

Evaluasi Kinerja Aplikasi Mobile Penghitung Kalori Makanan Berbasis Algoritma CNN-YOLO

Performance Evaluation of Food Calorie Counter Mobile Application Based on CNN-YOLO Algorithms

Nurul Khaerani Hamzidah*
Teknik Multimedia dan Jaringan,
Jurusan Teknik Informatika dan Komputer,
Politeknik Negeri Ujung Pandang (PNUP)
Makassar, Indonesia
nkhamzidah@poliupg.ac.id*

Mardawia Mabe Parenreng
Teknik Multimedia dan Jaringan,
Jurusan Teknik Informatika dan Komputer,
Politeknik Negeri Ujung Pandang (PNUP)
Makassar, Indonesia
mmparenreng@poliupg.ac.id

Ayu Ulandari
Teknik Multimedia dan Jaringan,
Jurusan Teknik Informatika dan Komputer,
Politeknik Negeri Ujung Pandang (PNUP)
Makassar, Indonesia
ayuulandari6@gmail.com

Nur Ichzan As
Teknik Multimedia dan Jaringan,
Jurusan Teknik Informatika dan Komputer,
Politeknik Negeri Ujung Pandang (PNUP)
Makassar, Indonesia
nurichzanas@poliupg.ac.id

Diterima : Februari 2025
Disetujui : Juni 2025
Dipublikasi : Juli 2025

Abstrak—Artikel ini membahas hasil kinerja aplikasi mobile penghitung kalori makanan. Aplikasi ini dapat membantu pengguna dalam mengatur dan mengetahui asupan kalori harian serta program diet sehat. Aplikasi ini berbasis teknik pengolahan citra menggunakan kombinasi CNN-YOLOv5. Adapun peran CNN adalah untuk mengklasifikasi data serta mengekstraksi fitur citra makanan yang telah terlabel dengan menggunakan metode *supervised learning* berdasarkan citra yang telah melalui tahapan proses training sedangkan YOLO berperan dalam mendeteksi data citra makanan dengan cepat dan tepat. Tahapan perancangannya terdiri dari perancangan desain UI, pembuatan UI, implementasi program, dan pengujian serta evaluasinya. Dalam menganalisis model digunakan 1.736 data latih, 149 data uji dan 206 data validasi dengan komputasi 150 epoch. Hasil analisis model diperoleh presisi 1.00, confidence 0.962, recall 0.99 serta F1 score 0.97. Hasil ini menunjukkan bahwa sistem sudah memenuhi syarat untuk digunakan dalam proses deteksi lebih lanjut. Hal ini dibuktikan dengan kemampuan aplikasi dalam mendeteksi citra makanan dengan akurasi 100% untuk semua kelas makanan secara real-time ataupun melalui upload citra. Hasil pengujian menunjukkan bahwa nilai confidence dipengaruhi oleh faktor jarak detektor ke objek, intensitas pencahayaan, resolusi kamera, kemiripan warna, variasi makanan serta adanya motif background wadah yang digunakan. Aplikasi dilengkapi dengan fitur dan tampilan UI yang menarik seperti kalkulator BMI yang informatif khususnya bagi pengguna yang sedang dalam program diet sehat. Penerapan kombinasi algoritma CNN-YOLOv5 terbukti mampu meningkatkan kinerja aplikasi secara konsisten dan akurat dalam mendeteksi jenis makanan beserta kandungan kalori dalam 100 gramnya sehingga layak dijadikan sebagai rujukan dalam membantu

monitoring asupan kalori harian dan membantu program diet sehat.

Kata Kunci—CNN-YOLO; kalori makanan; diet sehat; kalkulator BMI

Abstract— This article discusses the performance results of a mobile application for calculating food calories. This application can help users in managing and knowing their daily calorie intake and a healthy diet program. This application is based on image processing techniques using a combination of CNN-YOLOv5. The role of CNN is to classify data and extract labeled food image features using the supervised learning method based on images that have gone through the training process stages while YOLO plays a role in detecting food image data quickly and accurately. The design stages consist of UI design, UI creation, program implementation, and testing and evaluation. In analyzing the model, 1,736 training data, 149 test data and 206 validation data were used with 150 epochs of computation. The results of the model analysis obtained a precision of 1.00, confidence 0.962, recall 0.99 and F1 score 0.97. These results indicate that the system has met the requirements for use in further detection processes. This is evidenced by the application's ability to detect food images with 100% accuracy for all food classes in real-time or through image uploads. The test results show that the confidence value is influenced by the distance of the detector to the object, lighting intensity, camera resolution, color similarity, food variety and the background motif of the container used. The application is equipped with attractive features and UI displays such as an informative BMI calculator especially for users who are on a healthy diet program. The application of the CNN-YOLOv5 algorithm combination has been proven to be able to consistently and accurately improve application performance in detecting types of food and their calorie content in 100 grams so

that it is worthy of being used as a reference in helping to monitor daily calorie intake and help a healthy diet program.

Keywords—*CNN-YOLO; food kalori; healthy diet; BMI calculator.*

I. PENDAHULUAN

Obesitas merupakan permasalahan global yang berkontribusi pada 8% kematian secara global setiap tahunnya. Individu yang mengalami obesitas berisiko tinggi terhadap berbagai penyakit dan kondisi kesehatan serius, termasuk tekanan darah tinggi, dislipidemia, diabetes, serangan jantung, penyakit kandung empedu, osteoarthritis, masalah tidur, masalah pernapasan, gangguan kesehatan mental, dan penurunan kualitas hidup secara keseluruhan [1], [2]. Selain itu, obesitas juga menimbulkan kendala dalam menjaga postur tubuh yang benar saat bekerja atau tidur, yang dapat mengakibatkan cedera akibat stres. Meskipun faktor genetik memainkan peran dalam regulasi berat badan seseorang, ada faktor-faktor lain yang menjadi penyebab obesitas, seperti penggunaan obat-obatan tertentu, tingkat stres, kurangnya aktivitas fisik, dan kebiasaan tidur yang tidak teratur. Asupan makanan yang berlebihan merupakan salah satu penyebab utama kelebihan berat badan dan akumulasi lemak, dimana konsumsi kalori melebihi kebutuhan tubuh [3], [4], [5].

Untuk menangani permasalahan obesitas, berbagai program dan edukasi sudah pernah dilakukan oleh pemerintah melalui tenaga dan praktisi bidang kesehatan dan gizi diantaranya dengan sosialisasi dan penyuluhan, layanan konsultasi langsung ataupun via online hingga memberikan informasi iklan layanan masyarakat. Tetapi cara dan pendekatan ini masih dianggap kurang efektif karena masih kurangnya minat masyarakat dalam mengakses layanan dan memperhatikan himbauan akan pentingnya pola makan yang sehat. Akses layanan yang membutuhkan waktu dan tenaga juga menjadi faktor utama pendekatan program layanan ini kurang efektif dalam mengubah pola pikir masyarakat dalam hal menjaga pola makan [6], [7]. Jumlah obesitas semakin tidak terkontrolnya pola makan masih merajalela di kalangan masyarakat dunia khususnya di Indonesia semakin bertambah. Oleh karena itu, perlu penyediaan solusi praktis dan efektif bagi individu yang ingin mengubah gaya hidup sehat dalam mengatasi kurangnya perhatian masyarakat dalam mengatur pola makan melalui perhitungan asupan kalori harian [1], [2], [4], [8].

Dengan perkembangan teknologi digital, beberapa tahun terakhir ini, pendekatan berbasis aplikasi mobile telah banyak dilakukan di berbagai bidang termasuk bidang kesehatan. Aplikasi ini dapat digunakan setiap individu dalam monitoring jumlah asupan kalori harian makanan kapanpun. Selain fleksibel, murah dan mudah digunakan, aplikasi mobile dapat diakses oleh semua orang melalui ponsel androidnya masing-masing tanpa perlu lagi konsultasi ke ahli gizi ataupun konsultan kesehatan terkait asupan makanan sehat beserta nilai kolorinya yang tentunya membutuhkan waktu.

Terkait aplikasi mobile penghitung kalori harian, Sebelumnya, beberapa penelitian terkait perancangan aplikasi mobile ini telah dilakukan dengan hasil yang baik dari segi keakuratan informasi kalori harian dengan tampilan yang menarik. Perancangan aplikasi ini dapat menggunakan berbagai jenis metode salah satu diantaranya adalah CNN

dan YOLO. *Convolutional Neural Network* (CNN) merupakan salah satu metode dari *deep learning* yang merupakan pengembangan dari *Multilayer Perceptron* (MLP) yang dirancang untuk mengolah data dua dimensi, misalnya citra atau suara. CNN digunakan untuk mengklasifikasi data yang terlabel dengan menggunakan metode *supervised learning*. Cara kerja *supervised learning* yaitu terdapat data yang dilatih dan terdapat variabel yang ditargetkan sehingga tujuan dari metode ini adalah mengelompokkan suatu data kepada yang sudah ada [9], [10], [11].

Penggunaan algoritma CNN dan YOLO yang diintegrasikan ke dalam aplikasi telah banyak diterapkan sebelumnya dalam aplikasi penghitung nilai kalori harian. Penggunaan CNN dan YOLO dengan berbagai versi ini telah terbukti mampu meningkatkan kualitas deteksi sistem dari aplikasi. Hal ini ditunjukkan dalam hasil penelitian sebelumnya yang menghasilkan nilai akurasi sistem dan berbagai parameter misalnya nilai confidence, recall, F1 score yang tinggi [12], [13], [14], [15].

Berdasarkan rujukan dan ulasan penelitian sebelumnya maka pada penelitian ini akan dirancang pengembangan aplikasi mobile berbasis implementasi pengolahan citra menggunakan kombinasi CNN dan YOLO. Dalam konteks deteksi objek, CNN digunakan sebagai bagian dari arsitektur YOLO untuk memproses gambar dan mengekstrak fitur yang relevan. Selanjutnya, YOLO menggunakan fitur yang diekstrak oleh CNN untuk memprediksi kotak pembatas dan probabilitas kelas objek [1], [2], [5], [16], [17]. Diharapkan kombinasi CNN-YOLO memberikan kecepatan deteksi objek yang tinggi (real-time) dan akurasi yang baik. YOLO juga merupakan detektor objek tahap tunggal (single-stage detector), yang berarti ia tidak perlu melakukan beberapa tahap pemrosesan seperti detektor objek tradisional [9], [12], [18].

Beberapa hal yang menjadi pembeda pada penelitian sebelumnya diantaranya YOLO yang digunakan adalah YOLO versi 5. YOLOv5 dipilih karena keunggulannya dengan versi lainnya yaitu memiliki ukuran dan model yang lebih ringan serta sangat cocok diintegrasikan dengan perangkat mobile seperti android. Hal lainnya, yaitu hasil riset ini membahas lebih dalam evaluasi kinerja dari aplikasi dengan melakukan beberapa skema pengujian terkait parameter diantaranya pengaruh jarak detektor, pengaruh intensitas cahaya, pengaruh kualitas resolusi kamera. Tampilan fitur pada menu aplikasi ini lebih lengkap dengan tampilan desain UI yang menarik. Hasil evaluasi ini diharapkan juga memberikan informasi penting dalam meningkatkan kinerja aplikasi dan bisa menjadi rujukan bagi pengembangan aplikasi untuk penelitian selanjutnya.

Oleh karena itu, dalam artikel ini akan dibahas hasil pengujian dan evaluasi kinerja dari pengembangan aplikasi penghitung jumlah kalori makanan berbasis mobile. Dengan menggunakan metode *You Only Look Once* (YOLO) dalam pengenalan gambar, keunggulan aplikasi ini juga terletak pada kemampuannya untuk mendeteksi dan menghitung jumlah kalori makanan secara langsung (*real-time*). Diharapkan, hal ini dapat membantu orang yang ingin melakukan program diet, terutama remaja yang cenderung konsumtif, untuk lebih menyadari asupan kalori mereka dan secara efektif mengelola program diet.

Dengan demikian, aplikasi ini diharapkan dapat menjadi solusi yang terjangkau dan mandiri bagi individu

yang ingin mengatasi masalah obesitas dan menjalani program diet dengan pemahaman yang lebih baik.

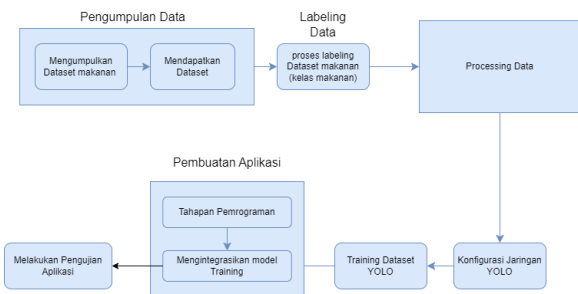
II. METODE

A. Peralatan dan Bahan

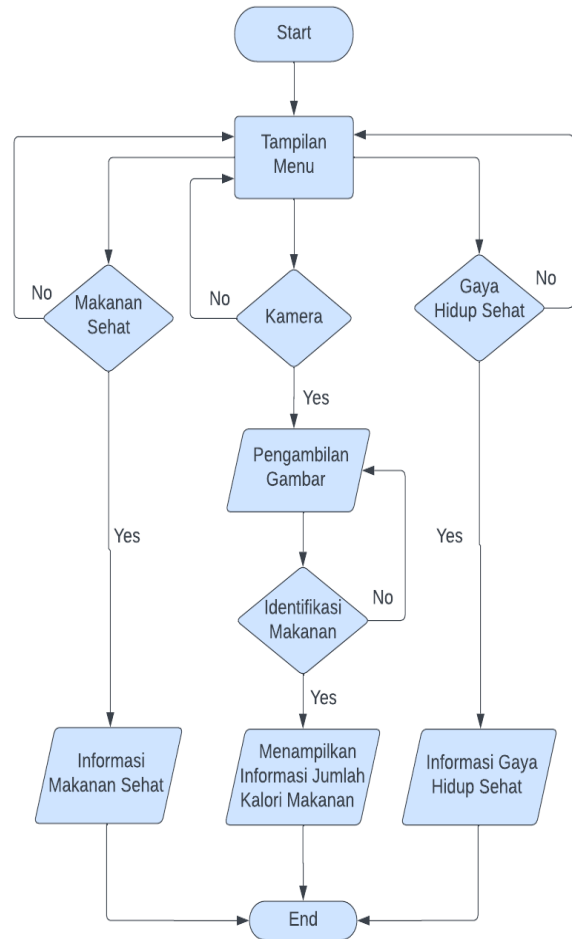
Peralatan dan bahan yang digunakan terdiri dari perangkat keras dan perangkat lunak yaitu laptop, handphone android, software android studio, software python, roboflow sedangkan bahan yang digunakan adalah dataset 12 kelas makanan yang terdiri dari 1468 data gambar berbagai makanan harian. Distribusi kelas masing-masing sampel beragam sehingga digunakan teknik augmentasi *transformasi geometri affine*. Teknik ini menggunakan beberapa parameter diantaranya kemiringan $\pm 45^\circ$, translasi 0.5, shearing 20, dan skala crop size 256.

B. Perancangan User Interface Aplikasi

Secara umum rancangan pembuatan aplikasi ini ditampilkan pada Gambar 1. Tahapannya dimulai dengan pengumpulan data berupa dataset makanan harian beserta jumlah kalorinya. Selanjutnya, beralih ke tahapan perancangan dan pembuatan sistem, kemudian dilanjutkan dengan pengujian sistem, dan tahap analisis hasil pengujian.



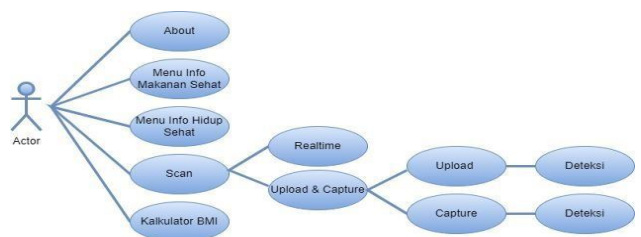
Gambar 1. Diagram Alur Sistem Aplikasi



Gambar 2. Diagram Alur Proses Deteksi Kalori Makanan

Berdasarkan bagan alir pada Gambar 2, alur tahapan proses deteksi diawali dengan tampilan fitur-fitur menu aplikasi. Selanjutnya, mengaktifkan fitur kamera ke objek yang akan dideteksi

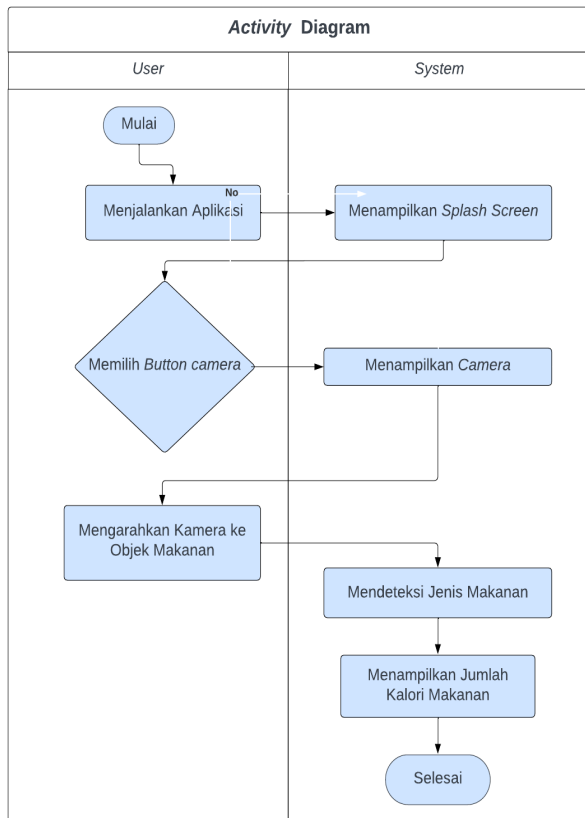
Berikut diagram use-case aplikasi ditampilkan pada Gambar 3.



Gambar 3. Diagram Use Case Aplikasi Deteksi Kalori Makanan

Dalam ilustrasi Gambar 3, terlihat fitur-fitur utama yang dalam. Penggunaan aplikasi dimulai dengan tampilan *splash screen* kemudian tampilan fitur menu lainnya yaitu, “Menu Info Makanan Sehat”, “Camera”, “Menu Info Hidup Sehat”, dan keluar. Pilihan fitur menu “Info Makanan Sehat”, akan menampilkan informasi mengenai berbagai makanan sehat, sedangkan ketika memilih menu “Scan atau Camera”, akan ditampilkan jendela kamera kemudian mendeteksi makanan dan apabila makanan berhasil terdeteksi maka sistem akan menampilkan nilai kalori makanan. Selanjutnya, pada saat menekan menu “Info Hidup Sehat” aplikasi akan

menampilkan informasi mengenai gaya hidup sehat. Adapun menu yang terakhir yaitu “Keluar” dimana fungsi dari menu tersebut adalah untuk mengeluarkan pengguna dari dalam aplikasi.



Gambar 4. Diagram Aktivitas Aplikasi Mobile Kalori

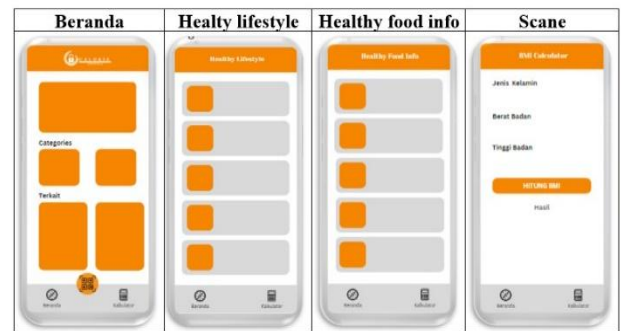
Dalam Gambar 4, memperlihatkan rangkaian kerja sistem secara keseluruhan. Saat layar pada aplikasi menampilkan *Splash Screen*. Selanjutnya pengguna memilih opsi kamera, untuk mengaktifkan kamera. Selanjutnya, pengguna mengarahkan kamera ke objek yang akan dideteksi hasil deteksi akan menampilkan nilai kalori makanan.

C. Pembuatan UI Aplikasi

Sebelum aplikasi dibuat, langkah perancangan user interface diperlukan sebagai dasar untuk pengembangan aplikasi tersebut. Berikut adalah tahapan dalam perancangan tampilan desain UI dari aplikasi:

1. Instalasi dan Desain Interface Aplikasi

Aplikasi yang digunakan adalah *Python* dan *Android Studio*. Berikut rancangan desain aplikasi yang ditampilkan pada Gambar 5:



Gambar 5. Desain Antarmuka Aplikasi

2. Implementasi

Tahapan implementasi dilakukan berdasarkan hasil rancangan dari tahapan sebelumnya. Aplikasi ini akan dijalankan menggunakan *smartphone* dengan platform *Android*. Prinsip kerja aplikasi berbasis pengolahan citra menggunakan algoritma *CNN-YOLO*. Objek yang akan dideteksi adalah beberapa jenis makanan konsumsi harian, seperti nasi, ayam goreng, telur, dan sayuran yang terlebih dahulu telah diketahui kandungan kalorinya dalam 100 gram. Berikut data kandungan kalori jenis makan harian dalam 100 gram ditampilkan pada Tabel 1.

TABEL 1. NILAI KALORI PER 100 GRAM 12 KELAS MAKANAN

No.	Nama Kelas Makanan	Kalori per 100 gram
1	Nasi	129 kal
2	Telur Dadar	93 kal
3	Mata Sapi	110 kal
4	Telur Rebus	78 kal
5	Ayam Goreng	260 kal
6	Tahu	80 kal
7	Tempe	225 kal
8	Capcay	67 kal
9	Sayur Sop	22 kal
10	Sayur Kangkung	98 kal
11	Sayur Bayam	36 kal
12	Tumis Buncis	65 kal

D. Skema dan Tahapan Pengujian

Tujuan pengujian aplikasi ini adalah untuk mengetahui kemampuan kinerja dari aplikasi serta sebagai bahan evaluasi untuk perbaikan dan pengembangan selanjutnya. Pengujian ini dilakukan melalui dua skenario yaitu secara real-time dan melalui proses unggah file gambar objek. Selain itu, kemampuan deteksi dari aplikasi ini juga akan diuji dengan beberapa parameter yaitu perubahan jarak deteksi, pencahayaan dan kualitas resolusi perangkat atau camera yang digunakan serta melakukan pengujian kinerja fitur-fitur dalam aplikasi melalui ujia blackbox. Semua skema pengujian ini akan menghasilkan data hasil deteksi dan nilai confidence (derajat kepercayaan). Nilai confidence yang diperoleh menunjukkan kemampuan akurasi sistem dalam mendeteksi objek atau gambar makanan. Berikut informasi mengenai pengujian yang akan dilakukan.

1. Uji Jarak Detektor ke Objek

Pengujian ini dilakukan secara real-time dan bertujuan untuk mengetahui pengaruh jarak detektor (saat pengambilan gambar) terhadap hasil deteksi dan nilai confidence dari aplikasi. Adapun variasi jarak yang digunakan adalah 10 cm, 15 cm dan 20 cm. Dari pengujian ini juga akan diperoleh informasi mengenai

nilai toleransi jarak ideal dari sistem aplikasi dalam mengcapture objek.

2. Uji Faktor Intensitas Pencahayaan

Pengujian ini juga bertujuan untuk mengetahui pengaruh faktor pencahayaan saat sistem melakukan proses deteksi objek. Pengujian ini juga akan dilakukan secara real-time dengan variasi nilai intensitas cahaya 6 lux, 18 lux, 130 lux. Ketiga variasi nilai ini mewakili level pencahayaan rendah, sedang dan tinggi. Pengujian ini juga akan memberikan informasi sejauh mana kemampuan sistem aplikasi merespon dan mendeteksi objek dengan perubahan intensitas cahaya.

3. Uji Faktor Kualitas Resolusi Camera Pendeteksi

Pengujian ini juga akan dilakukan secara real-time dengan menggunakan 3 (variasi) nilai resolusi kamera dengan perangkat android yang berbeda yaitu 15 MP, 23 MP dan 67 MP. Tujuan pengujian ini adalah untuk mengetahui pengaruh dan kemampuan sistem dalam merespon dan mendeteksi secara akurat dengan perubahan atau perbedaan kualitas resolusi kamera perangkat android yang digunakan.

4. Uji Fitur Kalkulator BMI

Pengujian fitur ini dilakukan untuk mengecek apakah fitur kalkulator BMI berfungsi dengan baik dan memberikan hasil tampilan dan rekomendasi yang sesuai dengan membandingkannya dengan beberapa aplikasi yang sudah ada yang bisa diakses secara online di internet.

BMI (*Body Mass Index*) atau indeks massa tubuh adalah suatu alat ukur yang digunakan untuk menentukan apakah berat badan seseorang sesuai dengan tinggi badannya. Untuk menghitung BMI atau Indeks Massa Tubuh (IMT), berat badan (dalam kilogram) dibagi dengan kuadrat tinggi badan (dalam meter) yang ditampilkan pada rumus (1):

$$BMI = \frac{\text{Berat Badan (kg)}}{\text{Tinggi Badan (cm}^2\text{)}} \quad (1)$$

Hasil perhitungan BMI kemudian dikategorikan sesuai pada Tabel 2:

TABEL 2. KATEGORI INDEKS MASSA TUBUH (BMI)

Nilai IMT (BMI)	Kategori
< 18,5	Kurus
18,5 – 24,9	Normal
25,0 - 29,9	Gemuk (<i>Overweight</i>)
>30,0	Gemuk (<i>Obesitas</i>)

5. Uji Black Box

Pengujian ini bertujuan untuk memeriksa fungsi-fungsi aplikasi sistem. Proses uji ini difokuskan pada hasil yang seharusnya diperoleh dari sistem yang akan diuji untuk mengecek sistem berfungsi seperti yang diharapkan. Adapun fitur yang akan di uji pada pengujian aplikasi ini adalah membuka aplikasi, about, menu *healthy lifestyle*, menu *healthy life info*, menu artikel, menu deteksi, *button home*, termasuk kalkulator BMI.

Setelah model diuji, hasil yang diperoleh akan dihitung menggunakan rumus (2) sampai dengan (5) berikut ini:

$$F_1 \text{ score} = 2 \left(\frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \right) \quad (2)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

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

$$\text{Accuracy} = \frac{TP+TN}{TP+FN+TN+FP} \quad (5)$$

TABEL 3. PARAMETER EVALUASI DETEKSI (TP, FP, FN, TN)

		Nilai Sebenarnya	
		True (Benar)	False (Salah)
Nilai Prediksi	True (Benar)	TP (True Positif) <i>Correct result</i>	FP (False Positif) <i>Unexpected result</i>
	False (Salah)	FN (False Negatif) <i>Missing result</i>	TN (True Negatif) <i>Correct absence of result</i>

III. HASIL DAN PEMBAHASAN

Bagian ini menampilkan data hasil pengujian yang telah dilakukan beserta ulasan mengenai evaluasi kinerja aplikasi berdasarkan parameter pengujian.

A. Hasil Training Dataset

Dalam penelitian ini, digunakan dataset gambar yang bersumber dari *website Roboflow*. Dalam menganalisis model dilakukan pelatihan data dengan jumlah data pelatihan 1.736, data uji 149, dan data *validasi* 206. Proses pelatihan dilakukan menggunakan komputasi *Google Colab Pro* dengan 150 *epoch*. Hasilnya ditampilkan pada Gambar 6.

```

all      5      14     0.988     1     0.995     0.736

Epoch  gpu_mem  box    obj    cls  labels  img_size
145/149  1.526   0.02154 0.01217 0.00125 32      416: 100% 0/0 [00:01:00:00, 7.12it/s]
Class  Images  Labels  P      R      mAP@0.5  mAP@0.5:0.95: 100% 1/1 [00:00:00:00, 29.42it/s]
all      5      14     0.99      1     0.995     0.71

Epoch  gpu_mem  box    obj    cls  labels  img_size
146/149  1.526   0.02377 0.01534 0.000745 46      416: 100% 0/0 [00:01:00:00, 7.10it/s]
Class  Images  Labels  P      R      mAP@0.5  mAP@0.5:0.95: 100% 1/1 [00:00:00:00, 26.54it/s]
all      5      14     0.992     1     0.995     0.75

Epoch  gpu_mem  box    obj    cls  labels  img_size
147/149  1.526   0.02575 0.0158 0.001076 35      416: 100% 0/0 [00:01:00:00, 7.06it/s]
Class  Images  Labels  P      R      mAP@0.5  mAP@0.5:0.95: 100% 1/1 [00:00:00:00, 29.52it/s]
all      5      14     0.995     1     0.995     0.792

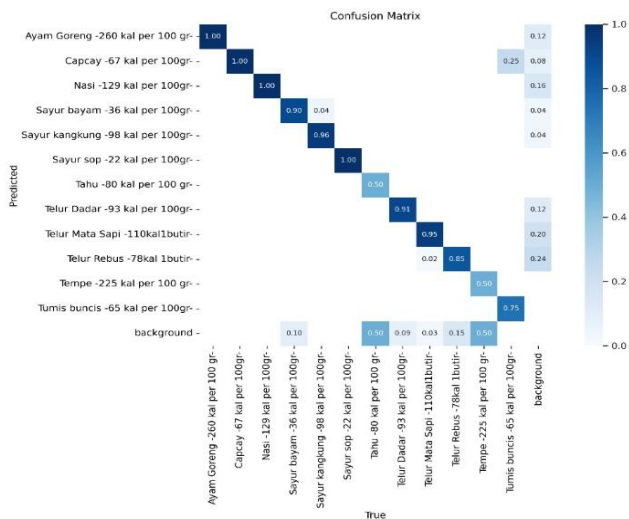
Epoch  gpu_mem  box    obj    cls  labels  img_size
148/149  1.526   0.02588 0.01521 0.0007734 39      416: 100% 0/0 [00:01:00:00, 6.94it/s]
Class  Images  Labels  P      R      mAP@0.5  mAP@0.5:0.95: 100% 1/1 [00:00:00:00, 25.51it/s]
all      5      14     0.998     1     0.995     0.803

Epoch  gpu_mem  box    obj    cls  labels  img_size
149/149  1.526   0.02392 0.01369 0.0007764 41      416: 100% 0/0 [00:01:00:00, 6.98it/s]
Class  Images  Labels  P      R      mAP@0.5  mAP@0.5:0.95: 100% 1/1 [00:00:00:00, 27.46it/s]
all      5      14     0.996     1     0.995     0.813

```

Gambar 6. Grafik Hasil Pelatihan Dataset YOLOv5

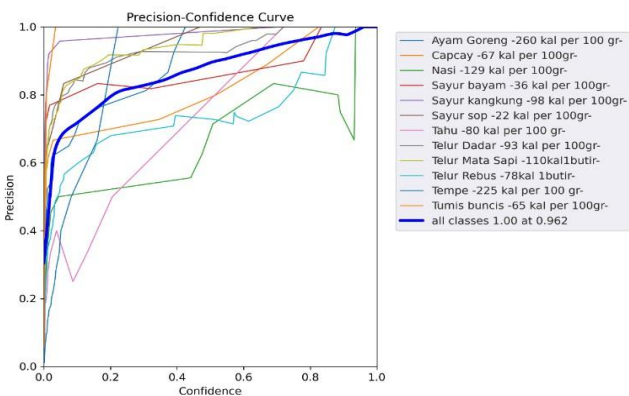
Pada Gambar 6, menampilkan hasil pelatihan dataset dengan menggunakan 150 *epoch* dengan nilai *GPU Mem* 1.52 GB, *Box loss* 0.02392, *Objectness Loss* 0.01369, dan *Classification Loss* 0.0007764 dengan *Size* 416 (menunjukkan pada *resolusi* gambar).



Gambar 7. Matriks Kebingungan (Confusion Matrix) Deteksi Makanan

Pada Gambar 7, menampilkan nilai *Confusion Matrix* sebagai informasi dan nilai perbandingan terhadap nilai target sebenarnya berdasarkan prediksi model. *Matriks* ini berbentuk tabel persegi yang menampilkan jumlah prediksi benar dan salah yang dihasilkan oleh model klasifikasi. Indikasi nilai menunjukkan seberapa baik model dalam mengidentifikasi makanan berdasarkan kalori per 100 gram. Diagonal utama, yang memanjang dari kiri atas ke kanan bawah, menunjukkan prediksi yang benar, di mana model berhasil mendeteksi jenis makanan dengan tepat. Angka yang berada di luar diagonal tersebut menunjukkan kesalahan sistem dalam deteksi jenis makanan. Misalnya, model salah menebak "Telur Rebus" sebagai "Telur Mata Sapi" dalam beberapa kasus. *Matriks* ini juga menginformasikan tingkat akurasi model bekerja optimal dan perlu perbaikan.

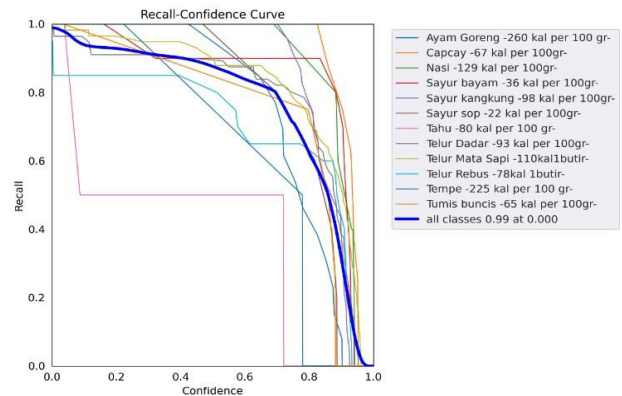
Selanjutnya pada Gambar 8, menampilkan parameter *Precision-Confidence Curve* adalah grafik yang menggambarkan hubungan antara presisi (*precision*) prediksi model dan tingkat keyakinan (*confidence*) dari prediksi tersebut. Grafik ini memberikan informasi mengenai kemampuan kinerja model dalam mengklasifikasi jenis makanan.



Gambar 8. Kurva Precision vs Confidence pada Model Deteksi Kalori

Pada Gambar 8, kurva garis biru diperoleh nilai *Precision* mencapai 1.00 pada *confidence* 0.962 untuk semua kelas makanan. Hal ini mengindikasikan bahwa model memiliki kemampuan prediksi yang sempurna untuk semua kelas makanan dengan derajat kepercayaan yang tinggi. Hasil ini

menunjukkan bahwa model dalam aplikasi bekerja dengan baik dalam mengklasifikasikan jenis makanan.



Gambar 9. Kurva Recall vs Confidence Model YOLOv5

Pada Gambar 6 menunjukkan kurva *Recall-Confidence* untuk berbagai jenis makanan beserta nilai kalorinya per 100 gramnya atau per butir. Parameter *Recall-Confidence Curve* adalah grafik yang digunakan untuk menilai performa model klasifikasi berdasarkan dua metrik utama: *recall* (sensitivitas) dan tingkat keyakinan (*confidence*). Masing-masing garis berwarna merepresentasikan performa model dalam mendeteksi jenis makanan tertentu berdasarkan tingkat kepercayaan (*confidence*) dan tingkat pengenalan kembali (*recall*). Misalnya, ayam goreng, capcay, nasi, sayur bayam, dan lainnya memiliki kurva masing-masing dengan warna berbeda. Garis biru tebal menunjukkan performa keseluruhan model untuk semua kelas makanan dengan *recall* sebesar 0.99 pada tingkat kepercayaan 0.01. Hasil ini menunjukkan bahwa model memiliki kemampuan yang sangat baik dalam mendeteksi makanan dengan tingkat kepercayaan yang tinggi.

Nilai *F1 score* dari hasil training data ini diperoleh 0.97 atau 97%. Nilai *F1 Score* ini adalah parameter yang perlu dipertimbangkan dalam menilai kinerja model klasifikasi. Parameter ini diperoleh dari kombinasi atau harmoni rata-rata dari *precision* dan *recall*. Nilai *F1 score* memberikan informasi mengenai keseimbangan antara *precision* dan *recall*, terutama saat ada ketidakseimbangan antara kelas positif dan negatif.

Hasil training data aplikasi menunjukkan nilai kinerja yang sangat baik, hal ini terbukti untuk semua parameter yang diperoleh yaitu nilai presisi, *confidence*, *recall* dan *f1 score*.

B. Menu Tampilan Antarmuka Aplikasi

Pada bagian ditampilkan User Interface (UI). Desain ini berdasarkan hasil desain merujuk pada rancangan desain sebelumnya pada Gambar 5. Berikut tampilan UI aplikasi dan dan penjelasan beberapa fitur menu aplikasi yang ditampilkan pada Gambar 10.



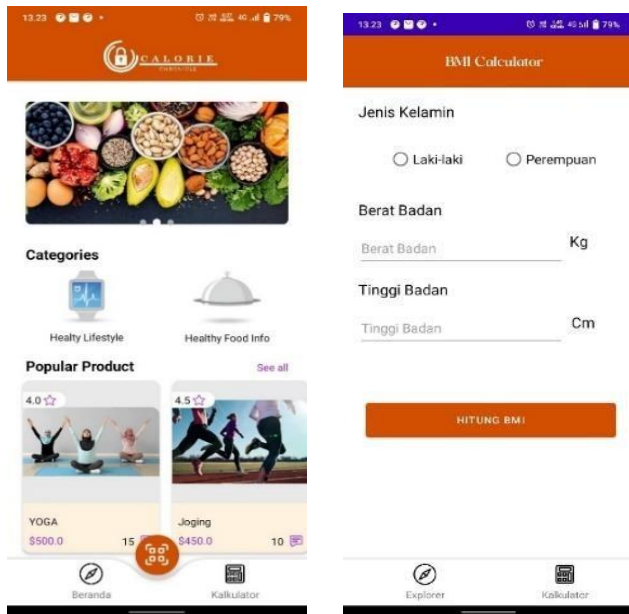
Gambar 10. Tampilan Awal Aplikasi dan Penjelasan Fitur

1. Tampilan Menu Utama

Menu utama adalah tampilan yang muncul setelah halaman tentang aplikasi. Menu ini berfungsi sebagai navigasi untuk mengakses berbagai fitur. Tampilan menu utama dapat dilihat pada Gambar 11.

2. Menu Kalkulator BMI

Menu ini berfungsi untuk memberikan informasi mengenai IMT. Dengan adanya fitur ini, pengguna dapat dengan mudah mengetahui kategori berat badan dan mendapatkan saran kesehatan yang sesuai. Fitur ini mendukung tujuan aplikasi untuk membantu pengguna dalam menjalankan program diet sehat dan mencapai berat badan ideal.



Gambar 11. Tampilan Menu Utama dan Kalkulator BMI Aplikasi

C. Hasil Pengujian Kinerja Aplikasi

1. Pengujian Jarak Detector (kamera) terhadap Objek dan Intensitas Pencahayaan

Jarak antara detektor dan pencahayaan diuji untuk mencapai deteksi yang optimal. Pengujian ini dilakukan dengan menentukan seberapa jauh jarak antara detektor dan objek, serta mengukur intensitas cahaya menggunakan Lux meter pada tiga kondisi cahaya yang berbeda. Pengujian jarak detektor dan pencahayaan terhadap objek dimulai dengan mendeteksi satu jenis makanan pada jarak 5 cm, 10 cm, dan 15 cm. Hasil dari pengujian ini ditampilkan pada Tabel 6.

TABEL 6. HASIL DETEKSI UJI JARAK DETEKTOR DAN INTENSITAS CAHAYA




Jarak Detektor ke Objek (cm)	Intensitas Cahaya (Lux)	Hasil Deteksi	Nilai Confidence
10	18		0.92 0.91 0.90
	130		0.95 0.94 0.90
	6		0.93 0.90 0.81
15	18		0.90 0.87 0.85
	130		0.95 0.91 0.90
	6		0.80 0.82 0.81
20	18		0.83 0.76 0.79
	130		0.82 0.77 0.70
	6		0.70 0.66 0.64

Berdasarkan data hasil uji pada Tabel 6 diperoleh hasil deteksi dengan akurasi 100% dengan nilai confidence yang bervariasi. Nilai confidence yang diperoleh dari percobaan ini berada pada kisaran 0.64 hingga 0.95. Adanya perbedaan nilai confidence dalam skema pengujian ini dipengaruhi oleh 2 (dua) faktor yaitu jarak kamera dan nilai intensitas cahaya yang diberikan ke objek yang akan dideteksi. Nilai confidence paling tinggi dihasilkan pada jarak 10 cm dengan nilai intensitas cahaya 130 lux sedangkan nilai confidence paling rendah dihasilkan pada jarak 20 cm dengan intensitas cahaya 6 lux. Berdasarkan hasil percobaan ini, diperoleh informasi bahwa aplikasi ini dapat mendeteksi objek dengan baik secara real-time ketika jarak kamera lebih dekat dengan objek dalam kondisi terintegrasi dengan intensitas pencahayaan yang tinggi atau kondisi lebih terang. Dalam uji ini, juga diperoleh jarak dan nilai intensitas cahaya yang optimum yaitu 10 cm dalam kondisi 130 lux. Data ini akan digunakan untuk skema pengujian kualitas resolusi kamera.

2. Pengujian Kualitas Kamera

Berikut data percobaan hasil deteksi aplikasi dengan variasi nilai resolusi kamera perangkat yang digunakan dalam mendeteksi objek makanan. Dalam pengujian ini digunakan 3 (tiga) perangkat android dengan kualitas resolusi kamera yang berbeda yaitu masing-masing 15 MP, 23 MP dan 67 MP. Variasi nilai resolusi yang digunakan ini sebagai sampel kualitas resolusi kamera rendah, sedang dan tinggi. Jarak deteksi yang diambil adalah 10 cm sebagai jarak optimum pengujian sebelumnya dengan kondisi pencahayaan 130 lux. Adapun secara detail hasil deteksinya ditampilkan pada Tabel 7 berikut ini.

TABEL 7. HASIL DETEKSI BERDASARKAN RESOLUSI KAMERA



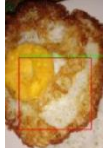
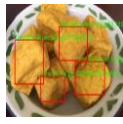


Nilai Resolusi (Megapixel, MP)	Hasil Deteksi	Nilai Confidence
15		0.90
23		0.93
67		0.95

Berdasarkan hasil pengujian pada Tabel 7, diperoleh data perangkat dengan nilai resolusi kamera 67 MP memiliki nilai confidence yang paling tinggi kemudian perangkat dengan resolusi 23 MP diperoleh 0.93 sedangkan kamera 15 MP menghasilkan nilai confidence 0.90. Nilai confidence yang dihasilkan oleh semua perangkat relatif sama dengan nilai lebih dari 0.90. Hal ini menunjukkan bahwa aplikasi ini secara konsisten dapat mendeteksi dengan baik objek makanan atau dengan kata lain aplikasi ini, cukup stabil digunakan pada berbagai jenis perangkat android yang memiliki kualitas resolusi kamera yang berbeda.




3. Pengujian Deteksi Klasifikasi Jenis Makanan










Pengujian ini terdiri dari 2 (dua) jenis yaitu dilakukan secara realtime dan melalui upload file gambar makanan yang ada dalam database atau diambil dari berbagai sumber di internet secara random. Pada pengujian real-time digunakan perangkat android dengan spesifikasi resolusi kamera 67 MP dengan jarak detektor kamera dengan objek yang digunakan adalah 10 cm dengan kondisi intensitas cahaya diabaikan. Percobaan dilakukan secara berulang sebanyak 3 (tiga) kali setiap sampel makanan. Berikut hasil deteksi dan nilai confidence disajikan pada Tabel 8, Tabel 9 dan Tabel 10.

TABEL 8. HASIL DETEKSI OBJEK SECARA REALTIME

Deteksi	Hasil Deteksi	Nilai Confidence
	Nasi	0.90 0.85 0.83
	Telur Dadar	0.97 0.92 0.87
	Telur Rebus	0.95 0.89 0.84
	Telur Mata Sapi	0.92 0.85 0.77
	Tahu Goreng	0.90 0.83 0.80
	Tempe Goreng	0.85 0.82 0.80
	Ayam Goreng	0.90 0.88 0.85

TABEL 9. HASIL DETEKSI UPLOAD GAMBAR (CITRA)

Deteksi	Hasil	Nilai Confidence
 <small>Nasi [129 kal per 100g], 0.9146976</small>	Nasi	0.91 0.88 0.81
 <small>Telur Mata Sapi [113kal 10sdtr telur], 0.61032313</small>	Telur Dadar	0.95 0.84 0.61
 <small>Telur Rebus [70kal 10sdtr telur], 0.734596</small>	Telur Rebus	0.90 0.88 0.72

Deteksi	Hasil	Nilai Confidence
 Telur Mata Sapi [116kal 100gr telur] 0.9022127	Telur Mata Sapi	0.94 0.90 0.70
 Tahu [80 kal per 100 gr] 0.7791152 Tahu [80 kal per 100 gr] 0.7782347 Tahu [80 kal per 100 gr] 0.72948914 Tahu [80 kal per 100 gr] 0.7091573 Tahu [80 kal per 100 gr] 0.81574185	Tahu	0.85 0.80 0.77
 Tempe [225 kal per 100 gr] 0.66879964	Tempe Goreng	0.80 0.75 0.66
 Ayam Goreng [200 kal per 100 gr] 0.64862833	Ayam Goreng	0.88 0.74 0.66
 Capcay [67 kal per 100gr] 0.92928016	Capcay	0.92 0.90 0.85
 Sayur sop [22 kal per 100gr] 0.674790463	Sayur Sop	0.87 0.71 0.78
 Sayur kangkung [88 kal per 100gr] 0.7021637	Sayur Kangkung	0.92 0.88 0.70
 Sayur bayam [36 kal per 100gr] 0.9208532	Sayur Bayam	0.92 0.86 0.81
 Tumis buncis [65 kal per 100gr] 0.7756585	Tumis Buncis	0.89 0.82 0.77

TABEL 10. HASIL DETEKSI MENU MIX (CAMPURAN)

Deteksi	Hasil	Kalori (kal)	Total Kalori (kal)	Nilai Confidence
	Nasi	129	694	0.74
	Tempe	225		0.70
	Ayam Goreng	260		0.57
	Tahu	80		0.53
	Nasi	129	275	0.95
	Telur Mata Sapi	110		0.91
	Sayur Bayam	36		0.46
	Nasi	129	487	0.85
	Ayam Goreng	260		0.71
	Sayur Kangkung	98		0.56
	Nasi	129	320	0.59
	Telur Dadar	93		0.58
	Sayur Kangkung	98		0.53

Berdasarkan data hasil pengujian pada Tabel 8, aplikasi dapat mendeteksi dengan tepat semua jenis makanan dengan nilai confidence yang relatif tinggi. Nilai confidence yang diperoleh berada pada kisaran 0.77 hingga 0.97. Data hasil deteksi juga menunjukkan nilai confidence yang cenderung relatif lebih rendah dari nilai 0.90 pada kelas makanan seperti telur mata sapi, tahu goreng, tempe goreng dan ayam goreng.

Hal yang sama juga diperoleh pada pengujian melalui upload file gambar, sistem aplikasi mampu mendeteksi dengan tepat setiap kelas. Nilai kepercayaan tertinggi terdapat pada kelas sayur bayam dengan nilai kepercayaan sebesar 0,92. Sementara itu, kelas telur dadar memiliki nilai kepercayaan yang lebih rendah yaitu 0,61. Nilai confidence yang relatif lebih rendah juga diperoleh pada sampel makanan yang disajikan dalam gorengan.

Pada Tabel 10 pengujian sampel makanan mix, aplikasi juga mampu mendeteksi semua sampel jenis makanan mix dengan tepat. Sistem aplikasi juga mampu menghitung masing-masing nilai kalori setiap jenis makanan dalam satu wadah penyajian dan mengkalkulasi secara total nilai kalorinya. Perhitungan estimasi kalori makanan ini berdasarkan kandungan kalori dalam 100 gram untuk setiap jenis makanan. Sama halnya dengan pengujian kondisi real-time dan upload gambar, nilai confidence yang relatif lebih rendah juga diperoleh pada percobaan ini mencapai 0.46.

Adanya perbedaan nilai confidence dan perolehan nilai confidence yang relatif lebih rendah disebabkan oleh faktor kemiripan warna dari sampel kelas makanan yang dideteksi. Jenis makanan yang disajikan dalam sajian goreng memiliki nilai confidence yang relatif lebih rendah dari jenis makan yang lain. Faktor lainnya adalah adanya tampilan warna background (warna motif piring) dan variasi beberapa jenis makanan yang disajikan dalam satu wadah membuat sistem aplikasi menghasilkan nilai confidence relatif lebih rendah. Meskipun demikian, aplikasi masih dapat mendeteksi semua sampel makanan dengan tepat dalam kondisi deteksi real-time, upload file gambar dan dalam kondisi sampel mix.

4. Pengujian Kalkulator BMI

Pada pengujian Kalkulator BMI, dilakukan pengujian langsung dengan memasukkan berat badan dan tinggi badan pengguna untuk menghasilkan nilai IMT (Indeks Massa Tubuh) yang mengkategorikan berat badan pengguna ke dalam golongan tertentu.

TABEL 11. HASIL PENGUJIAN KALKULATOR BMI

User	Jenis Kelamin	Berat (kg)	Tinggi (cm)	BMI (kg/cm ²)	Kategori
1	LK	53	163	19,95	Normal
2	LK	57	160	22,27	Normal
3	LK	98	167	35,14	Obesitas
4	LK	76	168	26,93	Berat Badan Berlebih
5	PR	44	156	18,08	Kurang
6	PR	45	157	18,26	Kurang
7	PR	60	156	24,65	Normal
8	PR	63	157	25,56	Berat Badan Berlebih

Pengujian fitur menu kalkulator BMI mengambil sampel sebanyak 8 user secara random dengan berat dan tinggi masing. Kalkulator BMI berhasil menampilkan nilai indeks BMI beserta kategorinya seperti yang ditampilkan pada Tabel 11. Hasil pengujian kalkulator BMI, menunjukkan bahwa perhitungan dari kalkulator BMI sudah sesuai dengan rumus BMI standar, yaitu berat badan dalam kilogram dibagi dengan tinggi badan dalam meter kuadrat. Hasil pengujian ini mengonfirmasi bahwa kalkulator BMI dapat digunakan secara akurat untuk menentukan indeks massa tubuh bagi user. Selain itu, menu fitur BMI dalam aplikasi ini juga dapat memberikan referensi dan acuan bagi pengguna dalam memberikan informasi rekomendasi mengenai kebutuhan kalori harian dan urgensi diet sehat.

5. Pengujian Metode *Black Box*

Pengujian menggunakan metode black box dilakukan untuk mengevaluasi fungsi setiap fitur guna mengetahui efektivitas dan kinerjanya, serta memastikan aplikasi dapat berfungsi dengan baik pada perangkat. Hasil pengujian pada Tabel 12 menunjukkan semua fitur berfungsi dengan baik sesuai yang diharapkan.

TABEL 12. HASIL PENGUJIAN SISTEM DENGAN METODE BLACK BOX

Skenario Pengujian	Bentuk Pengujian	Hasil Yang di Harapkan	Hasil
Membuka aplikasi	Menekan *.apk dari aplikasi	Running aplikasi dan menampilkan splashscreen aplikasi dan menu utama	Berhasil
About	Menggeser page	Menampilkan informasi seputar fitur yang ada di dalam aplikasi	Berhasil
Menu <i>Healthy Lifestyle</i>	Menekan menu <i>see all</i> pada halaman home	Menampilkan list penyakit yang dapat di deteksi	Berhasil
Menu <i>Healthy Life Info</i>	Menekan jenis penyakit	Menampilkan info detail terkait penyakit kuku	Berhasil
Menu Artikel	Menekan List Artikel	Menampilkan informasi kategori makanan	Berhasil
Menu deteksi	Menekan menu deteksi	Menampilkan metode deteksi	Berhasil
Button Home	Menekan Button Home	Menampilkan Home Page	Berhasil
Kalkulator BMI	Menekan Button Kalkulator	Menampilkan BMI kalkulator	Berhasil

IV. KESIMPULAN

Penerapan konsep pengolahan citra menggunakan algoritma CCN-YOLO versi 5 dalam perancangan dan implementasi pada aplikasi mobile penghitung nilai kalori makanan berhasil mendeteksi dengan akurat 100% untuk semua sampel jenis dan kelas beserta kalori total dari berbagai jenis makanan harian yang dijadikan sampel. Aplikasi ini khusus dirancang untuk mendeteksi kandungan kalori makanan per 100 gramnya. Nilai confidence optimum yaitu 0.97 dihasilkan pada jarak 10 cm, intensitas cahaya 108 lux dengan menggunakan kamera dengan resolusi 67 MP. Nilai confidence dalam penelitian ini dipengaruhi oleh beberapa faktor diantaranya jarak detektor, nilai intensitas pencahayaan, kemiripan warna sampel, adanya motif background wadah serta variasi mix makanan yang tersaji dalam satu wadah. Meskipun terdapat nilai confidence yang bervariasi, tetapi sistem aplikasi dapat secara konsisten dan akurat dalam mendeteksi jenis dan kelas makanan secara real-time. Hal ini menunjukkan bahwa kinerja sistem aplikasi ini berjalan dengan baik dan dapat digunakan sebagai salah satu rujukan bagi user untuk membantu dalam mengelola kebutuhan kalori makanan harian dan informasi mengenai diet sehat. Aplikasi hanya dapat memberikan prediksi nilai kalori makanan per 100 gramnya sehingga untuk pengembangan selanjutnya perlu optimasi untuk mendeteksi kalori makanan yang real dalam 1 (satu) porsi yang random. Selain itu, diperlukan data set yang lebih banyak dan beragam agar kondisi overfitting oleh sistem bisa diminimalisir.

UCAPAN TERIMA KASIH

Tim Penulis mengucapkan terima kasih kepada pengelola laboratorium multimedia cerdas pada program studi teknik multimedia sebagai tempat untuk melaksanakan penelitian sekaligus rekan diskusi bagi tim penulis.

REFERENSI

- [1] K. V. Dalakleidi, M. Papadelli, I. Kapolos, dan K. Papadimitriou, "Applying Image-Based Food-Recognition Systems on Dietary Assessment: A Systematic Review," *Advances in Nutrition*, vol. 13, no. 6, hlm. 2590–2619, Nov 2022.
- [2] R. U. Haque, R. H. Khan, A. S. M. Shihavuddin, M. M. M. Syeed, dan M. F. Uddin, "Lightweight and Parameter-Optimized Real-Time Food Calorie Estimation from Images Using CNN-Based Approach," *Applied Sciences (Switzerland)*, vol. 12, no. 19, Okt 2022.
- [3] A. Fakhrou, J. Kunthoth, dan S. Al Maadeed, "Smartphone-based food recognition system using multiple deep CNN models," *Multimed Tools Appl*, vol. 80, no. 21–23, hlm. 33011–33032, Sep 2021.
- [4] R. Agarwal, T. Choudhury, N. J. Ahuja, dan T. Sarkar, "Hybrid Deep Learning Algorithm-Based Food Recognition and Calorie Estimation," *J Food Process Preserv*, vol. 2023, 2023.
- [5] I. P. A. E. D. Eka Darma Udayana dan P. G. S. C. Nugraha, "Prediksi Citra Makanan Menggunakan Convolutional Neural Network untuk Menentukan Besaran Kalori Makanan," *Jurnal Teknologi Informasi dan Komputer*, vol. 6, no. 1, hlm. 30–8, 2020, doi: <https://doi.org/10.36002/jutik.v6i1.1001>.
- [6] Ratri Enggar Pawening, Meliana Eka Puteri, Agmelita Dwi Jianika, dan Fitriyah Hidayati, "Deteksi Objek pada

- Citra Makanan Sebagai Rekomendasi Diet Menggunakan Metode Mask R-CNN,” *Jurnal Ilmiah Teknik Informatika dan Komunikasi*, vol. 4, no. 1, hlm. 87–97, Des 2023.
- [7] N. Afif Bunyamin, A. Hendra Brata, dan R. K. Dewi, “Pengembangan Aplikasi Pengatur Kalori Harian menggunakan Tensorflow berbasis Android,” *Kurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 1, hlm. 452–460, 2022.
- [8] R. Agarwal, T. Choudhury, N. J. Ahuja, dan T. Sarkar, “Hybrid Deep Learning Algorithm-Based Food Recognition and Calorie Estimation,” *J Food Process Preserv*, vol. 2023.
- [9] G. C. Utami, C. R. Widiawati, dan P. Subarkah, “Detection of Indonesian Food to Estimate Nutritional Information Using YOLOv5,” *Teknika*, vol. 12, no. 2, hlm. 158–165, Jun 2023.
- [10] F. A. Puspitasari dan H. Pangaribuan, “Deteksi Kalori pada Citra Makanan dengan Algoritma Single Shot Multibox Detector,” *JURNAL COMASIE*, vol. 12, no. 02, 2025.
- [11] N. Nadiyah, E. M. Putri, M. Khairi, M. Furqan, dan B. Yusman, “Deteksi Objek Untuk Menghitung Perkiraan Kalori Makanan Menggunakan Metode R-CNN Mask Berbasis Web,” *Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi*, vol. 5, no. 1, hlm. 84–92, 2024.
- [12] T. Santya, C. Eko Suharyanto, P. Simanjuntak, dan A. Alfandianto, “Sistem Pakar Menentukan Maksimal Kalori Harian Berbasis Mobile,” vol. 1, no. 2, hlm. 70–77, 2019.
- [13] A. Istiqomah, R. F. Rachmadi, dan S. Nugroho, “Estimasi Kalori pada Citra Makanan Berbasis Deep Learning,” 2021. Diakses: 29 Desember 2024. [Daring]. Tersedia pada: <https://repository.its.ac.id/84021/>
- [14] R. Agarwal, T. Choudhury, N. J. Ahuja, dan T. Sarkar, “Hybrid Deep Learning Algorithm-Based Food Recognition and Calorie Estimation,” *J Food Process Preserv*, vol. 2023.
- [15] P. Pouladzadeh, P. Kuhad, S. V. B. Peddi, A. Yassine, dan S. Shirmohammadi, “Food calorie measurement using deep learning neural network,” dalam *Conference Record - IEEE Instrumentation and Measurement Technology Conference*, Institute of Electrical and Electronics Engineers Inc., Jul 2016.
- [16] A. Rouhafzay, G. Rouhafzay, dan J. Jbilou, “Image-based food monitoring and dietary management for patients living with diabetes: a scoping review of calorie counting applications,” *Front Nutr*, vol. 12, 2025.
- [17] N. Hamzidah, A. Jariyah, A. R. Ramadhani, N. Nurhasni, M. M. Parenreng, dan S. Suyuti, “Evaluation of image feature extraction using gray level co-occurrence matrix (GLCM) parameters and multilayer perceptron (MLP) algorithms in classifying typical batik motifs of South Sulawesi,” dalam *AIP Conference Proceedings*, American Institute of Physics, Jul 2024.
- [18] R. A. Shafa dan P. N. Andono, “Pendeteksi Visual Makanan dan Jumlah Kalorinya Menggunakan Algoritma Mask R-CNN Berbasis Both Telegram,” *JIPPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 10, no. 1, hlm. 641–651, Jan 2025.