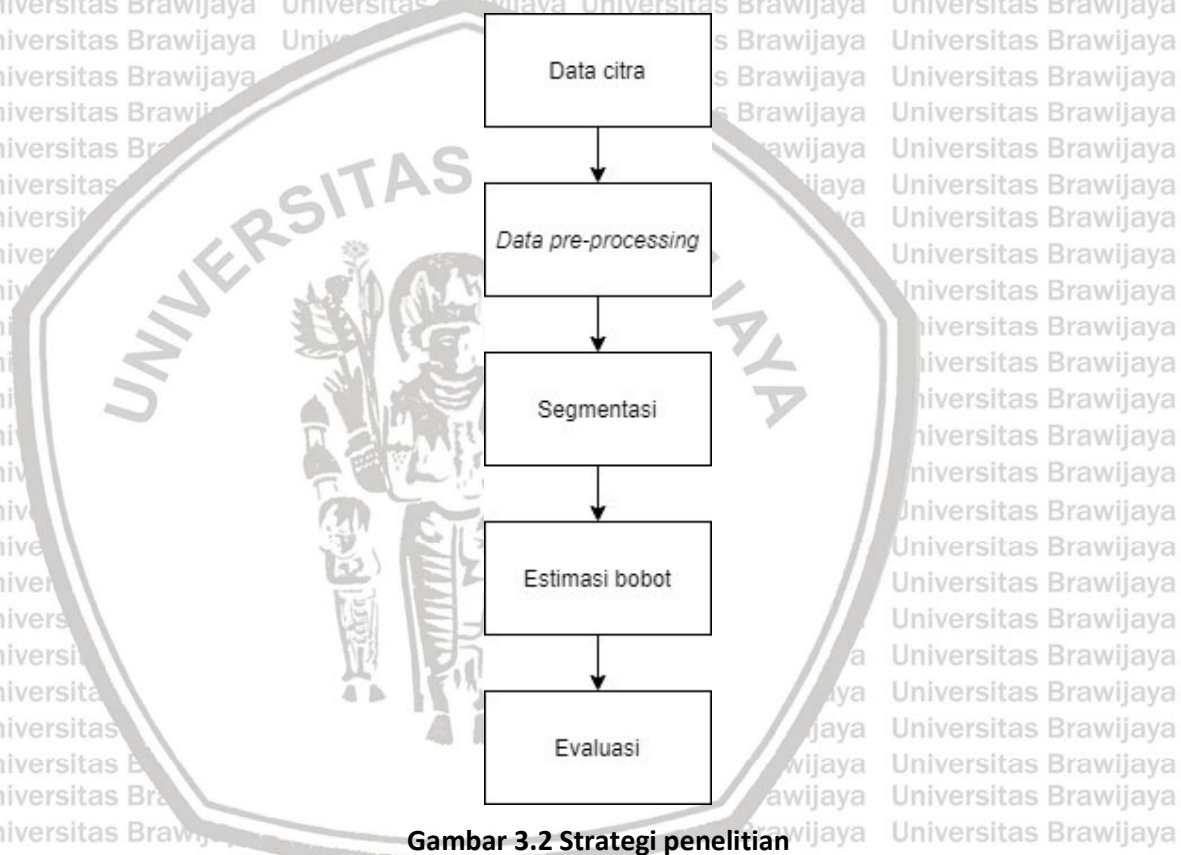
Data yang digunakan pada penelitian ini adalah data sekunder berupa makanan sebelum dan sesudah dimakan yang diperoleh dari dosen pembimbing. Dari data tersebut terdapat 4 kompartemen yang masing-masing kompartemen berisi jenis makanan yang berbeda yaitu nasi goreng, telur, dan mentimun. Total jumlah data yang didapatkan adalah sebanyak 31 data citra *tray box* dengan 124 kompartemen. *Dataset* dibuat dengan posisi kamera terletak di atas tengah *tray box* dengan jarak 30 cm dari atas. Sehingga didapatkan *dataset* dengan posisi yang sama dengan skala *landscape*. Sisa makanan pada *dataset* memiliki bentuk yang terarur karena *dataset* dibuat melalui pendekatan metode taksiran visual (*Comstock*). Contoh gambar dari *dataset* makanan sebelum dan sesudah dimakan yang akan digunakan dalam penelitian ini ditunjukkan pada Gambar 3.1.



Gambar 3.1 Contoh *dataset*

3.5 Strategi Penelitian

Implementasi dari penelitian ini adalah dengan pengembangan sebuah sistem yang berdasar pada algoritme *Improved K-Means* untuk melakukan segmentasi makanan pada citra makanan pada *tray box* yang digunakan untuk estimasi bobot sisa makanan. Algoritme tersebut akan digunakan untuk keperluan *clustering* dalam melakukan pengelompokan nilai piksel menjadi objek makanan dan *background*. Strategi dibuat dengan tujuan untuk menggambarkan konsep dari sistem yang akan dibangun agar sesuai dengan kebutuhan. Strategi dalam penelitian ini ditunjukkan pada Gambar 3.2.



Gambar 3.2 Strategi penelitian

Berikut adalah penjelasan dari strategi penelitian:

1. Data citra

Data yang digunakan pada penelitian ini adalah data sekunder berupa makanan yang diperoleh dari dosen pembimbing. Pada data citra tersebut terdapat 4 kompartemen untuk setiap data citra. Setiap kompartemen dapat berisi jenis makanan seperti nasi goreng, telur, dan mentimun. Total jumlah data yang didapatkan adalah sebanyak 31 data citra *tray box* dengan 124 kompartemen.

2. Data preprocessing

Masukan data citra nantinya akan dilakukan proses perubahan ukuran citra (*resize*). Perubahan ukuran citra dilakukan untuk mengurangi atau mempercepat waktu komputasi. Setelah proses *resize* selesai, selanjutnya diikuti dengan beberapa proses lain seperti *Median Filter*, konversi ruang warna *RGB* ke *LAB*, menggunakan momen warna a^* dan b^* dan mengurangi jumlah dimensi data.

3. Segmentasi

Penelitian sebelumnya memodifikasi inialisasi *centroids* (*IKM*) untuk meningkatkan performa dari metode *K-Means* tradisional. Pada penelitian ini akan dilakukan segmentasi citra dengan menggunakan algoritme *Improved K-Means* yang dimodifikasi (*IKM5*). Pada penelitian ini akan digunakan nilai $k=2$ untuk membedakan antara objek makanan dengan latar belakang. Selanjutnya akan dipilih centroid atau titik pusat awal menggunakan algoritme *Improved K-Means* yang dimodifikasi (*IKM5*). Modifikasi dilakukan dengan tujuan untuk meningkatkan hasil segmentasi citra.

4. Estimasi bobot

Pada penelitian ini akan menggunakan algoritme *Elementary leftover Food Estimation (EFLE)* untuk melakukan estimasi bobot sisa makanan. Untuk menghitung estimasi bobot dibutuhkan citra makanan sebelum dan sesudah dimakan.

5. Evaluasi

Pada penelitian ini akan menggunakan *Root Mean Square Error (RMSE)* sebagai evaluasi. Untuk menghitung besar perbedaan *RMSE* menggunakan *Elementary leftover Food Estimation (EFLE)* dijelaskan pada Persamaan (2.15). $RMSE_1$ akan diterapkan pada hasil segmentasi citra berdasarkan algoritme *Improved K-Means (IKM)* dan $RMSE_2$ akan diterapkan pada hasil segmentasi citra berdasarkan modifikasi algoritme *Improved K-Means (IKM5)*.

3.6 Spesifikasi Piranti Keras dan Piranti Lunak

Adapun piranti keras atau lunak yang dibutuhkan untuk membantu menyelesaikan laporan ini. Berikut adalah piranti keras dan piranti lunak yang digunakan:

3.6.1 Piranti Keras

Dalam pengembangan dan pengujian, piranti keras yang digunakan adalah laptop dengan spesifikasi sebagai berikut:

- Merk*: Acer
- Type*: A315-41-R9D3
- Processor*: Ryzen 5 2500U with Radeon Vega Mobile Gfx 2,00 GHz
- RAM*: 8.00 GB

e. HDD: 1TB

f. OS: Windows 10

3.6.2 Piranti Lunak

Dalam melakukan pengembangan dan pengujian digunakannya piranti lunak dengan spesifikasi sebagai berikut:

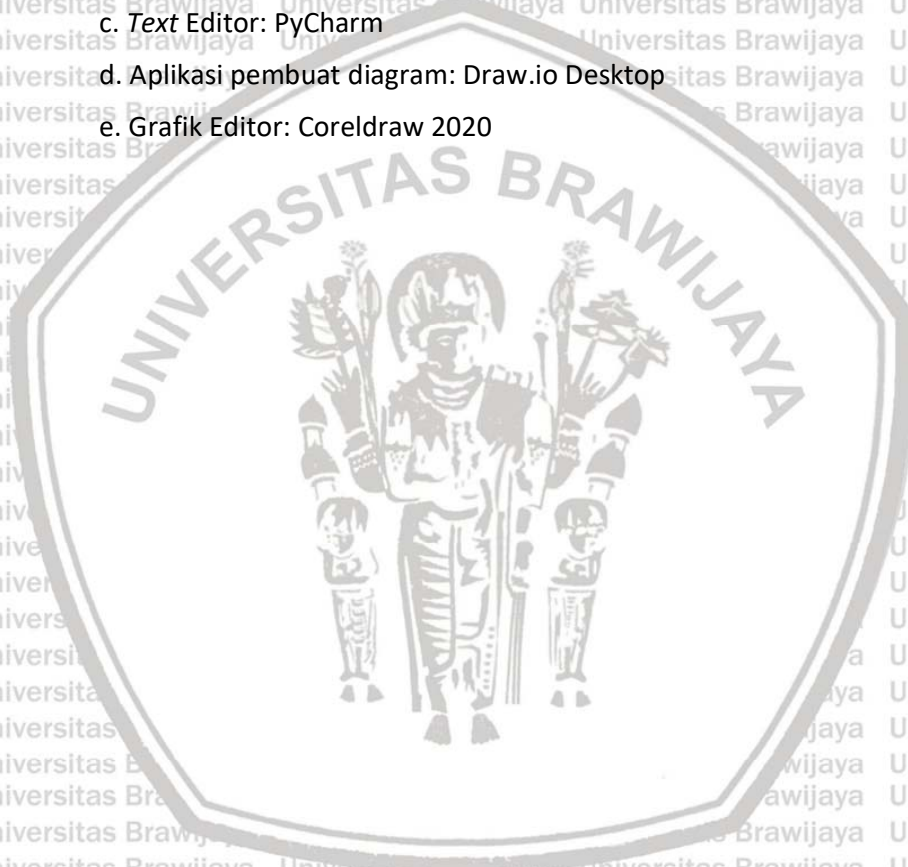
a. Sistem Operasi: Windows 10 Pro 64-bit

b. Bahasa pemrograman: Python 3.6

c. Text Editor: PyCharm

d. Aplikasi pembuat diagram: Draw.io Desktop

e. Grafik Editor: Coreldraw 2020

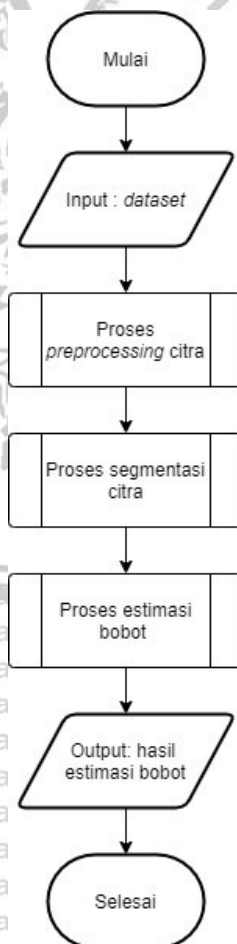


BAB 4 PERANCANGAN

Pada bab ini akan dijelaskan tentang perancangan algoritme, perancangan pengujian, dan perhitungan manual. Perancangan yang dihasilkan pada bab ini akan digunakan sebagai acuan dalam tahap implementasi segmentasi citra makanan menggunakan *clustering Improved K-Means* untuk estimasi bobot sisa makanan yang akan dijelaskan pada bab selanjutnya.

4.1 Perancangan Algoritme

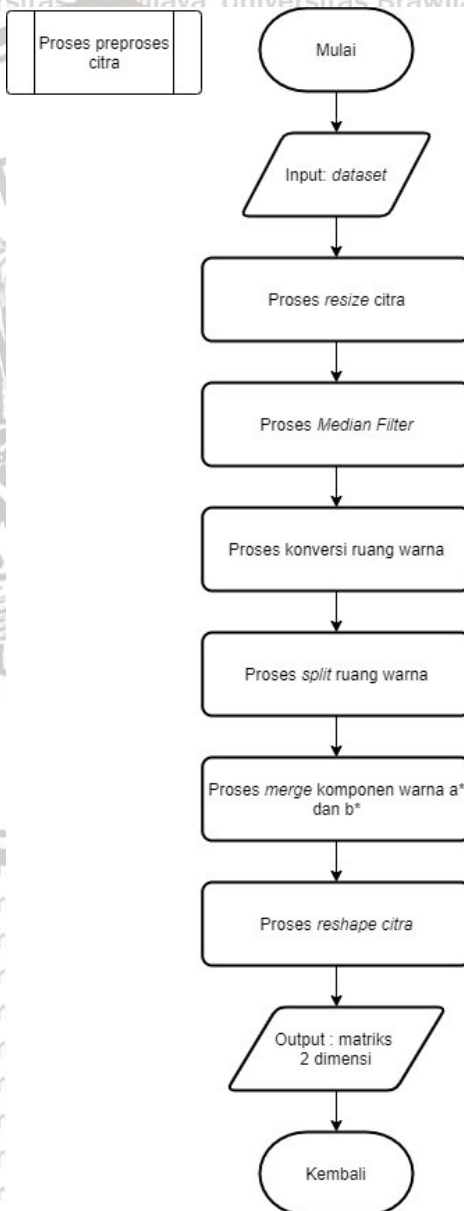
Perancangan algoritme menunjukkan bagaimana tahapan-tahapan dari penggunaan algoritme dalam melakukan estimasi bobot sisa makanan. Tahapan dimulai dari sistem menerima *input* berupa *dataset* citra makanan pada *tray box*, kemudian dilakukan *preprocessing* citra dan segmentasi citra menggunakan *Improved K-Means*, hingga dilakukan estimasi bobot makanan yang merupakan hasil *output* dari sistem. Pada Gambar 4.1 ditunjukkan perancangan algoritme pada penelitian ini.



Gambar 4.1 Diagram alir perancangan algoritme

4.1.1 Preprocessing citra

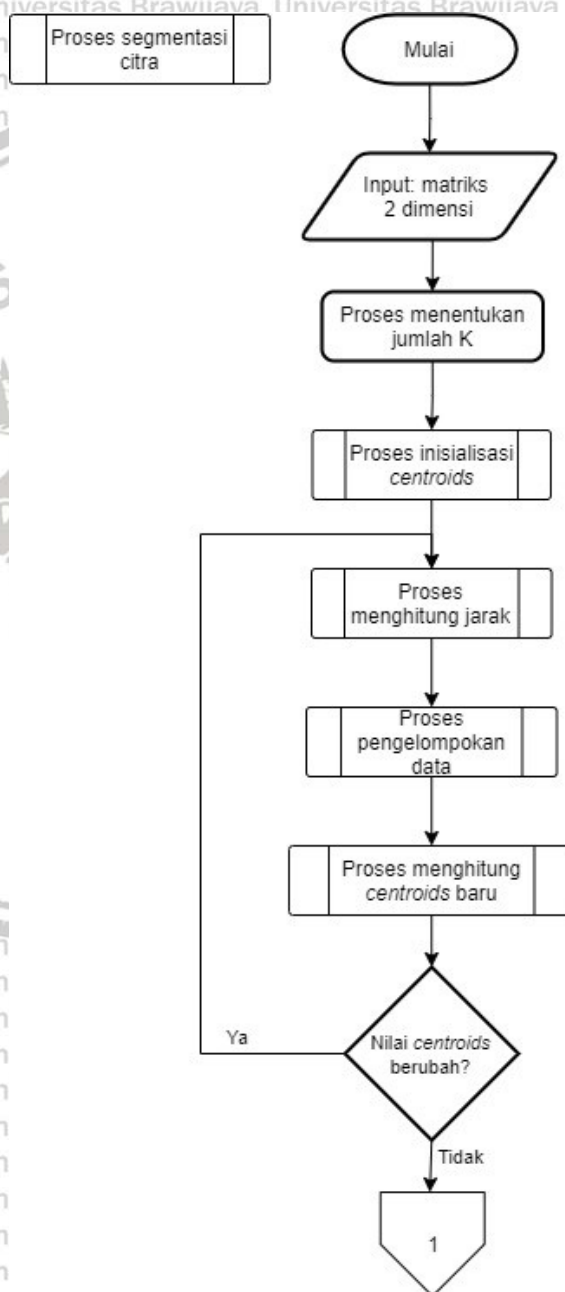
Preprocessing citra merupakan tahapan ketika *dataset* citra makanan pada *tray box* digunakan sebagai *input* dan akan menghasilkan *ouput* berupa matriks dengan ukuran 2 dimensi. Tahapan pada *preprocessing* citra meliputi memperkecil ukuran citra, *Median Filter* untuk menghilangkan *noise*, konversi *RGB* ke *LAB*, memecah warna *LAB*, menghapus komponen *L* untuk menghilangkan unsur pencahayaan dan mengurangi jumlah dimensi untuk mempermudah proses komputasi. Pada Gambar 4.2 ditunjukkan tahapan dari *preprocessing* citra.



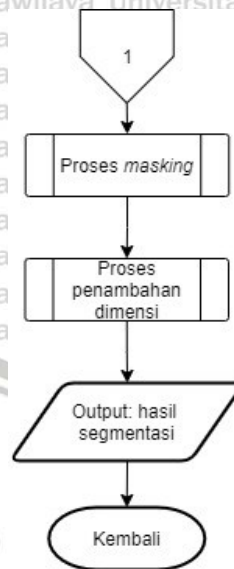
Gambar 4.2 Diagram alir *preprocessing* citra

4.1.2 Segmentasi citra dengan *Improved K-Means*

Gambar 4.3. menunjukkan diagram alir segmentasi citra dengan menggunakan metode *Improved K-Means*. Pada tahapan ini masukan berupa *dataset* citra makanan yang telah dilakukan *preprocessing* citra. Citra hasil *preprocessing* menghasilkan matriks citra 2 dimensi. Matriks tersebut di proses menggunakan *clustering Improved K-Means*, sehingga dihasilkan segmentasi citra berdasarkan *clustering*.

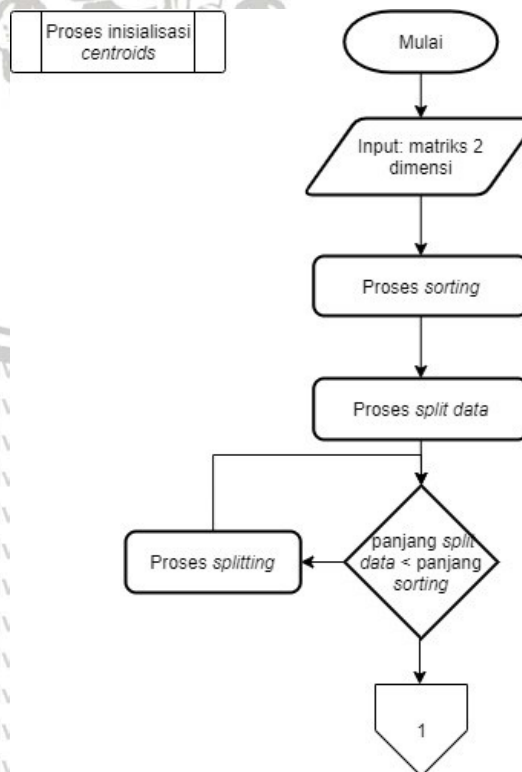


Gambar 4.3 Diagram alir segmentasi dengan *Improved K-Means*

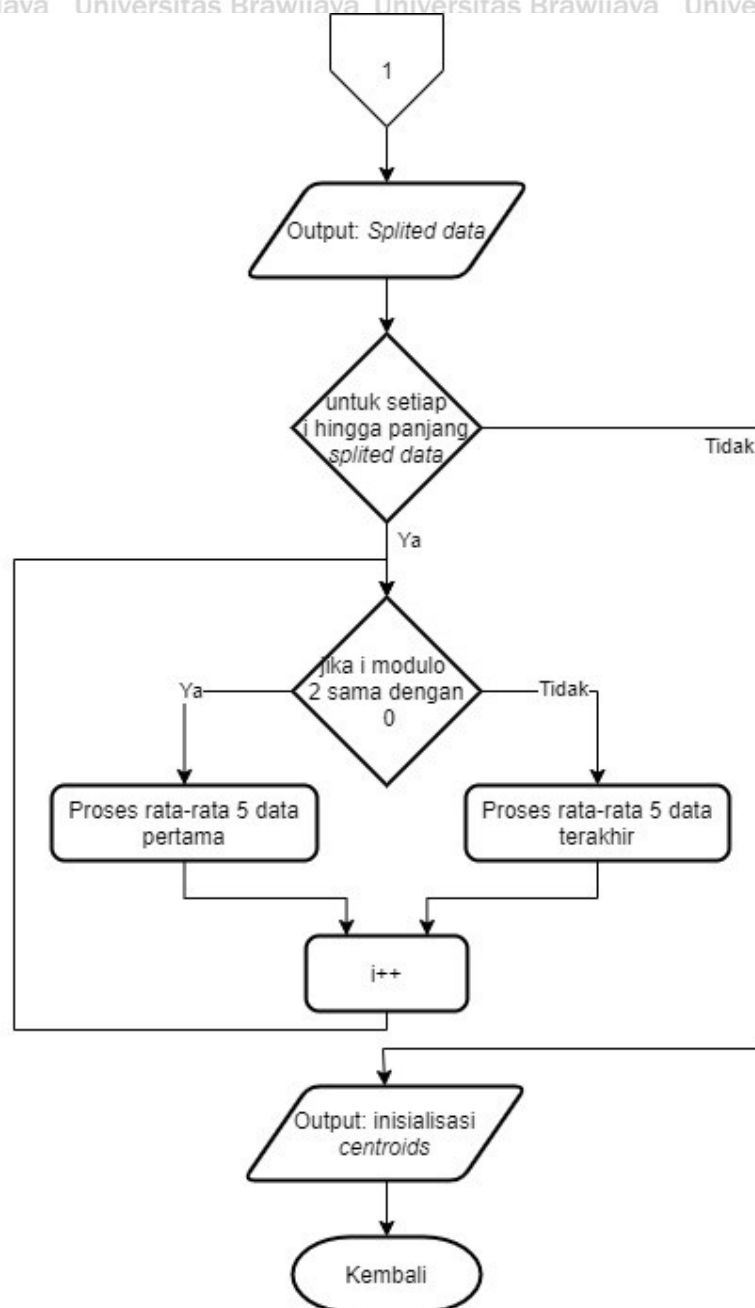


Gambar 4.3 Diagram alir segmentasi dengan *Improved K-Means* (lanjutan)

Gambar 4.4. menunjukkan diagram alir menentukan inisialisasi *centroids*. Pada tahapan ini masukan berupa matriks citra 2 dimensi. Matriks tersebut di proses dengan mengatur urutan data mulai dari terendah hingga yang tertinggi dan membagi data menjadi K bagian. Data yang telah dibagi kemudian diambil 5 data pertama dan 5 data terakhir untuk diambil rata-rata, sehingga dihasilkan inisialisasi *centroids* dengan nilai tetap.

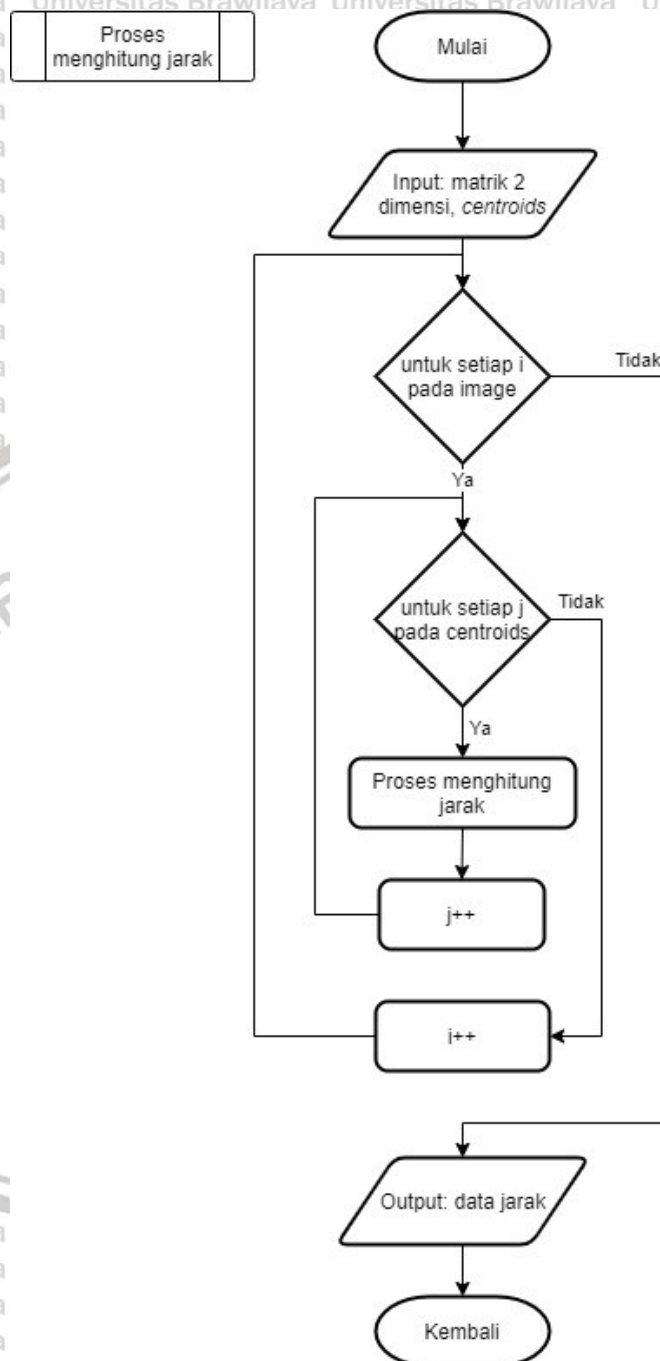


Gambar 4.4 Diagram alir inisialisasi *centroids*



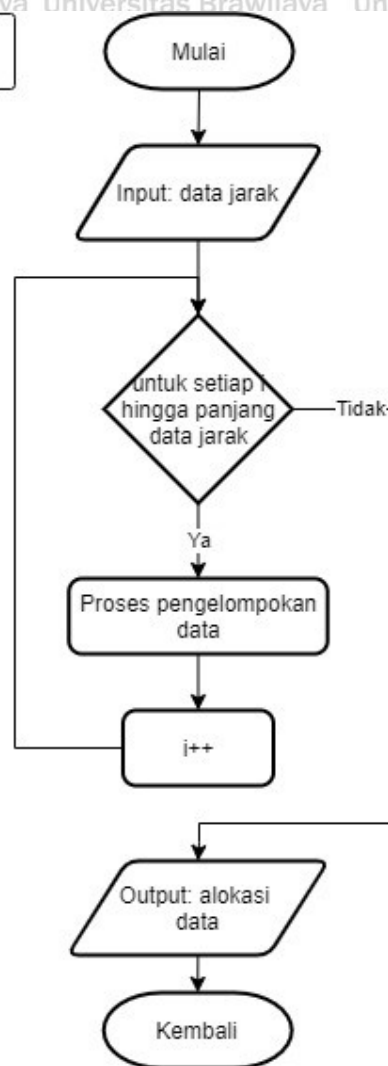
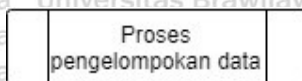
Gambar 4.4 Diagram alir inisialisasi *centroids* (lanjutan)

Gambar 4.5. menunjukkan diagram alir hitung jarak antara titik data dengan *centroids*. Untuk proses perhitungan jarak menggunakan *Euclidean distance* ditunjukkan pada persamaan (2.8) untuk menghasilkan jarak antar kedua data.



Gambar 4.5 Diagram alir hitung jarak titik data dengan centroids

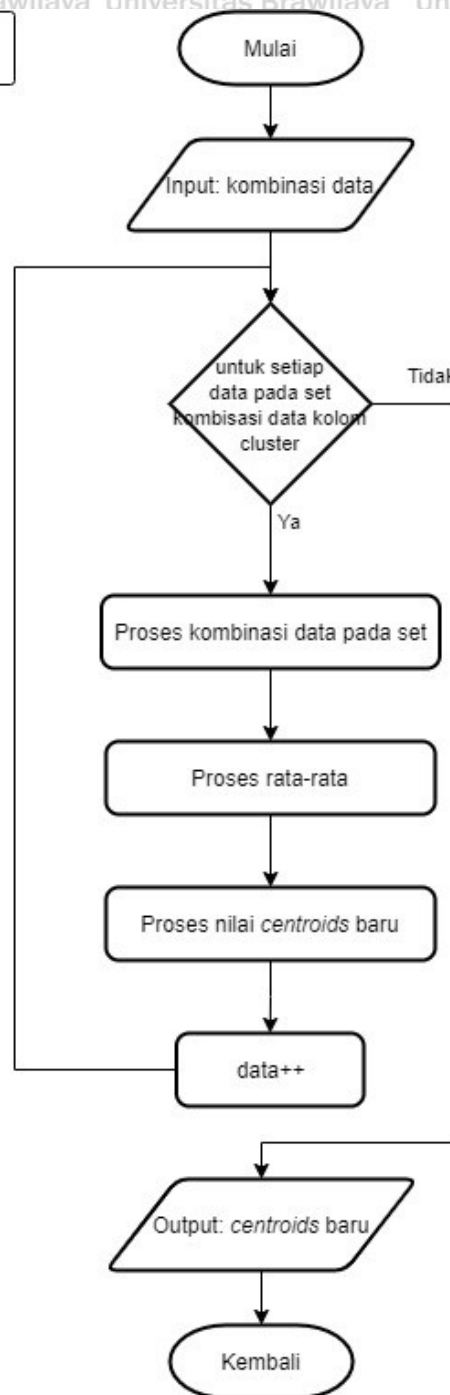
Gambar 4.6. menunjukkan diagram alir pengelompokan data yang dilakukan dengan menandai setiap titik data yang memiliki jarak terdekat dengan *centroids*. Pada tahapan ini masukan berupa data jarak antara titik data dengan *centroids*. Untuk setiap titik data yang memiliki jarak minimum pada *centroids* akan ditandai sebagai anggota dari sebuah *subcluster*. Pengelompokan data dilakukan dengan menggunakan persamaan (2.9) untuk menghasilkan alokasi data sesuai dengan *clustering*.



Gambar 4.6 Diagram alir tandai setiap titik data yang memiliki jarak terdekat dengan *centroids*

Gambar 4.7. menunjukkan diagram alir hitung *centroids* baru. Pada tahapan ini masukan berupa data gabungan antara matriks citra dua dimensi dengan alokasi data. Data gabungan digunakan untuk mempermudah dalam mencari data sesuai dengan alokasi atau pengelompokan data. Untuk mencari *centroids* baru menggunakan persamaan (2.10). Setiap data pada *subcluster* diambil rata-rata dan hasil operasi tersebut digunakan sebagai *centroids* yang baru.

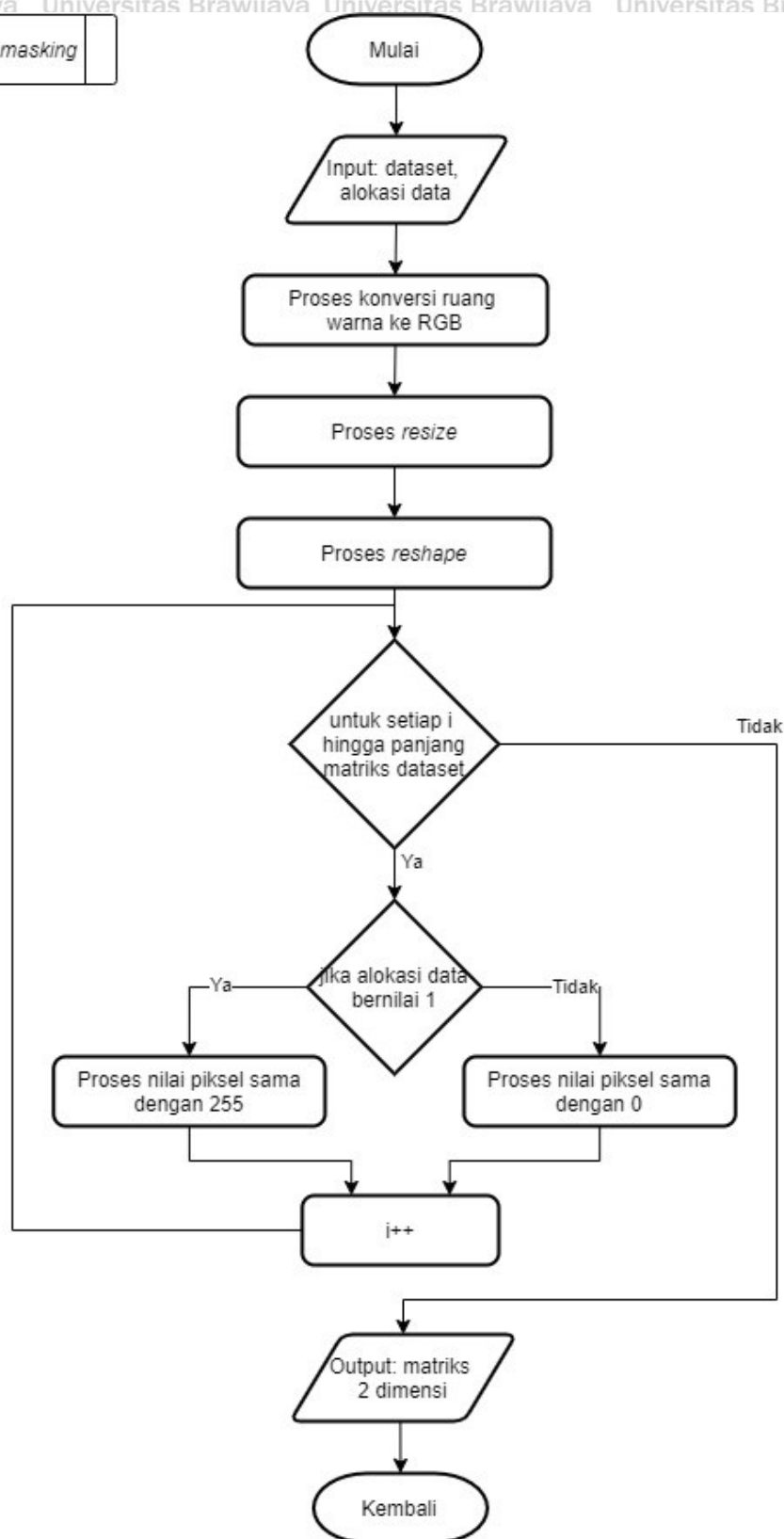
Proses hitung
centroids baru



Gambar 4.7 Diagram alir hitung centroids baru

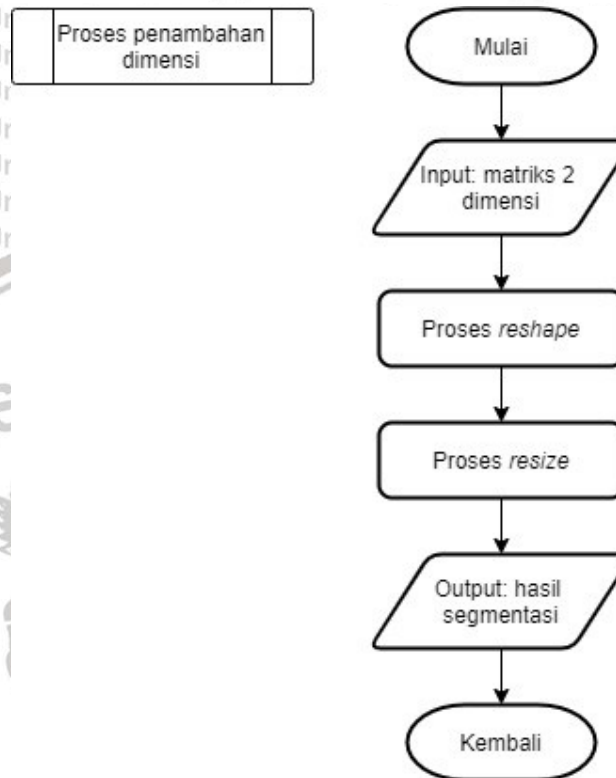
Gambar 4.8. menunjukkan diagram alir *masking* citra. Pada tahap ini masukan berupa citra makanan. Citra tersebut diubah menjadi matriks 2 dimensi untuk mempermudah proses komputasi. Nilai dari matriks tersebut kemudian dilakukan *masking* sehingga dihasilkan pola segmentasi berdasarkan *clustering*. *masking* merupakan suatu proses untuk menyembunyikan atau mengganti nilai objek sesungguhnya dengan suatu nilai yang telah ditetapkan.

Proses *masking*



Gambar 4.8 Diagram alir *masking* citra

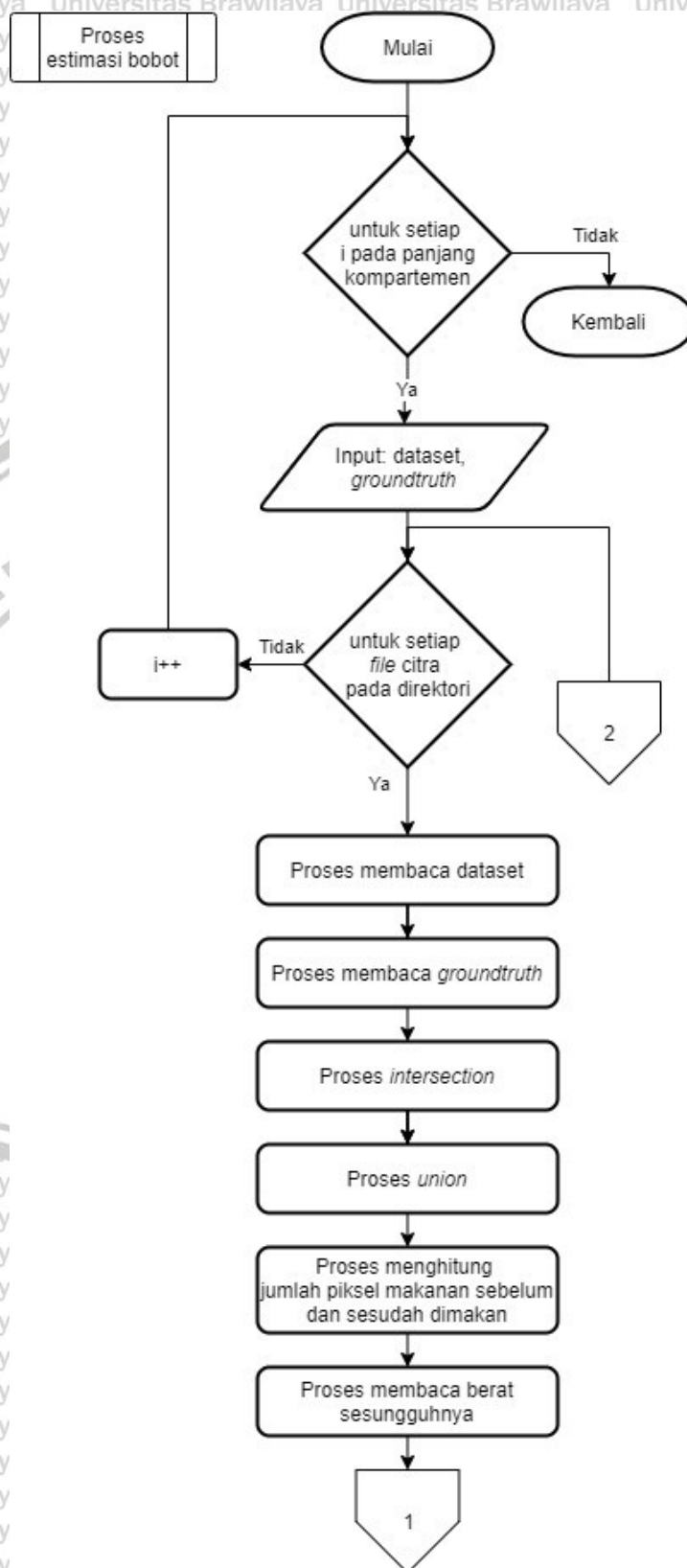
Gambar 4.9. menunjukkan diagram alir penambahan dimensi. Pada tahap ini masukkan berupa matriks citra 2 dimensi. Matriks tersebut dilakukan proses penambahan jumlah dimensi dan mengembalikan ukuran asli citra dengan tujuan agar ukuran data keluaran sama dengan ukuran data masukan awal.



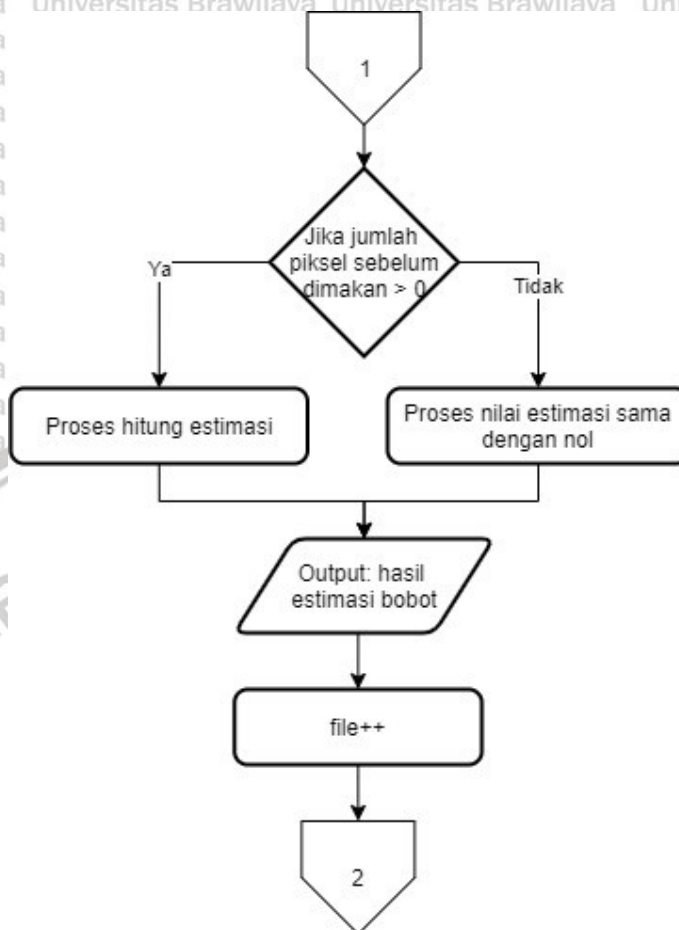
Gambar 4.9 Diagram alir penambahan dimensi

4.1.3 Estimasi bobot makanan

Estimasi bobot makanan dilakukan dengan menerima masukan berupa data citra kompartemen. Nilai estimasi didapati dengan cara bobot sesungguhnya dari makanan sebelum dimakan dibagi dengan jumlah piksel yang tidak bernilai 0 pada citra sebelum dimakan dikali dengan jumlah piksel yang tidak bernilai 0 pada citra sesudah dimakan. Diagram alir estimasi bobot makanan dapat dilihat pada Gambar 4.10.



Gambar 4.10 Diagram alir estimasi bobot makanan



Gambar 4.10 Diagram alir estimasi bobot makanan (lanjutan)

4.2 Perancangan Pengujian

Pada penelitian ini akan digunakan pengujian terkait algoritme yang digunakan. Pengujian yang dilakukan berupa pengujian hasil segmentasi citra dan pengujian estimasi bobot sisa makanan. Hasil pengujian akan dicatat ke dalam Tabel 4.1 berupa nilai akurasi dan Tabel 4.2 berupa bobot.

Tabel 4.1 Contoh hasil pengujian segmentasi citra menggunakan *IoU*

Nama citra	<i>IoU</i> kompartemen				<i>IoU</i> keseluruhan
	Nasi	Telur	Mentimun	Blank	
citra.jpg	...	...	...	...	...
...	...	...	...	...	...
...	...	...	...	...	...
citrn.jpg	...	...	...	...	...
Rata-rata	...	...	...	...	...

Tabel 4.2 Contoh pengujian estimasi bobot makanan menggunakan *EFLE*

Nama citra	Estimasi bobot kompartemen (gram)			
	Nasi	Telur	Mentimun	Blank
citra.jpg	...	...	...	...
...	...	...	...	...
citrان.jpg	...	...	...	...

4.3 Perhitungan Manual

Perhitungan manual segmentasi citra dengan *clustering Improved K-Means* dilakukan menggunakan contoh data penelitian dari matriks citra *RGB* yang ditunjukkan pada Tabel 4.3.

Tabel 4.3 Contoh data penelitian

RGB image pixel input		
R	G	B
104	107	90
108	109	93
112	113	97
115	116	102
117	115	102
112	114	103
113	115	104
115	116	108
116	115	110
117	117	109
116	119	112
117	119	116
117	119	118
121	121	123
119	119	119
118	120	119
120	121	123
120	121	126
123	122	130
121	120	126
119	120	122
121	122	127
122	122	132
121	121	133
122	120	133

Kemudian dilakukan proses inisialisasi *centroids* dengan tahapan sebagai berikut:

- a. Tahap pertama dalam inisialisasi *centroids* adalah dengan mengurutkan data dari nilai terendah hingga nilai tertinggi yang ditunjukkan pada Tabel 4.4.

Tabel 4.4 Tahap mengurutkan data dari kecil ke besar

data yang sudah terurut		
<i>R</i>	<i>G</i>	<i>B</i>
104	107	90
108	109	93
112	113	97
112	114	102
113	115	102
115	115	103
115	115	104
116	116	108
116	116	109
117	117	110
117	119	112
117	119	116
117	119	118
118	119	119
119	120	119
119	120	122
120	120	123
120	120	123
121	121	126
121	121	126
121	121	127
121	121	130
122	122	132
122	122	133
123	122	133

- b. Setelah data terurut, data dibagi menjadi *K* bagian. Pada penelitian ini nilai *K* yang digunakan adalah 2. Tahap pembagian data ditunjukkan pada Tabel 4.5 dan Tabel 4.6.

Tabel 4.5 Tahap pembagian data untuk *K* = 1

<i>K</i> = 1		
<i>R</i>	<i>G</i>	<i>B</i>
104	107	90
108	109	93
112	113	97
112	114	102

113	115	102
115	115	103
115	115	104
116	116	108
116	116	109
117	117	110
117	119	112
117	119	116

Tabel 4.6 Tahap pembagian data untuk K = 2

K = 2		
R	G	B
117	119	118
118	119	119
119	120	119
119	120	122
120	120	123
120	120	123
121	121	126
121	121	126
121	121	127
121	121	130
122	122	132
122	122	133
123	122	133

c. Tahap selanjutnya adalah dengan mengambil 5 data pertama dari data bagian pertama dan 5 data terakhir dari data bagian kedua. Tahap pengambilan 5 data pada setiap bagian ditunjukkan pada Tabel 4.7 dan Tabel 4.8.

Tabel 4.7 Tahap mengambil 5 data pertama untuk K = 1

Ambil 5 data pertama		
R	G	B
104	107	90
108	109	93
112	113	97
112	114	102
113	115	102

Tabel 4.8 Tahap mengambil 5 data terakhir untuk $K = 2$

Ambil 5 data terakhir		
R	G	B
121	121	127
121	121	130
122	122	132
122	122	133
123	122	133

d. Tahap terakhir adalah dengan mengambil nilai rata-rata pada setiap bagian untuk dijadikan sebagai inisialisasi *centroids* yang ditunjukkan pada Tabel 4.9.

Tabel 4.9 Tahap melakukan rata-rata

Initial Centroids		
R	G	B
109.8	111.6	96.8
121.8	121.6	131

Setelah didapati inisialisasi *centroids* tahap selanjutnya adalah dengan menghitung jarak antara piksel dengan *centroids* menggunakan *Euclidean distance*, D1 merepresentasikan jarak piksel dengan *centroids* baris pertama, dan D2 merepresentasikan jarak piksel dengan *centroids* baris kedua. Selanjutnya pembentukan *cluster* ditentukan oleh jarak terdekat antara piksel dengan *centroids*. *Cluster 0* merepresentasikan piksel objek *background* dan *cluster 1* merepresentasikan piksel objek makanan. Hasil perhitungan jarak dan penentuan *cluster* ditunjukkan pada Tabel 4.10.

Tabel 4.10 Hasil perhitungan jarak dan penentuan *cluster*

Jarak		
D1	D2	Cluster
10.05187	47.02127	0
4.943683	42.34619	0
2.615339	36.41428	0
8.569714	30.30841	0
9.509995	30.1264	0
7.002857	30.62352	0
8.581375	29.15476	0
13.10878	24.62925	0
14.97465	22.76401	0
15.16047	22.9826	0
18.00667	20.03497	0
21.8	15.96246	1
23.5805	14.09965	1

30.004	8.062258	1
25.14438	12.59365	1
25.11255	12.68858	1
29.64524	8.221922	1
32.32708	5.347897	1
37.21075	1.612452	1
32.38271	5.310367	1
28.11121	9.560335	1
33.8473	4.09878	1
38.67868	1.095445	1
39.04152	2.236068	1
39.11317	2.569047	1

Menghitung nilai *centroids* baru dengan cara mendapatkan nilai rata-rata *value* dari setiap *channel* warna pada setiap *cluster*. Hasil *centroids* baru ditunjukkan pada Tabel 4.11.

Tabel 4.11 Hasil *centroids* baru

<i>R</i>	<i>G</i>	<i>B</i>
113.1818	114.1818	102.7273
120.0714	120.5	124.7857

BAB 5 PEMBAHASAN

Pada bab ini akan dijelaskan mengenai implementasi dari perancangan yang telah dibuat pada bab sebelumnya ke dalam bahasa pemrograman. Terdapat tiga implementasi yang akan dilakukan antara lain implementasi *preprocessing* citra, segmentasi citra makanan menggunakan *Improved K-Means*, dan estimasi bobot sisa makanan.

5.1 Implementasi *preprocessing* citra

Preprocessing citra bertujuan untuk mempercepat dan mempermudah proses komputasi dengan mengurangi ukuran citra dan mengurangi dimensi ukuran citra. Selain itu dilakukan *Median Filter* untuk menghilangkan *noise* pada citra dan merubah warna *RGB* ke *LAB* untuk menghilangkan fitur *lightness* pada citra. Data masukan berupa citra makanan. Kode sumber 5.1 menunjukkan kode sumber dari *preprocessing* citra.

Kode sumber 5.1 *Preprocessing* citra

Kode sumber <i>preprocessing</i> citra	
1	import cv2
2	import pathlib
3	import numpy as np
4	import pandas as pd
5	from xlwt import Workbook
6	import xlr2
7	# membaca setiap file yang memiliki format.jpg pada directory
8	dataset_directory = pathlib.Path('E:\[Code]8 - skripsi\Dataset')
9	for files in dataset_directory.rglob('*.jpg'):
10	ori_image = cv2.imread(str(files))
11	#Preprocessing
12	sizeBP = 15
13	image = resizeImage(ori_image, resizeByPercent=sizeBP)
14	image = cv2.medianBlur(image, 3)
15	imageLab = cv2.cvtColor(image, cv2.COLOR_BGR2LAB)
16	L, a, b = cv2.split(imageLab)
17	merged = cv2.merge((a, b))
18	height, width, dim = image.shape
19	image = (merged / 255).reshape(height * width, 2)

Keterangan:

1. Baris 1 – 6 adalah deklarasi dari *library* yang digunakan
2. Baris 7 adalah komentar
3. Baris 8 adalah deklarasi alamat direktori citra yang akan di *preprocessing*
4. Baris 9 adalah perulangan untuk setiap *file* berformat *jpg* yang ada didalam *folder*
5. Baris 10 adalah proses membaca *file* yang disimpan ke dalam variabel *ori_image*
6. Baris 11 adalah komentar
7. Baris 12 – 13 adalah proses mengurangi ukuran citra hingga 15% dari ukuran semula
8. Baris 14 adalah proses *Median Filter* dengan kernel berukuran 3 x 3
9. Baris 15 adalah proses merubah format warna citra dari *BGR* ke *LAB*
10. Baris 16 adalah proses memisahkan warna *LAB* menjadi 3 bagian yaitu *L, a, b*

11. Baris 17 adalah menggabungkan komponen warna dari a^* dan b^* yang disimpan ke dalam variabel *merged*
12. Baris 18 adalah deklarasi ukuran dari citra yang disimpan ke dalam variabel *height*, *width*, dan *dim*
13. Baris 19 adalah proses normalisasi data dan mengurangi ukuran dimensi citra menjadi 2 dimensi yang disimpan ke dalam variabel *image*

5.2 Implementasi segmentasi dengan *Improved K-Means*

Pada subab ini implementasi akan dibagi menjadi beberapa tahapan antara lain implementasi menentukan jumlah *K cluster*, implementasi menentukan *centroids*, implementasi hitung jarak terpendek antara titik data dengan *centroids* dan implementasi kondisi proses berhenti.

5.2.1 Implementasi menentukan jumlah *K cluster*

Menentukan jumlah *K cluster* bertujuan untuk mendapatkan citra menjadi *K* bagian sehingga dapat dilakukan proses selanjutnya yaitu *clustering*. Kode sumber 5.2 menunjukkan kode sumber menentukan jumlah *K cluster*.

Kode sumber 5.2 Menentukan jumlah *K cluster*

Kode sumber menentukan jumlah K kluster	
1	<code>K = 2</code>
2	<code>Threshold = 0.001</code>
3	<code>MaxIteration = 50</code>

Keterangan:

1. Baris 1 adalah deklarasi nilai *K* yaitu 2
2. Baris 2 adalah deklarasi batas perubahan nilai dari *centroids* yang disimpan ke dalam variabel *Threshold*
3. Baris 3 adalah deklarasi iterasi maksimum sebuah proses dilakukan yang disimpan ke dalam variabel *MaxIteration*

5.2.2 Implementasi menentukan *centroids*

Menentukan *centroids* bertujuan untuk mendapatkan titik pusat dari sebuah *cluster* sehingga data dapat dikelompokkan berdasarkan titik pusat. Kode sumber 5.3 ditunjukkan kode sumber menentukan *centroids*.

Kode sumber 5.3 Menentukan *centroids*

Kode sumber menentukan centroids	
1	<code>def initialCentroids(image, K):</code>
2	<code> sorting = np.sort(image, axis=0)</code>
3	<code> splitting = len(sorting) / K</code>
4	<code> splited data = []</code>
5	<code> temp = 0.0</code>
6	<code> while temp < len(sorting):</code>
7	<code> splited data.append(sort_it[int(temp):int(temp + splitting)])</code>
8	<code> temp += splitting</code>
9	<code> centroids = []</code>
10	<code> for i in range(len(splited data)):</code>
11	<code> if i%2 == 0:</code>
12	<code> centroids.append(np.mean(splited_data[i] [:5],</code>


```

13         axis=0))
14     else:
15         centroids.append(np.mean(splited_data[i][-5:],
16                                 axis=0))
17     return np.array(centroids)

```

Keterangan:

1. Baris 1 merupakan deklarasi dari fungsi *initialCentroids()* dengan menerima parameter berupa citra dan K
2. Baris 2 adalah proses mengatur urutan data dari kecil hingga besar yang disimpan ke dalam variabel *sort_it*
3. Baris 3 adalah proses membagi data menjadi K bagian yang disimpan ke dalam variabel *average*
4. Baris 4 adalah deklarasi *list* kosong yang disimpan ke dalam variabel *splited_data*
5. Baris 5 adalah deklarasi nilai nol yang disimpan ke dalam variabel *temp*
6. Baris 6 adalah deklarasi perulangan ketika nilai *temp* kurang dari panjang *sort_it*
7. Baris 7 adalah memasukkan nilai *sort_it* ke dalam *list* yang telah dideklarasikan sebelumnya yaitu *splited_data*
8. Baris 8 adalah melakukan operasi tambah antara *temp* dengan *average*
9. Baris 9 adalah deklarasi *list* kosong yang disimpan ke dalam variabel *centroids*
10. Baris 10 – 13 adalah memasukkan 5 data pertama pada *splited_data* ke dalam *list centroids* ketika k bernilai ganjil
11. Baris 14 – 16 adalah memasukkan 5 data terakhir pada *splited_data* ke dalam *list centroids* ketika k bernilai genap
12. Baris 17 adalah mengembalikan nilai *centroids*.

5.2.3 Implementasi hitung jarak antara titik data dengan titik pusat

Hitung jarak antara titik data dengan *centroids* bertujuan untuk mengalokasikan titik data yang memiliki jarak terdekat dengan *centroids* sebagai bagian dari *subcluster* titik pusat tersebut. Kode sumber 5.4 ditunjukkan kode sumber hitung jarak antara titik data dengan titik pusat.

Kode sumber 5.4 Hitung jarak antara titik data dengan titik pusat

```

Kode sumber hitung jarak antara titik data dengan titik pusat
1 def nearestDistance(image, centroids):
2     alocation_data = []
3     for i in image:
4         distances = []
5         for j in centroids:
6             distances.append(EuclideanDistance(i,j))
7         alocation_data.append(np.argmin(distances))
8     return alocation_data

```

Keterangan:

1. Baris 1 merupakan deklarasi dari fungsi *nearestDistance()* dengan menerima parameter berupa citra dan *centroids*
2. Baris 2 adalah deklarasi *list* kosong yang disimpan ke dalam variabel *alocation_data*

3. Baris 3 – 6 adalah memasukkan nilai jarak antara titik data dengan *centroids* ke dalam *list distance*
4. Baris 7 adalah memasukkan jarak terpendek antara titik data dengan *centroids* ke dalam *list alocation_data*
5. Baris 8 adalah mengembalikan nilai *alocation_data*

5.2.4 Implementasi hitung *centroids* baru

Menentukan *centroids* baru bertujuan untuk mendapatkan titik pusat yang lebih optimum dibandingkan titik pusat sebelumnya, sehingga data dapat dikelompokkan secara maksimum. Kode sumber 5.5 ditunjukkan kode sumber menentukan *centroids* pada.

Kode sumber 5.5 Hitung *centroids* baru

```
Kode sumber hitung centroids baru
1 def newCentroids(image, clusters):
2     new_centroids = []
3     combine_data = pd.concat([pd.DataFrame(image),
4                               pd.DataFrame(clusters, columns=['cluster'])],
5                               axis=1)
6     for data in set(combine_data['cluster']):
7         image_data = combine_data[combine_data['cluster'] ==
8                                   data][combine_data.columns[:-1]]
9         mean = image_data.mean(axis=0)
10        new_centroids.append(mean)
11    return new_centroids
```

Keterangan:

1. Baris 1 merupakan deklarasi dari fungsi *newCentroids()* dengan menerima parameter berupa citra dan *clusters*
2. Baris 2 adalah deklarasi list kosong yang disimpan ke dalam variabel *new_centroids*
3. Baris 3 adalah proses penggabungan data *image* dengan data *clusters*
4. Baris 6 adalah deklarasi perulangan untuk setiap *array* di dalam kumpulan *array combine_data* pada kolom *cluster*
5. Baris 7 – 9 adalah proses menghitung rata-rata data yang memiliki nilai yang sama dengan *array combine_data* pada kolom *cluster*
6. Baris 10 adalah memasukkan rata-rata ke dalam *list new_centroids*
7. Baris 11 adalah mengembalikan nilai *new_centroids*

5.2.5 Implementasi kondisi proses berhenti

Kondisi proses berhenti bertujuan untuk membatasi berjalannya proses *Improved K-Means* untuk didapatkan hasil yang dirasa sudah mencapai optimum. Kode sumber 5.6 ditunjukkan kode sumber kondisi proses berhenti.

Kode sumber 5.6 Kondisi proses berhenti

```
Kode sumber kondisi proses berhenti
1 def stopCondition(old_centroid, curr_centroid):
2     return np.array_equal(np.array(old_centroid),
3                             np.array(curr_centroid))
```

Keterangan:

1. Baris 1 merupakan deklarasi dari fungsi *stopCondition()* dengan menerima parameter berupa *centroids* lama dan *centroids* saat ini
2. Baris 2 – 3 merupakan mengembalikan nilai perubahan *centroids*

5.3 Implementasi estimasi bobot makanan

Estimasi bobot makanan bertujuan untuk mendapatkan hasil estimasi bobot makanan yang telah dikonsumsi oleh seseorang. Kode sumber 5.7 ditunjukkan kode sumber estimasi bobot makanan.

Kode sumber 5.7 Estimasi bobot makanan

```
Kode sumber estimasi bobot makanan
1  wb = Workbook()
2  sheet1 = wb.add_sheet('Sheet 1')
3  row = 0
4  for i in range(4):
5      image_full = cv2.imread('E:\[Code]8 -skripsi\Output_traypart/
6          full_' + str(i)+ '.jpg')
7      gt_full = cv2.imread('E:\[Code]8 - skripsi\Groundtruth_part/
8          full_' + str(i) + '.jpg')
9      image_directory = pathlib.Path('E:\[Code]8 - skripsi\
10         Output_traypart')
11     for files in image_directory.rglob('*' + str(i) + '.jpg'):
12         image = cv2.imread(str(files))
13         gt = cv2.imread('E:\[Code]8 - skripsi\Groundtruth_part/'
14             + str(files.name))
15         image_full = cv2.bitwise_and(image_full, gt_full)
16         image = cv2.bitwise_and(image, gt)
17         weight_data = ("E:\[Code]8 - skripsi\beratdata.xlsx")
18         wb_xlrd = xlrd2.open_workbook(weight_data)
19         sheet_xlrd = wb_xlrd.sheet_by_index(0)
20         weight = sheet_xlrd.cell_value(1, i)
21         food = np.sum(image_full != 0)
22         foodleft = np.sum(image != 0)
23         if food > 0:
24             estimation = round(weight / food * foodleft, 2)
25         else:
26             estimation = 0
27         sheet1.write(row, 0, files.name)
28         sheet1.write(row, 1, estimation)
29         wb.save('E:\[Code]8 - skripsi\estimasi.xls')
30     row += 1
```

Keterangan:

1. Baris 1 adalah deklarasi variabel *wb*
2. Baris 2 adalah deklarasi variabel *sheet1*
3. Baris 3 adalah inisialisasi variabel *row* dengan nilai nol
4. Baris 4 adalah melakukan proses perulangan sebanyak 4 kali. Dimana 4 merupakan jumlah kompartemen pada *tray box*.
5. Baris 5 – 6 adalah proses membaca *file* yang disimpan ke dalam variabel *image_full*
6. Baris 7 – 8 adalah proses membaca *file* yang disimpan ke dalam variabel *gt_full*
7. Baris 9 – 12 adalah proses membaca *file* yang memiliki format *.jpg* dari direktori
8. Baris 13 – 14 adalah proses membaca *file* yang disimpan ke dalam variabel *gt*

9. Baris 15 – 16 adalah proses *intersection*

10. Baris 17 – 20 adalah proses membaca data berat makanan yang sesungguhnya

11. Baris 21 – 22 adalah proses menghitung jumlah piksel dari citra yang memiliki nilai tidak sama dengan nol

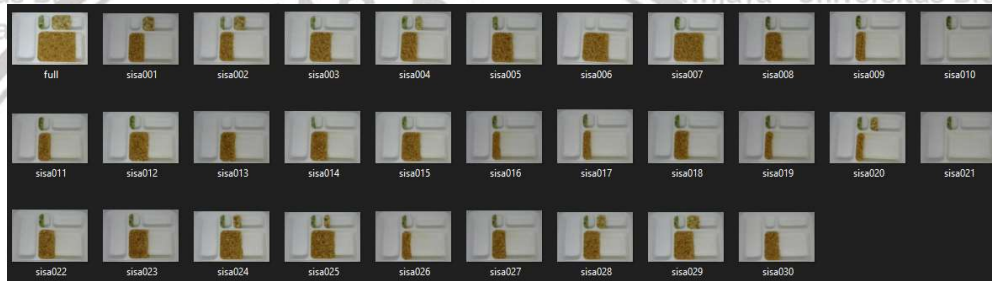
12. Baris 23 – 26 adalah proses melakukan estimasi bobot makanan yang disimpan pada variabel *prediction*

13. Baris 27 – 30 adalah proses menyimpan hasil estimasi bobot makanan pada *file excel*



BAB 6 PENGUJIAN DAN ANALISIS

Pada tahap ini akan dijelaskan mengenai pengujian beserta analisisnya. Pengujian yang dilakukan pada penelitian ini adalah pengujian hasil segmentasi dan pengujian estimasi bobot makanan berdasarkan hasil segmentasi. Setelah pengujian dilakukan maka akan didapatkan ringkasan hasil pengujian yang bisa digunakan untuk analisis evaluasi akhir. Seluruh pengujian menggunakan nilai $k=2$. Tujuan dari penggunaan $k=2$ agar hasil segmentasi citra dapat membedakan antara objek makanan dengan *background*. Untuk perhitungan evaluasi hasil segmentasi menggunakan nilai akurasi. Keseluruhan data yang digunakan dalam pengujian berjumlah 31 data citra *tray box* yang terdiri dari 124 citra kompartemen untuk setiap data. Contoh *dataset* yang digunakan sebagai pengujian dapat dilihat pada Gambar 6.1.



Gambar 6.1 *Dataset* citra makanan

Pada *dataset* citra makanan yang ditunjukkan pada Gambar 6.1 dapat dilihat bahwa masing-masing citra memiliki bentuk yang berbeda-beda. Dari data tersebut akan dilakukan pengujian beserta analisisnya untuk melihat sejauh apakah *Improved K-Means* dapat melakukan estimasi bobot makanan berdasarkan hasil segmentasi citra.

6.1 Pengujian hasil segmentasi citra

Pengujian ini dilakukan untuk melihat pengaruh penggunaan *Improved K-Means* (*IKM*) dan *Improved K-Means* yang dimodifikasi dengan 5 data (*IKM5*) terhadap ketepatan mengetahui akurasi dari hasil segmentasi. Untuk mengetahui akurasi dari hasil segmentasi dilakukan dengan menggunakan metode *Intersection Over Union* pada persamaan (2.11), dalam proses tersebut *intersection* didapat dari irisan piksel citra hasil segmentasi dengan *groundtruth* dan *union* didapat dari gabungan piksel citra hasil segmentasi dengan *groundtruth*. Untuk *groundtruth* dari sebuah citra ditunjukkan pada Gambar 6.2.



Gambar 6.2 Contoh citra dan *groundtruth*

Dari hasil pengujian pada Tabel 6.1 dapat dilihat bahwa nilai *IoU* pada kompartemen mentimun memiliki rata-rata nilai yang lebih rendah dibandingkan dengan rata-rata nilai kompartemen mentimun pada Tabel 6.2. Berdasarkan Gambar 6.3 ketika menggunakan *Improved K-Means (IKM)* maka proses inisialisasi *centroids* akan kurang maksimal, hal tersebut dikarenakan nilai inisialisasi didapat dari rata-rata data yang telah dibagi menjadi 2 bagian. Pada bagian pertama, data makanan dan data *background* masih dalam satu bagian, hal tersebut mengakibatkan inisialisasi *centroids* pada bagian pertama tidak sepenuhnya merepresentasikan nilai dari data makanan. Ketika menggunakan *IKM5* maka proses inisialisasi didapat dengan melakukan rata-rata untuk 5 data pertama yang digunakan untuk inisialisasi nilai *centroids* pertama dan 5 data terakhir untuk nilai kedua, sehingga inisialisasi *centroids* lebih maksimal karena masing-masing nilai merepresentasikan setiap bagian antara makanan dan *background*.

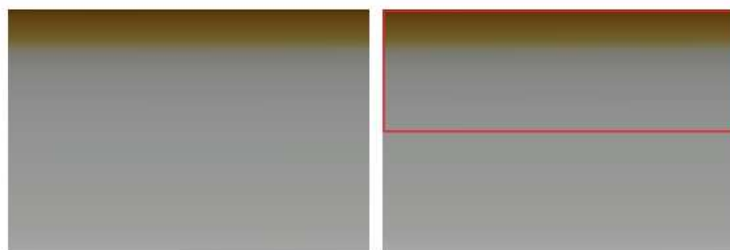
Tabel 6.1 Akurasi hasil segmentasi (IKM) terhadap *groundtruth*

Nama citra	<i>IoU</i> pada setiap kompartemen				<i>IoU</i> keseluruhan
	Nasi	Telur	Mentimun	Kosong	
full.jpg	0.98	0.94	0.72	1	0.88
sisa001.jpg	0.96	0.93	1	1	0.945
sisa002.jpg	0.96	0.93	0.73	1	0.873333333
sisa003.jpg	0.98	1	0.63	1	0.805
sisa004.jpg	0.96	0.91	0.85	1	0.906666667
sisa005.jpg	0.97	1	0.84	1	0.905
sisa006.jpg	0.99	1	1	1	0.99
sisa007.jpg	0.97	1	0.83	1	0.9
sisa008.jpg	0.95	1	0.77	1	0.86
sisa009.jpg	0.91	1	0.86	1	0.885
sisa010.jpg	1	1	0.46	1	0.46
sisa011.jpg	0.97	1	0.83	1	0.9
sisa012.jpg	0.97	1	0.83	1	0.9
sisa013.jpg	0.97	1	1	1	0.97
sisa014.jpg	0.97	1	0.66	1	0.815
sisa015.jpg	0.97	1	0.86	1	0.915
sisa016.jpg	0.91	1	0.86	1	0.885
sisa017.jpg	0.91	1	0.9	1	0.905
sisa018.jpg	0.96	1	0.81	1	0.885
sisa019.jpg	0.86	1	0.81	1	0.835
sisa020.jpg	0.93	0.92	0.86	1	0.903333333
sisa021.jpg	1	1	0.44	1	0.44
sisa022.jpg	0.98	1	0.8	1	0.89
sisa023.jpg	0.98	1	0.77	1	0.875
sisa024.jpg	0.98	0.91	0.8	1	0.896666667
sisa025.jpg	0.98	0.91	0.78	1	0.89

sisa026.jpg	0.94	1	0.79	1	0.865
sisa027.jpg	0.97	1	0.8	1	0.885
sisa028.jpg	0.97	0.94	0.85	1	0.92
sisa029.jpg	0.98	0.95	0.8	1	0.91
sisa030.jpg	0.97	1	1	1	0.97
Rata-rata:	0.961	0.979	0.805	1	0.866612903

Tabel 6.2 Akurasi hasil segmentasi (IKM5) terhadap *groundtruth*

Nama citra	IoU pada setiap kompartemen				IoU keseluruhan
	Nasi	Telur	Mentimun	Kosong	
full.jpg	0.98	0.94	0.72	1	0.88
sisa001.jpg	0.96	0.93	1	1	0.945
sisa002.jpg	0.96	0.93	0.73	1	0.873333333
sisa003.jpg	0.98	1	0.63	1	0.805
sisa004.jpg	0.96	0.91	0.85	1	0.906666667
sisa005.jpg	0.97	1	0.84	1	0.905
sisa006.jpg	0.99	1	1	1	0.99
sisa007.jpg	0.97	1	0.83	1	0.9
sisa008.jpg	0.95	1	0.77	1	0.86
sisa009.jpg	0.91	1	0.86	1	0.885
sisa010.jpg	1	1	0.93	1	0.93
sisa011.jpg	0.97	1	0.83	1	0.9
sisa012.jpg	0.97	1	0.83	1	0.9
sisa013.jpg	0.97	1	1	1	0.97
sisa014.jpg	0.97	1	0.66	1	0.815
sisa015.jpg	0.97	1	0.86	1	0.915
sisa016.jpg	0.92	1	0.86	1	0.89
sisa017.jpg	0.91	1	0.9	1	0.905
sisa018.jpg	0.96	1	0.81	1	0.885
sisa019.jpg	0.86	1	0.81	1	0.835
sisa020.jpg	0.93	0.92	0.86	1	0.903333333
sisa021.jpg	1	1	0.89	1	0.89
sisa022.jpg	0.98	1	0.8	1	0.89
sisa023.jpg	0.98	1	0.77	1	0.875
sisa024.jpg	0.98	0.91	0.8	1	0.896666667
sisa025.jpg	0.98	0.91	0.78	1	0.89
sisa026.jpg	0.94	1	0.77	1	0.855
sisa027.jpg	0.97	1	0.8	1	0.885
sisa028.jpg	0.97	0.94	0.85	1	0.92
sisa029.jpg	0.98	0.95	0.8	1	0.91
sisa030.jpg	0.97	1	1	1	0.97
Rata-rata:	0.962	0.979	0.834	1	0.896129032



Gambar 6.3 Contoh ilustrasi pengurutan data

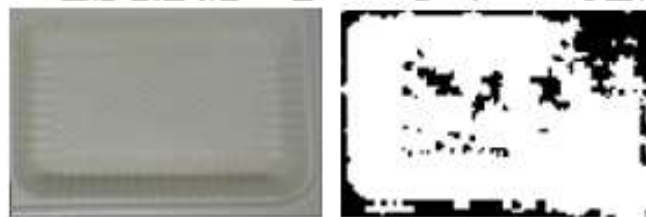
6.2 Pengujian hasil estimasi bobot sisa makanan

Pada Lampiran 2 menunjukkan data bobot sisa makanan yang sesungguhnya. Lampiran 3-4 merupakan keluaran dari pengujian hasil estimasi bobot sisa makanan menggunakan *EFLE*. Hasil dari estimasi bobot sisa makanan digunakan untuk melihat seberapa jauh metode yang diusulkan dapat melakukan estimasi bobot makanan, sehingga tidak diperlukannya penimbangan menggunakan alat khusus ataupun taksiran visual yang membutuhkan penaksir yang terlatih dan teliti. Tabel 6.3 menunjukkan *RMSE* estimasi bobot sisa makanan pada *IKM*. Terdapat jarak yang signifikan antara *RMSE* dengan bobot sesungguhnya dan *EFLE*, terutama pada kompartemen yang berisi kosong. Masalahnya adalah dengan menggunakan *IKM* yang tidak dapat mengatasi seluruh area dari item makanan. Gambar 6.4 menunjukkan contoh masalah dari hasil segmentasi citra pada kompartemen nasi saat menggunakan *IKM*.

Tabel 6.3 *RMSE* estimasi bobot sisa makanan pada *IKM*

Nama citra	Estimasi bobot kompartemen			
	Nasi	Telur	Mentimun	Kosong
sisa001.jpg	226.80	5.71	0.00	0.00
sisa002.jpg	195.16	4.54	3.65	0.00
sisa003.jpg	237.78	0.00	2.96	0.00
sisa004.jpg	96.24	3.76	4.16	0.00
sisa005.jpg	390.06	0.00	9.36	0.00
sisa006.jpg	124.32	0.00	0.00	0.00
sisa007.jpg	72.59	0.00	1.72	0.00
sisa008.jpg	72.93	0.00	2.37	0.00
sisa009.jpg	184.42	0.00	7.40	0.00
sisa010.jpg	0.00	0.00	24.11	0.00
sisa011.jpg	202.21	0.00	8.70	0.00
sisa012.jpg	211.12	0.00	4.88	0.00
sisa013.jpg	119.03	0.00	0.00	0.00
sisa014.jpg	144.00	0.00	2.66	0.00
sisa015.jpg	62.41	0.00	9.49	0.00

sisa016.jpg	113.42	0.00	8.64	0.00
sisa017.jpg	27.88	0.00	7.62	0.00
sisa018.jpg	114.28	0.00	1.90	0.00
sisa019.jpg	109.41	0.00	9.12	0.00
sisa020.jpg	146.41	17.98	10.96	0.00
sisa021.jpg	0.00	0.00	27.46	0.00
sisa022.jpg	267.00	0.00	9.36	0.00
sisa023.jpg	245.24	0.00	5.43	0.00
sisa024.jpg	153.02	37.70	5.95	0.00
sisa025.jpg	79.21	20.88	4.08	0.00
sisa026.jpg	332.70	0.00	4.75	0.00
sisa027.jpg	315.42	0.00	7.24	0.00
sisa028.jpg	262.76	58.06	5.52	0.00
sisa029.jpg	190.44	57.15	4.67	0.00
sisa030.jpg	252.17	0.00	0.00	0.00



Gambar 6.4 Contoh masalah dari hasil segmentasi citra

Tabel 6.4 menunjukkan *Root Mean Square Error (RMSE)* estimasi berat sisa makanan pada *IKM5*. Terdapat jarak yang signifikan antara *RMSE* dengan bobot sesungguhnya dan *Elementary leftover Food Estimation (EFLE)*. Pada Tabel 6.5 ditunjukkan perbandingan *RMSE* antara *IKM* dengan *IKM5*, dan pengurangan galat tertinggi mencapai 14,89% dengan *RMSE* terkecil dicapai sebesar 2,19. Ini membuktikan bahwa segmentasi citra menggunakan *IKM5* sangat berguna untuk mengatasi masalah *background* yang terlalu bertekstur.

Tabel 6.4 *RMSE* estimasi bobot sisa makanan pada *IKM5*

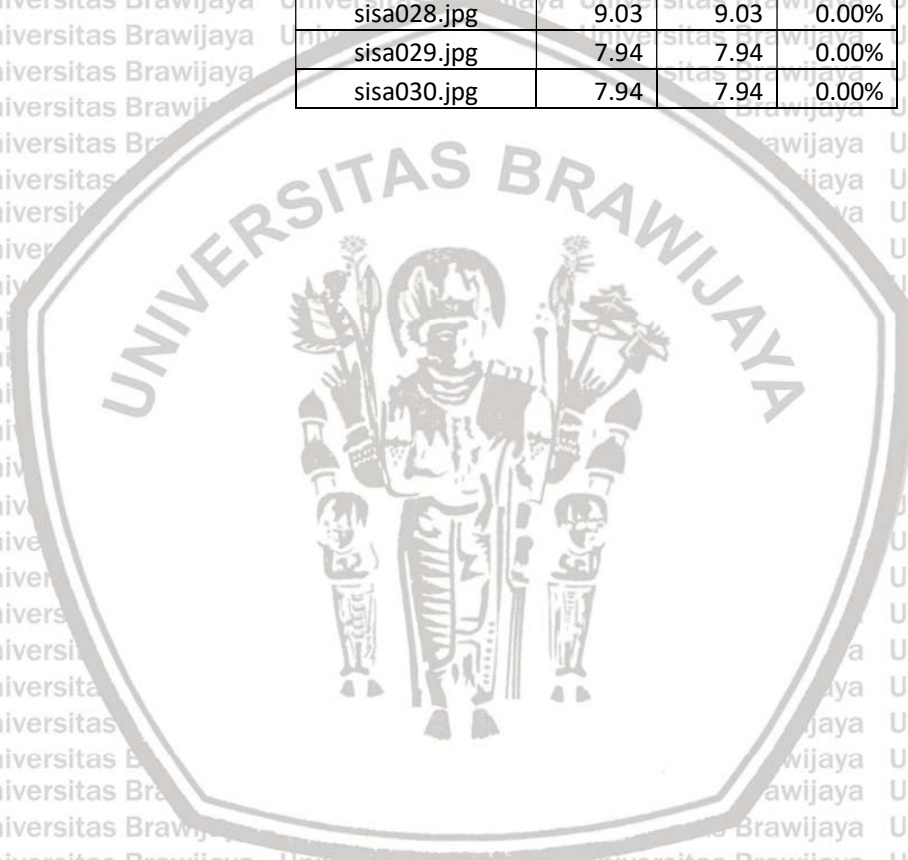
Nama citra	Estimasi bobot kompartemen			
	Nasi	Telur	Mentimun	Kosong
sisa001.jpg	226.80	5.71	0.00	0.00
sisa002.jpg	195.16	4.54	3.65	0.00
sisa003.jpg	237.78	0.00	2.96	0.00
sisa004.jpg	96.24	3.76	4.16	0.00
sisa005.jpg	390.06	0.00	9.36	0.00
sisa006.jpg	124.32	0.00	0.00	0.00
sisa007.jpg	72.59	0.00	1.72	0.00
sisa008.jpg	72.93	0.00	2.37	0.00
sisa009.jpg	184.42	0.00	7.40	0.00

sisa010.jpg	0.00	0.00	19.18	0.00
sisa011.jpg	202.21	0.00	8.70	0.00
sisa012.jpg	211.12	0.00	4.88	0.00
sisa013.jpg	119.03	0.00	0.00	0.00
sisa014.jpg	144.00	0.00	2.66	0.00
sisa015.jpg	62.41	0.00	9.49	0.00
sisa016.jpg	113.42	0.00	8.64	0.00
sisa017.jpg	27.88	0.00	7.62	0.00
sisa018.jpg	114.28	0.00	1.90	0.00
sisa019.jpg	109.41	0.00	9.12	0.00
sisa020.jpg	146.41	17.98	10.96	0.00
sisa021.jpg	0.00	0.00	19.89	0.00
sisa022.jpg	267.00	0.00	9.36	0.00
sisa023.jpg	245.24	0.00	5.43	0.00
sisa024.jpg	153.02	37.70	5.95	0.00
sisa025.jpg	79.21	20.88	4.08	0.00
sisa026.jpg	329.42	0.00	3.92	0.00
sisa027.jpg	315.42	0.00	7.24	0.00
sisa028.jpg	262.76	58.06	5.52	0.00
sisa029.jpg	190.44	57.15	4.67	0.00
sisa030.jpg	252.17	0.00	0.00	0.00

Tabel 6.5 Perbedaan *RMSE* antara *IKM* dengan *IKM5*

Nama Citra	<i>IKM</i>	<i>IKM</i>	Diff
sisa001.jpg	7.62	7.62	0.00%
sisa002.jpg	7.13	7.13	0.00%
sisa003.jpg	7.76	7.76	0.00%
sisa004.jpg	5.10	5.10	0.00%
sisa005.jpg	9.99	9.99	0.00%
sisa006.jpg	5.58	5.58	0.00%
sisa007.jpg	4.31	4.31	0.00%
sisa008.jpg	4.34	4.34	0.00%
sisa009.jpg	6.92	6.92	0.00%
sisa010.jpg	2.46	2.19	10.79%
sisa011.jpg	7.26	7.26	0.00%
sisa012.jpg	7.35	7.35	0.00%
sisa013.jpg	5.46	5.46	0.00%
sisa014.jpg	6.06	6.06	0.00%
sisa015.jpg	4.24	4.24	0.00%
sisa016.jpg	5.52	5.52	0.00%
sisa017.jpg	2.98	2.98	0.00%
sisa018.jpg	5.39	5.39	0.00%

sisa019.jpg	5.44	5.44	0.00%
sisa020.jpg	6.62	6.62	0.00%
sisa021.jpg	2.62	2.23	14.89%
sisa022.jpg	8.31	8.31	0.00%
sisa023.jpg	7.92	7.92	0.00%
sisa024.jpg	7.01	7.01	0.00%
sisa025.jpg	5.10	5.10	0.00%
sisa026.jpg	9.18	9.13	0.61%
sisa027.jpg	8.98	8.98	0.00%
sisa028.jpg	9.03	9.03	0.00%
sisa029.jpg	7.94	7.94	0.00%
sisa030.jpg	7.94	7.94	0.00%



BAB 7 PENUTUP

7.1 Kesimpulan

Dari hasil pengujian dan analisis yang telah dilakukan pada bab sebelumnya maka didapat kesimpulan hasil penelitian adalah sebagai berikut.

1. Berdasarkan segmentasi citra makanan di *tray box* menggunakan *clustering Improved K-Means* didapat bahwa metode yang diusulkan mampu melakukan segmentasi citra makanan jika dilihat berdasarkan hasil evaluasinya. Hal ini disebabkan segmentasi dilakukan berdasarkan *clustering* menyebabkan objek dan *background* pada citra dapat dibedakan dengan tingkat rata-rata akurasi tertinggi sebesar 97%.
2. Pada estimasi bobot sisa makanan, *Improved K-Means* yang dimodifikasi dapat menyelesaikan masalah ketika kompartemen yang berisi kosong tidak dapat tersegmen dengan baik dengan menggunakan *Improved K-Means*. *RMSE* dari *IKM* ke *IKM5* menghasilkan penurunan galat hingga 14,89% dengan *RMSE* terkecil dicapai sebesar 2,19.

7.2 Saran

Berikut adalah saran yang didapat berdasarkan hasil penelitian yang telah dilakukan dengan tujuan dapat diimplementasikan pada penelitian selanjutnya.

1. Diperlukan uji coba menggunakan *preprocessing* ruang warna lain seperti *RGB*, *HSV* atau lainnya dengan tujuan mendapatkan perbandingan hasil.
2. Pengambilan data secara manual pada *IKM5* perlu ditingkatkan dengan tujuan agar lebih adaptif.

DAFTAR REFERENSI

- Amelia, A., 2018. *K-Means Clustering: From A to Z*. [Online] Available at: <https://towardsdatascience.com/K-Means-clustering-from-a-to-z-f6242a314e9a>
- Anisa, A. F. et al. (2017) 'Permasalahan gizi masyarakat dan upaya perbaikannya', *Gizi Masyarakat*, 40, pp. 1–22.
- Dubey, S. R. et al. (2013) 'Infected Fruit Part Detection using K-Means Clustering Segmentation Technique', *International Journal of Interactive Multimedia and Artificial Intelligence*, 2(2), p. 65. doi: 10.9781/ijimai.2013.229.
- Ediyanto, Mara, N. and Intisari, N. S. (2013) 'Pengklasifikasian Karakteristik Dengan Metode K-Means Cluster Analysis', *Buletin Ilmiah Mat. Stat. dan Terapannya (Bimaster)*, 02(2), pp. 133–136.
- FAO (2019) 'Pangan Sehat Itu Penting', p. 4. Available at: <http://www.fao.org/3/ca4694id/CA4694ID.pdf>.
- Gustientiedina, G., Adiya, M. H. and Desnelita, Y. (2019) 'Penerapan Algoritma K-Means Untuk Clustering Data Obat-Obatan', *Jurnal Nasional Teknologi dan Sistem Informasi*, 5(1), pp. 17–24. doi: 10.25077/teknosi.v5i1.2019.17-24.
- Kadir, A. and Susanto, A. (2012) 'Abdul Kadir Abd AdhiS dhiSusanto', *Pengolahan Citra*, pp. 1–700.
- Mardalena, I. and Suyani, E. (2016) 'Keperawatan Ilmu Gizi', *Kementerian Kesehatan Republik Indonesia*, p. 182.
- Maulana, L., Bihanda, Y. G. and Sari, Y. A. (2020) 'Color space and color channel selection on image segmentation of food images', *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, 6(2), p. 141. doi: 10.26594/register.v6i2.2061.
- `pandas.DataFrame.head` — *pandas 1.3.0 documentation* (no date). Available at: <https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.head.html> (Accessed: 13 July 2021).
- Rosebrock, A., 2016. *Intersection Over Union (IoU) for object detection*. [Online] Available at: <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-IoU-for-object-detection/>
- Sari, Y. A. et al. (2019) 'Automatic Food Leftover Estimation in Tray Box Using Image Segmentation', *Proceedings of 2019 4th International Conference on Sustainable Information Engineering and Technology, SIET 2019*, pp. 212–216. doi: 10.1109/SIET48054.2019.8986104.
- Sari, Y. A., Maligan, J. M. and Bihanda, Y. G. (2020) 'Multiple Food or Non-Food Detection in Single Tray Box Image using Fraction of Pixel Segmentation for Developing Smart Nutrition Box Prototype', *International Journal of Innovative Technology and Exploring Engineering*, 9(3S), pp. 132–136. doi: 10.35940/ijitee.c1030.0193s20.

Sari, Y. A., Maligan, J. M. and Prakoso, A. F. (2020) 'Improving the *Elementary leftover Food Estimation* Algorithm by Using *Clustering* Image Segmentation in Nutrition Intake Problem', *CENIM 2020 - Proceeding: International Conference on Computer Engineering, Network, and Intelligent Multimedia 2020*, pp. 435–439. doi: 10.1109/CENIM51130.2020.9298005.

Sinaga, A. S. (2019) 'SEGMENTASI RUANG WARNA $L^*a^*b^*$ ', *Jurnal Mantik Penusa*, 3(1), pp. 43–46.

Sona, K. & Rajiv, B., 2016. *Efficient Improved K means Clustering for Image Segmentation*, June, IV(6), pp. 10625-10629.

Wakhidah, N. (2010) 'Clustering Menggunakan K-Means Algorithm', *Jurnal Transformatika*, 8(1), p. 33. doi: 10.26623/transformatika.v8i1.45.



LAMPIRAN-LAMPIRAN

Lampiran-lampiran yang terdapat pada bagian ini meliputi Lampiran 1-4

Lampiran 1 tabel *dataset* citra makanan pada *tray box*



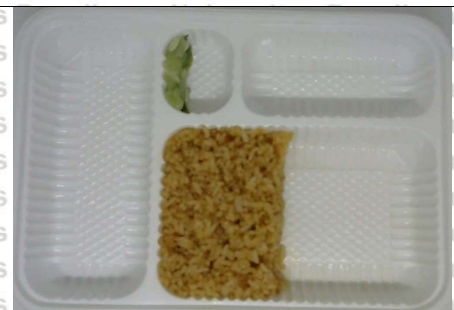
Full.jpg



Sisa001.jpg



Sisa002.jpg



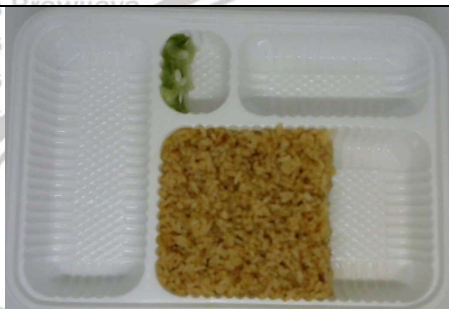
Sisa003.jpg



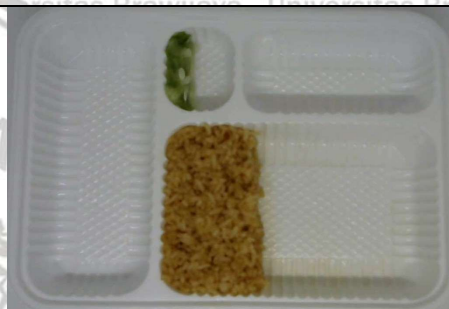
Sisa004.jpg



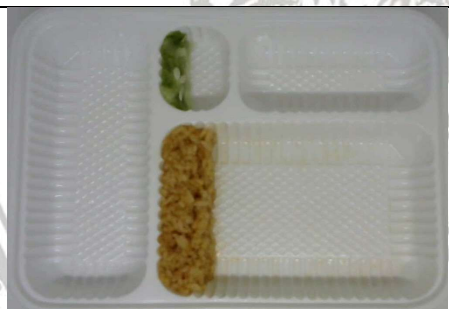
Sisa005.jpg



Sisa006.jpg



Sisa007.jpg



Sisa008.jpg



Sisa009.jpg



Sisa010.jpg



Sisa011.jpg



Sisa012.jpg





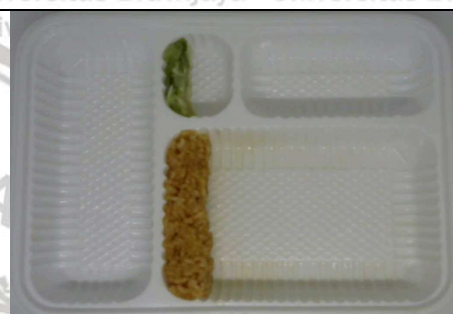
Sisa013.jpg



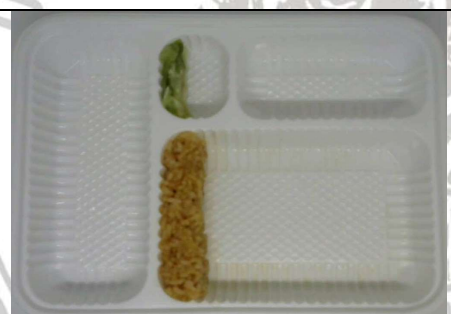
Sisa014.jpg



Sisa015.jpg



Sisa016.jpg



Sisa017.jpg



Sisa018.jpg



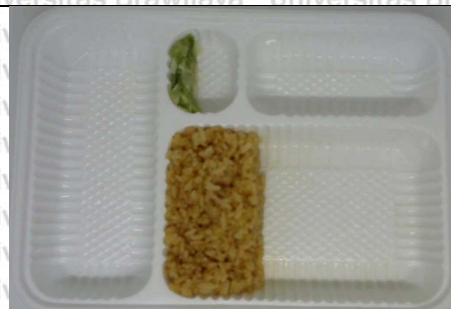
Sisa019.jpg



Sisa020.jpg



Sisa021.jpg



Sisa022.jpg



Sisa023.jpg



Sisa024.jpg



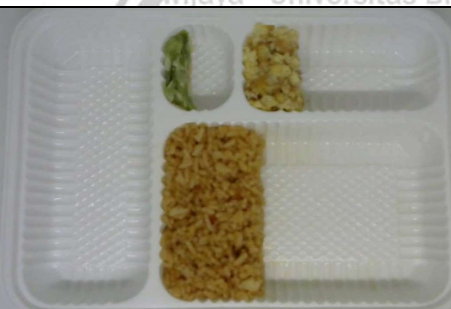
Sisa025.jpg



Sisa026.jpg



Sisa027.jpg



Sisa028.jpg



Lampiran 2 tabel data berat sesungguhnya

Nama citra	Data berat sesungguhnya (gram)			
	Nasi	Telur	Mentimun	Blank
full.jpg	188	38	10	0
sisa001.jpg	61	20	0	0
sisa002.jpg	52	20	5	0
sisa003.jpg	91	0	5	0
sisa004.jpg	70	10	8	0
sisa005.jpg	84	0	7	0
sisa006.jpg	118	0	0	0
sisa007.jpg	141	0	8	0
sisa008.jpg	74	0	7	0
sisa009.jpg	31	0	7	0
sisa010.jpg	0	0	7	0
sisa011.jpg	47	0	7	0
sisa012.jpg	82	0	7	0
sisa013.jpg	64	0	0	0
sisa014.jpg	73	0	4	0
sisa015.jpg	86	0	6	0
sisa016.jpg	28	0	6	0
sisa017.jpg	43	0	6	0
sisa018.jpg	59	0	7	0
sisa019.jpg	25	0	6	0
sisa020.jpg	25	19	6	0
sisa021.jpg	0	0	6	0
sisa022.jpg	66	0	6	0
sisa023.jpg	83	0	6	0
sisa024.jpg	99	9	6	0
sisa025.jpg	113	6	6	0
sisa026.jpg	25	0	6	0
sisa027.jpg	51	0	6	0

sisa028.jpg	68	11	6	0
sisa029.jpg	89	17	6	0
sisa030.jpg	57	0	0	0

Lampiran 3 tabel hasil estimasi menggunakan IKM

Nama citra	Estimasi bobot setiap kompartemen			
	Nasi	Telur	Mentimun	Kosong
sisa001.jpg	76.06	22.39	0	0
sisa002.jpg	65.97	22.13	6.91	0
sisa003.jpg	106.42	0	6.72	0
sisa004.jpg	79.81	11.94	10.04	0
sisa005.jpg	103.75	0	10.06	0
sisa006.jpg	129.15	0	0	0
sisa007.jpg	149.52	0	9.31	0
sisa008.jpg	82.54	0	8.54	0
sisa009.jpg	44.58	0	9.72	0
sisa010.jpg	0	0	11.91	0
sisa011.jpg	61.22	0	9.95	0
sisa012.jpg	96.53	0	9.21	0
sisa013.jpg	74.91	0	0	0
sisa014.jpg	85	0	5.63	0
sisa015.jpg	93.9	0	9.08	0
sisa016.jpg	38.65	0	8.94	0
sisa017.jpg	37.72	0	8.76	0
sisa018.jpg	69.69	0	8.38	0
sisa019.jpg	35.46	0	9.02	0
sisa020.jpg	37.1	14.76	9.31	0
sisa021.jpg	0	0	11.24	0
sisa022.jpg	82.34	0	9.06	0
sisa023.jpg	98.66	0	8.33	0
sisa024.jpg	111.37	15.14	8.44	0
sisa025.jpg	121.9	10.57	8.02	0
sisa026.jpg	43.24	0	8.18	0
sisa027.jpg	68.76	0	8.69	0
sisa028.jpg	84.21	18.62	8.35	0
sisa029.jpg	102.8	24.56	8.16	0
sisa030.jpg	72.88	0	0	0

Lampiran 4 tabel hasil estimasi menggunakan IKM5

Nama citra	Estimasi bobot setiap kompartemen			
	Nasi	Telur	Mentimun	Kosong

sis001.jpg	76.06	22.39	0	0
sis002.jpg	65.97	22.13	6.91	0
sis003.jpg	106.42	0	6.72	0
sis004.jpg	79.81	11.94	10.04	0
sis005.jpg	103.75	0	10.06	0
sis006.jpg	129.15	0	0	0
sis007.jpg	149.52	0	9.31	0
sis008.jpg	82.54	0	8.54	0
sis009.jpg	44.58	0	9.72	0
sis010.jpg	0	0	11.38	0
sis011.jpg	61.22	0	9.95	0
sis012.jpg	96.53	0	9.21	0
sis013.jpg	74.91	0	0	0
sis014.jpg	85	0	5.63	0
sis015.jpg	93.9	0	9.08	0
sis016.jpg	38.65	0	8.94	0
sis017.jpg	37.72	0	8.76	0
sis018.jpg	69.69	0	8.38	0
sis019.jpg	35.46	0	9.02	0
sis020.jpg	37.1	14.76	9.31	0
sis021.jpg	0	0	10.46	0
sis022.jpg	82.34	0	9.06	0
sis023.jpg	98.66	0	8.33	0
sis024.jpg	111.37	15.14	8.44	0
sis025.jpg	121.9	10.57	8.02	0
sis026.jpg	43.15	0	7.98	0
sis027.jpg	68.76	0	8.69	0
sis028.jpg	84.21	18.62	8.35	0
sis029.jpg	102.8	24.56	8.16	0
sis030.jpg	72.88	0	0	0