

**ESTIMASI KALORI PADA JAJANAN PASAR DI INDONESIA
MENGUNAKAN MASK R-CNN DAN REGRESI LINEAR
BERGANDA**

TESIS

**Karya tulis sebagai salah satu syarat
untuk memperoleh gelar Magister dari
Institut Teknologi Bandung**

**Oleh
NADYA ADITAMA
NIM: 23520039
(Program Studi Magister Informatika)**



**INSTITUT TEKNOLOGI BANDUNG
Januari 2022**

ABSTRAK

ESTIMASI KALORI PADA JAJANAN PASAR DI INDONESIA MENGUNAKAN MASK R-CNN DAN REGRESI LINEAR BERGANDA

Oleh
Nadya Aditama
NIM: 23520039
(Program Studi Magister Informatika)

Terdapat dua permasalahan dalam membangun model estimasi kalori berbasis citra dengan segmentasi *Mask R-CNN* untuk mendapatkan bentuk objek makanan dan prediksi berat makanan dengan model regresi linear. Permasalahan pertama adalah model regresi linear sederhana mempunyai nilai *R Squared* yang lebih rendah dibandingkan dengan regresi linear berganda menurut penelitian Abdelhady dkk. (2019) sehingga diperlukan penambahan fitur lainnya dalam pengukuran. Permasalahan kedua adalah adanya posisi tumpang tindih pada objek makanan sehingga tidak didapatkan bentuk objek secara utuh. Oleh karena itu, dalam penelitian ini model *Mask R-CNN* akan dilatih dengan objek yang dianotasi secara *amodal* sehingga diharapkan model dapat membentuk hasil segmentasi yang mensegmen bagian yang terhalang seperti yang telah dievaluasi pada penelitian Qi dkk. (2019) dalam kasus *amodal instance segmentation* pada *dataset KINS*.

Dalam penelitian ini, *dataset* citra jajanan pasar di Indonesia telah dibangun. Makanan yang terdapat dalam *dataset* ini adalah tahu, tempe, bakwan, cireng, bolu, dan serabi. *Dataset* diambil secara manual dengan jumlah makanan di piring yang beragam serta posisi yang berbeda-beda, baik posisi yang tumpang tindih maupun yang tidak tumpang tindih. Model dibagi menjadi tiga bagian, yaitu model deteksi, model estimasi, dan model gabungan antara deteksi dan estimasi. Dalam model gabungan, objek akan disegmentasi terlebih dahulu oleh model *Mask R-CNN*, kemudian dari hasil segmentasi akan diambil informasi luas, perimeter, panjang, dan lebar dari objek untuk memprediksi berat makanan dengan model regresi linear berganda. Informasi berat akan dikonversikan ke satuan kilokalori.

Dalam pembangunan model deteksi, hasil penelitian menunjukkan bahwa, model ResNeXt-101-FPN mempunyai mAP validasi yang tidak berbeda jauh dengan model ResNet-101-FPN dalam mensegmentasi objek yang diannotasi secara *amodal*, yaitu 91.74% (ResNet-101-FPN) dan 91.47% (ResNeXt-101-FPN). Dalam model estimasi, model regresi linear berganda dengan empat fitur yang diusulkan mempunyai nilai *R Squared* sebesar 0.804 dan rata-rata nilai MAE seluruh kelas dalam prediksi data uji model estimasi sebesar 5.254.

Dalam model gabungan, model Mask R-CNN yang terbaik adalah model dengan *backbone* ResNeXt-101-FPN. Model ini berhasil mendeteksi dan mensegmentasi makanan dengan rata-rata *F1 Score* sebesar 0.821 dalam *IoU threshold* di atas 0.85 pada skenario gambar yang mengandung objek tumpang tindih dan 0.994 dalam *IoU threshold* di atas 0.9 pada skenario gambar yang mengandung objek tidak tumpang tindih. Model regresi linear berganda yang diusulkan mendapatkan rata-rata nilai MAE seluruh kelas sebesar 8.354 pada skenario objek yang tumpang tindih dan rata-rata sebesar 11.256 pada skenario objek yang tidak tumpang tindih.

Meskipun seperti itu, model ini mempunyai kelemahan. Pada model deteksi, masih dijumpai *false positive* diantara objek yang tumpang tindih dan hasil segmentasi yang belum terlalu mirip dengan *ground truth* objeknya. Selain itu, pada saat memprediksi gambar yang tidak ada pada data latih, masih terdapat deteksi *false positive* diantara objek yang tumpang tindih. Secara keseluruhan, task *amodal instance segmentation* dalam memprediksi kalori makanan yang tumpang tindih mampu membantu dalam mendapatkan fitur untuk estimasi kalori dengan rata-rata nilai MAE yang tidak terlalu besar. Selain itu, pada kasus tumpang tindih, regresi linear berganda dengan fitur yang diusulkan tidak sebaik model yang lainnya pada beberapa kelas makanan. Hal ini disebabkan karena kurang sempurnanya hasil segmentasi pada objek tumpang tindih. Model regresi linear berganda lebih tepat digunakan pada pengukuran kalori pada objek tidak tumpang tindih.

Kata kunci: *Mask R-CNN*, Regresi Linear Berganda, Estimasi Kalori, *Amodal Instance Segmentation*, Jajanan Pasar Indonesia.

ABSTRACT

INDONESIAN STREET FOOD CALORIE ESTIMATION USING MASK R-CNN DAN MULTIPLE LINEAR REGRESSION

By

Nadya Aditama

NIM: 23520039

(Master's Program in Informatics)

There are two problems in building image-based calorie estimation model using Mask R-CNN to get the food shape and food weight prediction using linear regression model. First, simple linear regression model has lower R Squared score than multiple linear regression according to Abdelhady et al. (2019) so it is necessary to add more feature in measurement. Second, there are some occluded food objects so the system cannot get the real food shape. Therefore, in this research, Mask R-CNN model will be trained with the amodally annotated object dataset, so the model is expected to form the segmentation result that segment the occluded part of the object as evaluated in Qi et al. (2019) research in amodal instance segmentation task on KINS Dataset.

In this research, an image dataset of Indonesian street food has been created. The food that are used in this dataset are tahu, tempe, bakwan, cireng, bolu, and serabi. The dataset was taken manually with the various amount of food in plates and different positions, both occluded and non-occluded positions. The calorie estimation model is divided into three models, the detection model, the estimation model, and the combined model between detection and estimation. In the combined model, the image will be segmented first by the Mask R-CNN model. From the segmentation results, information on the area, perimeter, length, and width of the object will be taken to predict the weight of the food using multiple linear regression models. Weight information will be converted to kilocalorie units.

In the development of detection model, the results showed that the ResNeXt-101-FPN model had a validation mAP that was not much different from the ResNet-101-FPN model in segmenting amodal annotated objects. The mAP is 91.74% for ResNet-101-FPN and 91.47% for ResNeXt-101-FPN. In the estimation model, a multiple linear regression model with four proposed features has an R Squared score of 0.804, and the average MAE score for all classes in the prediction of the estimation model on test data is 5.254.

In the combined model, the best Mask R-CNN model is the model with the ResNeXt-101-FPN backbone. This model succeeded in detecting and segmenting food with an average F1 Score of 0.821 in the IoU threshold above 0.85 in the scenario of images containing occluded objects and 0.994 in the IoU threshold above 0.9 in the

scenario of images containing non-occluded objects. The proposed multiple linear regression model gets an average MAE value for all classes 8.354 in the occluded object scenario and an average MAE value for all classes 11.256 in the non-occluded object scenario.

Even so, this model has drawbacks. In the detection model, there are still false positives between the occluded objects, and segmentation results are not very similar to the ground truth object. When predicting object that is not on training data, there still false positive detection between the occluded objects. Overall, the amodal instance segmentation task in predicting occluded food calories can help in getting features for calorie estimation with an average MAE value that is not too large. In addition, in the scenario of the occluded object, the multiple linear regression with the proposed features is not as good as the other models in some food class. This is due to imperfect segmentation results on occluded objects. A multiple linear regression model is more appropriate to use in measuring calories on non-overlapping objects.

Keywords: Mask R-CNN, Multiple Linear Regression, Calorie Estimation, Amodal Instance Segmentation, Indonesian Street Food.

ESTIMASI KALORI PADA JAJANAN PASAR DI INDONESIA MENGUNAKAN MASK R-CNN DAN REGRESI LINEAR BERGANDA

Oleh
Nadya Aditama
NIM: 23520039
(Program Studi Magister Informatika)

Institut Teknologi Bandung

Menyetujui
Tim Pembimbing

Tanggal 24 Januari 2022

Ketua

A handwritten signature in black ink, appearing to read 'Rinaldi Munir', with a small circled 'M' at the end. The signature is written over a horizontal line.

(Dr. Ir. Rinaldi Munir, M. T.)

PEDOMAN PENGGUNAAN TESIS

Tesis Magister yang tidak dipublikasikan terdaftar dan tersedia di Perpustakaan Institut Teknologi Bandung, dan terbuka untuk umum dengan ketentuan bahwa hak cipta ada pada penulis dengan mengikuti aturan HaKI yang berlaku di Institut Teknologi Bandung. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau peringkasan hanya dapat dilakukan seizin penulis dan harus disertai dengan kaidah ilmiah untuk menyebutkan sumbernya.

Sitasi hasil penelitian Tesis ini dapat ditulis dalam bahasa Indonesia sebagai berikut:

Aditama, N. (2022): *Estimasi Kalori Pada Jajanan Pasar Di Indonesia Menggunakan Mask R-CNN Dan Regresi Linear Berganda*, Tesis Program Magister, Institut Teknologi Bandung.

dan dalam bahasa Inggris sebagai berikut:

Aditama, N. (2022): *Indonesian Street Food Calorie Estimation Using Mask R-CNN and Multiple Linear Regression*, Master's Thesis, Institut Teknologi Bandung.

Memperbanyak atau menerbitkan sebagian atau seluruh tesis haruslah seizin Dekan Sekolah Pascasarjana, Institut Teknologi Bandung.

Dipersembahkan kepada orang tua tercinta yang senantiasa mendukung lahir dan batin.

KATA PENGANTAR

Alhamdulillah, puji syukur kita panjatkan kepada Allah SWT, atas kehendak-Nya penulis dapat menyelesaikan Tesis yang berjudul “Estimasi Kalori Pada Jajanan Pasar Di Indonesia Menggunakan *Mask* R-CNN Dan Regresi Linear Berganda”. Penulis menyadari bahwa ada beberapa pihak yang telah mendukung, membimbing, dan membantu pengerjaan dan penulisan Tesis ini. Oleh karena itu, dengan hormat penulis mengucapkan terimakasih kepada:

1. Bapak Dr. Ir. Rinaldi Munir, M.T. selaku pembimbing Tesis yang senantiasa membimbing penulis dalam pengerjaan dan penulisan Tesis.
2. Bapak Ir. Rila Mandala, MEng, PhD. dan Bapak Dr.tech. Wikan Danar Sunindyo, S.T., M.Sc. selaku penguji Tesis yang telah memberikan masukan dan perbaikan dalam Tesis ini.
3. Ibu Dr. Masayu Leylia Khodra, S.T, M.T. selaku dosen wali dari penulis yang senantiasa membantu dan membimbing penulis dalam perkuliahan.
4. Orang tua penulis, yaitu Bapak Hen Restu Adi dan Ibu Neni Herawati yang senantiasa mendukung dan memberikan motivasi dan semangat dalam pengerjaan Tesis dan perkuliahan.
5. Ibu Sri Wahyuni selaku admin Tata Usaha Magister Informatika ITB yang telah membantu penulis dalam administrasi perkuliahan.
6. Rekan-rekan S2 Informatika yang senantiasa mendukung dan membantu penulis dalam pengerjaan Tesis dan perkuliahan.

Penulis menyadari bahwa masih terdapat banyak kekurangan pada penulisan dan pengerjaan Tesis ini. Oleh karena itu penulis berharap agar pembaca dapat mengembangkan penelitian ini menjadi lebih baik dari sebelumnya. Semoga Tesis ini dapat memberikan kontribusi ilmu bagi masyarakat di Indonesia.

DAFTAR ISI

ABSTRAK	i
ABSTRACT	iii
HALAMAN PENGESAHAN	v
PEDOMAN PENGGUNAAN TESIS	vi
HALAMAN PERUNTUKAN	vi
KATA PENGANTAR	viii
DAFTAR ISI	ix
DAFTAR LAMPIRAN.....	xi
DAFTAR GAMBAR DAN ILUSTRASI.....	xii
DAFTAR TABEL.....	xiii
DAFTAR SINGKATAN DAN LAMBANG	xv
Bab I Pendahuluan	1
I.1 Latar Belakang	1
I.2 Masalah Penelitian	3
I.3 Tujuan Penelitian	4
I.4 Hipotesis	4
I.5 Batasan Masalah	5
I.6 Metodologi.....	5
Bab II Tinjauan Pustaka.....	7
II.1 Citra Digital	7
II.2 Segmentasi	8
II.3 <i>Convolutional Neural Network</i>	10
II.4 <i>Mask Regional Convolutional Neural Network</i>	13
II.5 Regresi	16
II.6 Penelitian Terkait	17
II.6.1 He dkk. (2017)	17
II.6.2 Qi dkk. (2019).....	18
II.6.3 Abdelhardy dkk. (2019)	19
II.6.4 Yogaswara dkk. (2019).....	20
II.6.5 Chiang dkk. (2019)	22
Bab III Analisis Masalah dan Rancangan Solusi	24
III.1 Analisis Masalah	24
III.2 Rancangan Solusi	25
III.3 Alur Pembangunan Solusi	27
III.3.1 Akuisisi <i>Dataset</i>	28
III.3.2 Model Deteksi	30
III.3.3 Model Estimasi	30
III.3.4 Model Gabungan.....	31
III.3.5 Evaluasi Model	33

Bab IV	Implementasi dan Uji Coba.....	36
IV.1	<i>Dataset</i>	36
IV.2	Eksperimen Modul Deteksi	38
IV.3	Eksperimen Model Estimasi.....	40
IV.4	Eksperimen Modul Gabungan.....	43
	IV.4.1 Pengujian dengan data uji yang mengandung objek tumpang tindih	44
	IV.4.2 Pengujian dengan data uji yang seluruhnya tidak tumpang tindih	56
IV.5	Hasil Rekap Eksperimen Modul Gabungan	63
 Bab V	 Kesimpulan dan Saran	 66
	V.1 Kesimpulan	66
	V.2 Saran	67
 DAFTAR PUSTAKA		 69
LAMPIRAN	71

DAFTAR LAMPIRAN

Lampiran A Contoh Dataset	72
A1. Gambar yang Mengandung Objek Tumpang Tindih	72
A2. Gambar yang Mengandung Objek Tidak Tumpang Tindih.....	72
 Lampiran B Distribusi Objek.....	74
B1. Distribusi Objek di Data Latih	74
B2. Distribusi Objek di Data Validasi	74
B3. Distribusi Objek di Data Uji.....	74

DAFTAR GAMBAR DAN ILUSTRASI

Gambar II.1	Representasi citra dalam bentuk pixel (a) citra <i>array</i> dua dimensi (b) citra <i>array</i> tiga dimensi.....	7
Gambar II.2	Ilustrasi jenis segmentasi (a) gambar awal (b) <i>semantic segmentation</i> (c) <i>instance segmentation</i> (d) <i>panoptic segmentation</i> (Chen dkk., 2020)	8
Gambar II.3	Bentuk <i>amodal instance segmentation</i> (Qi dkk., 2019)	9
Gambar II.4	Arsitektur umum pada <i>Convolutional Neural Network</i> (Alom dkk., 2018).....	10
Gambar II.5	Bentuk <i>residual learning</i> (He dkk., 2016).....	11
Gambar II.6	Sebelah kiri adalah bentuk arsitektur ResNet dan sebelah kanan adalah bentuk arsitektur ResNeXt (Xie dkk., 2017).....	12
Gambar II.7	Arsitektur Mask R-CNN secara umum (Gonzalez dkk., 2019).....	14
Gambar II.8	Ilustrasi arsitektur backbone dengan FPN (Lin dkk., 2017).....	15
Gambar II.9	Bentuk arsitektur pada cabang mask (He dkk., 2017).....	16
Gambar II.10	Hasil segmentasi makanan (Yogaswara dkk., 2019).....	21
Gambar III.1	Contoh anotasi pada bagian objek yang tertumpang tindih	25
Gambar III.2	Proses <i>Mask R-CNN</i> secara detail dengan contoh gambar masukan	26
Gambar III.3	Alur Pembangunan Solusi.....	28
Gambar III.4	Contoh Pengambilan Gambar	29
Gambar III.5	Contoh pengambilan gambar dengan posisi objek tumpang tindih.	29
Gambar III.6	Alur pelatihan arsitektur Mask R-CNN	30
Gambar III.7	Alur pembangunan model regresi linear berganda	30
Gambar III.8	Alur estimasi kalori secara keseluruhan.....	31
Gambar III.9	Proses estimasi kalori dengan menggunakan model regresi linear dengan fitur luas area.....	32
Gambar III.10	Proses estimasi kalori dengan menggunakan model regresi linear dengan fitur yang diusulkan.	33
Gambar IV.1	Pengambilan gambar pada masing-masing item makanan dilakukan sebanyak 10 kali	36

DAFTAR TABEL

Tabel II.1	Arsitektur ResNet-50 dan ResNeXt-50 secara keseluruhan (Xie dkk., 2017).....	12
Tabel II.2	<i>Error</i> klasifikasi ResNeXt, Inception, dan Resnet pada kasus ImageNet-1K.	13
Tabel II.3	Hasil evaluasi <i>object detection</i> COCO (He dkk., 2017).....	18
Tabel II.4	Hasil evaluasi arsitektur <i>backbone</i> dalam model Mask R-CNN (He dkk., 2017).....	18
Tabel II.5	Perbandingan model instance segmentation dalam task <i>amodal instance segmentation</i> (Qi dkk., 2019).....	19
Tabel II.6	Informasi statistik regresi pada penelitian (Abdelhady dkk., 2019)19	
Tabel II.7	Hasil Evaluasi Mask R-CNN dengan <i>backbone</i> ResNet-101 (Yogaswara dkk., 2019)	20
Tabel II.8	Perbandingan <i>error</i> antara <i>ground truth</i> dengan hasil prediksi (Yogaswara dkk., 2019)	21
Tabel II.9	Nilai <i>Precision</i> , <i>Recall</i> , dan <i>F1 Measure</i> dari NMS dan <i>Union Postprocessing</i> (Chiang dkk., 2019)	22
Tabel II.10	Nilai <i>absolute error</i> dan <i>relative error</i> pada setiap kategori makanan (Chiang dkk., 2019)	23
Tabel IV.1	Informasi jumlah makanan dan gambar beserta kalori makanan per jenis.....	37
Tabel IV.2	Jumlah gambar makanan dengan posisi tumpang tindih dan tidak tumpang tindih.....	37
Tabel IV.3	Jumlah gambar makanan yang dikelompokkan berdasarkan jumlah makanan dalam satu piring dan posisinya.	37
Tabel IV.4	Pembagian data latih, data validasi, dan data uji.....	38
Tabel IV.5	Spesifikasi perangkat keras untuk pelatihan model <i>Mask R-CNN</i>	39
Tabel IV.6	Spesifikasi parameter untuk pelatihan model <i>Mask R-CNN</i>	39
Tabel IV.7	Hasil kinerja berdasarkan model <i>backbone pretrained</i> COCO.....	39
Tabel IV.8	Nilai AP per kelas pada model <i>backbone</i> ResNet-101-FPN dan model <i>backbone</i> ResNeXt-101-FPN.	40
Tabel IV.9	Pembagian data untuk pembangunan modul estimasi.....	41
Tabel IV.10	Nilai <i>R Squared</i> modul estimasi berat makanan.....	42
Tabel IV.11	Nilai MAE pada <i>ground truth</i> data uji	43
Tabel IV.12	Nilai <i>Confusion Matrix</i> berdasarkan <i>threshold</i> IoU segmentasi pada data uji yang mengandung objek tumpang tindih.....	44
Tabel IV.13	Nilai <i>Precision</i> , <i>Recall</i> , dan <i>F1 Score</i> berdasarkan <i>threshold</i> IoU segmentasi pada data uji yang mengandung objek tumpang tindih.	45
Tabel IV.14	Nilai MAE pada makanan yang terdeteksi dengan IoU 0.85 pada kasus objek tumpang tindih.	46
Tabel IV.15	Nilai <i>Confusion Matrix</i> berdasarkan jumlah makanan per kelas....	47
Tabel IV.16	Kesalahan segmentasi pada permasalahan tergabungnya segmentasi pada lebih dari satu makanan.	48
Tabel IV.17	Kesalahan segmentasi pada permasalahan deteksi <i>false positive</i>	49
Tabel IV.18	Kesalahan segmentasi pada permasalahan kualitas segmentasi	50

Tabel IV.19	Hasil deteksi makanan lebih dari empat dan objek yang tidak ada di data latih.	51
Tabel IV.20	Evaluasi kalori terhadap gambar input pada kasus yang benar	53
Tabel IV.21	Nilai <i>confusion matrix</i> berdasarkan <i>threshold</i> IoU dengan data uji yang tidak tumpang tindih.	57
Tabel IV.22	Nilai <i>precision</i> , <i>recall</i> , dan <i>F1 Score</i> berdasarkan <i>threshold</i> IoU dengan data uji yang tidak tumpang tindih.....	58
Tabel IV.23	Nilai MAE pada makanan yang terdeteksi dengan IoU 0.85.	59
Tabel IV.24	Evaluasi kalori terhadap gambar input pada kasus yang benar	60
Tabel IV.25	Rata-rata <i>F1 Score</i> pada dua skenario pengujian	63
Tabel IV.26	Rata-rata nilai MAE seluruh kelas pada kasus tumpang tindih dan kasus tidak tumpang tindih.....	64

DAFTAR SINGKATAN DAN LAMBANG

SINGKATAN	Nama	Pemakaian pertama kali pada halaman
AP	<i>Average Precision</i>	17
CNN	<i>Convolutional Neural Network</i>	2
FCN	<i>Fully Convolutional Network</i>	15
FN	<i>False Negative</i>	34
FP	<i>False Positive</i>	34
FPN	<i>Feature Pyramid Network</i>	14
IoU	<i>Intersection over Union</i>	33
R-CNN	<i>Regional Convolutional Neural Network</i>	2
MAE	<i>Mean Absolute Error</i>	42
mAP	<i>Mean Average Precision</i>	14
ResNet	<i>Residual Network</i>	11
ROI	<i>Region of Interest</i>	15
RPN	<i>Region Proposal Network</i>	15
TP	<i>True Positive</i>	34
LAMBANG		
β_n	Nilai koefisien ke-n	17
ε	Error acak	17
x_n	Variabel bebas ke-n	17
y	Variabel terikat	17

Bab I Pendahuluan

I.1 Latar Belakang

Menurut WHO (*World Health Organization*), obesitas merupakan akumulasi dari lemak berlebih yang dapat menyebabkan risiko kesehatan. Obesitas dapat mengakibatkan risiko penyakit jantung dan *stroke* yang dapat mengakibatkan kematian. Obesitas juga dapat menjadi pemicu diabetes yang dapat mengakibatkan kebutaan, amputasi anggota tubuh, dan kebutuhan cuci darah (*World Health Organization*, 2020).

Pengendalian obesitas dapat dilakukan dengan mengetahui jumlah kalori yang terdapat pada makanan, sehingga masyarakat dapat mengatur makanan berdasarkan informasi tersebut. Dalam kehidupan sehari-hari, masyarakat di Indonesia suka mengonsumsi jajanan pasar, seperti kue basah dan gorengan. Namun jika konsumsi jajanan pasar tersebut tidak dikontrol, maka hal tersebut akan menyebabkan obesitas. Oleh karena itu, diperlukan teknologi yang dapat mengestimasi kalori dari jajanan pasar.

Salah satu cara untuk mengakses informasi makanan secara mudah adalah dengan menggunakan *task food recognition*, pengenalan makanan melalui citra. *Task* tersebut masih menjadi *task* yang menantang karena tingkat variasi makanan di dunia sangat tinggi.

Food Recognition untuk masakan di Indonesia telah diimplementasikan oleh dua penelitian, yaitu dari Wibisono dkk. (2020) yang mengenali makanan tradisional di Indonesia dan Giovany dkk. (2020) yang mengenali makanan umum di Indonesia. Namun, kedua penelitian tersebut baru sampai pada tahap pengenalan jenis makanan saja. Teknologi *food recognition* perlu dilengkapi dengan estimasi kalori agar dapat mengetahui kalori dari suatu makanan.

Penelitian terkait *food recognition* dengan estimasi kalori telah diimplementasikan oleh Ege dan Yanai (2017b). Pada penelitian tersebut dilakukan pengenalan dan

estimasi kalori secara konkuren dengan arsitektur VGG16 yang melakukan *sharing layer fully connected* untuk task pengenalan dan estimasi. Penelitian tersebut hanya mendeteksi kalori dari satu jenis makanan dan tidak memperhitungkan berat makanan. Kemudian pada penelitian Ege dan Yanai (2017a), dikembangkanlah sistem tersebut dengan menggunakan *Faster R-CNN*, sehingga dapat mendeteksi makanan yang terdapat pada lebih dari satu piring. Penelitian tersebut juga tidak memperhitungkan berat makanan.

Faster R-CNN telah diperluas menjadi *Mask R-CNN*, yaitu arsitektur CNN yang dikembangkan oleh He dkk. (2017). *Mask R-CNN* merupakan perluasan dari *Faster R-CNN* yang dapat mengenali, mendeteksi, serta mengeluarkan hasil segmentasi suatu objek dalam periode yang bersamaan, dengan fitur yang dihasilkan dari model *backbone*. *Mask R-CNN* telah diimplementasikan pada task *food recognition* dengan estimasi kalori. Penggunaan metode tersebut dalam task *food recognition* dengan estimasi kalori telah diimplementasikan oleh Yogaswara dkk. (2019) dan Chiang dkk. (2019).

Terdapat bermacam-macam pendekatan estimasi kalori pada makanan. Penelitian dari Ege dan Yanai (2017b) dan Ege dan Yanai (2017a) mengestimasi kalori makanan dengan menggunakan arsitektur CNN berbasis regresi tanpa menggunakan informasi berat. Penelitian Yogaswara dkk. (2019) mengestimasi kalori dengan menghitung volume makanan terlebih dahulu, namun penelitian tersebut hanya mengukur kalori makanan yang berbentuk persegi. Penelitian Chiang dkk. (2019) mengestimasi kalori dengan menggunakan regresi linear dengan variabel bebas luas area segmentasi dan variabel terikatnya adalah berat makanan dalam satuan gram. Berat makanan tersebut akan dikonversikan dalam satuan kalori.

Terdapat dua permasalahan dari metode yang diusulkan oleh Chiang dkk. (2019). Pertama adalah masalah fitur yang digunakan dalam model regresi linear. Fitur yang digunakan dalam model regresi linear Chiang dkk. (2019) hanya menggunakan luas area segmentasi sebagai variabel bebas. Penelitian tersebut

belum mempertimbangkan fitur geometri lainnya. Menurut penelitian Abdelhady et al. (2019) dalam mengukur bobot domba, model regresi linear antara luas area segmentasi dengan bobot domba mempunyai nilai *R Squared* yang lebih rendah dibandingkan model regresi linear berganda antara panjang dan lebar domba dengan bobot domba. Sehingga perlu dilakukan penambahan fitur lain untuk meningkatkan keakuratan pengukuran kalori.

Permasalahan yang kedua adalah adanya tumpang tindih pada objek sehingga tidak didapatkan bentuk objek secara utuh yang dapat mempengaruhi kinerja dari pengukuran bobot makanan. Oleh karena itu, dilakukan pelatihan dengan objek yang dianotasi secara amodal, yaitu anotasi segmentasi berdasarkan bentuk objek yang sesungguhnya meskipun ada objek lain yang menghalangi, sehingga diharapkan model dapat membentuk hasil segmentasi yang mensegmen bagian yang terhalang. *Mask R-CNN* juga dapat melaksanakan task tersebut. Metode ini telah diuji pada penelitian Qi dkk. (2019) dalam kasus *amodal instance segmentation*. Namun pada model ini tidak disebutkan model *backbone* yang digunakan dalam evaluasi. Oleh karena itu, pada penelitian ini juga akan diobservasi model *backbone* yang tepat pada model *Mask R-CNN* dalam *task amodal instance segmentation*.

Pada penelitian ini, akan dibangun sistem estimasi kalori pada jajanan pasar di Indonesia. Penelitian tersebut akan mengadaptasi metode *Mask R-CNN* yang dilatih dengan objek yang dianotasi secara *amodal* untuk mendapatkan bentuk dari makanan yang terhalang objek lain. Untuk mengestimasi kalori makanan akan dibangun model regresi linear berganda dengan variabel bebasnya berupa fitur geometri citra yang ditambahkan selain luas area dan variabel terikatnya adalah bobot makanan. Informasi bobot tersebut akan dikonversikan dalam satuan kalori.

I.2 Masalah Penelitian

Masalah penelitian utama dalam penelitian ini adalah bagaimana cara menghasilkan model estimasi kalori yang seakurat mungkin dalam berbagai posisi,

termasuk objek yang tumpang tindih dengan objek lain. Sub masalah dalam penelitian ini adalah sebagai berikut.

1. Bagaimana membangun *dataset* citra jajanan pasar di Indonesia dengan informasi bobot dan kalori makanan?
2. Bagaimana hasil dan evaluasi kinerja dari model yang diusulkan?

I.3 Tujuan Penelitian

Tujuan yang akan dicapai pada penelitian ini adalah sebagai berikut

1. Penelitian ini bertujuan untuk membangun *dataset* citra berupa jajanan pasar yang ada di Indonesia.
2. Penelitian ini bertujuan untuk menghasilkan dan menguji model yang dapat mengenali, mensegmentasi objek makanan, dan mengestimasi kalori pada jajanan pasar di Indonesia dalam berbagai posisi, terutama pada kasus tumpang tindih.
3. Penelitian ini bertujuan untuk menguji dan menganalisis kinerja model, baik dalam mengenali jajanan pasar di Indonesia maupun melakukan estimasi kalori.

I.4 Hipotesis

Premis-1: Berdasarkan penelitian yang dilakukan oleh Abdelhady dkk. (2019), model regresi linear antara luas area segmentasi dengan bobot domba mempunyai nilai *R Squared* yang lebih rendah dibandingkan model regresi linear berganda antara tinggi dan lebar domba dengan bobot domba, sehingga model regresi linear berganda memberikan hasil prediksi berat yang lebih bagus.

Premis-2: Berdasarkan evaluasi yang dilakukan oleh Qi dkk. (2019), *Mask R-CNN* telah terbukti dapat melakukan *task amodal instance segmentation* yang dapat mensegmentasi seluruh bagian objek, termasuk dengan bagian yang terhalang objek lain dengan menggunakan *dataset* dengan objek yang dianotasi secara *amodal*.

Berdasarkan premis yang telah disebutkan, dapat dirumuskan hipotesis sebagai berikut

Hipotesis-1: Model regresi linear berganda dapat meningkatkan keakuratan pengukuran.

Hipotesis-2: Kemampuan *Mask R-CNN* dalam melakukan task *amodal instance segmentation* dengan melatih *dataset* dengan objek yang diannotasi secara amodal dapat mengatasi permasalahan pengukuran kalori makanan dalam keadaan tumpang tindih.

I.5 Batasan Masalah

Berikut ini adalah batasan masalah yang ditentukan pada penelitian ini.

1. Penelitian berfokus pada jajanan pasar di Indonesia yang mewakili tiga bentuk makanan (bulat, persegi, dan asimetris).
2. Latar belakang yang digunakan hanya terbatas pada piring saja.
3. Dalam sebuah piring maksimal terdapat empat objek makanan untuk data latih dan data validasi, namun pada kasus pengujian terdapat pengujian lebih dari empat objek makanan.
4. Pengambilan citra makanan dilakukan secara manual dengan menggunakan *smartphone*.
5. Jarak kamera untuk setiap citra bersifat *fixed*.
6. Pengambilan gambar berfokus pada tampak atas.
7. Posisi objek yang diukur adalah posisi yang berdekatan, berjauhan, dan saling tumpang tindih.
8. Dalam kasus objek tumpang tindih, objek tertutupi secara parsial.
9. Tebal seluruh makanan per kelas sudah diseragamkan.

I.6 Metodologi

Tahap pengerjaan tesis adalah sebagai berikut

1. Persiapan *Dataset*

Pada tahap preparasi *dataset*, pengumpulan data dilakukan dengan cara mengumpulkan jajanan pasar beserta informasi kalorinya. Pengumpulan citra diambil secara manual menggunakan kamera *smartphone*, dengan posisi yang beragam. dan informasi kalori yang diambil pada penelitian ini berasal dari *database* kalori makanan dari *fatsecret.id*.

2. Desain Eksperimen

Pada tahap desain eksperimen, ditentukan skenario pengujian, parameter yang dikontrol, dan metrik evaluasi yang akan digunakan sebagai perbandingan *baseline* model.

3. Pembangunan Model

Pada tahap pembangunan model, dilakukan pembangunan model berdasarkan metode yang diusulkan.

4. Eksperimen

Pada tahap eksperimen, dilakukan pelatihan pada model dengan *dataset* yang telah didapatkan. Setelah pelatihan, akan dilakukan pengujian berdasarkan skenario yang telah ditentukan.

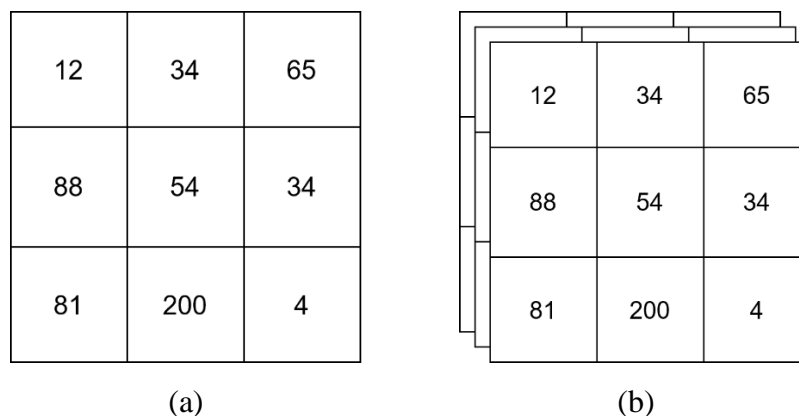
5. Evaluasi

Pada tahap evaluasi, akan dicatat kinerja model dan melakukan analisis pada keluaran dari skenario pengujian.

Bab II Tinjauan Pustaka

II.1 Citra Digital

Citra digital merupakan data yang termasuk ke dalam jenis data tidak terstruktur. Pada dasarnya, citra digital mengandung nilai *array* dua dimensi dengan keterangan jumlah baris *array* disebut *height* dan jumlah kolom *array* disebut *width* (Birchfield, 2016). Setiap elemen *array* disebut dengan *pixel*, yang merupakan singkatan dari *picture element* (Birchfield, 2016). *Pixel* tersebut mengandung nilai yang berkisar antara 0-256. Citra tidak hanya merupakan *array* dua dimensi, namun juga dapat berupa *array* tiga dimensi. Representasi citra dalam bentuk *pixel* ditunjukkan pada Gambar II.1.



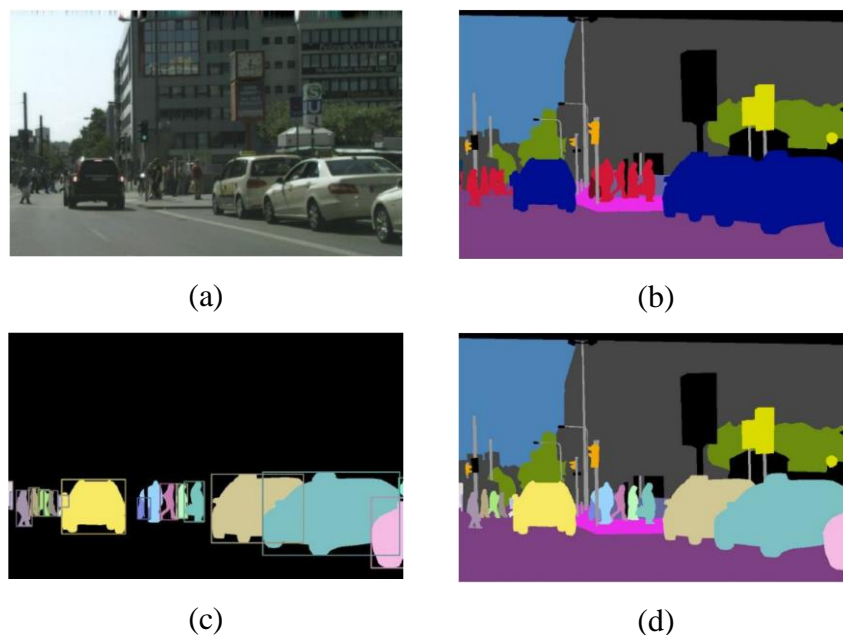
Gambar II.1 Representasi citra dalam bentuk pixel (a) citra *array* dua dimensi (b) citra *array* tiga dimensi.

Citra digital tidak hanya mempunyai ukuran panjang dan lebar, namun juga mempunyai kedalaman yang beragam seperti pada Gambar II.1 (b). Nilai kedalaman pada citra disebut dengan *channel*. *Channel* merepresentasikan warna yang terdapat dalam suatu citra. Citra yang hanya mengandung *array* dua dimensi atau satu *channel* disebut dengan citra *grayscale* atau citra keabuan. Citra keabuan mengandung warna putih, hitam, dan abu-abu. Sedangkan contoh citra dengan tiga *channel* adalah citra RGB (*Red Green Blue*), yang mengandung kombinasi warna merah, hijau, dan biru.

Saat ini, terdapat bermacam-macam jenis citra digital mulai dari yang berkualitas tinggi maupun berkualitas rendah. Data citra tersebut juga mempunyai informasi penting, seperti informasi bentuk, warna, tekstur dan sebagainya. Oleh karena itu, munculah teknik untuk memperbaiki kualitas citra dengan pengolahan citra digital dan teknik untuk mengambil informasi penting suatu citra dengan analisis citra. Pada penelitian ini, akan digunakan suatu teknik analisis citra, yaitu segmentasi.

II.2 Segmentasi

Segmentasi merupakan sebuah proses untuk menentukan *pixel* mana saja dalam sebuah citra yang merupakan proyeksi dari objek yang sama dalam suatu tempat (Birchfield, 2016). Segmentasi dapat diformulasikan sebagai masalah dalam mengklasifikasikan *pixel* dengan label semantik, atau membaginya menjadi objek yang berbeda, atau keduanya (Minaee dkk., 2021). Jenis segmentasi dibagi menjadi tiga, yaitu *semantic segmentation*, *instance segmentation*, dan *panoptic segmentation*. Ilustrasi dari ketiga jenis segmentasi ditunjukkan pada Gambar II.2.

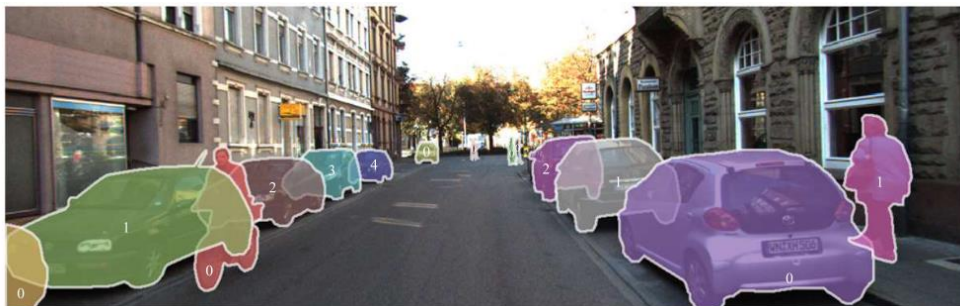


Gambar II.2 Ilustrasi jenis segmentasi (a) gambar awal (b) *semantic segmentation* (c) *instance segmentation* (d) *panoptic segmentation* (Chen dkk., 2020)

Semantic Segmentation merupakan teknik segmentasi yang melakukan pelabelan tingkat pixel dengan kumpulan kategori objek dalam seluruh *pixel* gambar (Minaee

dkk., 2021). *Semantic segmentation* hanya mengelompokkan objek tanpa melihat objek tersebut sebagai *instance* yang berbeda. Sebagai contoh pada Gambar II.2 (b), tiga mobil yang saling tumpang tindih akan dianggap sebagai satu *instance* mobil yang ditandai dengan satu *region*. *Instance segmentation* merupakan pengembangan dari *semantic segmentation*, yaitu menganggap objek yang tumpang tindih dengan kelas yang sama sebagai *instance* yang berbeda. *Instance segmentation* mensegmentasikan objek per *instance*. Dapat dilihat pada Gambar II.2 (c), tiga mobil yang saling berdekatan ditandai dengan warna *region* yang berbeda. *Panoptic Segmentation* merupakan gabungan antara *semantic segmentation* dan *instance segmentation*, yaitu melakukan klasifikasi per *pixel*, namun juga melihat objek per *instance*. Sebagai contoh pada Gambar II.2 (d), seluruh lingkungan citra akan dikelompokkan, baik yang berupa objek maupun yang berupa *background*.

Selain tiga jenis segmentasi tersebut, terdapat satu jenis segmentasi lain yang disebut *amodal instance segmentation*. *Amodal instance segmentation* melakukan segmentasi pada setiap *instance* objek yang melibatkan bagian yang teroklusi untuk meniru kemampuan manusia dalam interpretasi objek (Qi dkk., 2019). Bentuk dari *amodal instance segmentation* ditunjukkan pada Gambar II.3.

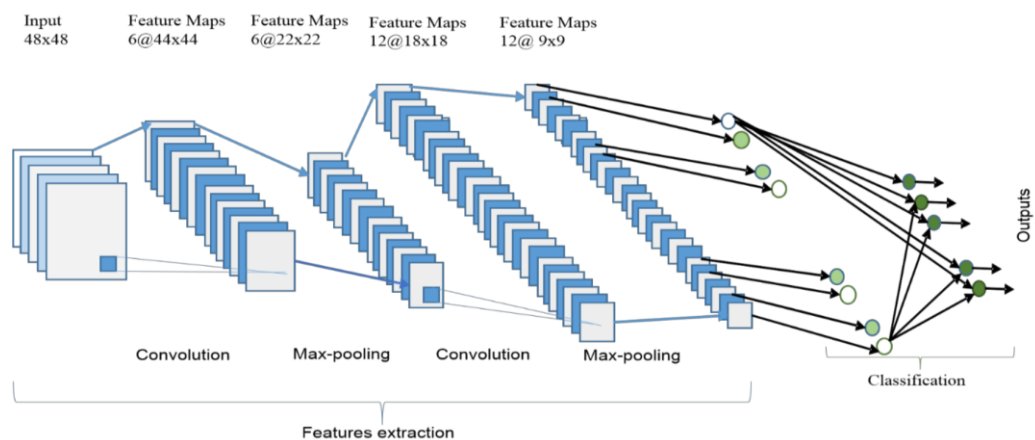


Gambar II.3 Bentuk *amodal instance segmentation* (Qi dkk., 2019)

Pada penelitian ini, akan digunakan teknik *instance segmentation* untuk melakukan segmentasi hanya pada objek makanan. Teknik *instance segmentation* yang digunakan adalah *Mask R-CNN*. Selain dapat melakukan *instance segmentation*, *Mask R-CNN* juga dapat melakukan segmentasi secara amodal yang sudah diuji pada penelitian Qi dkk. (2019).

II.3 *Convolutional Neural Network*

Convolutional Neural Network (CNN) merupakan struktur *neural network* yang pertama kali diusulkan oleh Fukushima pada tahun 1988 yang masih jarang diimplementasikan saat itu karena keterbatasan perangkat keras untuk pelatihan (Alom dkk., 2018). Namun seiring berkembangnya zaman, arsitektur CNN telah digunakan dalam melakukan pengenalan objek berbasis visual. Jenis *layer* dari CNN secara umum ditunjukkan pada Gambar II.4



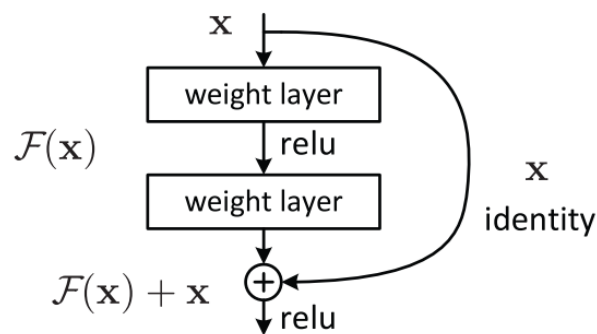
Gambar II.4 Arsitektur umum pada *Convolutional Neural Network* (Alom dkk., 2018)

Arsitektur CNN terdiri dari dua bagian, yaitu bagian *feature extraction*, dan *classification*. Pada *feature extraction*, dilakukan ekstraksi *feature map* dengan menggunakan *convolutional layer*. Pada *convolutional layer* dilakukan operasi konvolusi pada citra dengan sejumlah *kernel* yang dipelajari secara otomatis seiring dengan proses pelatihan. Hasil dari operasi konvolusi dari *convolutional layer* akan menjadi inputan untuk *layer* selanjutnya. Kemudian terdapat *pooling layer*, yaitu *layer* yang digunakan untuk mereduksi dimensi dari *feature map* yang dihasilkan dengan cara melakukan pergeseran *kernel* seperti pada *convolutional layer*, dan memangkas dimensi dengan mencari nilai *max*, *min*, atau *average*.

Setelah melakukan ekstraksi *feature map*, akan dilakukan prediksi keluaran pada *classification layer*. *Classification layer* merupakan *fully connected layer* yang

menghitung nilai keluaran dari *feature* yang telah diekstrak dari *convolutional network* (Alom dkk., 2018).

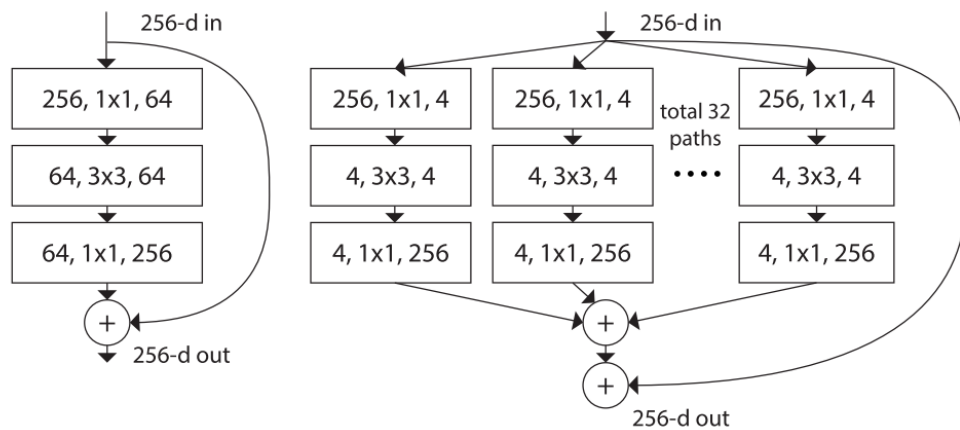
Arsitektur CNN yang akan digunakan dalam penelitian ini adalah ResNet dan ResNeXt. ResNet merupakan sebuah arsitektur CNN yang disusun secara bertumpuk dan bertujuan untuk mengatasi masalah *vanishing gradient* pada saat melatih model CNN dengan *layer* yang mempunyai kedalaman yang tinggi. ResNet memperkenalkan *residual learning* yang ditujukan pada Gambar II.5.



Gambar II.5 Bentuk *residual learning* (He dkk., 2016)

Dalam residual learning, dilakukan penjumlahan antara input dari tumpukan *layer* sebelumnya yang disebut *identity* dengan output pada tumpukan *layer* saat ini. Dalam koneksi pada Gambar II.5, tidak ada parameter tambahan dan kompleksitas komputasi (He dkk., 2016). Sehingga dalam ResNet hanya ditambahkan operasi penjumlahan antara output dari tumpukan *layer* dengan input yang sebelumnya.

Pengembangan dari arsitektur ResNet adalah ResNeXt. ResNeXt merupakan pengembangan dari ResNet yang berhasil menduduki peringkat kedua dalam ILSVRC tahun 2016 dalam kasus klasifikasi pada ImageNet (Xie dkk., 2017). ResNeXt mengenalkan sebuah *hyperparameter* baru yang disebut dengan *cardinality*. Perbedaan arsitektur ResNet dan ResNext ditujukan pada Gambar II.6.



Gambar II.6 Sebelah kiri adalah bentuk arsitektur ResNet dan sebelah kanan adalah bentuk arsitektur ResNeXt (Xie dkk., 2017)

Pada Gambar II.6, perbedaan arsitektur ResNet dan ResNeXt terletak pada adanya percabangan dengan topologi yang sama dalam setiap *stage*. Dalam ResNeXt, *cardinality* merupakan jumlah cabang yang mempunyai topologi yang sama dalam satu *stage*. Hasil ekstraksi seluruh cabang akan diaggregasikan dengan melakukan penjumlahan. Untuk melihat konsep perbedaan ResNet dan ResNeXt secara keseluruhan, struktur arsitektur ResNet-50 dan ResNeXt-50 ditujukan pada Tabel II.1.

Tabel II.1 Arsitektur ResNet-50 dan ResNeXt-50 secara keseluruhan (Xie dkk., 2017)

Stage	Output	ResNet-50	ResNeXt-50 (32 x 4d)
Conv1	112 x 112	7 x 7, 64, <i>stride</i> 2	7 x 7, 64, <i>stride</i> 2
Conv2	56 x 56	3 x 3 <i>max pool</i> , <i>stride</i> 2	3 x 3 <i>max pool</i> , <i>stride</i> 2
		$\begin{bmatrix} 1 \times 1, & 64 \\ 3 \times 3, & 64 \\ 1 \times 1, & 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, & 128 \\ 3 \times 3, & 128, C = 32 \\ 1 \times 1, & 256 \end{bmatrix} \times 3$
Conv3	28 x 28	$\begin{bmatrix} 1 \times 1, & 128 \\ 3 \times 3, & 128 \\ 1 \times 1, & 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, & 256 \\ 3 \times 3, & 256, C = 32 \\ 1 \times 1, & 512 \end{bmatrix} \times 4$
Conv4	14 x 14	$\begin{bmatrix} 1 \times 1, & 256 \\ 3 \times 3, & 256 \\ 1 \times 1, & 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, & 512 \\ 3 \times 3, & 512, C = 32 \\ 1 \times 1, & 1024 \end{bmatrix} \times 6$
Conv5	7 x 7	$\begin{bmatrix} 1 \times 1, & 512 \\ 3 \times 3, & 512 \\ 1 \times 1, & 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, & 1024 \\ 3 \times 3, & 1024, C = 32 \\ 1 \times 1, & 2048 \end{bmatrix} \times 3$
	1 x 1	<i>Global average pool</i> 1000- fc, <i>softmax</i>	<i>Global average pool</i> 1000-d fc, <i>softmax</i>
# params.		25.5 × 10	25.0 × 10

Perbandingan *error* klasifikasi antara ResNeXt dengan arsitektur lain pada data validasi ImageNet-1K ditunjukkan pada Tabel II.2.

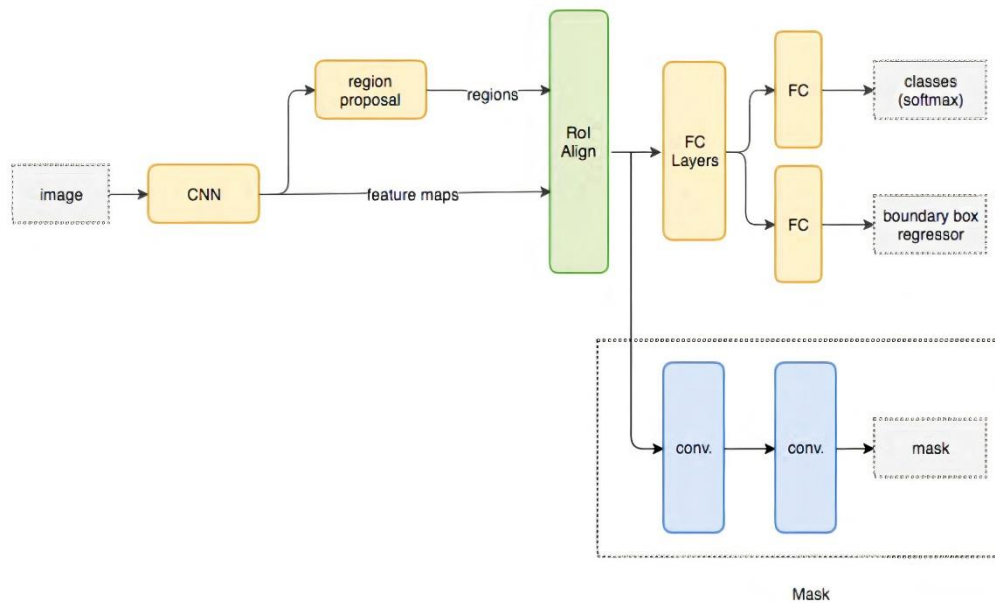
Tabel II.2 *Error* klasifikasi ResNeXt, Inception, dan Resnet pada kasus ImageNet-1K.

	224 x 224		320x320/299x299	
	Top-1 err	Top-5 err	Top-1 err	Top-5 err
ResNet-101	22.0	6.0	-	-
ResNet-200	21.7	5.8	20.1	4.8
Inception-v3	-	-	21.2	5.6
Inception-v4	-	-	20.0	5.0
Inception-ResNet-v2	-	-	19.9	4.9
ResNeXt-101 (64 x 4d)	20.4	5.3	19.1	4.4

Berdasarkan eksperimen pada Tabel II.2, dapat dibuktikan bahwa meningkatkan *cardinality* pada ResNeXt dapat meningkatkan kinerja klasifikasi secara efektif dibandingkan dengan meningkatkan kedalaman atau lebar *layer* suatu arsitektur. Hal tersebut dapat dibuktikan dengan hasil penelitian yang menyatakan bahwa arsitektur ResNeXt-101 dengan 101 *layer* dapat mencapai *error* yang lebih rendah dibandingkan ResNet-200 dengan 200 *layer*.

II.4 *Mask Regional Convolutional Neural Network*

Mask Regional Convolutional Neural Network (Mask R-CNN) adalah perluasan dari Faster R-CNN yang dikembangkan oleh penelitian (He dkk., 2017) dalam tim FAIR (*Facebook AI Research*). Mask R-CNN mengembangkan arsitektur Faster R-CNN, yang menambahkan cabang segmentasi objek yang terdeteksi. Bentuk arsitektur Mask R-CNN secara umum ditunjukkan pada Gambar II.7.



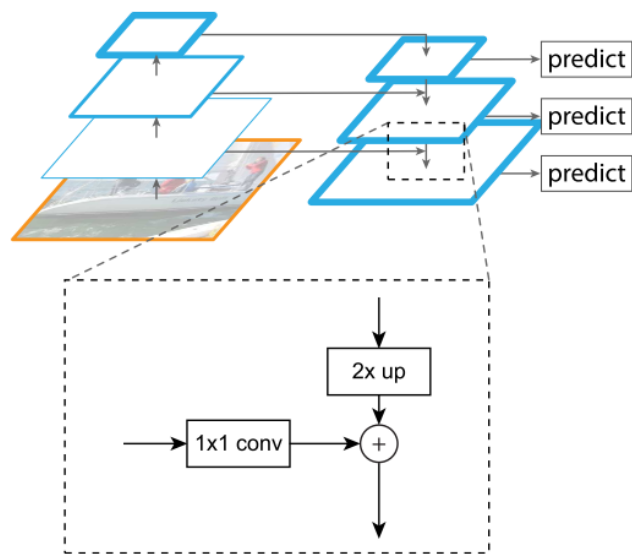
Gambar II.7 Arsitektur Mask R-CNN secara umum (Gonzalez dkk., 2019)

Bagian-bagian penting yang terdapat pada arsitektur Mask R-CNN adalah sebagai berikut

1. *Backbone* Model

Backbone merupakan arsitektur CNN utama yang digunakan sebagai ekstraksi fitur suatu citra. Pada penelitian dari He dkk. (2017), arsitektur yang diujikan adalah ResNet dan ResNeXt dengan kedalaman 50 dan 101 *layer*. Namun, tidak menutup kemungkinan bahwa terdapat arsitektur lain yang dapat dijadikan sebagai *backbone*, seperti penelitian dari Gonzalez dkk. (2019) yang menggunakan MobileNetV1. Meskipun arsitektur tersebut lebih ringan daripada ResNet, namun nilai mAP dari ResNet masih lebih baik dibandingkan MobileNetV1 dalam melakukan deteksi objek.

Semakin dalam arsitektur *neural network*, maka *feature map* yang dihasilkan akan semakin kecil. Oleh karena itu, arsitektur *backbone* dilengkapi dengan *Feature Pyramid Network* (FPN). Representasi arsitektur CNN yang ditambahkan FPN ditunjukkan pada Gambar II.8.



Gambar II.8 Ilustrasi arsitektur backbone dengan FPN (Lin dkk., 2017)

FPN bertujuan untuk menghasilkan *feature map* dengan berbagai skala. Pada Gambar II.7, jika arsitektur tersebut merupakan ResNet, masing-masing *stage* dari ResNet akan menghasilkan sebuah *feature map*.

2. RPN (*Region Proposal Network*)

RPN (*Region Proposal Network*) merupakan bagian *layer* yang melakukan pemilihan kandidat *bounding box*. Dalam RPN, dilakukan *sliding window* pada *feature map* untuk mengetahui apakah *region* yang berada di dalam *bounding box* tersebut merupakan objek atau bukan. Selain menentukan suatu objek, RPN memetakan lokasi dari *bounding box* tersebut ke gambar asli objek.

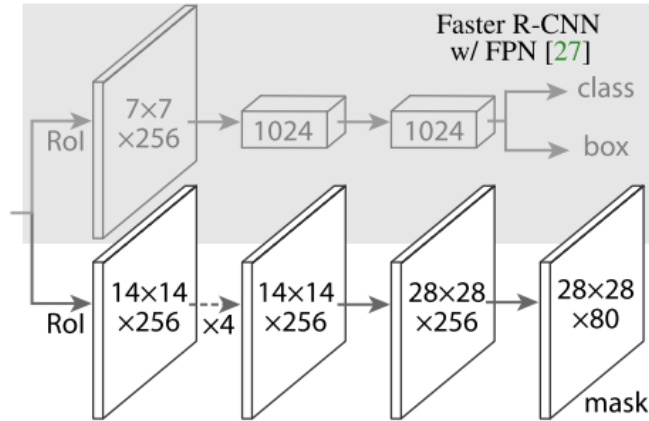
3. ROI Align

Hasil *feature map* dan hasil *region* akan di proses di bagian ROI Align. Pada ROI Align, akan dilakukan *bilinear interpolation* pada *feature map* masing-masing *region*. Hasil ROI align akan menjadi inputan untuk *Fully Connected Layer* pada bagian klasifikasi dan regresi *bounding box*. Hasil tersebut juga akan menjadi inputan pada cabang *mask*.

4. Proses Segmentasi

Pada saat melakukan segmentasi objek, teknik yang digunakan adalah FCN (*Fully Convolutional Network*). FCN melakukan klasifikasi secara *pixelwise*, yaitu dilakukan klasifikasi per $m \times m$ *pixel* untuk setiap *region*

sehingga akan membentuk segmentasi pada objek. Arsitektur FCN yang digunakan pada penelitian ini ditunjukkan pada Gambar II.9.



Gambar II.9 Bentuk arsitektur pada cabang mask (He dkk., 2017)

Pada arsitektur Gambar II.9, *layer* terdiri dari *convolutional layer* sebanyak 4 *layer* yang bersifat identik, dengan ukuran filter sebesar 3 x 3 dan jumlah *channel* sebanyak 256. Kemudian dilakukan *deconvolution* dalam *deconvolution layer* dengan ukuran filter sebesar 2 x 2. Setelah itu terdapat *convolutional layer* lagi dengan ukuran filter sebesar 1 x 1 untuk menghasilkan hasil segmentasi dari objek. Jumlah *channel* pada *layer* yang terakhir disesuaikan dengan jumlah kelas dalam kasus. Setelah itu akan dipilih satu *channel* berdasarkan prediksi kelas yang dihasilkan pada cabang klasifikasi. Misalnya kelas yang diprediksi bernilai 2 maka *channel* yang diambil adalah *channel* yang kedua.

II.5 Regresi

Regresi merupakan sebuah teknik statistik yang digunakan untuk menyelidiki dan memodelkan hubungan antar variabel (Montgomer dkk., 2021). Regresi bertujuan untuk memprediksi nilai dari variabel berdasarkan perubahan dari variabel lain. Salah satu jenis regresi adalah regresi linear. Regresi linear memodelkan relasi antara variabel dengan persamaan linear (Abdelhady dkk., 2019). Regresi linear terdiri dari dua jenis, yaitu regresi linear sederhana (*linear regression*), dan regresi linear berganda (*multiple linear regression*). Dalam regresi linear sederhana,

sebuah variabel bebas digunakan untuk memprediksi variabel respons. Jika variabel bebas yang digunakan lebih dari satu, maka model tersebut disebut dengan regresi linear berganda. Model regresi linear berganda merepresentasikan hubungan dengan persamaan (II.1)

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n + \varepsilon \quad (\text{II.1})$$

Dengan keterangan y adalah variabel respons, x adalah variabel bebas, β_0 sampai β_n adalah nilai koefisien, dan ε adalah nilai error acak. Pada penelitian ini, regresi linear berganda akan digunakan untuk mengestimasi berat makanan berdasarkan luas area segmentasi, panjang, lebar, dan perimeter objek.

II.6 Penelitian Terkait

II.6.1 He dkk. (2017)

Pada tahun 2017, penelitian He dkk. (2017) dalam tim FAIR (*Facebook AI Research*) telah membangun model Mask R-CNN dalam melakukan object detection. Model Mask R-CNN menambahkan sebuah cabang baru, yaitu cabang segmentasi pada arsitektur yang sebelumnya, yaitu Faster R-CNN.

Pada penelitian ini, dilakukan berbagai pengujian dari berbagai kasus dataset. Salah satunya adalah kasus object detection COCO. Pengujian arsitektur Mask R-CNN sendiri dilakukan pada dua *backbone*, yaitu ResNet dan ResNeXt. Pada pengujian tersebut, ditambahkan network FPN (*Feature Pyramid Network*) untuk mengekstrak fitur dari berbagai skala. Selain itu, perbandingan juga dilakukan dengan menggunakan arsitektur MNC dan FCIS, yang telah menjadi pemenang dari COCO 2015 dan 2016. Berikut ini adalah hasil kinerja Mask R-CNN dengan dalam kasus *object detection* COCO berdasarkan nilai AP (*Average Precision*) dari hasil segmentasi pada penelitian He dkk. (2017) yang ditunjukkan pada Tabel II.3.

Tabel II.3 Hasil evaluasi *object detection* COCO (He dkk., 2017)

	<i>backbone</i>	AP	AP50	AP75	APS	APM	APL
MNC	ResNet-101-C4	24.6	44.3	24.8	4.7	25.9	43.6
FCIS + OHEM	ResNet-101-C5-dilated	29.2	49.5	-	7.1	31.3	50.0
FCIS++ OHEM	Resnet-101-C5-dilated	33.6	54.5	-	-	-	-
Mask R-CNN	ResNet-101-C4	33.1	54.9	34.8	12.1	35.6	51.1
	ResNet-101-FPN	35.7	58.0	37.8	15.5	38.1	52.4
	ResNeXt-101-FPN	37.1	60.0	39.4	16.9	39.9	53.5

Berdasarkan perbandingan pada Tabel II.3., Mask R-CNN dengan *backbone* ResNeXt-101 yang ditambahkan dengan FPN dapat melakukan segmentasi yang paling baik diantara metode-metode lainnya. Selain itu, pada penelitian ini juga melakukan observasi dengan mengganti arsitektur *backbone* dalam model Mask R-CNN dengan hasil yang ditujukan oleh Tabel II.4.

Tabel II.4 Hasil evaluasi arsitektur *backbone* dalam model Mask R-CNN (He dkk., 2017)

<i>backbone</i>	AP	AP50	AP75
ResNet-50-C4	30.3	51.2	31.5
ResNet-101-C4	32.7	54.2	34.3
ResNet-50-FPN	33.6	55.2	35.3
ResNet-101-FPN	35.4	57.3	37.5
ResNeXt-101-FPN	36.7	59.5	38.9

Pada hasil tersebut, dapat dilihat bahwa semakin dalam jaringan pada arsitektur *backbone*, maka hasil kinerja akan semakin membaik.

II.6.2. Qi dkk. (2019)

Qi dkk. (2019) membuat sebuah dataset yang disebut KINS (KITTI INStance dataset). KINS merupakan sebuah dataset citra yang diaugmentasi dari dataset KITTI yang dianotasi secara amodal, yaitu menganotasi objek yang saling tumpang tindih secara lengkap. Dalam penelitian ini dilakukan pengujian dengan beberapa model *instance segmentation* dalam melakukan task *Amodal Instance Segmentation* dengan menggunakan dataset tersebut. Hasil eksperimen ditujukan pada Tabel II.5.

Tabel II.5 Perbandingan model instance segmentation dalam task *amodal instance segmentation* (Qi dkk., 2019)

Model	Amodal Seg (mAP)
MNC	18.5
FCIS	23.5
Mask R-CNN	29.3
PANet	30.4

Berdasarkan hasil eksperimen tersebut, Mask R-CNN dapat menghasilkan hasil yang *reasonable* dalam melakukan *amodal instance segmentation*, meskipun nilai mAP dari metode tersebut masih dikalahkan oleh PANet. Namun, penelitian tersebut tidak menyebutkan *backbone* model yang digunakan dalam evaluasi model Mask R-CNN.

II.6.3 Abdelhardy dkk. (2019)

Penelitian mengenai regresi linear dalam memprediksi berat suatu objek berdasarkan citra telah dilakukan oleh Abdelhady dkk. (2019). Pada penelitian tersebut, dilakukan prediksi bobot domba berdasarkan informasi tinggi dan lebar domba dengan menggunakan regresi linear berganda. Statistik regresi dari tinggi dan lebar domba ditunjukkan pada Tabel II.6.

Tabel II.6 Informasi statistik regresi pada penelitian (Abdelhady dkk., 2019)

<i>Multiple R</i>	0.99673
<i>R Squared</i>	0.993471
<i>Adjusted R Squared</i>	0.983793
<i>Standard Error</i>	3.586779

Berdasarkan tabel tersebut, *Multiple R* merupakan korelasi yang menunjukkan seberapa kuat suatu hubungan linear. *R Squared* menunjukkan seberapa dekat data dengan garis regresi. *Adjusted R Squared* menunjukkan kekuatan *explanatory* dari persamaan regresi. *Standard Error* menunjukkan standar deviasi dari *error* (Abdelhady et al., 2019). Pada Tabel II.6, dapat dilihat bahwa nilai tinggi dan lebar domba berkorelasi kuat dengan nilai bobot domba berdasarkan nilai *Multiple R*. Selain itu, jarak data dengan garis regresi sangat dekat berdasarkan nilai *R Squared*.

Penelitian ini menghasilkan akurasi sebesar 98.75%, dengan nilai *R Squared* sebesar 0.99. Penelitian tersebut juga membandingkan pengukuran bobot domba dengan menggunakan informasi luas segmentasi domba. Namun model regresi tersebut menghasilkan nilai *R Squared* yang sangat kecil, yaitu 0.21702. Sehingga dari penelitian tersebut dapat disimpulkan bahwa penggunaan fitur geometri lain selain luas segmentasi dapat meningkatkan kinerja pengukuran bobot domba.

II.6.4 Yogaswara dkk. (2019)

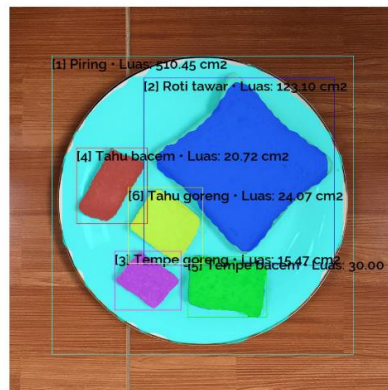
Penelitian dari Yogaswara dkk. (2019) melakukan task *food recognition* dengan menggunakan *Mask R-CNN* untuk melakukan pengenalan dan segmentasi pada citra makanan. Pada penelitian tersebut, estimasi kalori dilakukan dengan mencari volume suatu makanan terlebih dahulu dengan mengalikan luas area segmentasi makanan dengan ketebalan makanan. Diasumsikan ketebalan makanan bersifat konstan per kategori. Kemudian volume akan dikalikan dengan massa jenis makanan yang telah diukur sebelumnya untuk mendapatkan berat makanan. Hasil berat makanan tersebut akan dikonversikan dalam satuan kalori.

Pada penelitian ini, Yogaswara dkk. melakukan evaluasi dengan menggunakan *backbone* ResNet-101. Hasil *Average Precision* dari *backbone* ResNet-101 ditunjukkan pada Tabel II.7.

Tabel II.7 Hasil Evaluasi Mask R-CNN dengan *backbone* ResNet-101 (Yogaswara dkk., 2019)

<i>backbone</i>	AP	AP50	AP75
ResNet-101	0.8941	0.9678	0.9638

Hasil evaluasi untuk estimasi kalori pada penelitian ini diukur dari hasil segmentasi yang ditunjukkan pada Gambar II.10.



Gambar II.10 Hasil segmentasi makanan (Yogaswara dkk., 2019)

Perbandingan *error* antara *ground truth* dengan hasil prediksi berdasarkan hasil segmentasi pada Gambar II.9 ditujukan pada Tabel II.8.

Tabel II.8 Perbandingan *error* antara *ground truth* dengan hasil prediksi (Yogaswara dkk., 2019)

Kelas	Area <i>Ground Truth</i> (cm2)	Massa <i>Ground Truth</i> (g)	Kalori <i>Ground Truth</i> (cal)	Area Prediksi (cm2)	Massa Prediksi (g)	Kalori Prediksi (cal)	<i>Accuracy</i> (%)
<i>Fried Tofu</i>	22.5	24	26.64	24.07	24.77	27.5	92.98
<i>Fried Tempeh</i>	16	14	33.04	15.47	13.49	31.8	96.25
<i>Braised Spiced Tofu</i>	18	33	48.51	20.72	33.45	49.2	98.58
<i>Braised Spiced Tempeh</i>	26.5	37	116.18	30.0	36.92	115.9	99.76
<i>White Bread</i>	133	38	97.28	123.1	37.93	97.1	99.81

Berdasarkan penelitian tersebut, Mask R-CNN dapat mendeteksi, mengenali, dan mensegmentasi secara baik makanan-makanan yang berada di atas piring. Selain itu, hasil estimasi kalori pada sistem ini cukup dekat dengan nilai kalori yang sesungguhnya.

II.6.5 Chiang dkk. (2019)

Penelitian Chiang dkk. (2019) melakukan task tersebut dengan Mask R-CNN untuk melakukan pengenalan dan segmentasi pada citra makanan. Penelitian ini melakukan observasi pada penggunaan NMS (*Non Maximum Suppresion*) dan *Union Postprocessing* dalam objek yang berdekatan dan sedikit *overlap*. Berikut ini adalah hasil eksperimen dari penelitian ini ditunjukan pada Tabel II.9.

Tabel II.9 Nilai *Precision*, *Recall*, dan *F1 Measure* dari NMS dan *Union Postprocessing* (Chiang dkk., 2019)

No	Kelas	NMS			<i>Union Postprocessing</i>		
		<i>Precision</i>	<i>Recall</i>	<i>F1 Measure</i>	<i>Precision</i>	<i>Recall</i>	<i>F1 Measure</i>
1	<i>Salad</i>	100	100	100	100	100	100
2	<i>Fruit</i>	100	98.19	99.09	100	98.36	99.17
3	<i>Toast</i>	95.92	92.17	94.01	100	100	100
4	<i>Egg</i>	100	100	100	100	100	100
5	<i>Sausage</i>	100	96.93	98.44	100	100	100
6	<i>Chicken Cutlet</i>	100	94.65	97.25	100	95.88	97.90
7	<i>Bacon</i>	98.98	99.49	99.23	98.98	99.49	99.23
8	<i>French Toast</i>	100	97.03	98.49	100	99.26	99.63
9	<i>Omelete</i>	100	100	100	100	100	100
10	<i>Hash Brown</i>	99.60	100	99.80	99.6	100	99.8
11	<i>Pancake</i>	97.63	95.38	96.49	100	100	100
12	<i>Ham</i>	93.38	87.58	90.38	94.08	88.82	91.37
13	<i>Patty</i>	93.71	88.16	90.85	93.71	88.16	90.85
14	<i>Sandwich</i>	96.97	94.12	90.52	96.97	94.12	95.52
15	<i>French Fries</i>	97.44	100	98.70	97.44	100	98.70
16	<i>Hamburger</i>	95.16	90.77	92.91	95.16	90.77	92.91
	Total	98.48	96.31	97.38	99.09	97.91	98.50

Berdasarkan hasil eksperimen pada Tabel II.8., penggunaan *Union Posprocessing* dapat meningkatkan kinerja deteksi pada makanan yang posisinya *overlap* dengan yang lainnya.

Dalam melakukan estimasi berat makanan, dilakukan regresi linear dengan variabel bebasnya adalah jumlah *pixel* segmentasi, dan variable terikatnya adalah berat

makanan. Agar estimasi *error* tidak terlalu besar, jumlah *pixel* dibagi dengan 10.000 dan berat makanan dibagi dengan 100. Regresi linear dilakukan pada 8 kategori makanan, yaitu *salad*, *fruit*, *french toast*, *sausage*, *bacon*, *ham*, *patty*, dan *french fries*. Evaluasi dilakukan dengan menghitung *absolute error* dan *relative error*. Nilai *error* ditunjukkan pada Tabel II.10.

Tabel II.10 Nilai *absolute error* dan *relative error* pada setiap kategori makanan (Chiang dkk., 2019)

Kategori Makanan	<i>Absolute Error</i>	<i>Relative Error</i>
<i>Salad</i>	2.71	0.34
<i>Fruit</i>	8.45	0.11
<i>Toast</i>	15.98	0.19
<i>Sausage</i>	8.00	0.11
<i>Bacon</i>	2.50	0.08
<i>Ham</i>	1.79	0.07
<i>Patty</i>	6.53	0.06
<i>French Fries</i>	9.83	0.04

Berdasarkan nilai *absolute error* dan *relative error* tersebut, hasil estimasi kalori yang dihasilkan dengan menggunakan model regresi cukup dekat dengan nilai kalori yang sesungguhnya.

Bab III Analisis Masalah dan Rancangan Solusi

III.1 Analisis Masalah

Berdasarkan studi literatur dan tujuan penelitian yang telah dijabarkan, masalah yang akan ditangani pada penelitian ini adalah sebagai berikut

- a) Ketersediaan *dataset* jajanan pasar Indonesia.

Dalam kasus *food recognition* pada makanan Indonesia, *dataset* makanan di Indonesia telah tersedia pada penelitian (Wibisono dkk., 2020) dan (Giovany dkk., 2020). Namun kedua *dataset* tersebut hanya mempunyai keterangan gambar makanan tanpa adanya informasi detail dari makanan tersebut, seperti berat makanan dan kalori makanan. Kedua *dataset* tersebut juga tidak memperhatikan porsi makanan secara detail. Oleh karena itu pada penelitian ini akan dibangun *dataset* yang berfokus pada jajanan pasar Indonesia yang menyertakan informasi berat dan kalori di dalamnya.

- b) Penggunaan fitur geometri tambahan untuk estimasi kalori makanan

Dalam kasus estimasi kalori yang sebelumnya yang dilakukan oleh (Chiang dkk., 2019), dibangun model regresi linear dengan variabel bebasnya adalah luas area segmentasi dan variabel terikatnya adalah berat makanan dan menghasilkan estimasi dengan *error* yang sangat kecil. Namun, menurut penelitian (Abdelhady dkk., 2019) mengenai pengukuran bobot domba, *multiple linear regression* yang menggunakan variabel bebas tinggi dan lebar domba mempunyai nilai *R Square* yang lebih baik dibandingkan dengan regresi linear yang mengandalkan luas area segmentasi. Oleh karena itu, pada penelitian ini akan dilakukan observasi fitur geometri yang cocok dalam membangun model *multiple linear regression* untuk estimasi kalori makanan.

- c) Oklusi objek

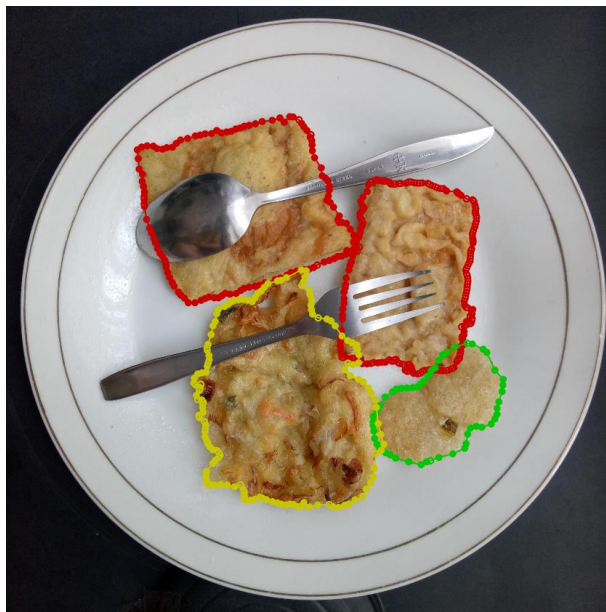
Dalam kasus segmentasi objek, terdapat permasalahan yang disebut oklusi. Oklusi merupakan suatu permasalahan dimana suatu objek akan dihalangi oleh objek lain. Hal ini dapat berpengaruh pada bentuk segmentasi yang didapatkan dari model sehingga akan terjadi estimasi kalori yang berbeda jauh dengan hasil. Oleh karena itu, akan dilakukan anotasi data secara

amodal, dengan menandai *ground truth* segmentasi pada bagian yang terhalang objek. Kemudian model *Mask R-CNN* akan dilatih dengan *dataset* tersebut untuk mendapatkan bentuk objek secara utuh.

III.2 Rancangan Solusi

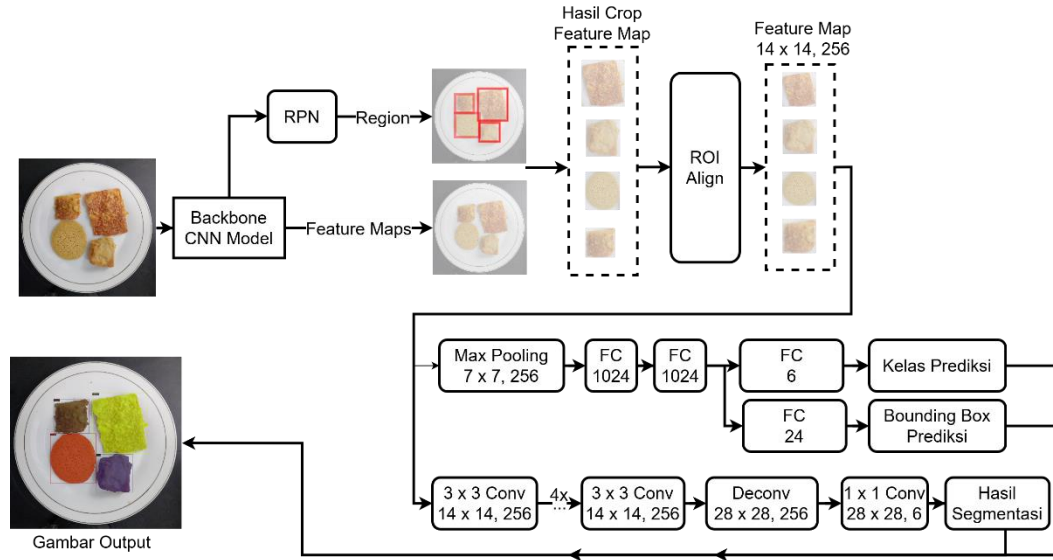
Dataset akan diambil secara manual dengan menggunakan kamera *smartphone* dengan resolusi sebesar 1000 x 1000 *pixel*. Jarak kamera pada setiap pengambilan gambar bersifat statis dan waktu pengambilan dilakukan pada siang hari. Informasi kalori yang digunakan berasal dari *database* kalori *fatsecret.co.id*, yang menampilkan satuan kilokalori makanan per 100 gram. Setelah itu, akan dilakukan anotasi *ground truth* dan kelas pada masing-masing objek makanan.

Pada kasus oklusi objek, dilakukan anotasi objek secara *amodal*, yaitu melakukan anotasi segmentasi berdasarkan bentuk objek yang sesungguhnya, sehingga tidak hanya bagian objek yang terlihat saja yang ditandai dengan *ground truth*, namun juga bagian yang terhalang dengan objek lain. Berdasarkan hal tersebut, diharapkan model dapat membentuk hasil segmentasi dengan mengisi bagian objek yang terhalang oleh objek lain. Contoh bentuk anotasi pada objek yang bertumpang tindih dapat dilihat pada Gambar III.1.



Gambar III.1 Contoh anotasi pada bagian objek yang tertumpang tindih

Pada penelitian ini, modul dibagi menjadi dua, yaitu modul pengenalan dan modul estimasi kalori. Pada modul pengenalan akan digunakan arsitektur *Mask R-CNN* yang merujuk pada Gambar II.5. Proses secara detail dari arsitektur yang diusulkan dengan menggunakan contoh gambar masukan ditujukan pada Gambar III.2.



Gambar III.2 Proses *Mask R-CNN* secara detail dengan contoh gambar masukan

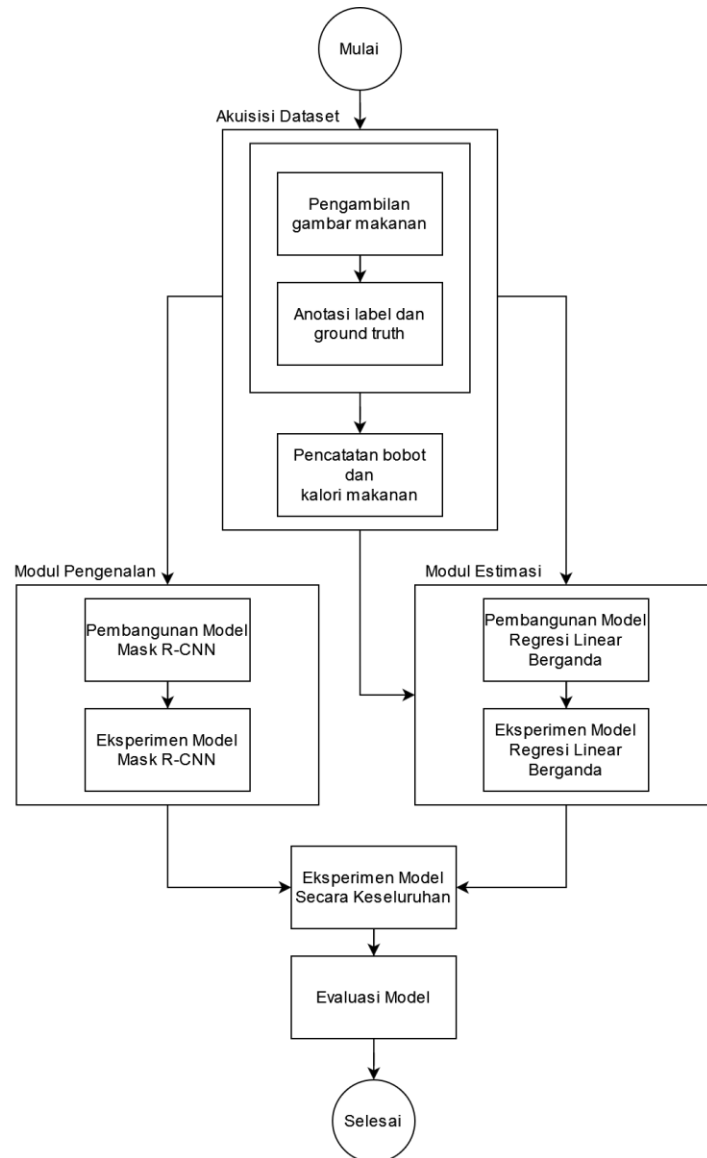
Dengan keterangan, Conv adalah *Convolutional layer*, Deconv adalah *Deconvolution Layer*, FC adalah *Fully Connected Layer*. Pada evaluasi model *Mask R-CNN* dalam dataset KINS pada task *amodal instance segmentation* yang dilakukan oleh Qi dkk. (2019), tidak disebutkan model *backbone* yang digunakan. Oleh karena itu pada penelitian ini akan dilakukan observasi dengan berbagai backbone model *Mask R-CNN* yang telah diteliti pada penelitian He dkk. (2017). Pada modul estimasi kalori, akan dibangun model regresi berganda per kelas, dikarenakan setiap makanan mempunyai karakteristik yang berbeda. Fitur geometri yang akan digunakan sebagai variabel bebas untuk model regresi ini adalah panjang, lebar, perimeter, dan luas segmentasi. Model regresi linear berganda ditujukan pada persamaan (III.1)

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \varepsilon \quad (\text{III.1})$$

Dengan keterangan, x_1 adalah panjang objek, x_2 adalah lebar objek, x_3 adalah perimeter objek, dan x_4 adalah luas area segmentasi objek. Sedangkan y adalah bobot dari makanan dalam satuan gram. Perimeter objek didapatkan dari jumlah *pixel* luar yang mengelilingi objek. Luas area segmentasi didapatkan dari jumlah *pixel* dalam bentuk segmentasi. Panjang dan lebar objek didapatkan dari jarak titik tengah panjang dan lebar *bounding box* yang berotasi, dengan asumsi bahwa panjang mempunyai jarak yang lebih besar dibandingkan lebar. Akan dilakukan observasi dengan cara menghitung nilai *R Squared* antara keempat variabel tersebut dengan variabel bobot makanan. Setelah mendapatkan hasil estimasi bobot, dilakukan konversi bobot dalam gram ke satuan kilokalori.

III.3 Alur Pembangunan Solusi

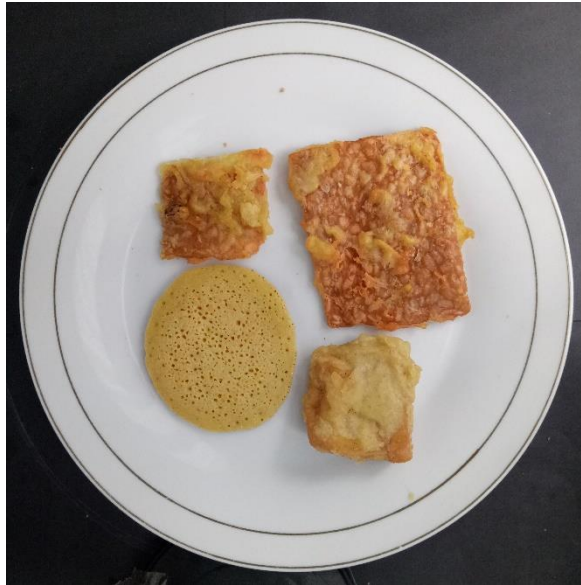
Pada penelitian ini, alur pembangunan solusi secara umum ditujukan pada Gambar III.3. Pembangunan solusi dimulai dari akusisi dataset, pembangunan modul pengenalan, pembangunan modul estimasi, kemudian eksperimen model, dan evaluasi model.



Gambar III.3 Alur Pembangunan Solusi

III.3.1 Akuisisi Dataset

Dalam modul akuisisi *dataset*, dilakukan pengambilan gambar dengan menggunakan kamera *smartphone*. Resolusi gambar adalah sebesar 1000 x 1000 *pixel*. Makanan akan diletakan di atas piring dan difoto dengan posisi tampak atas dengan jarak yang bersifat *fixed*, yaitu ± 30 cm. Waktu pengambilan gambar dilakukan di siang hari. Kemudian masing-masing citra makanan akan dilakukan anotasi label dan *ground truth* segmentasi. Label jajanan pasar yang akan dideteksi adalah tempe goreng tepung, tahu goreng, bakwan, cireng, bolu, dan serabi. Contoh pengambilan gambar ditujukan pada Gambar III.4.



Gambar III.4 Contoh Pengambilan Gambar

Dilakukan juga pengambilan gambar dengan posisi yang berbeda, yaitu dengan objek yang berjauhan, berdekatan, dan tumpang tindih. Contoh pengambilan gambar dengan oklusi ditunjukkan pada Gambar III.5.



Gambar III.5 Contoh pengambilan gambar dengan posisi objek tumpang tindih.

Setelah melakukan pengambilan gambar, setiap objek makanan akan diukur bobotnya dengan menggunakan timbangan dengan skala 0.1 gram. Kemudian akan dikonversikan ke dalam satuan kilokalori. *Database* kalori yang digunakan berasal dari *database* kalori dari *fatsecret.co.id*, salah satu *database* yang menyajikan informasi kalori dari setiap makanan. *Fatsecret.co.id* mempunyai informasi mengenai makanan dari seluruh dunia, termasuk makanan dari Indonesia, dan mempunyai informasi kalori makanan per 100 gram.

III.3.2 Model Deteksi

Pada model deteksi, dilakukan pembangunan modul terlebih dahulu dengan melakukan pelatihan Mask R-CNN dengan *backbone* ResNeXt. Alur pelatihan arsitektur Mask R-CNN ditujukan pada Gambar III.6.

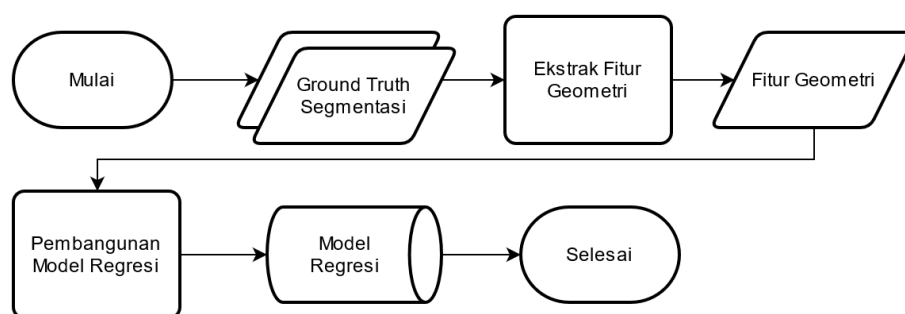


Gambar III.6 Alur pelatihan arsitektur Mask R-CNN

Pada model ini akan dilakukan eksperimen dengan mengobservasi model *backbone* pada model *Mask R-CNN*. Hasil keluaran dari modul tersebut adalah Model *Mask R-CNN* yang telah dilatih. Model *Mask R-CNN* akan mengeluarkan kelas dari makanan, *bounding box*, dan hasil segmentasi yang akan digunakan di modul estimasi kalori.

III.3.3 Model Estimasi

Pada modul estimasi, dilakukan pembangunan model regresi linear berganda. Alur pembangunan modul estimasi ditujukan pada Gambar III.7.



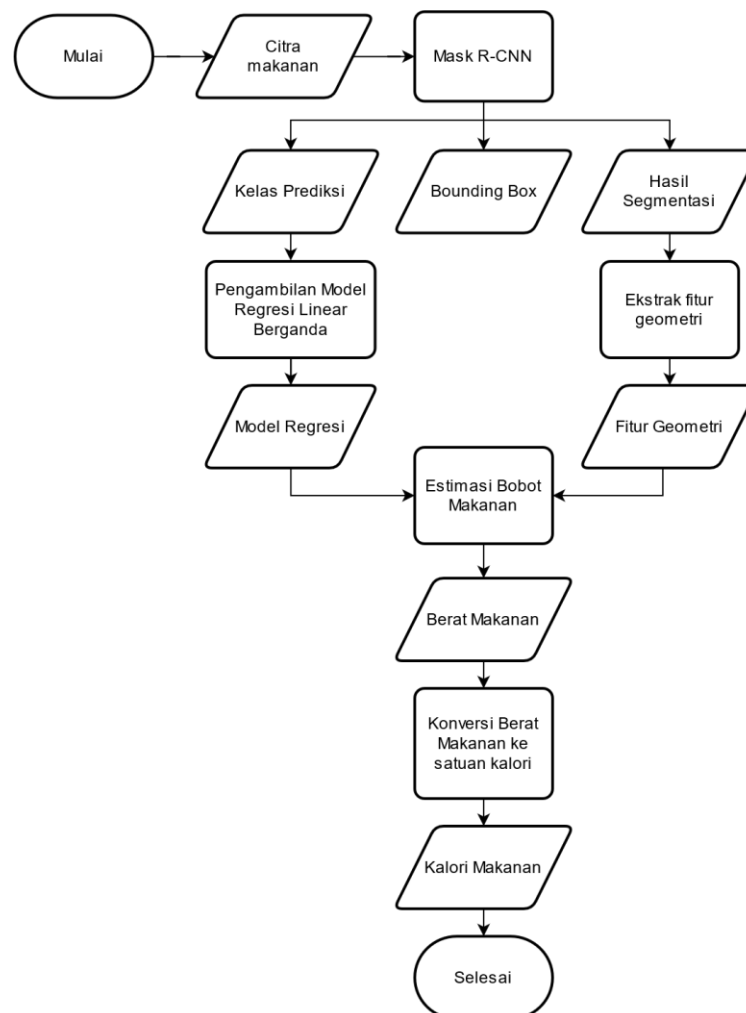
Gambar III.7 Alur pembangunan model regresi linear berganda

Pada modul ini akan dibangun model regresi linear berganda untuk masing-masing kelas makanan. Hal ini dilakukan karena masing-masing makanan mempunyai ciri khas dan massa jenis yang berbeda, sehingga dibutuhkan model regresi linear berganda yang berbeda di setiap kelas. Pertama-tama akan dilakukan ekstraksi fitur

geometri yang terdiri dari luas area segmentasi, panjang objek, lebar objek, dan perimeter objek. Kemudian dilakukan pembangunan model regresi linear berganda antara fitur geometri tersebut dengan bobot makanan. Kemudian, model regresi linear berganda akan dibandingkan kinerjanya dengan model regresi linear sederhana antara luas area segmentasi dengan bobot makanan. Hasil keluaran dari modul ini adalah model regresi linear berganda dari masing-masing kelas.

III.3.4 Model Gabungan

Setelah melakukan pembangunan model, akan dilakukan integrasi antara dua model. Alur model keseluruhan setelah dilakukan integrasi ditujukan pada Gambar III.8.



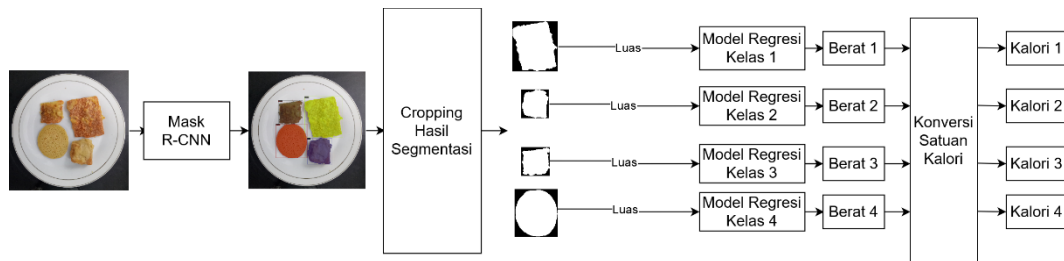
Gambar III.8 Alur estimasi kalori secara keseluruhan

Masukan dari sistem tersebut adalah berupa citra makanan. Kemudian dilakukan deteksi dengan model *Mask R-CNN* yang telah dibangun. Hasil keluaran dari model *Mask R-CNN* tersebut adalah kelas prediksi, *bounding box*, dan hasil segmentasi.

Kemudian dari kelas prediksi yang telah didapatkan, akan diambil model regresi berdasarkan nama makanan yang telah diprediksi. Kemudian dilakukan ekstraksi fitur geometri pada hasil segmentasi. Setelah itu dilakukan estimasi bobot makanan dengan menggunakan model regresi yang telah didapat. Setelah mendapatkan bobot, akan dilakukan konversi satuan gram ke kilokalori dengan persamaan (III.2)

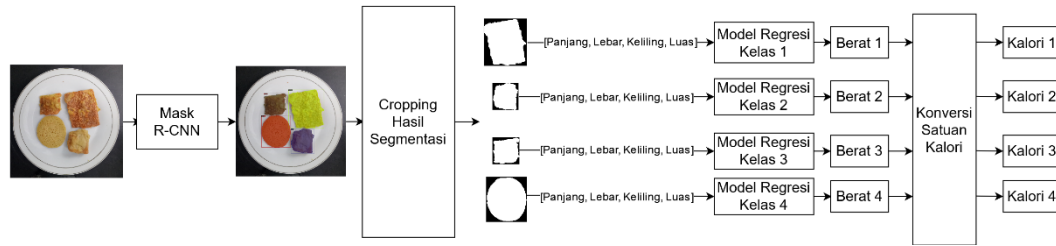
$$Kilokalori = \frac{\text{berat makanan}}{100} * \text{nilai kilokalori per 100 gram} \quad (\text{III.2})$$

Sehingga model gabungan ini akan menghasilkan nilai kalori per makanan yang berada di atas piring. Dalam proses estimasi dengan menggunakan contoh gambar input ditujukan pada Gambar III.9 untuk penggunaan model regresi linear dengan luas area segmentasi.



Gambar III.9 Proses estimasi kalori dengan menggunakan model regresi linear dengan fitur luas area

Proses estimasi dengan menggunakan contoh gambar input ditujukan pada Gambar III.9 untuk penggunaan model regresi linear berganda dengan fitur yang diusulkan.



Gambar III.10 Proses estimasi kalori dengan menggunakan model regresi linear dengan fitur yang diusulkan.

III.3.5 Evaluasi Model

Pada penelitian ini, skenario pengujian yang dilakukan adalah sebagai berikut

- Pengujian makanan yang posisinya tidak tumpang tindih
- Pengujian makanan yang posisinya tumpang tindih.

Dalam evaluasi model, akan dilakukan evaluasi pada masing-masing modul. Pada modul deteksi, akan dilakukan pengukuran kinerja dengan menggunakan metrik evaluasi mAP (*mean Average Precision*), yaitu salah satu metrik standar yang digunakan untuk mengukur kinerja *object detection*. Nilai mAP mempunyai interval dari 0 sampai 1. Semakin besar nilai mAP, maka model akan semakin baik dalam mendeteksi. Perhitungan mAP dilakukan dengan cara menghitung nilai AP (*Average Precision*) pada masing-masing kelas. Untuk menghitung AP yang dilakukan adalah menghitung IoU (*Intersection over Union*) dari masing-masing hasil deteksi dengan persamaan (III.3).

$$IoU = \frac{\text{Area yang beririsan dengan ground truth}}{\text{Area gabungan dengan ground truth}} \quad (III.3)$$

Kemudian dilakukan perhitungan *precision* dan *recall* secara keseluruhan dengan persamaan (III.4) dan (III.5)

$$Precision = \frac{TP}{TP + FP} \quad (III.4)$$

$$Recall = \frac{TP}{TP + FN} \quad (III.5)$$

Dengan keterangan TP adalah *True Positive*, yang merupakan hasil deteksi yang mempunyai nilai IoU lebih dari *threshold* yang ditentukan. FP adalah *False Positive*, yang diasumsikan sebagai hasil deteksi yang mempunyai nilai IoU yang kurang dari *threshold*. FN adalah *False Negative*, yang merupakan *ground truth* yang tidak *overlap* dengan area hasil prediksi. Berdasarkan hasil *precision* dan *recall*, akan dibangun *precision and recall curve*. Nilai AP didapatkan dengan menghitung luas daerah dibawah kurva tersebut. Nilai AP yang telah didapat dari masing-masing kelas akan dirata-ratakan sehingga disebut mAP yang ditujukan pada persamaan (III.6)

$$mAP = \sum_{i=1}^n \frac{AP_i}{n} \quad (III.6)$$

Dalam model estimasi kalori, metrik evaluasi yang digunakan adalah *R Squared*, untuk menghitung jarak kedekatan antara data dengan garis regresi dengan menggunakan persamaan (III.7).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (III.7)$$

Dengan keterangan y_i adalah nilai sebenarnya, \hat{y}_i adalah nilai prediksi, dan \bar{y} adalah rata-rata dari nilai yang sebenarnya. Dalam model gabungan, akan dihitung nilai *precision*, *recall*, dan *F1 Score* (III.10) yang bertujuan untuk mengetahui seberapa baiknya deteksi pada gambar uji.

$$F1\ Score = 2 \times \frac{precision \times recall}{precision + recall} \quad (III.8)$$

Untuk mengukur *error* dalam melakukan pengukuran kalori pada masing-masing objek makanan, akan dilakukan perhitungan *absolute error* pada hasil prediksi kalori dengan kalori yang sesungguhnya dengan menggunakan persamaan (III.9).

$$\text{absolute error} = |\text{nilai prediksi} - \text{nilai sebenarnya}| \quad (\text{III.9})$$

Akan dihitung juga *relative error* dengan menggunakan persamaan (III.10)

$$\text{relative error} = \frac{|\text{nilai prediksi} - \text{nilai sebenarnya}|}{\text{nilai sebenarnya}} \quad (\text{III.10})$$

Bab IV Implementasi dan Uji Coba

IV.1 Dataset

Pengambilan *dataset* dilakukan secara manual, dengan jarak ± 30 cm dan dilakukan di tampak atas. Ukuran gambar adalah 1000×1000 *pixel*. Jenis makanan terdiri dari 6 jenis, yaitu bakwan, bolu, cireng, serabi, tahu goreng, dan tempe goreng tepung. Makanan tersebut telah mewakili tiga bentuk makanan, yaitu bulat, persegi, dan asimetris. Setiap makanan akan dilakukan pemotretan sebanyak 10 kali dengan posisi yang berbeda-beda untuk pembangunan model estimasi kalori dengan regresi linear berganda karena dalam setiap posisi ada kemungkinan bentuk makanan akan sedikit berbeda. Contoh pengambilan gambar satu item makanan dengan 10 kali posisi yang berbeda ditunjukkan pada Gambar IV.1.



Gambar IV.1 Pengambilan gambar pada masing-masing item makanan dilakukan sebanyak 10 kali

Jumlah makanan pada tiap masing-masing jenis dan informasi kalori dari masing-masing makanan per 100 gram yang diambil dari website *fatsecret.co.id* ditunjukkan pada Tabel IV.1.

Tabel IV.1 Informasi jumlah makanan dan gambar beserta kalori makanan per jenis

ID Kategori	Nama Makanan	Kalori (per 100 gram)	Bentuk Makanan	Jumlah Makanan
0	bakwan	228 kkal	Asimetris	28
1	bolu	297 kkal	Persegi	25
2	cireng	348 kkal	Asimetris	30
3	serabi	216 kkal	Bulat	29
4	Tahu	271 kkal	Persegi	28
5	tempe	241 kkal	Persegi	28

Kemudian, dilakukan juga pengambilan dataset dengan lebih dari satu makanan, dengan posisi makanan yang tumpang tindih dan tidak tumpang tindih. Dalam satu gambar yang tumpang tindih, minimal terdapat satu makanan yang dihalangi oleh objek lain, baik berupa makanan maupun alat makan. Dalam keseluruhan gambar tumpang tindih, rasio tertutupnya objek dengan objek lain berkisar antara 10% sampai dengan 50%, walaupun terdapat beberapa objek yang rasionya melebihi 50%. Jumlah gambar ditujukan pada Tabel IV.2.

Tabel IV.2 Jumlah gambar makanan dengan posisi tumpang tindih dan tidak tumpang tindih

Posisi	Total Gambar
Tumpang Tindih	878
Tidak Tumpang Tindih	768

Pengelompokkan jumlah makanan dalam satu piring ditujukan pada Tabel IV.3.

Tabel IV.3 Jumlah gambar makanan yang dikelompokkan berdasarkan jumlah makanan dalam satu piring dan posisinya.

Jumlah Makanan	Posisi Makanan	Total Gambar
1	Tidak Tumpang Tindih	172
2	Tidak Tumpang Tindih	336
	Tumpang Tindih	155
3	Tidak Tumpang Tindih	162

Jumlah Makanan	Posisi Makanan	Total Gambar
	Tumpang Tindih	441
4	Tidak Tumpang Tindih	98
	Tumpang Tindih	274
Lebih dari 4	Tumpang Tindih	8

Untuk eksperimen dengan modul deteksi, akan dilakukan pembagian *dataset* secara acak dengan jumlah pembagian ditujukan pada Tabel IV.4. Dalam modul deteksi akan digunakan data latih dan data validasi untuk melakukan eksperimen terhadap model berdasarkan model *backbone Mask R-CNN*. Untuk pengujian model gabungan akan menggunakan data uji yang sudah diberikan informasi kalori pada setiap objeknya.

Tabel IV.4 Pembagian data latih, data validasi, dan data uji

Pembagian Data	Posisi	Total Gambar
Latih	Tumpang Tindih	652
	Tidak Tumpang Tindih	616
Validasi	Tumpang Tindih	81
	Tidak Tumpang Tindih	77
Uji	Tumpang Tindih	145
	Tidak Tumpang Tindih	75

Untuk pengujian data tumpang tindih, ditambahkan masing-masing 4 item objek per kelas yang tidak ada di data latih. Hal ini bertujuan untuk melihat seberapa bagus model dalam mensegmentasi dan melakukan estimasi kalori dengan objek yang tidak ada di data latih. Sehingga untuk pengujian dicampurkan gambar dengan objek yang tidak ada di data latih yang sudah termasuk dalam total gambar 145 untuk objek tumpang tindih.

IV.2 Eksperimen Modul Deteksi

Pada modul deteksi, dilakukan pelatihan model *Mask R-CNN* dengan melakukan observasi pada bagian model *backbone*. Spesifikasi perangkat keras yang digunakan dalam melakukan pelatihan model ditujukan pada Tabel IV.5.

Tabel IV.5 Spesifikasi perangkat keras untuk pelatihan model *Mask R-CNN*

Perangkat Keras	Spesifikasi
<i>Processor</i>	6 Core
<i>Memory</i>	56 GB
GPU	NVIDIA Tesla M60

Framework yang digunakan untuk melakukan pelatihan model *Mask R-CNN* adalah *framework* Detectron2. Kemudian dilakukan observasi pada model *pretrained backbone* *Mask R-CNN* ResNet dan ResNeXt dengan *pretrained* model dari COCO yang dilakukan pada penelitian He dkk. (2017). Konfigurasi yang digunakan ditunjukkan pada Tabel IV.6

Tabel IV.6 Spesifikasi parameter untuk pelatihan model *Mask R-CNN*

Parameter	Konfigurasi
<i>Optimizer</i>	<i>Stochastic Gradient Descent</i>
<i>Learning rate</i>	0.0001
<i>Momentum</i>	0.9
Iterasi	10000

Pada eksperimen ini, akan digunakan *optimizer* SGD (Stochastic Gradient Descent) dengan menggunakan *learning rate* 0.0001 dan momentum 0.9. Model ini dilatih dengan iterasi sebesar 10000 iterasi. Hasil kinerja model *Mask R-CNN* berdasarkan *backbone* model ditunjukkan pada Tabel IV.7

Tabel IV.7 Hasil kinerja berdasarkan model *backbone pretrained* COCO

Model <i>Backbone</i>	Waktu Pelatihan	mAP Pelatihan	mAP Validasi
ResNet-50-C4	02:38:43	89.53 %	88.30 %
ResNet-101-C4	03:33:45	90.19 %	88.84 %
ResNet-50-FPN	02:19:38	93.49 %	90.31 %
ResNet-101-FPN	03:28:23	93.95 %	91.74 %
ResNeXt-101-FPN	07:05:31	94.53%	91.47%

Berdasarkan hasil tersebut, dapat terlihat bahwa semakin dalam jaringan ResNet, maka model akan dapat memprediksi gambar dengan baik. Penambahan FPN

dalam model *backbone* juga dapat meningkatkan kinerja dari masing-masing sistem. Model ResNet 101 FPN dan Model ResNeXt 101 FPN mempunyai nilai mAP Validasi dan mAP Pelatihan yang cukup dekat. Sehingga pada kedua model ini, dilihat nilai mAP per kelas yang ditunjukkan pada Tabel IV.8.

Tabel IV.8 Nilai AP per kelas pada model *backbone* ResNet-101-FPN dan model *backbone* ResNeXt-101-FPN.

Model <i>Backbone</i>	AP kelas	AP Pelatihan	AP Validasi
ResNet-101-FPN	Bakwan	93.95	92.06
	Bolu	90.43	88.95
	Cireng	93.04	91.0
	Serabi	95.87	95.40
	Tahu	95.63	93.40
	Tempe	94.79	89.65
ResNeXt-101-FPN	Bakwan	95.18	93.65
	Bolu	91.46	82.62
	Cireng	93.42	91.29
	Serabi	96.59	94.84
	Tahu	95.72	94.61
	Tempe	94.82	91.82

Dapat terlihat bahwa model ResNeXt mempunyai nilai AP validasi yang lebih baik dibandingkan dengan model ResNet pada kelas bakwan, cireng, tahu, dan tempe. Perbedaan dari kedua model backbone ini terletak pada waktu pelatihannya. Model ResNeXt membutuhkan waktu sekitar 7 jam untuk melakukan pelatihan sedangkan model ResNet hanya membutuhkan waktu sekitar 3 jam untuk melakukan pelatihan, sehingga dapat dikatakan model ResNeXt tergolong model yang kompleks. Untuk pengecekan model kembali, akan dilakukan kembali pengujian dengan data uji pada model gabungan untuk melihat lebih lanjut peformansi model backbone ResNeXt dan ResNet.

IV.3 Eksperimen Model Estimasi

Pada model estimasi, dilakukan observasi pada fitur regresi linear berganda antara fitur yang digunakan dengan berat makanan. Konversi kalori akan dilakukan pada saat pembangunan modul gabungan. Terdapat tiga skenario model regresi linear, yaitu

1. Model regresi linear dengan fitur luas yang dilakukan oleh penelitian Chiang dkk. (2019).
2. Model regresi linear dengan fitur panjang dan lebar objek seperti pada penelitian Abdelhady dkk. (2019) yang membangun regresi linear dengan tinggi dan lebar domba.
3. Model regresi linear dengan fitur luas, panjang, lebar, dan perimeter objek, yang merupakan metode yang diusulkan.

Kemudian, pada pembangunan dan eksperimen model estimasi dilakukan pembagian data yang terdiri dari pembagian data latih dan data uji. Pembagian data ditujukan pada Tabel IV.9.

Tabel IV.9 Pembagian data untuk pembangunan modul estimasi

Nama Makanan	Pembagian Data	Total Item	Total Gambar
bakwan	Latih	23	230
	Uji	5	50
bolu	Latih	20	200
	Uji	5	50
cireng	Latih	25	250
	Uji	5	50
serabi	Latih	24	240
	Uji	5	50
tahu	Latih	23	230
	Uji	5	50
tempe	Latih	23	230
	Uji	5	50
Total Keseluruhan		168	1680

Kemudian, dilakukan pembangunan pada model regresi linear pada data latih dalam masing-masing jenis makanan. Sebagai contoh, kelas bakwan akan mempunyai model regresi linearnya sendiri dan kelas lainnya juga akan mempunyai model regresi linearnya sendiri. Untuk mengukur korelasi antara fitur dan berat dilakukan perhitungan Nilai *R Squared* pada tiap model regresi yang telah dibangun. Berat makanan ditujukan pada Tabel IV.10.

Tabel IV.10 Nilai *R Squared* modul estimasi berat makanan

Jenis Regresi Linear	Fitur	Kelas	<i>R Squared</i> (Berat)	Rata-rata <i>R Squared</i>
Regresi Linear	Luas	Bakwan	0.6581	0.785216667
		Bolu	0.9524	
		Cireng	0.7464	
		Serabi	0.9588	
		Tahu	0.5537	
		Tempe	0.8419	
Regresi Linear Berganda	Panjang dan Lebar	Bakwan	0.6687	0.79535
		Bolu	0.9470	
		Cireng	0.7857	
		Serabi	0.9404	
		Tahu	0.5783	
		Tempe	0.8396	
Regresi Linear Berganda	Panjang, Lebar, Luas dan Perimeter	Bakwan	0.6811	0.80425
		Bolu	0.9573	
		Cireng	0.7886	
		Serabi	0.9621	
		Tahu	0.5880	
		Tempe	0.8484	

Berdasarkan nilai *R Squared* tersebut, dapat terlihat bahwa penggunaan regresi linear berganda dengan fitur panjang, lebar, luas dan perimeter mempunyai nilai korelasi yang paling besar dibandingkan dengan hanya menggunakan fitur luas atau panjang dan lebar saja. Artinya, bila menggunakan fitur geometri lebih dari satu, maka pengaruh fitur pada variabel berat makanan akan semakin kuat. Rata-rata yang didapatkan dalam model *R Squared* dengan fitur yang diusulkan adalah 0.804. Meskipun nilai *R Squared* pada model regresi linear berganda lebih baik dibandingkan model regresi linear sederhana, namun nilai ini belum tentu dapat menjadi acuan yang tepat dalam pengukuran kinerja prediksi dari *ground truth* segmentasi.

Kemudian dilakukan pengujian dengan data uji yang telah dibagi sebelumnya. Fitur segmentasi diambil dari *ground truth* segmentasi dan akan dilakukan perhitungan terhadap nilai MAE (*Mean Absolute Error*), yaitu rata-rata dari nilai *absolute error* masing-masing objek. Perbandingan *error* berat makanan ditujukan pada Tabel IV.11.

Tabel IV.11 Nilai MAE pada *ground truth* data uji

Jenis Regresi Linear	Fitur	Kelas	MAE (Berat)	Rata-Rata MAE
Regresi Linear	Luas	Bakwan	6.3240	5.291933
		Bolu	3.0605	
		Cireng	7.8405	
		Serabi	4.2299	
		Tahu	4.9583	
		Tempe	5.3384	
Regresi Linear Berganda	Panjang dan Lebar	Bakwan	6.2522	5.31165
		Bolu	4.3188	
		Cireng	7.4662	
		Serabi	4.3082	
		Tahu	5.4221	
		Tempe	4.1024	
Regresi Linear Berganda	Panjang, Lebar, Luas dan Perimeter	Bakwan	6.2035	5.254567
		Bolu	3.1728	
		Cireng	7.2903	
		Serabi	4.2586	
		Tahu	5.7266	
		Tempe	4.8756	

Berdasarkan hasil tersebut, dapat terlihat bahwa regresi linear berganda dapat menurunkan *error* pengukuran berat makanan dari tiga kelas, yaitu tempe, bakwan, dan cireng. Namun, pada kasus tempe, dapat terlihat bahwa nilai MAE pada regresi linear berganda yang hanya menggunakan panjang dan lebar saja hasilnya lebih bagus dibandingkan dengan kedua model regresi linear lainnya. Walaupun nilai MAE bolu dan serabi regresi linear dengan luas lebih kecil dibandingkan regresi linear berganda dengan panjang, lebar, luas, dan perimeter, namun nilai MAE hanya berbeda sekitar 0.028 untuk serabi dan 0.1123 untuk bolu sehingga kedua model sama-sama dapat memprediksi dengan hasil yang bagus untuk kedua kelas tersebut.

IV.4 Eksperimen Modul Gabungan

Pada modul gabungan, akan dilakukan pengujian dengan menggunakan pembagian data uji yang telah dibagi sebelumnya. Model gabungan akan membandingkan model deteksi Mask R-CNN dengan *backbone* ResNet-101-FPN dan ResNeXt-101-FPN dan akan diobservasi model dengan kinerja yang paling baik. Kemudian, akan dilakukan observasi kembali mengenai model regresi linear yang paling baik

fiturnya. Oleh karena itu, skenario pengujian dibagi menjadi dua, diantaranya adalah sebagai berikut.

IV.4.1 Pengujian dengan data uji yang mengandung objek tumpang tindih

Pengujian kinerja deteksi dilakukan dengan menghitung nilai *confusion matrix* dengan *threshold* IoU segmentasi 0.85 dan 0.9. Pemilihan nilai tersebut dipertimbangkan karena dalam estimasi kalori dibutuhkan kualitas segmentasi yang mendekati *ground truth* agar mendapatkan hasil estimasi kalori yang lebih akurat. Selain itu, dilakukan observasi terhadap model *backbone* ResNet-101-FPN dan ResNeXt-101-FPN. Dalam pengujian, dilakukan perhitungan *confusion matrix* untuk mendapatkan nilai TP, FN, dan FP. Nilai *confusion matrix* ditunjukkan pada Tabel IV.12.

Tabel IV.12 Nilai *Confusion Matrix* berdasarkan *threshold* IoU segmentasi pada data uji yang mengandung objek tumpang tindih.

Mask R-CNN ResNet-101-FPN								
No	Kelas	Jumlah <i>Instance</i>	<i>Threshold</i> IoU 0.9			<i>Threshold</i> IoU 0.85		
			TP	FP	FN	TP	FP	FN
1	Bakwan	75	46	25	29	50	21	25
2	Bolu	96	72	22	24	85	9	11
3	Cireng	84	61	30	23	70	21	14
4	Serabi	77	65	15	12	70	10	7
5	Tahu	82	59	36	23	70	25	12
6	Tempe	72	47	24	25	57	14	15
Mask R-CNN ResNeXt-101-FPN								
No	Kelas	Jumlah <i>Instance</i>	<i>Threshold</i> IoU 0.9			<i>Threshold</i> IoU 0.85		
			TP	FP	FN	TP	FP	FN
1	Bakwan	75	47	24	28	52	19	23
2	Bolu	96	66	26	30	79	13	17
3	Cireng	84	65	27	19	71	21	13
4	Serabi	77	65	12	12	68	9	9
5	Tahu	82	58	28	24	70	16	12
6	Tempe	72	50	22	22	61	11	11

Berdasarkan nilai tersebut, maka didapat nilai *Precision*, *Recall*, dan *F1 Score* dari setiap kelas yang ditunjukkan pada Tabel IV.13.

Tabel IV.13 Nilai *Precision*, *Recall*, dan *F1 Score* berdasarkan *threshold* IoU segmentasi pada data uji yang mengandung objek tumpang tindih.

Mask R-CNN ResNet-101-FPN							
No	Kelas	<i>Threshold</i> IoU 0.9			<i>Threshold</i> IoU 0.85		
		<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>	<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>
1	Bakwan	0.648	0.613	0.63	0.704	0.667	0.685
2	Bolu	0.766	0.75	0.758	0.904	0.885	0.895
3	Cireng	0.67	0.726	0.697	0.769	0.833	0.8
4	Serabi	0.813	0.844	0.828	0.875	0.909	0.892
5	Tahu	0.621	0.72	0.667	0.737	0.854	0.791
6	Tempe	0.662	0.653	0.657	0.803	0.792	0.797
Rata-Rata		0.697	0.718	0.706	0.799	0.823	0.81
Mask R-CNN ResNeXt-101-FPN							
No	Kelas	<i>Threshold</i> IoU 0.9			<i>Threshold</i> IoU 0.85		
		<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>	<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>
1	Bakwan	0.662	0.627	0.644	0.732	0.693	0.712
2	Bolu	0.717	0.688	0.702	0.859	0.823	0.84
3	Cireng	0.707	0.774	0.739	0.772	0.845	0.807
4	Serabi	0.844	0.844	0.844	0.883	0.883	0.883
5	Tahu	0.674	0.707	0.69	0.814	0.854	0.833
6	Tempe	0.694	0.694	0.694	0.847	0.847	0.847
Rata-Rata		0.716	0.722	0.719	0.818	0.824	0.821

Berdasarkan pengujian tersebut, kedua model dapat mensegmentasi benda beroklusi dengan IoU antara *ground truth* dengan hasil segmentasi minimal 0.85. Model dengan *F1 Score* yang terbaik adalah model ResNeXt-101-FPN, yaitu 0.821. Oleh karena itu, dalam pengujian estimasi kalori akan dipilih model ResNeXt-101-FPN dan akan diambil deteksi makanan yang mempunyai IoU dengan *ground truth* lebih dari 0.85.

Kemudian akan dihitung nilai MAE dari setiap kelas dalam melakukan prediksi kalori. Dalam model ini diambil hasil prediksi yang melebihi IoU *threshold* 0.85 untuk diprediksi kalorinya. Nilai MAE ditunjukkan pada Tabel IV.14.

Tabel IV.14 Nilai MAE pada makanan yang terdeteksi dengan IoU 0.85 pada kasus objek tumpang tindih.

No	Kelas	Jumlah	MAE Regresi Linear		
			Fitur Luas	Fitur Panjang, Lebar	Fitur Panjang Lebar, Luas, Perimeter
1	Bakwan	52	7.519	7.953	7.464
2	Bolu	79	6.8	7.993	7.241
3	Cireng	71	9.927	11.934	10.76
4	Serabi	68	3.958	4.422	7.673
5	Tahu	70	11.832	10.393	9.862
6	Tempe	61	10.091	9.848	9.972
Rata-rata			8.354	8.757	8.829

Berdasarkan pengujian tersebut, dapat terlihat bahwa kelas bakwan dan tahu mempunyai nilai MAE regresi linear berganda dengan fitur panjang, lebar, luas, dan perimeter yang paling kecil diantara kedua model regresi lainnya. Kelas cireng, bolu dan serabi mempunyai nilai MAE yang paling kecil dalam model regresi linear dengan fitur area. Hal ini kemungkinan disebabkan karena kualitas segmentasi yang belum cukup baik sehingga penggunaan empat fitur yang diusulkan menjadi tidak terlalu bagus.

Meskipun segmentasi ResNeXt-101-FPN mempunyai *F1 Score* yang lebih baik dibandingkan ResNet-101-FPN, akan tetapi terdapat beberapa kesalahan segmentasi dalam model ResNeXt-101-FPN ini. Untuk pengecekan lebih lanjut pada model telah dilakukan juga pengujian segmentasi berdasarkan jumlah makanan dalam piring. Dalam pengujian, diujikan juga makanan yang lebih dari

empat makanan dalam piring. Nilai *confusion matrix* dari setiap pengujian jumlah makanan per kelas ditunjukkan pada Tabel IV.15.

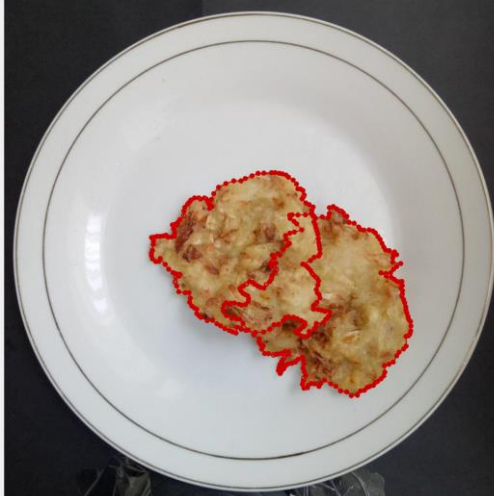
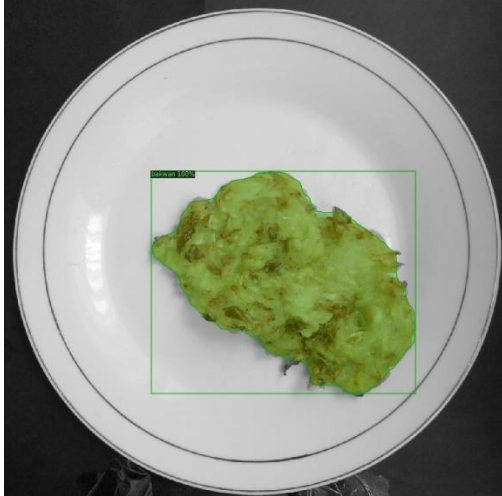


Tabel IV.15 Nilai *Confusion Matrix* berdasarkan jumlah makanan per kelas.

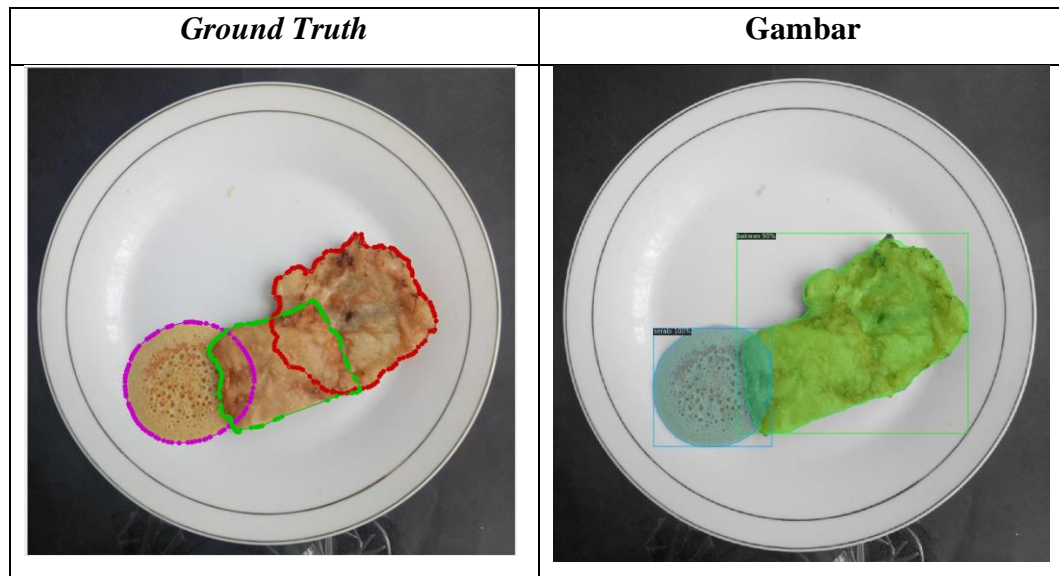
Jumlah makanan	kelas	Jumlah Instance	TP	FP	FN
2	Bakwan	9	5	3	4
	Bolu	6	5	1	1
	Cireng	13	13	1	0
	Serabi	6	6	0	0
	Tahu	6	4	2	2
	Tempe	10	10	0	0
3	Bakwan	37	28	10	9
	Bolu	33	30	2	3
	Cireng	29	26	4	3
	Serabi	41	40	2	1
	Tahu	31	27	2	4
	Tempe	27	22	6	5
4	Bakwan	25	17	6	8
	Bolu	51	38	10	13
	Cireng	26	22	4	4
	Serabi	24	19	4	5
	Tahu	29	27	10	2
	Tempe	29	24	2	5
Lebih dari 4	Bakwan	4	2	0	2
	Bolu	6	6	0	0
	Cireng	16	10	12	6
	Serabi	6	3	3	3
	Tahu	16	12	2	4
	Tempe	6	5	3	1

Berdasarkan nilai *confusion matrix* pada Tabel IV.15, dapat terlihat untuk dua kelas dengan rasio tumpang tindih yang parsial dapat mensegmentasi dengan baik objek yang tumpang tindih. Namun dapat terlihat juga bahwa untuk kelas bakwan dalam skenario jumlah makanan 2,3, dan 4 mempunyai nilai *false positive* yang besar. Hal ini disebabkan karena bentuk bakwan yang asimetris dan masalah pada pencahayaan, sehingga objek bakwan akan terlihat sebagai satu objek. Tidak hanya bakwan, namun juga kelas lainnya. Ketika sudut pencahayaan kurang pas, maka

kedua objek tersebut akan tersegmentasi sebagai satu objek. Tidak hanya di kelas yang sama, namun di kelas yang berbeda juga mempunyai masalah yang serupa dikarenakan warna objek yang mirip dan sudut cahaya yang kurang sesuai. Kesalahan segmentasi tersebut ditunjukkan pada Tabel IV.16.

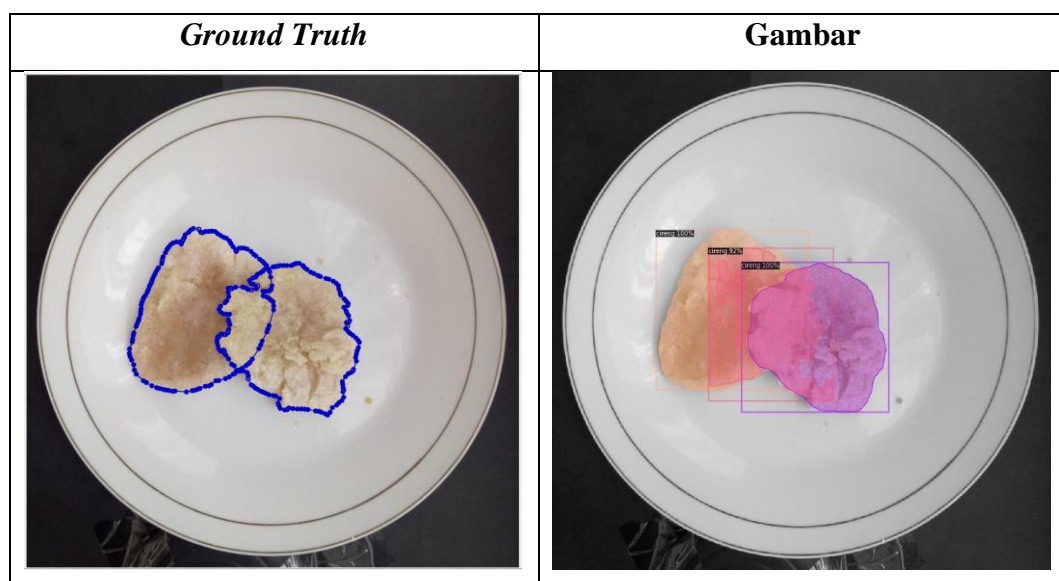
Tabel IV.16 Kesalahan segmentasi pada permasalahan tergabungnya segmentasi pada lebih dari satu makanan.

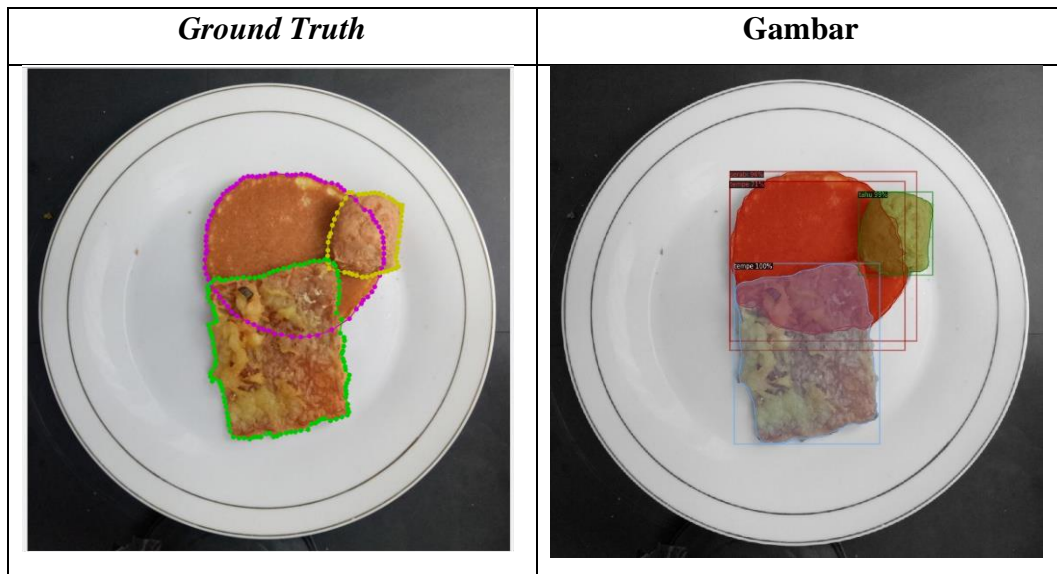
<i>Ground Truth</i>	Gambar
	
	



Kemudian, terdapat deteksi *false positive* diantara beberapa objek yang tumpang tindih, baik di kelas yang sama maupun pada kelas yang berbeda. Meskipun deteksi tersebut diabaikan dalam estimasi kalori, namun hal tersebut tergolong mengganggu tampilan hasil segmentasi objek sehingga hal tersebut merupakan permasalahan dan hal tersebut dapat meningkatkan *error* deteksi. Kemungkinan besar hal tersebut terjadi karena pelatihan yang melibatkan bagian yang tertumpang tindih oleh objek lain. Contoh gambar ditujukan pada Tabel IV.17.

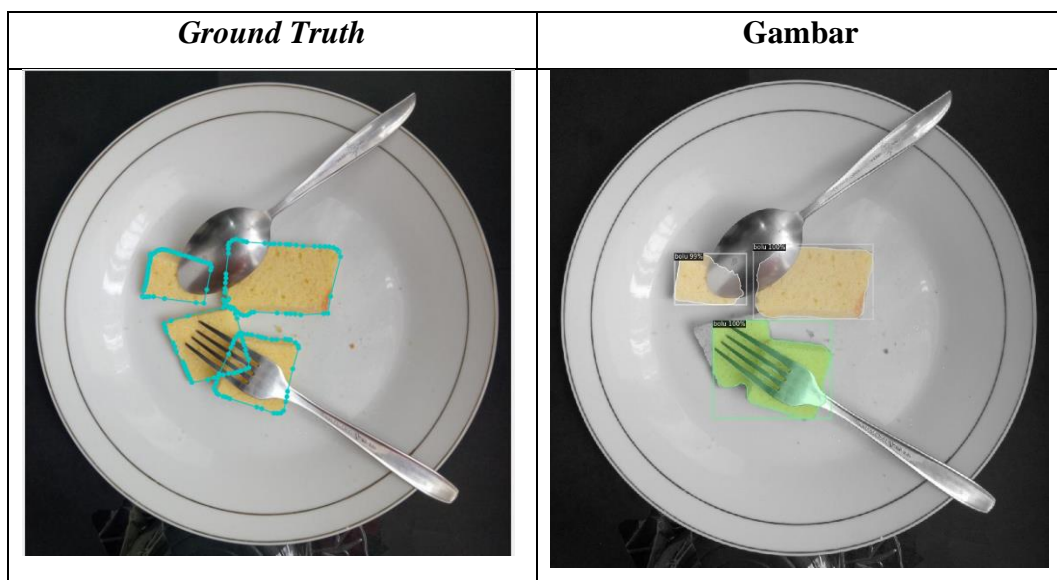
Tabel IV.17 Kesalahan segmentasi pada permasalahan deteksi *false positive*.

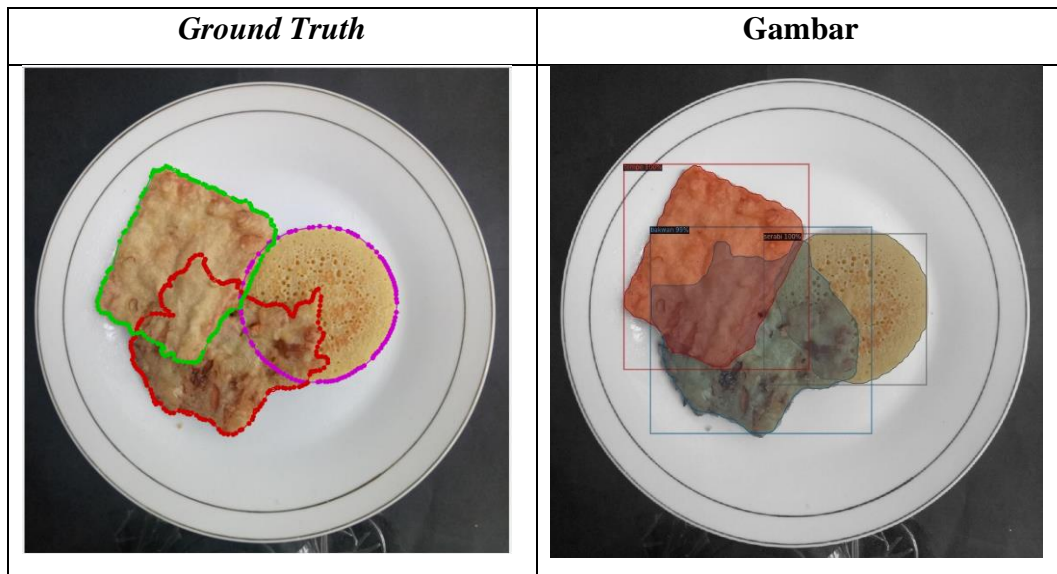




Kemudian terdapat permasalahan kualitas dari segmentasi. Beberapa gambar belum bisa menghasilkan hasil segmentasi yang baik dan masih belum dapat mendekati *ground truth*, terutama pada *item* makanan yang asimetris yang tidak ada di data latih. Contoh kesalahan ditunjukkan pada Tabel IV.18.

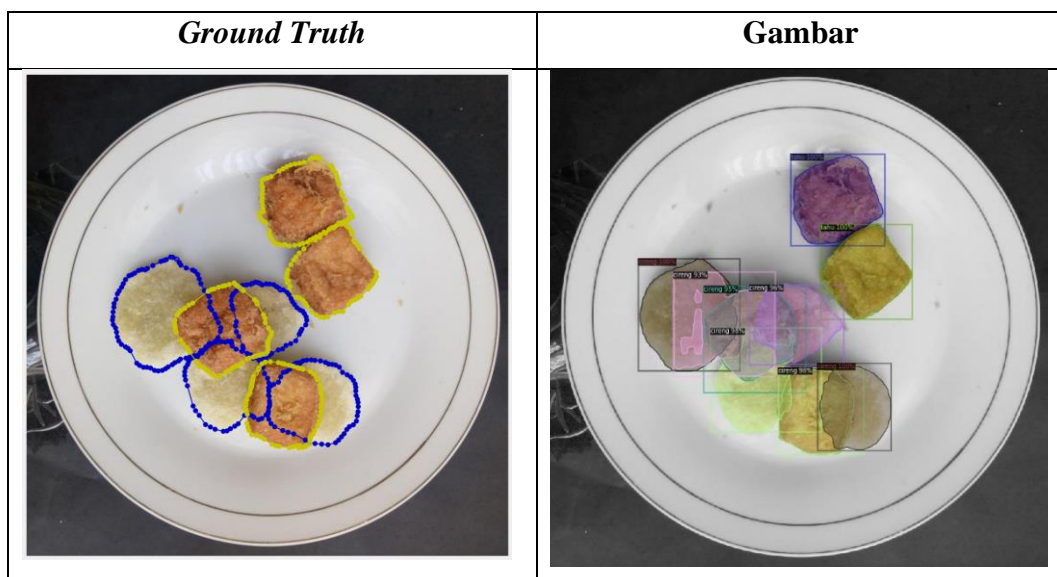
Tabel IV.18 Kesalahan segmentasi pada permasalahan kualitas segmentasi


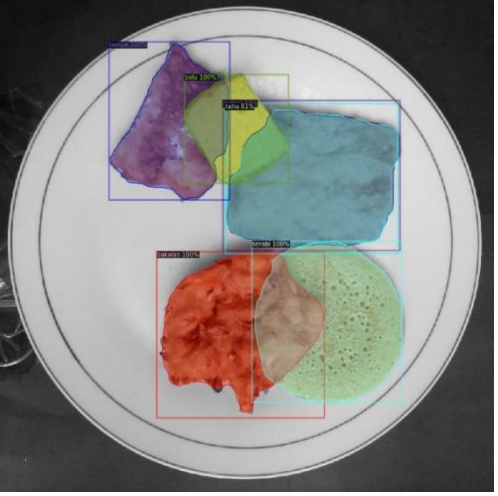




Kemudian, dilakukan juga pengujian dengan makanan yang lebih dari 4 makanan, dan dilakukan juga pengujian dengan *item* makanan yang tidak ada di data latih. Untuk hasil segmentasinya beberapa makanan sudah bagus, walaupun masih terdapat deteksi *false positive* diantara objek yang saling tumpang tindih. Contoh hasil segmentasi ditunjukkan pada Tabel IV.19.

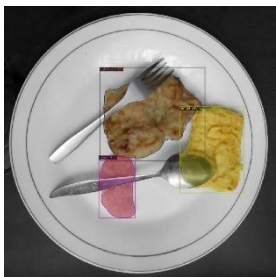
Tabel IV.19 Hasil deteksi makanan lebih dari empat dan objek yang tidak ada di data latih.

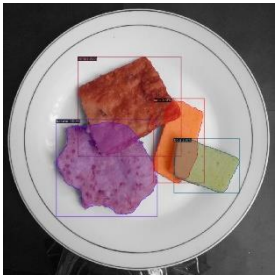


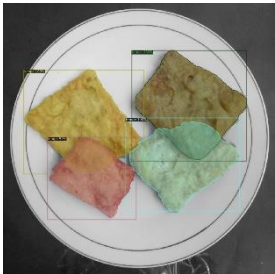
<i>Ground Truth</i>	Gambar
	

Berdasarkan hasil tersebut, dapat terlihat bahwa pada gambar yang lebih dari empat objek makanan, beberapa hasil segmentasinya masih sesuai dengan *ground truth* untuk objek yang tidak tumpang tindih. Namun, untuk objek yang tumpang tindih, terdapat deteksi *false positive* ketika ada dua atau lebih makanan yang saling tumpang tindih. Artinya, model masih dapat mengeneralisasikan data baru dengan baik walaupun masih terdapat kesalahan-kesalahan segmentasi. Beberapa contoh hasil input dan nilai kalori ditujukan pada Tabel IV.20.

Tabel IV.20 Evaluasi kalori terhadap gambar input pada kasus yang benar

Hasil Segmentasi	Informasi	Model Regresi Linear		
		Fitur Panjang Lebar	Fitur Panjang Lebar, Luas, Perimeter	Fitur Luas
	Kelas : tempe	Berat : 30.25 gram	Berat : 33.76 gram	Berat : 28.05 gram
		Kalori : 72.92 kkal	Kalori : 81.37 kkal	Kalori : 67.61 kkal
	Berat : 32.9	<i>Absolute Error :</i> 6.36	<i>Absolute Error :</i> 2.08	<i>Absolute Error :</i> 11.67
	Kalori : 79.289	<i>Relative Error :</i> 0.08	<i>Relative Error :</i> 0.02	<i>Relative Error :</i> 0.14
	Kelas : bakwan	Berat : 42.65 gram	Berat : 44.19 gram	Berat : 44.55 gram
		Kalori : 97.26 kkal	Kalori : 100.76 kkal	Kalori : 101.59 kkal
	Berat : 43.7	<i>Absolute Error :</i> 2.37	<i>Absolute Error :</i> 1.12	<i>Absolute Error :</i> 1.95
	Kalori : 99.636	<i>Relative Error :</i> 0.02	<i>Relative Error :</i> 0.01	<i>Relative Error :</i> 0.019
	Kelas : cireng	Berat : 19.38 gram	Berat : 19.89 gram	Berat : 17.16 gram
		Kalori : 67.45 kkal	Kalori : 69.22 kkal	Kalori : 59.74 kkal
	Berat : 10.3	<i>Absolute Error :</i> 31.61	<i>Absolute Error :</i> 33.38	<i>Absolute Error :</i> 23.90
	Kalori : 35.844	<i>Relative Error :</i> 0.88	<i>Relative Error :</i> 0.93	<i>Relative Error :</i> 0.66
	Kelas : bolu	Berat : 20.49 gram	Berat : 20.30 gram	Berat : 20.35 gram

Hasil Segmentasi	Informasi	Model Regresi Linear		
		Fitur Panjang Lebar	Fitur Panjang Lebar, Luas, Perimeter	Fitur Luas
	Berat : 19.5 Kalori : 57.915	Kalori : 60.85 kkal <i>Absolute Error :</i> 2.94 <i>Relative Error :</i> 0.05	Kalori : 60.29 kkal <i>Absolute Error :</i> 2.37 <i>Relative Error :</i> 0.041	Kalori : 60.46 kkal <i>Absolute Error :</i> 2.55 <i>Relative Error :</i> 0.044
	Kelas : bolu Berat : 15.0 Kalori: 44.55	Berat : 26.59 gram Kalori : 78.98 kkal <i>Absolute Error :</i> 34.43 <i>Relative Error :</i> 0.77	Berat : 24.64 gram Kalori: 73.18 kkal <i>Absolute Error :</i> 28.63 <i>Relative Error :</i> 0.64	Berat : 24.27 gram Kalori: 72.10 kkal <i>Absolute Error :</i> 27.55 <i>Relative Error :</i> 0.61
	Kelas : bakwan Berat: 53.9 Kalori: 122.892	Berat : 49.23 gram Kalori : 112.26 kkal <i>Absolute Error :</i> 10.63 <i>Relative Error :</i> 0.086	Berat : 50.65 gram Kalori : 115.49 kkal <i>Absolute Error :</i> 7.39 <i>Relative Error :</i> 0.0601	Berat : 51.22 gram Kalori : 116.79 kkal <i>Absolute Error :</i> 6.09 <i>Relative Error :</i> 0.0496
	Kelas : tempe Berat : 42.5 Kalori : 102.425	Berat: 43.01 gram Kalori: 103.66 kkal <i>Absolute Error :</i>	Berat: 46.59 gram Kalori: 112.28 kkal <i>Absolute Error :</i>	Berat: 43.55 gram Kalori: 104.96 kkal <i>Absolute Error :</i>

Hasil Segmentasi	Informasi	Model Regresi Linear		
		Fitur Panjang Lebar	Fitur Panjang Lebar, Luas, Perimeter	Fitur Luas
		1.24	9.86	2.54
		<i>Relative Error :</i> 0.01	<i>Relative Error :</i> 0.09	<i>Relative Error :</i> 0.02
	Kelas : tempe Berat : 54.7 Kalori : 131.827	Berat : 50.38 gram Kalori : 121.42 kkal <i>Absolute Error :</i> 10.39 <i>Relative Error :</i> 0.078	Berat : 56.35 gram Kalori: 135.81 kkal <i>Absolute Error :</i> 3.99 <i>Relative Error :</i> 0.030	Berat : 52.29 gram Kalori: 126.04 kkal <i>Absolute Error :</i> 5.78 <i>Relative Error :</i> 0.043
	Kelas : tempe Berat : 52.4 Kalori: 126.283	Berat : 49.91 gram Kalori : 120.30 kkal <i>Absolute Error :</i> 5.97 <i>Relative Error :</i> 0.047	Berat : 53.02 gram Kalori: 127.78 kkal <i>Absolute Error :</i> 1.49 <i>Relative Error :</i> 0.011	Berat : 49.21 gram Kalori : 118.61 kkal <i>Absolute Error :</i> 7.66 <i>Relative Error :</i> 0.060
	Kelas : Tempe Berat : 45.4 Kalori : 109.414	Berat : 45.01 gram Kalori : 108.49 kkal <i>Absolute Error :</i> 0.916 <i>Relative Error :</i> 0.008	Berat : 47.13 gram Kalori : 113.60 kkal <i>Absolute Error :</i> 4.191 <i>Relative Error :</i> 0.038	Berat : 42.70 gram Kalori : 102.91 kkal <i>Absolute Error :</i> 6.496 <i>Relative Error :</i> 0.059

Hasil Segmentasi	Informasi	Model Regresi Linear		
		Fitur Panjang Lebar	Fitur Panjang Lebar, Luas, Perimeter	Fitur Luas
	Kelas : tempe Berat : 31.9 Kalori : 76.87	Berat : 25.74 gram	Berat : 29.67 gram	Berat : 26.06 gram
		Kalori : 62.03 kkal	Kalori : 71.52 kkal	Kalori : 62.81 kkal
		<i>Absolute Error :</i> 14.84	<i>Absolute Error :</i> 5.35	<i>Absolute Error :</i> 14.06
		<i>Relative Error :</i> 0.193	<i>Relative Error :</i> 0.069	<i>Relative Error :</i> 0.182

IV.4.2 Pengujian dengan data uji yang seluruhnya tidak tumpang tindih

Selain pengujian dengan data uji yang tumpang tindih, dilakukan juga pengujian dengan data uji yang seluruhnya tidak tumpang tindih. Sama dengan pengujian dengan data uji yang tumpang tindih, dilakukan observasi *confusion matrix* terhadap *IoU threshold ground truth* dengan hasil prediksi sebesar 0.85 dan 0.9. Hasil observasi ditunjukkan pada Tabel IV.21.

Tabel IV.21 Nilai *confusion matrix* berdasarkan *threshold* IoU dengan data uji yang tidak tumpang tindih.

Mask R-CNN ResNet-101-FPN								
No	Kelas	Jumlah <i>Instance</i>	<i>Threshold</i> IoU 0.9			<i>Threshold</i> IoU 0.85		
			TP	FP	FN	TP	FP	FN
1	Bakwan	43	41	0	2	41	0	2
2	Bolu	22	22	0	0	22	0	0
3	Cireng	30	30	0	0	30	0	0
4	Serabi	23	23	0	0	23	0	0
5	Tahu	20	20	5	0	20	5	0
6	Tempe	23	20	0	3	20	0	3
Mask R-CNN ResNeXt-101-FPN								
No	Kelas	Jumlah <i>Instance</i>	<i>Threshold</i> IoU 0.9			<i>Threshold</i> IoU 0.85		
			TP	FP	FN	TP	FP	FN
1	Bakwan	43	42	0	1	42	0	1
2	Bolu	22	22	0	0	22	0	0
3	Cireng	30	30	0	0	30	0	0
4	Serabi	23	23	0	0	23	0	0
5	Tahu	20	20	1	0	20	1	0
6	Tempe	23	23	0	0	23	0	0

Berdasarkan nilai tersebut, maka didapat nilai *Precision*, *Recall*, dan *F1 Score* dari setiap kelas yang ditunjukkan pada Tabel IV.22.

Tabel IV.22 Nilai *precision*, *recall*, dan *F1 Score* berdasarkan *threshold* IoU dengan data uji yang tidak tumpang tindih.

Mask R-CNN ResNet-101-FPN							
No	Kelas	<i>Threshold</i> IoU 0.9			<i>Threshold</i> IoU 0.85		
		<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>	<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>
1	Bakwan	1	0.953	0.976	1	0.953	0.976
2	Bolu	1	1	1	1	1	1
3	Cireng	1	1	1	1	1	1
4	Serabi	1	1	1	1	1	1
5	Tahu	0.8	1	0.889	0.8	1	0.889
6	Tempe	1	0.87	0.93	1	0.87	0.93
Rata-rata		0.967	0.971	0.966	0.967	0.971	0.966
Mask R-CNN ResNeXt-101-FPN							
No	Kelas	<i>Threshold</i> IoU 0.9			<i>Threshold</i> IoU 0.85		
		<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>	<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>
1	Bakwan	1	0.977	0.988	1	0.977	0.988
2	Bolu	1	1	1	1	1	1
3	Cireng	1	1	1	1	1	1
4	Serabi	1	1	1	1	1	1
5	Tahu	0.952	1	0.976	0.952	1	0.976
6	Tempe	1	1	1	1	1	1
Rata-rata		0.992	0.996	0.994	0.992	0.996	0.994

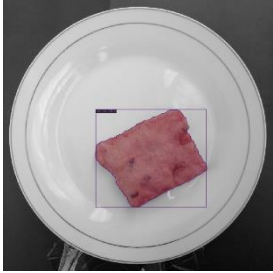
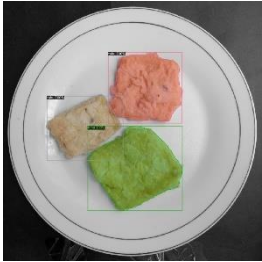
Berdasarkan hasil pengujian tersebut, kedua model dapat melakukan deteksi dan segmentasi dengan sangat baik pada masing-masing kelas dengan *threshold* IoU lebih dari 0.9. Ketika mengurangi IoU *threshold* menjadi 0.85, nilai *precision*, *recall*, dan *F1 Score* dari model tidak berubah. Artinya seluruh deteksi yang benar sudah mencapai dan melebihi IoU 0.9. Model yang dipilih adalah model ResNeXt-101-FPN. Sama seperti pengujian pada objek tumpang tindih, dilakukan juga estimasi kalori pada gambar yang berhasil terdeteksi dengan baik di atas IoU *threshold* 0.9. Hasil nilai MAE kalori ditujukan pada Tabel IV.23.

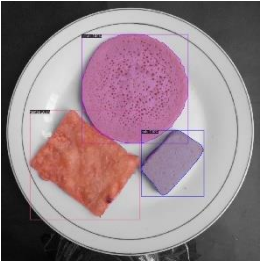
Tabel IV.23 Nilai MAE pada makanan yang terdeteksi dengan IoU 0.85.

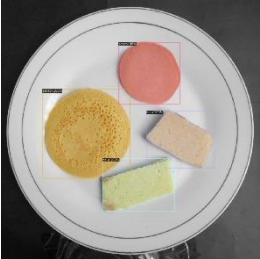
No	Kelas	Jumlah	MAE Regresi Linear		
			Fitur Luas	Fitur Panjang, Lebar	Fitur Panjang Lebar, Luas, Perimeter
1	Bakwan	42	9.041	9.353	8.618
2	Bolu	22	7.497	9.257	8.171
3	Cireng	30	19.704	18.825	17.287
4	Serabi	23	5.956	5.878	6.6
5	Tahu	20	13.958	13.657	12.958
6	Tempe	23	13.799	12.677	13.901
Rata-rata			11.659	11.608	11.256

Berdasarkan pengujian tersebut, dapat terlihat bahwa kelas bakwan, cireng, dan tahu mempunyai nilai MAE regresi linear berganda dengan fitur panjang, lebar, luas, dan perimeter yang paling kecil diantar ketiga metode. Namun, untuk kelas tempe dan serabi regresi linear berganda dengan fitur panjang dan lebar mempunyai nilai MAE yang paling kecil dibandingkan kedua model regresi linear lainnya. Hal ini disebabkan karena tempe dan serabi mempunyai bentuk yang simetris, yaitu bentuk persegi pada tempe dan bentuk bulat untuk serabi. Meskipun bolu dan tahu mempunyai bentuk yang mendekati simetris, yaitu bentuk persegi, namun nilai MAE lebih baik pada empat fitur yang diusulkan pada tahu dan lebih baik di fitur area pada bolu. Contoh evaluasi kalori terhadap gambar input ditujukan pada Tabel IV.24.

Tabel IV.24 Evaluasi kalori terhadap gambar input pada kasus yang benar

Hasil Segmentasi	Informasi	Model Regresi Linear		
		Fitur Panjang, Lebar	Fitur Panjang Lebar, Luas, Perimeter	Fitur Luas
	Kelas : tempe	Berat : 46.93 gram	Berat : 49.58 gram	Berat : 47.12 gram
		Kalori: 113.11 kkal	Kalori: 119.50 kkal	Kalori: 113.57 kkal
	Berat : 48.1	<i>Absolute Error :</i> 2.80	<i>Absolute Error :</i> 3.58	<i>Absolute Error :</i> 2.34
	Kalori: 115.921	<i>Relative Error :</i> 0.024	<i>Relative Error :</i> 0.03	<i>Relative Error :</i> 0.02
	Kelas : tahu	Berat : 30.68 gram	Berat : 30.18 gram	Berat : 29.36 gram
		Kalori : 83.16 kkal	Kalori : 81.80 kkal	Kalori : 79.58 kkal
	Berat : 22.2	<i>Absolute Error :</i> 23.0	<i>Absolute Error :</i> 21.63	<i>Absolute Error :</i> 19.42
	Kalori: 60.162	<i>Relative Error :</i> 0.38	<i>Relative Error :</i> 0.35	<i>Relative Error :</i> 0.32
	Kelas : tahu	Berat : 38.36 gram	Berat : 38.81 gram	Berat : 38.72 gram
		Kalori : 103.95 kkal	Kalori : 105.17 kkal	Kalori: 104.95 kkal
	Berat : 31.0	<i>Absolute Error :</i> 19.94	<i>Absolute Error :</i> 21.16	<i>Absolute Error :</i> 20.94
	Kalori: 84.01	<i>Relative Error :</i> 0.23	<i>Relative Error :</i> 0.25	<i>Relative Error :</i> 0.24
	Kelas : Tahu	Berat : 44.10 gram	Berat : 40.74 gram	Berat : 47.49 gram

	Berat : 41.2 Kalori : 111.652	Kalori : 119.51 kkal <i>Absolute Error :</i> 7.86 <i>Relative Error :</i> 0.07	Kalori: 110.42 kkal <i>Absolute Error :</i> 1.22 <i>Relative Error :</i> 0.01	Kalori: 128.72 kkal <i>Absolute Error :</i> 17.06 <i>Relative Error :</i> 0.15
	Kelas : tempe Berat : 45.6 Kalori : 109.896	Berat : 45.88 gram Kalori: 110.58 kkal <i>Absolute Error :</i> 0.69 <i>Relative Error :</i> 0.0062	Berat : 49.71 gram Kalori : 119.8 kkal <i>Absolute Error :</i> 9.90 <i>Relative Error :</i> 0.090	Berat : 49.19 gram Kalori : 118.56 kkal <i>Absolute Error :</i> 8.66 <i>Relative Error :</i> 0.078
	Kelas : serabi Berat : 35.2 Kalori : 76.032	Berat : 52.77 gram Kalori: 113.98 kkal <i>Absolute Error :</i> 37.9 <i>Relative Error :</i> 0.49	Berat : 56.09 gram Kalori : 121.16 kkal <i>Absolute Error :</i> 45.13 <i>Relative Error :</i> 0.59	Berat : 56.06 gram Kalori : 121.11 kkal <i>Absolute Error :</i> 45.07 <i>Relative Error :</i> 0.59
	Kelas : bolu Berat : 19.5 Kalori: 57.915	Berat : 21.10 gram Kalori : 62.68 kkal <i>Absolute Error :</i> 4.76	Berat: 21.11 gram Kalori : 62.71 kkal <i>Absolute Error :</i> 4.79	Berat: 21.15 gram Kalori : 62.81 kkal <i>Absolute Error :</i> 4.90

		<i>Relative Error :</i> 0.08	<i>Relative Error :</i> 0.082	<i>Relative Error :</i> 0.084
	Kelas : bolu Berat : 19.5 Kalori: 57.915	Berat : 21.14 gram Kalori : 62.79 kkal <i>Absolute Error :</i> 4.88 <i>Relative Error :</i> 0.084	Berat : 21.06 gram Kalori : 62.57 kkal <i>Absolute Error :</i> 4.65 <i>Relative Error :</i> 0.0804	Berat : 20.88 gram Kalori : 62.03 kkal <i>Absolute Error :</i> 4.12 <i>Relative Error :</i> 0.07
	Kelas : bolu Berat : 15.0 Kalori: 44.55	Berat : 26.10 gram Kalori : 77.53 kkal <i>Absolute Error :</i> 32.98 <i>Relative Error :</i> 0.74	Berat : 24.58 gram Kalori : 73.02 kkal <i>Absolute Error :</i> 28.47 <i>Relative Error :</i> 0.63	Berat : 23.66 gram Kalori : 70.29 kkal <i>Absolute Error :</i> 25.74 <i>Relative Error :</i> 0.57
	Kelas : serabi Berat : 13.7 Kalori : 29.592	Berat : 15.11 gram Kalori : 32.65 kkal <i>Absolute Error :</i> 3.06 <i>Relative Error :</i> 0.103	Berat : 13.55 gram Kalori : 29.27 kkal <i>Absolute Error :</i> 0.314 <i>Relative Error :</i> 0.0106	Berat : 14.06 gram Kalori : 30.38 kkal <i>Absolute Error :</i> 0.793 <i>Relative Error :</i> 0.026
	Kelas : serabi Berat : 35.2	Berat : 35.33 gram Kalori : 76.33 kkal	Berat : 33.55 gram Kalori : 72.48 kkal	Berat : 34.13 gram Kalori : 73.73 kkal

	Kalori : 76.032	<i>Absolute Error :</i> 0.29	<i>Absolute Error :</i> 3.55	<i>Absolute Error :</i> 2.29
		<i>Relative Error :</i> 0.0039	<i>Relative Error :</i> 0.046	<i>Relative Error :</i> 0.03

Berdasarkan pengujian tersebut, dapat terlihat bahwa model sudah dapat mendeteksi dengan baik makanan-makanan tersebut, meskipun pada saat pelatihan model melibatkan objek yang saling tumpang tindih. *Error* yang dihasilkan pada beberapa *item* makanan sudah cukup kecil, meskipun masih terdapat sejumlah objek yang mengeluarkan prediksi kalori yang kurang tepat karena karakteristik dari makanan yang beragam.

IV.5 Hasil Rekap Eksperimen Modul Gabungan

Dalam pembangunan model deteksi, dievaluasi empat model *pretrained backbone* Mask R-CNN dan model tersebut dilatih dengan menggunakan *dataset* dengan anotasi secara *amodal*, yang melibatkan bagian tumpang tindih. Model yang mempunyai kinerja yang tidak berbeda jauh dalam data validasi adalah model ResNet-101-FPN dan model ResNeXt-101-FPN.

Pada pembangunan model gabungan, telah dilakukan pengujian dengan dua skenario, yaitu pengujian dengan data uji yang mengandung objek tumpang tindih dan pengujian dengan data uji yang mengandung objek tidak tumpang tindih. Dilakukan pengujian terhadap dua model *backbone* deteksi Mask R-CNN, yaitu *backbone* ResNet-101-FPN dan ResNeXt-101-FPN. Perbandingan nilai *F1 Score* ditunjukkan pada Tabel IV.25.

Tabel IV.25 Rata-rata *F1 Score* pada dua skenario pengujian

Model <i>Backbone</i>	Rata-rata <i>F1 Score</i>	
	Tumpang Tindih	Tidak Tumpang Tindih
ResNet-101-FPN	0.81	0.966
ResNeXt-101-FPN	0.821	0.994

Model yang terpilih adalah model *Mask R-CNN* dengan *backbone* ResNeXt-101-FPN. Model *Mask R-CNN* tersebut mampu mendeteksi dan mensegmentasi gambar yang mengandung objek yang tumpang tindih dalam *IoU threshold* lebih dari 0.85, dengan rata-rata nilai *precision* sebesar 0.818, rata-rata nilai *recall* sebesar 0.824, dan rata-rata nilai *F1 Score* sebesar 0.821. Sedangkan untuk data yang tidak tumpang tindih mampu mendeteksi dan mensegmentasi objek-objek tersebut dengan *IoU threshold* lebih dari 0.9, dengan rata-rata nilai *precision* sebesar 0.992, rata-rata nilai *recall* sebesar 0.996, dan rata-rata nilai *F1 Score* sebesar 0.994. Meskipun seperti itu, model tersebut masih mempunyai kelemahan, baik dalam kualitas segmentasi maupun terdapat deteksi *false positive*.

Meskipun model *backbone* ResNeXt-101-FPN merupakan model yang mempunyai nilai *F1 Score* yang paling baik, namun terdapat beberapa kualitas segmentasi masih tergolong belum memenuhi *ground truth* dan terdapat deteksi *false positive* diantara objek yang beroklusi. Model tersebut sudah dapat mensegmentasi objek makanan yang tidak ada di data latih dengan baik, meskipun masih terdapat hasil deteksi yang kurang baik.

Dalam melakukan prediksi kalori, dilakukan perhitungan terhadap nilai MAE pada seluruh kelas. Rata-rata nilai MAE dari seluruh kelas ditunjukkan pada Tabel IV.26.

Tabel IV.26 Rata-rata nilai MAE seluruh kelas pada kasus tumpang tindih dan kasus tidak tumpang tindih

Skenario	MAE		
	Fitur Luas	Fitur Panjang dan Lebar	Fitur Panjang, Lebar, Perimeter, dan Luas
Tumpang Tindih	8.354	8.757	8.829
Tidak Tumpang Tindih	11.659	11.608	11.256

Berdasarkan nilai tersebut, model gabungan antara model *Mask R-CNN* yang diannotasi secara *amodal* dengan model regresi linear dapat menghasilkan estimasi kalori dengan *error* yang tidak terlalu besar, baik dalam keadaan tumpang tindih maupun yang tidak tumpang tindih. Namun, model regresi yang paling baik

berdasarkan nilai rata-rata MAE dari seluruh kelas dalam memprediksi makanan tumpang tindih adalah model regresi dengan fitur luas. Salah satu yang mempengaruhi hal tersebut adalah kurang sempurnanya hasil segmentasi objek sehingga ketika menggunakan model regresi linear berganda dengan banyak fitur hasilnya menjadi tidak terlalu bagus.

Pada pengujian dengan keadaan tidak tumpang tindih, model regresi linear dengan empat fitur mempunyai nilai MAE yang paling kecil diantara yang lainnya. Meskipun seperti itu, model regresi linear dengan fitur luas dan fitur panjang dan lebar dapat mengurangi *error* lebih baik dibandingkan dengan model regresi linear yang diusulkan dalam beberapa kelas dan beberapa *item* lainnya.

Bab V Kesimpulan dan Saran

V.1 Kesimpulan

Dalam penelitian ini, telah dibangun model estimasi kalori berbasis citra dengan menggunakan *Mask R-CNN* yang diannotasi secara amodal, yang melibatkan bagian yang tumpang tindih dan regresi linear berganda dengan fitur panjang, lebar, luas, dan perimeter objek. Berdasarkan penelitian yang telah dilakukan, dapat diambil beberapa kesimpulan berikut

1. Pada penelitian ini, telah dibangun dataset jajanan pasar di Indonesia yang terdiri dari 6 kelas, yaitu bakwan, bolu, cireng, serabi, tahu, dan tempe. Untuk model deteksi, pembagian data dibagi menjadi data latih dan data validasi dengan berbagai posisi, baik posisi tumpang tindih maupun tidak tumpang tindih. Untuk model estimasi, dilakukan pengambilan gambar sebanyak 10 kali untuk setiap makanan dan dibagi menjadi data latih dan data validasi untuk setiap kelas. Untuk model gabungan, digunakan data uji dengan campuran objek yang sudah ada di data latih dan objek yang belum ada di data latih.
2. Hipotesis-1 menyatakan bahwa model regresi linear berganda dapat meningkatkan keakuratan pengukuran. Hasil penelitian menunjukkan bahwa model regresi linear berganda dengan empat fitur yang diusulkan mempunyai rata-rata nilai *R Squared* sebesar 0.804, yang paling tinggi diantara dua model regresi linear lainnya dan rata-rata MAE seluruh kelas dalam prediksi data uji model estimasi sebesar 5.254, yang paling kecil dibandingkan model regresi lainnya. Model regresi linear berganda dengan empat fitur yang diusulkan mempunyai rata-rata nilai MAE sebesar 11.256 pada skenario objek tidak tumpang tindih, yang paling kecil diantara kedua model regresi linear lainnya. Akan tetapi terdapat kelas yang lebih baik di regresi linear berganda dengan hanya menggunakan fitur panjang dan lebar saja seperti tempe dan serabi. Pada skenario objek tumpang tindih, regresi linear dengan fitur luas mempunyai rata-rata MAE lebih kecil dibandingkan dengan regresi linear berganda, yaitu 8.354, yang disebabkan karena kualitas segmentasi yang kurang baik, sehingga dengan menggunakan gabungan antara panjang, lebar, luas, dan keliling menghasilkan hasil yang

tidak terlalu bagus. Oleh karena itu model regresi linear berganda lebih tepat digunakan untuk estimasi kalori makanan dalam keadaan objek yang tidak tumpang tindih.

3. Hipotesis-2 menyatakan bahwa kemampuan *Mask R-CNN* dalam melakukan task *amodal instance segmentation* dengan melatih *dataset* dengan objek yang diannotasi secara amodal dapat mengatasi permasalahan pengukuran kalori makanan dalam keadaan tumpang tindih. Model *Mask R-CNN* yang terbaik terdiri dari dua model *backbone*, yaitu ResNet-101-FPN dan ResNeXt-101-FPN. Kedua model tersebut mempunyai mAP Validasi yang tidak berbeda jauh, 91.74% (ResNet-101-FPN) dan 91.47% (ResNeXt-101-FPN). Dalam pengujian model gabungan, *Mask R-CNN* dengan *backbone* ResNeXt-101-FPN mendapatkan rata-rata *F1 Score* dari keseluruhan kelas sebesar 0.821 pada skenario gambar yang mengandung objek tumpang tindih dalam *IoU threshold* 0.85 dan *F1 Score* sebesar 0.994 pada skenario gambar yang mengandung objek tidak tumpang tindih dalam *IoU threshold* 0.9. Meskipun seperti itu, model ini masih punya kelemahan, yaitu masih terdapat deteksi *false positive* diantara objek yang tumpang tindih serta hasil segmentasi yang masih belum mirip dengan bagian yang terhalang objek lain, terutama pada objek yang asimetris. Model dapat mensegmentasi objek yang tidak ada di data latih dengan baik walaupun masih terdapat kesalahan segmentasi yang sama seperti masih ada deteksi *false positive* dan hasil segmentasi yang tidak sesuai dengan *ground truth*.

V.2 Saran

Dalam penelitian ini, saran yang dapat diberikan dalam penelitian ini adalah sebagai berikut

1. Dalam model *Mask R-CNN* masih terdapat masalah pada deteksi *false positive* dan kualitas segmentasi yang masih tidak terlalu bagus dalam segmentasi objek tumpang tindih. Perlu dilakukan eksperimen lebih lanjut dengan metode *amodal instance segmentation* yang lain agar dapat menghasilkan bentuk objek makanan yang menyerupai *ground truth* objek.

Selain itu, dapat diobservasi juga model *instance segmentation* yang lain dalam melakukan task *amodal instance segmentation*.

2. Dalam sebuah bentuk, terdapat beberapa fitur geometri lain selain fitur luas, perimeter, panjang dan lebar. Oleh karena itu perlu adanya observasi pada fitur geometri yang lainnya untuk menghasilkan estimasi yang baik. Penggunaan model regresi yang lainnya juga perlu dipertimbangkan.
3. Dalam penelitian ini, model regresi linear yang dibangun masih bergantung dengan kelas makanan sehingga perlu membuat lebih dari satu model regresi linear. Oleh karena itu perlu dibuat sebuah model yang dapat memprediksi kalori makanan yang tidak bergantung pada kelas makanan.
4. Dalam penelitian ini, jumlah gambar dan item makanan yang digunakan masih tergolong kurang. Oleh karena itu perlu adanya eksperimen kembali dengan jumlah dataset yang lebih besar lagi dan jumlah item makanan yang lebih banyak serta kelas yang beragam. Selain itu, dapat dipertimbangkan juga metode untuk estimasi kalori dengan jarak kamera yang bersifat dinamis.

DAFTAR PUSTAKA

- Abdelhady, A. S., Hassanien, A. E., Awad, Y. M., El-Gayar, M., dan Fahmy, A. (2019): Automatic sheep weight estimation based on k-means clustering and multiple linear regression, *International Conference on Advances in Intelligent Systems and Computing*, 546-555.
- Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., Esesn, B. C. V., Awwal, A. A. S., dan Asari, V. K. (2018): The history began from alexnet: a comprehensive survey on deep learning approaches, arXiv preprint arXiv:1803.01164.
- Birchfield, S. (2016): *Image Processing and Analysis*, Cengage Learning, Amerika Serikat, 2-11.
- Chen, C., Wang, B., Xiaoxuan Lu, C., Trigoni, N., dan Markham, A. (2020): A Survey on Deep Learning for Localization and Mapping: Towards the Age of Spatial Machine Intelligence. ArXiv preprint arXiv:2006.12567.
- Chiang, M. L., Wu, C. A., Feng, J. K., Fang, C. Y., dan Chen, S. W. (2019): Food calorie and nutrition analysis system based on mask r-cnn, *2019 IEEE 5th International Conference on Computer and Communications*, 1721–1728.
- Ege, T., dan Yanai, K. (2017a): Estimating food calories for multiple-dish food photos, *2017 4th Asian Conference on Pattern Recognition (ACPR)*, 646–651.
- Ege, T., dan Yanai, K. (2017b): Simultaneous estimation of food categories and calories with multi-task CNN, *2017 15th IAPR International Conference on Machine Vision Applications (MVA)*, 198–201.
- Giovany, S., Putra, A., Hariawan, A. S., Wulandhari, L. A., dan Irwansyah, E. (2020): Indonesian food image recognition using convolutional neural network, *Computer Science On-line Conference*, **985**, 208–217.
- Gonzalez, S., Arellano, C., dan Tapia, J. E. (2019): Deepblueberry : quantification of blueberries in the wild using instance segmentation, *IEEE Access*, **7**, 105776-105788.
- He, K., Gkioxari, G., Dollár, P., dan Girshick, R. (2017): Mask r-cnn, *Proceedings of the IEEE International Conference on Computer Vision*, 2961–2969.
- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., dan Belongie, S. (2017): Feature pyramid networks for object detection, *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.
- Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., dan Terzopoulos, D. (2021): Image segmentation using deep learning: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1.
- Montgomery, D. C., Peck, E. A., dan Vining, G. G. (2021): *Introduction to Linear Regression Analysis*, Wiley, Britania Raya, 1.
- Qi, L., Jiang, L., Liu, S., Shen, X., dan Jia, J. (2019): Amodal instance segmentation with kins dataset, *Proceeding of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 3014–3023.
- Wibisono, A., Wisesa, H. A., Rahmadhani, Z. P., Fahira, P. K., Mursanto, P., dan Jatmiko, W. (2020): Traditional food knowledge of Indonesia: a new high-quality food dataset and automatic recognition system, *Journal of Big Data*, **7(1)**.
- World Health Organization. (2020): Obesity and overweight. diperoleh melalui situs internet: <https://www.who.int/news-room/fact-sheets/detail/obesity-and->




overweight . Diunduh pada tanggal 18 Maret 2021

- Xie, S., Girshick, R., Dollár, P., Tu, Z., dan He, K. (2017): Aggregated residual transformations for deep neural networks, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1492–1500.
- Yogaswara, R. D., Yuniarno, E. M., dan Wibawa, A. D. (2019): Instance-aware semantic segmentation for food calorie estimation using mask r-cnn, *2019 International Seminar on Intelligent Technology and Its Application, (ISITIA)*, 416–421.

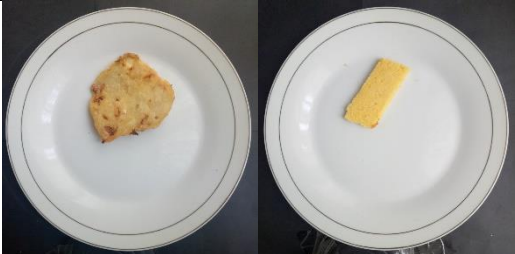
LAMPIRAN


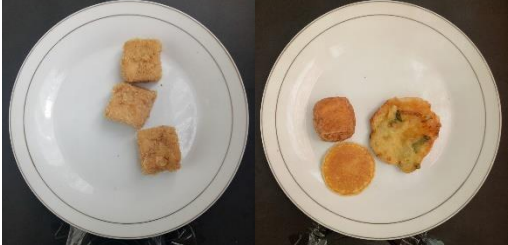
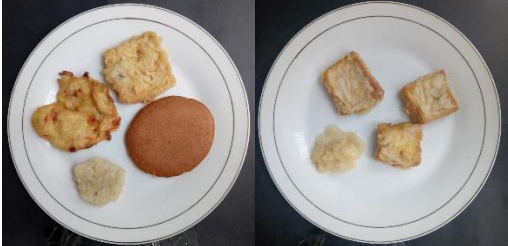
Lampiran A Contoh Dataset

A1. Gambar yang Mengandung Objek Tumpang Tindih

Jumlah Makanan dalam Piring	Gambar	
2		
3		
4		

A2. Gambar yang Mengandung Objek Tidak Tumpang Tindih

Jumlah Makanan dalam Piring	Gambar	
1		

Jumlah Makanan dalam Piring	Gambar
2	
3	
4	

Lampiran B Distribusi Objek

B1. Distribusi Objek di Data Latih

Kelas	Jumlah Objek
bakwan	467
bolu	623
cireng	706
serabi	514
tahu	544
tempe	600

B2. Distribusi Objek di Data Validasi

Kelas	Jumlah Objek
bakwan	59
bolu	71
cireng	98
serabi	65
tahu	65
tempe	65

B3. Distribusi Objek di Data Uji

Kelas	Jumlah Objek
bakwan	118
bolu	118
cireng	114
serabi	100
tahu	102
tempe	95